

Πανεπιστήμιο Αιγαίου
Σχολή Κοινωνικών Επιστημών: Τμήμα Πολιτισμικής Τεχνολογίας
και Επικοινωνίας

Διατριβή για την απόκτηση διδακτορικού διπλώματος από το Τμήμα
Πολιτισμικής Τεχνολογίας και Επικοινωνίας

Κωνσταντίνος Καλογερόπουλος του Αντωνίου

ΖΗΤΗΜΑΤΑ ΑΡΧΑΙΟΛΟΓΙΚΗΣ ΘΕΩΡΙΑΣ ΚΑΙ ΨΗΦΙΟΠΟΙΗΣΗΣ ΑΝΑΣΚΑΦΙΚΩΝ
ΔΕΔΟΜΕΝΩΝ

Τριμελής Επιτροπή Παρακολούθησης:

Σοφία Δασκαλοπούλου, Ομότιμος Καθηγήτρια Τμήματος Πολιτισμικής
Τεχνολογίας και Επικοινωνίας, Πανεπιστημίου Αιγαίου.

Γεράσιμος Παυλογεωργάτος, Αναπληρωτής Καθηγητής του Πανεπιστημίου
Αιγαίου.

Εμμανουήλ Στεφανάκης, Καθηγητής Κλασικής Αρχαιολογίας και
Νομισματικής Τμήματος Μεσογειακών Σπουδών, Πανεπιστημίου Αιγαίου.

Μυτιλήνη 2019

Στον Ιωάννη

Θα ήθελα να ευχαριστήσω, με την ευκαιρία ολοκλήρωσης του παρόντος, την Δρα Σοφία Δασκαλοπούλου Ομότιμο Καθηγήτρια του Τμήματος Πολιτισμικής Τεχνολογίας και Επικοινωνίας, του Πανεπιστημίου Αιγαίου, για την ενθάρρυνσή της καθόλη τη διάρκεια της έρευνας και τις υποδείξεις της σε θεωρητικά και πρακτικά ζητήματα σχετικά με την εκπόνηση της διατριβής. Επίσης, τον Καθηγητή Κλασικής Αρχαιολογίας και Νομισματικής του Τμήματος Μεσογειακών Σπουδών, του Πανεπιστημίου Αιγαίου, Δρα Μανόλη Ι. Στεφανάκη, για την αμέριστη συμπαράσταση και τη στήριξή του στη διαμόρφωση και εφαρμογή του λογισμικού Digit ως διευθυντής ανασκαφών. Επίσης, θα ήθελα να ευχαριστήσω τον Δρα Νικόλα Βερνίκο Ομότιμο Καθηγητή του Τμήματος Πολιτισμικής Τεχνολογίας και Επικοινωνίας, του Πανεπιστημίου Αιγαίου, για τις ατελείωτες και διεγερτικές θεωρητικές συζητήσεις μας σε ζητήματα πολιτισμικού και ανθρωπολογικού ενδιαφέροντος. Τέλος θα ήθελα να ευχαριστήσω τον Αναπληρωτή Καθηγητή του Πανεπιστημίου Αιγαίου Γεράσιμο Παυλογεωργάτο, για τις εμβόλιμες παρατηρήσεις του, σε όλο το φάσμα της διατριβής.

Περιεχόμενα

Μέρος 1ο	
Αντικείμενο της έρευνας	6
Περίληψη	8
Δομή	10
Βιβλιογραφική έρευνα	11
Κεφάλαιο 1	21
1.1. Πολιτισμική τεχνολογία και αρχαιολογία πεδίου	21
1.2. Ιστορικό της έρευνας	22
1.3. Κατηγορίες αρχαιολογικής δραστηριότητας	30
1.4. Ζητήματα διαχείρισης της πληροφορίας	37
Κεφάλαιο 2: Γενικά και ειδικά ζητήματα αρχαιολογικής θεωρίας	43
2.1. Προβληματική των θεωρητικών κλάδων της αρχαιολογίας	43
2.2. Πολιτισμική εξελικτική αρχαιολογία	49
2.3. Πολιτισμική ιστορική αρχαιολογία	52
2.4. Διαδικαστική αρχαιολογία	55
2.5. Μεταδιαδικαστική αρχαιολογία	68
2.6. Ειδικά ζητήματα αρχαιολογικής θεωρίας	82
2.7. Αναλυτικοί χάρτες πληροφόρησης	94
Κεφάλαιο 3: Αρχαιολογική μεθοδολογία	101
3.1. Αρχαιολογικό αρχείο	101
3.2. Παράγοντες σχηματισμού και μετασχηματισμού του αρχαιολογικού αρχείου	106
3.3. Παράγοντες μετασχηματισμού του αρχαιολογικού αρχείου	109
3.4. Περιβαλλοντικοί παράγοντες μετασχηματισμού	111
3.5. Περιβαλλοντικοί παράγοντες διατήρησης του αρχαιολογικού αρχείου	115
3.6. Το αρχαιολογικό πλαίσιο	118
3.7. Στρωματογραφία συσχέτιση και επαλληλία	122
3.8. Αρχαιολογική έρευνα	127
3.9. Μέθοδοι ανεύρεσης αρχαιολογικών δεδομένων	130
3.10. Η αρχαιολογική ανασκαφή	153
3.11. Εργαλεία της αρχαιολογικής έρευνας	162
Κεφάλαιο 4: Μεταανασκαφική μεθοδολογία	166

4.1.	Μελέτες κεραμεικής	167
4.2.	Ανάλυση λίθινων εργαλείων	170
4.3.	Μεταλλουργική ανάλυση	171
4.4.	Σκελετικά κατάλοιπα	172
4.5.	Αναλύσεις πανίδας	174
4.6.	Αναλύσεις μαλακίων και ασπονδύλων	175
4.7.	Ανάλυση χλωρίδας	177
4.8.	Παλαιοοικολογικές μελέτες	178
4.9.	Ξύλο	180
4.10.	Συσσωρεύσεις απολιθωμάτων	181
4.11.	Γεωαρχειολογία	182
4.12.	Μέθοδοι χρονολόγησης	185
Μέρος 2ο		
Κεφάλαιο 5: Digit		200
5.1.	Ο τόπος	201
5.2.	Η αρχαιολογική θεωρία στην ανασκαφή της Κυμισάλας	206
5.3.	Digit	218
5.4.	Εργαλεία	226
5.5.	Συμπεράσματα, δυνατότητες-μελλοντικές προοπτικές	235
Βιβλιογραφία		237
Διαδίκτυο		272
Παράρτημα 1 Ο τρόπος λειτουργίας του λογισμικού		276
Παράρτημα 2 Οι πίνακες της βάσης δεδομένων Digit		285
Παράρτημα 3 Υλοποίηση της κλάσης DigitSystem		305

Αντικείμενο της έρευνας

Αντικείμενο της παρούσας έρευνας είναι η συνδυαστική θεώρηση αρχαιολογικής θεωρίας και εφαρμογής λογισμικού και η διερεύνηση της εφαρμογής της θεωρίας στη διαμόρφωση ιεραρχικού μοντέλου οντοτήτων. Ο στόχος είναι η χρήση του μοντέλου σε σχεσιακή βάση δεδομένων για την κατασκευή μιας κλάσης. Σκοπός είναι η υλοποίηση αυτής της κλάσης με την κατασκευή λογισμικού, που μπορεί να χρησιμοποιηθεί στο πεδίο για την καταγραφή της αρχαιολογικής έρευνας.

Για τη δημιουργία του ιεραρχικού μοντέλου είναι αναγκαία η αξιοποίηση της αρχαιολογικής θεωρίας στο μέγιστο δυνατό εύρος της, προκειμένου να διερευνηθούν και να καταγραφούν όλες οι παράμετροι που διαμορφώνουν ή επηρεάζουν τις ιεραρχημένες οντότητες αρχαιολογική γεωγραφική θέση, αρχαιολογική θέση, ανασκαφική φάση, ανασκαφικό πλαίσιο και αρχαιολογικά ευρήματα.

Όσον αφορά στην αρχαιολογική οντολογία είναι λιγοστές, έως μηδενικές, οι προσπάθειες διαμόρφωσης ενός συνεπούς μοντέλου σε σχέση με την αρχαιολογική θεωρία, που θα μπορούσε να χρησιμοποιηθεί γενικά σε εφαρμογές της υπολογιστικής αρχαιολογίας. Συνεπώς, τα ερευνητικά ερωτήματα σε σχέση με τον στόχο και τους σκοπούς της παρούσας έρευνας έχουν ως εξής:

1. Τι είναι αρχαιολογική θεωρία και μέθοδος;
2. Σε ποιο βαθμό εμπλέκεται η αρχαιολογική θεωρία στην κατασκευή της αρχαιολογικής οντολογίας;
3. Με ποιο τρόπο είναι δυνατόν να χρησιμοποιηθεί η αρχαιολογική οντολογία στην κατασκευή σχεσιακής βάσης δεδομένων;
4. Ποιοι είναι οι παράγοντες διαμόρφωσης της κλάσης αρχαιολογία,
5. Ποιες είναι οι υπερκλάσεις της κλάσης αρχαιολογία,
6. Τι περιλαμβάνει η υποκλάση αρχαιολογική θεωρία
7. Τι περιλαμβάνει η υποκλάση αρχαιολογική μέθοδος,
8. Ποια είναι τα απαραίτητα εργαλεία για την κατασκευή ημερολογιο-κεντρικού λογισμικού για χρήση στο πεδίο;

9. Ποιες είναι οι ανάγκες καταγραφής στην αρχαιολογική γεωγραφική θέση της Κυμισάλας και πώς συνδέονται με τη διαμόρφωση του εν λόγω λογισμικού;
10. Ποια είναι η συμβολή της ηλεκτρονικής καταγραφής στο πεδίο;
11. Ποιοι είναι οι παράγοντες καλής πρακτικής και με ποιο τρόπο μπορούν να συμβάλλουν στην ανάπτυξη του συγκεκριμένου λογισμικού;

Η απάντηση των παραπάνω ερευνητικών ερωτημάτων διαμορφώνει και το εννοιολογικό πλαίσιο της παρούσας έρευνας, Η δεκαετής εφαρμογή της στο πεδίο αιτιολογεί τις επιλογές των παραμέτρων των αρχικών οντοτήτων (αρχαιολογική γεωγραφική θέση, αρχαιολογική θέση, ανασκαφική φάση, ανασκαφικό πλαίσιο και αρχαιολογικά ευρήματα) και την πλήρη διασύνδεσή τους με την αρχαιολογική θεωρία και μέθοδο.

Περίληψη

Στην παρούσα διατριβή επιχειρείται μέσω της ανάγνωσης γενικών και ειδικών ζητημάτων της αρχαιολογικής θεωρίας, η καταγραφή εκείνων των στοιχείων που μπορούν να αξιοποιηθούν από τη σύγχρονη τεχνολογία για την πλήρη καταγραφή των δεδομένων που προκύπτουν από την ανασκαφική διαδικασία. Επιπλέον, επιχειρείται η διαμόρφωση σχετικού λογισμικού (software) και σχεσιακής βάσης δεδομένων (relational database), που μπορούν να χρησιμοποιηθούν στο πεδίο ως εργαλεία εισαγωγής πρωτογενών δεδομένων και εξαγωγής πληροφοριών –οπτικοποιημένων ή μη- και στατιστικών στοιχείων σχετικών με τη διαδικασία της ανάλυσης και εξαγωγής συμπερασμάτων, έως την τελική δημοσίευση του ανασκαφικού έργου.

Με κεντρικό άξονα το ανασκαφικό ημερολόγιο ως βασικό εργαλείο καταγραφής της στρωματογραφικής αλληλουχίας, η παρούσα διατριβή επεκτείνεται σε διεπιστημονικά ζητήματα της αρχαιολογικής θεωρίας που σχετίζονται με το πρακτικό έργο της ανασκαφικής μεθοδολογίας, από την επιφανειακή επιθεώρηση, τον σχεδιασμό και τη μελέτη της ανασκαφικής διαδικασίας, έως τη λήψη απόφασης για ανασκαφή και συνεπακόλουθα τη μεταανασκαφική ανάλυση και διαχείριση των καταχωρισμένων δεδομένων που προκύπτουν από την ημερολογιακή καταγραφή και το αρχαιολογικό εργαστήριο.

Στη συνέχεια αναλύονται οι παράμετροι και τα εργαλεία δημιουργίας του συστήματος διαχείρισης περιεχομένου (CMS) *Digit* και αναλύονται οι σχέσεις των πινάκων της βάσης δεδομένων (MySQL), όπως και οι δυνατότητες διασύνδεσης του συστήματος με βιβλιοθήκες οπτικοποίησης δεδομένων (data visualization), στατιστικής ανάλυσης και γεωγραφικών συστημάτων πληροφοριών (GIS).

Βάσει των παραπάνω η απαιτούμενη μέθοδος για την ολοκλήρωση μιας τέτοιας έρευνας προανατολίστηκε στην:

1. Καταγραφή και σύγκριση σύγχρονων και παλαιότερων θεωρητικών τάσεων στο γνωστικό αντικείμενο της αρχαιολογίας.
2. Ανάλυση των βασικών εννοιών που συνθέτουν το εννοιολογικό πλαίσιο του «αρχαιολογικού έργου»
3. Καταγραφή των μεθόδων που χρησιμοποιούνται στο ανασκαφικό έργο.

4. Ανάλυση βασικών εργαλείων που χρησιμοποιούνται για την κατασκευή μιας σχεσιακής βάσης δεδομένων.
5. Συνοπτική περιγραφή της γλώσσας προγραμματισμού και του περιβάλλοντος διαμόρφωσης του χρησιμοποιούμενου λογισμικού.
6. Περιγραφή του τύπου πειραματικής εφαρμογής του λογισμικού στην περίπτωση της ανασκαφής της Κυμισάλας.
7. Αναφορά στη θεωρία πίσω από την ανασκαφή της Κυμισάλας και την κατασκευή του λογισμικού Digit.

Για την πρακτική εφαρμογή των παραπάνω αξιοποιήθηκαν δεδομένα από την πανεπιστημιακή ανασκαφή της Κυμισάλας (KARP) υπό τη διεύθυνση του Καθ. Μανόλη Στεφανάκη, του Τμήματος Μεσογειακών Σπουδών και της Δρος Βασιλικής Πατσιαδά της ΕΠΚΑ Δωδεκανήσου.

Δομή

Η έρευνα δομείται σε 2 μέρη. Στο πρώτο μέρος διερευνώνται στα κεφάλαια 1, 2, 3 και 4 γενικά και ειδικά ζητήματα αρχαιολογικής θεωρίας, η ανασκαφική μεθοδολογία και θεωρητικά ζητήματα μεταανασκαφικής ανάλυσης σχετικά με την ανάλυση των δεδομένων. Στο δεύτερο μέρος, στο κεφάλαιο 4, παρουσιάζονται θεωρητικά και τεχνικά ζητήματα που σχετίζονται με την ανάπτυξη του λογισμικού digit. Ακολουθεί η βιβλιογραφία και τα σχετικά παραρτήματα.

Ειδικότερα στο εισαγωγικό πρώτο κεφάλαιο αναφέρονται οι σχέσεις πολιτισμικής τεχνολογίας και αρχαιολογίας πεδίου, παρουσιάζεται το ιστορικό της έρευνας γύρω από την εφαρμογή υπολογιστικών συστημάτων στην αρχαιολογία. Επίσης, αναλύονται οι κατηγορίες αρχαιολογικής δραστηριότητας και παρουσιάζονται ζητήματα διαχείρισης της πληροφορίας με συνοπτική εισαγωγή στη θεωρία του ορίζοντα πληροφόρησης.

Στο δεύτερο κεφάλαιο καταγράφονται οι σύγχρονες θεωρητικές τάσεις γύρω από γενικά ζητήματα (πολιτισμική, εξελικτική, διαδικαστική, μετα-διαδικαστική αρχαιολογία) και ειδικά ζητήματα, όπως η φαινομενολογία και οι αναλυτικοί χάρτες πληροφόρησης. Επίσης, αναλύονται περαιτέρω οι πιθανές χρήσεις της θεωρίας του ορίζοντα πληροφόρησης όχι μόνο στην αρχαιολογία πεδίου, αλλά και σε συναφείς επιστήμες πολιτισμικού ενδιαφέροντος.

Στο τρίτο κεφάλαιο παρουσιάζονται κλασικές και σύγχρονες θεωρήσεις για την αρχαιολογική μεθοδολογία, καθώς και σύγχρονες συζητήσεις για το αρχαιολογικό αρχείο και τους παράγοντες σχηματισμού και μετασχηματισμού της αρχαιολογικής μαρτυρίας, τη στρωματογραφία, το αρχαιολογικό πλαίσιο, την αρχαιολογική έρευνα και τα αρχαιολογικά εργαλεία σε μια ευρύτερη προοπτική του όρου εργαλεία.

Στο τέταρτο κεφάλαιο αναπτύσσονται ζητήματα μεταανασκαφικής ανάλυσης, που περιλαμβάνουν μελέτες κεραμεικής, ανάλυση λίθινων εργαλείων, μεταλλουργική ανάλυση, ζητήματα σκελετικών καταλοίπων, αναλύσεις πανίδας και χλωρίδας, αναλύσεις μαλακίων και ασπονδύλων, παλαιοοικολογικές μελέτες κ.ά. Εντέλει παρουσιάζεται το θεωρητικό υπόβαθρο και τεχνικά στοιχεία μεθόδων χρονολόγησης.

Στο πέμπτο κεφάλαιο παρουσιάζεται συνοπτικά ιστορικό του συγκεκριμένου πεδίου και το εννοιολογικό πλαίσιο για τη διαμόρφωση της συγκεκριμένης σχεσιακής βάσης δεδομένων και οι σχέσεις των οντοτήτων που τη συγκροτούν. Στη συνέχεια παρουσιάζονται τα εργαλεία που χρησιμοποιήθηκαν για την κατασκευή του λογισμικού Digit και επεξήγηση των βασικών του παραμέτρων.

Βιβλιογραφική έρευνα

Η βιβλιογραφική έρευνα κινήθηκε σε τέσσερις άξονες. Όσον αφορά στο εννοιολογικό πλαίσιο της αρχαιολογικής θεωρίας υφίσταται σημαντική παραγωγή κειμένων που σχετίζονται κυρίως με ζητήματα ερμηνείας και προσπάθειας ορισμού της έννοιας αρχαιολογική θεωρία. Σημαντική είναι, επίσης, η γενική και εξειδικευμένη βιβλιογραφία σε ζητήματα που άπτονται της μεθοδολογίας της αρχαιολογίας πεδίου και των εφαρμογών της. Εκτεταμένη είναι και η βιβλιογραφία σχετικά με τις διεπιστημονικές προεγγίσεις της αρχαιολογίας πεδίου. Εντέλει ιδιαίτερα μεγάλη σε όγκο είναι η βιβλιογραφία για τις σχεσιακές βάσεις δεδομένων, για τις γλώσσες προγραμματισμού όσο και τις τεχνικές κατασκευής ενός συστήματος διαχείρισης περιεχομένου (CMS). Μικρή είναι, ωστόσο, η βιβλιογραφία που σχετίζεται με τον ρόλο του ημερολογίου και τη συσχέτισή του με τις θεωρητικές δομές της αρχαιολογικής επιστήμης.

1. Αρχαιολογία και Μέθοδος

Βασικό έργο για τη συνδυαστική θεώρηση αρχαιολογικής θεωρίας και μεθόδου είναι το *Archaeology, Theories Methods and Practice* των Colin Renfrew and Paul Bahn στην αναθεωρημένη έκδοσή του 2016 από τους Thames & Hudson¹. Δεδομένου ότι υφίστανται σύνθετες σχέσεις μεταξύ θεωρίας, μεθόδου και πρακτικής στην αρχαιολογία στο εν λόγω έργο παρουσιάζονται συγκεκριμένα πρακτικά παραδείγματα με συναφείς τεχνικές και θεωρητικές προσεγγίσεις, χωρίς ενασχόληση με αμφιλεγόμενα ζητήματα ερμηνείας της σύγχρονης αρχαιολογίας, είτε στον τομέα της θεωρίας είτε της πολιτικής.

¹ Renfrew and Bahn 2016.

Με τη σειρά του το *Archaeological theory: who sets the agenda?* σε επιμέλεια των Norman Yoffee και Andrew Sherratt² αναδεικνύει το γεγονός ότι η αρχαιολογική θεωρία δεν είναι ανεξάρτητη από τα ζητήματα που χρειάζεται να επιλυθούν, καθώς προκύπτουν από συγκεκριμένα προβλήματα. Στο συγκεκριμένο έργο διερευνάται κατά πόσο οι θεωρητικές θέσεις σχετίζονται με τους κύριους τομείς που μελετούν οι αρχαιολόγοι. Δεδομένου ότι δεν υπάρχει μια ενιαία σχολή αρχαιολογίας, το ζήτημα της αρχαιολογικής θεωρίας χαρακτηρίζεται συχνά από ανταγωνιστικές θεωρητικές στάσεις που προκύπτουν από διαφορετικά δεδομένα και συνακόλουθα από διαφορετικές ερμηνείες. Η αρχαιολογία χρειάζεται ένα κοινό έδαφος μεταξύ των πρακτικών της. Τούτο σημαίνει ότι χρειάζεται να ταιριάζει η αρχαιολογική θεωρία στα διάφορα επιμέρους προβλήματα με τα οποία πρέπει να ασχοληθούν οι αρχαιολόγοι. Οι διάφορες θεωρητικές σχολές καθορίζουν τι αποτελεί έγκυρη ερμηνεία στην αρχαιολογική έρευνα εδραιωμένες σε ένα μόνο τμήμα του εμπειρικού υλικού, αντιδρώντας ενίοτε σε προηγούμενες θεωρητικές θέσεις, χωρίς να διασφαλίζεται ότι δεν πρόκειται με τη σειρά τους να αντικατασταθούν εξαιτίας μελλοντικών θεωρητικών μετατοπίσεων.

Αναμφίβολα οι ορισμοί για το τι είναι η αρχαιολογία ποικίλουν ιδιαίτερα. Το πρόβλημα περιγράφει ο Michael B. Schiffer ήδη από το 1978 στο *Advances in Archaeological Method and Theory*³. Από την παραδοσιακή αντίληψη της αρχαιολογίας ως μελέτης του παρελθόντος της ανθρωπότητας από τα υλικά κατάλοιπα, έως τη σύγχρονη διεπιστημονική αντιμετώπιση της αρχαιολογικής επιστήμης με προηγμένη εφαρμοσμένη τεχνολογία, οι αντιλήψεις μας για το αρχαιολογικό έργο έχουν πολλαπλασιαστεί υπέρμετρα και είναι πιθανώς εξαιρετικά δύσκολο να περιληφθούν όλες οι θεωρητικές αναλύσεις σε ένα και μοναδικό δοκίμιο. Παρά την πεποίθηση των αρχαιολόγων ότι έχουν την απάντηση, το ζήτημα *τι είναι* η αρχαιολογία σε γενικό επίπεδο παραμένει άλυτο. Από αυτή την άποψη είναι πιθανώς προτιμότερο να εστιάζονται οι προσπάθειες των θεωρητικών της αρχαιολογίας στον εντοπισμό εκείνων των περιοχών, ιδιαίτερα στη μέθοδο και τη θεωρία, όπου η πρόοδος στην έρευνα φαίνεται να προχωράει με μεγάλη ταχύτητα, αναθεωρώντας παλαιότερες μεθόδους,

² Yoffee and Sherratt 1993.

³ Schiffer 1978.

διαμορφώνοντας νέες προοπτικές. Η πρόοδος στην αρχαιολογική μέθοδο και θεωρία είναι το μέσο για τη διαμόρφωση και τη διάδοση νέων μεθοδολογικών και θεωρητικών αντιλήψεων και η διερεύνηση τέτοιων περιοχών, θα βοηθήσει πιθανώς να εντοπιστεί το *τι είναι* και *τι δεν είναι* η αρχαιολογία. Προς αυτή την κατεύθυνση, το δεκάτομο πλέον *Advances in Archaeological Method and Theory*, σε επιμέλεια M.B. Schiffer⁴, προσφέρει τη σημαντική οπτική σύγχρονων συζητήσεων για τη φύση του αρχαιολογικού έργου.

Σημαντική συμβολή στον σχεδιασμό της παρούσας διατριβής και στην ελληνική απόδοση των αρχαιολογικών όρων υπήρξε το *Αρχαιολογία στον Ελληνικό Χώρο*, Τόμος Α', με συγγραφείς την Αλεξάνδρα Κουκουζέλη, την Ελένη Μανακίδου και τον Κωνσταντίνο Σμπόνια, πιθανώς το πλέον οργανωμένο πανεπιστημιακό εγχειρίδιο (Ελληνικό Ανοικτό Πανεπιστήμιο) σε ζητήματα αρχαιολογικής θεωρίας και μεθοδολογίας⁵.

2. Μεθοδολογία της αρχαιολογίας πεδίου και εφαρμογές

Όσον αφορά στην αρχαιολογία πεδίου, θεμελιώδεις είναι οι θεωρήσεις του Peter L. Drewett στο *Field Archaeology: An Introduction*, που εκδόθηκε το 1999⁶, στο οποίο περιγράφονται οι διαφορετικές τεχνικές που ακολουθούνται στη διάρκεια του ανασκαφικού έργου. Ασχέτως του τρόπου με τον οποίο αντιμετωπίζονται από διαφορετικές σχολές τα ζητήματα πεδίου, ή του τρόπου με τον οποίο οφείλει να προσεγγίζει σε πραγματική βάση ο αρχαιολόγος το ανασκαφικό έργο και τις νόρμες που επιβάλλει η εκάστοτε Κρατική Αρχαιολογική Υπηρεσία ή ο εκάστοτε χρηματοδότης, στο συγκεκριμένο έργο αναδεικνύεται η ανάγκη για ευελιξία στην προσέγγιση διαφορετικών τεχνικών και σφαιρική εκτίμηση των ζητημάτων που ανακύπτουν. Ο ρόλος του αρχαιολόγου στην προκειμένη περίπτωση δεν είναι η απλή τεχνική επίβλεψη ανασκαφικού έργου στη βάση κάποιου δόγματος, αλλά ο αναστοχασμός για την αντιμετώπιση των μοναδικών προβλημάτων που παρουσιάζονται σε κάθε ιδιαίτερο πεδίο.

Ποιοι είναι οι στόχοι της αρχαιολογίας πεδίου; Πώς συνδέονται με τους στόχους της αρχαιολογίας; Στο επίπεδο της θεωρίας τελικός στόχος της αρχαιολογίας είναι η μελέτη της ανθρώπινης συμπεριφοράς και της πολιτιστικής

⁴ Schiffer 1978, 1980, 1981, 1982, 1983, 1984, 1985, 1986, 1987, 1988.

⁵ Κουκουζέλη, Α. Μανακίδου, Ε. Σμπόνιας, Κ. 2003.

⁶ Drewett, 1999.

αλλαγής στο παρελθόν⁷. Ένας τέτοιος ευρύς ορισμός επιτρέπει τη διάκριση σε πολλά είδη αρχαιολογίας, καθένα με το δικό του σύνολο στόχων. Η αρχαιολογία, σε αντίθεση με τους πολιτισμούς και τις κοινωνίες που βρίσκονται υπό την εστίασή της, έχει υποστεί πολλές αλλαγές τον περασμένο αιώνα. Με κάθε αλλαγή, νέοι στόχοι αντικαθιστούν ή απλώς ενισχύουν τους παλαιότερους⁸. Στο *Field methods in archaeology* οι Thomas R. Hester, Harry J. Shafer και Kenneth L. Feder εξετάζουν με ανανεωμένη οπτική την αρχαιολογία πεδίου, επικεντρωμένοι στη μελλοντική προοπτική της, έτσι όπως εξελίσσεται μέσω της διεπιστημονικής φύσης της αρχαιολογικής έρευνας στο πεδίο⁹.

Την τελευταία εικοσαετία έχουν γίνει βαθιές αλλαγές στην οργάνωση και την πρακτική της αρχαιολογίας σε παγκόσμιο επίπεδο. Σε αυτές τις αλλαγές είναι εμφανής η επίδραση των κυβερνητικών πολιτικών στον τρόπο με τον οποίο αναπτύσσεται η αρχαιολογία, ο καθορισμός προτύπων, τόσο δεοντολογικών όσο και πρακτικών, η αυξημένη έμφαση στην προανασκαφική έρευνα, η χρήση προηγμένων γεωφυσικών ερευνών και η ταχεία ανάπτυξη και χρήση των τεχνολογιών πληροφορικής στις αρχαιολογικές έρευνες, οδήγησαν σε πολλές επίσημες και ανεπίσημες συζητήσεις για όλες τις πτυχές του αρχαιολογικού επιστημονικού πεδίου. Στο *Techniques of Archaeological Excavation* ο Philip Barker συζητά την ταχεία ανάπτυξη νέων τεχνικών και μεθόδων που έρχονται και παρέρχονται την τελευταία εικοσαετία, επιμένει, ωστόσο, πως ο ανθρώπινος παράγοντας είναι ακόμη ο ισχυρότερος ερμηνευτής και καταγραφέας στο πεδίο και είναι εκείνος που παράγει την νέα μαρτυρία για το ανθρώπινο παρελθόν¹⁰.

Οι Hannah Cobb, Oliver J.T. Harris, Cara Jones και Philip Richardson στο *Reconsidering Archaeological Fieldwork*, εξετάζουν τις εντάσεις που περιβάλλουν τη διχοτόμηση υποκειμένου / αντικειμένου στην αρχαιολογία, ιδίως μεταξύ θεωρίας και πρακτικής. Στη σύγχρονη βιβλιογραφία επικρατεί μια τάση αμφισβήτησης στην παραγωγή ερμηνειών που βασίζονται στις διχοτομίες για την κατανόηση του παρελθόντος. Τέτοιες προσεγγίσεις, οι οποίες διαχωρίζουν το κοινωνικό από τον υλικό και τον πολιτισμό από τη φύση, κατηγορήθηκαν ότι επέβαλαν έναν ιδιαίτερα σύγχρονο τρόπο διερεύνησης του κόσμου στο

⁷ Trigger 1989 371

⁸ Trigger 1989, 370-411.

⁹ Hester, Shafer and Feder 2016.

¹⁰ Barker 1993.

παρελθόν¹¹. Μία τέτοια επιβολή, ωστόσο, απεμπολεί την αντίληψη ότι οι άνθρωποι και τα πράγματα στο παρελθόν ήταν αμοιβαία συστατικά ενός συστήματος. Βάσει αυτής της κυρίαρχης προοπτικής που εξακολουθεί σήμερα να βασίζεται σε τέτοιες διχοτομίες, τόσο το παρόν όσο και το παρελθόν διαχωρίζονται εξαπλουστευτικά, σε ενεργητικά, σκεπτόμενα υποκείμενα (που διαθέτουν δράση) και παθητικά αντικείμενα (που δε διαθέτουν)¹², εμποδίζοντας σε μεγάλο βαθμό τη διαμόρφωση μιας ενοποιημένης εικόνας του αρχαίου κόσμου, στον οποίο τα υποκείμενα και αντικείμενα αλληλεπιδρούν με πολλούς τρόπους.

Σημαντική συμβολή στη μετανασκαφική ανάλυση είναι το *Archaeologist's Laboratory The Analysis of Archaeological Data*, του E.B. Banning, ως εισαγωγή στις βασικές εργαστηριακές και αναλυτικές τεχνικές στην αρχαιολογία. Δεν είναι κείμενο για τη στατιστική ή την αρχαιομετρία, αν και περιέχει κάποιες στατιστικές εφαρμογές σε αρχαιομετρικά προβλήματα. Ορισμένα βιβλία καλύπτουν μεθόδους ανασκαφής, άλλα στατιστικές εφαρμογές, άλλα ταξινόμηση, ανάλυση της πανίδας, παλαιοβοτανική ή αρχαιομετρία, ενώ άλλα ασχολούνται με χωρικές αναλύσεις ή εφαρμογές ηλεκτρονικών υπολογιστών, αλλά πολύ λίγα με τα κοινά θέματα που συνδέουν όλα αυτά τα πράγματα σε έναν πυρήνα αρχαιολογικής πρακτικής¹³.

3. Σχεσιακές βάσεις δεδομένων, γλώσσες προγραμματισμού, συστήματα διαχείρισης περιεχομένου (CMS).

Η βιβλιογραφία για τις σχεσιακές βάσεις δεδομένων, τις γλώσσες προγραμματισμού και τα συστήματα διαχείρισης περιεχομένου είναι εξαιρετικά πλούσια. Δεδομένου ότι το συγκεκριμένο λογισμικό υπέστη σειρά τροποποιήσεων επί σειρά ετών, εδώ καταγράφεται η κύρια βιβλιογραφική διερεύνηση για την ανάπτυξη της τελευταίας μορφής του, δηλαδή βιβλιογραφία σχετική με τη MySQL¹⁴ ως σχεσιακή βάση δεδομένων και τα εργαλεία της, σχετική

¹¹ Βλ. Olsen 2010, επίσης Thomas 2004 και Webmoor and Witmore 2008.

¹² Cobb et al 2012.

¹³ Banning 2006.

¹⁴ Kofler 2006.

βιβλιογραφία με το Visual Studio (2015), τη γλώσσα προγραμματισμού C#¹⁵ και το περιβάλλον Asp.NET Webforms¹⁶. Άλλα πλαίσια που χρησιμοποιήθηκαν και απαντώνται στη βιβλιογραφική και διαδικτυακή έρευνα είναι η HTML5¹⁷, το CSS3¹⁸, το JQuery-3.3.1¹⁹, το JQuery-ui-1.12.1²⁰, ο CKEditor²¹, το ImageResizer.3.4.3²², όλα ανοικτού πηγαίου κώδικα.

Δεδομένου ότι έγινε μετάβαση από το VS 2013 στο VS 2015 χρειάστηκε σχετική βιβλιογραφική έρευνα, για να γίνουν κατανοητές οι διαφορές ανάμεσα σε φαινομενικά ίδια εργαλεία. Για παράδειγμα, η διεπαφή χρήστη για το Visual Studio 2015 (VS2015) διαθέτει αρκετές ομοιότητες με εκείνη του Visual Studio 2013 (VS2013) από άποψη εμφάνισης. Ωστόσο, υφίστανται διαφορές, τις οποίες χρειάζεται να γνωρίζει κανείς προκειμένου να πλοηγείται εύκολα στο Ενσωματωμένο Περιβάλλον Ανάπτυξης (Integrated Development Environment - IDE)²³.

Η διαμόρφωση ενός συστήματος διαχείρισης περιεχομένου συνοδεύεται, όσον αφορά τουλάχιστον στην αρχαιολογία με νόρμες καλής πρακτικής και συντήρησης των δεδομένων που προκύπτουν από την έρευνα. Η αρχαιολογία από αντικειμενοστρεφές επιστημονικό πεδίο έχει εξελιχθεί σε αναλυτικό πεδίο, το οποίο περιλαμβάνει για τις ανάγκες της ανάλυσης δεδομένα, δηλαδή γραπτά κείμενα, σχέδια, φωτογραφίες και ψηφιακά δεδομένα. Εξαρχής είναι ευθύνη των αρχαιολόγων η διατήρηση των ευρημάτων αλλά και των αρχείων που τηρούν, ώστε να είναι μεν ακριβείς στη τελική τους δημοσίευση, αλλά παράλληλα να είναι βιώσιμα τα αποτελέσματα των ερευνών τους σε βάσεις δεδομένων, που διασφαλίζουν την επανεξέταση των συμπερασμάτων και ερμηνειών υπό το πρίσμα νέων δεδομένων που συλλέγονται από μελλοντικούς αρχαιολόγους.

Ο συνδυασμός της αρχαιολογικής θεωρίας και της ψηφιακής τεχνολογίας με την πρακτική γνώση των επαγγελματιών του χώρου περιγράφεται στο *Digital archaeology: bridging method and theory* των Thomas L. Evans και Patrick

¹⁵ Skeet 2014.

¹⁶ Asp.NET Webforms, διαθέσιμο online.

¹⁷ McDaniel 2011.

¹⁸ Sawyer McFarland 2012.

¹⁹ Libby 2015.

²⁰ JQuery-ui-1.12.1 διαθέσιμο online.

²¹ CKEditor4, διαθέσιμο online .

²² ImageResizer 3.4.3. διαθέσιμο online.

²³ Martin 2016.

Daly²⁴. Οι συγγραφείς συζητούν τους τρόπους με τους οποίους μπορεί να ενσωματωθεί η πληροφορική στην αρχαιολογία. Το βιβλίο ασχολείται με στοιχεία της αρχαιολογικής θεωρίας και αποκαλύπτει πώς οι υπολογιστές μπορούν να χρησιμοποιηθούν για την επανένταξη των θεωρητικών ζητημάτων στην εργασία πεδίου και την ανάλυση δεδομένων. Εκκινώντας από την ιστορία της ανάπτυξης της πληροφορικής στο πεδίο, οι συγγραφείς παρέχουν παραδείγματα για το πώς και γιατί έχουν εφαρμοστεί διαφορετικές τεχνολογίες στην αρχαιολογική θεωρία και μέθοδο. Η διερεύνηση των παραδειγμάτων περιλαμβάνει τα γεωγραφικά συστήματα πληροφοριών (GIS), τη μοντελοποίηση της εικονικής πραγματικότητας, τη δημοσίευση και αρχειοθέτηση του διαδικτύου και την ψηφιακή καταγραφή *in situ*. Εξετάζονται, επίσης τρόποι, με τους οποίους η τεχνολογία μπορεί πλέον να χρησιμοποιηθεί στο κλασικό μοντέλο ανασκαφής και πώς επηρεάζει η ψηφιακή τεχνολογία και οι εξελίξεις της τη μελέτη της αρχαιολογίας στο σύνολό της, από το σχεδιασμό έως την τελική δημοσίευση.

Η κατασκευή μιας βάσης δεδομένων και η διαχείριση της αρχαιολογικής ψηφιακής πληροφορίας εμπλέκουν τον τρόπο με τον οποίο οι αρχαιολόγοι αναζητούν και βρίσκουν πληροφορίες για την τεκμηρίωση του έργου τους. Οι συζητήσεις για τη συμμετοχή, την προσβασιμότητα, την προνομιακή θέση της ψηφιακής πληροφόρησης και την επακόλουθη εμφάνιση μιας ιδιαίτερης αλλά συχνά κακώς καθορισμένης ψηφιακής πολιτιστικής πληροφόρησης είναι πολλές. Παρά τις νέες τεχνολογίες, υπάρχουν προβλήματα και ανησυχίες που δεν μπορούν να απαντηθούν. Γιατί οι μηχανές αναζήτησης και άλλες τεχνολογίες πληροφοριών δεν παρέχουν απαντήσεις σε όλες τις πιθανές ερωτήσεις; Ή μήπως υπάρχει έλλειψη εκείνων των κατάλληλων δεξιοτήτων και ικανοτήτων για την αναζήτηση πληροφοριών; Οποσδήποτε, σε εύρος δεκαετίας υπάρχουν σημαντικές αλλαγές στον τρόπο με τον οποίο αναζητούνται οι πληροφορίες, πού, πότε και από ποιον. Η αλλαγή αυτή καθεαυτή δεν είναι κάτι νέο, αλλά υπάρχουν ιδιαιτερότητες που έχουν κάνει την αλλαγή διαφορετική και πλέον ριζοσπαστική από τις αλλαγές που συνέβαιναν εδώ και δεκαετίες.

Η τεχνολογία και ειδικά οι υποθέσεις για το πώς λειτουργεί η τεχνολογία για τον κοινωνικό ιστό έχουν αλλάξει, αλλάζοντας και την υφιστάμενη κουλτούρα της πληροφορίας. Την αλλαγή περιγράφει ο Isto Huvila στο έργο του

²⁴ Evans and Daly 2006.

*Information Services and Digital Literacy: In search of the boundaries of knowing*²⁵, εξετάζοντας παράλληλα και τα θεμέλια της θεωρίας της D.H. Sonnenwald για τον ορίζοντα πληροφόρησης²⁶. Η Sonnenwald διερευνά αφενός τις αλλαγές αλλά και τις δεξιότητες που απαιτούνται για να αντιμετωπιστεί το μεταβαλλόμενο τοπίο της πληροφορίας, αφετέρου συνδέει αυτό το τοπίο με το αρχαιολογικό έργο.

Όσον αφορά στο λογισμικό Digit, από τη διεθνή βιβλιογραφία αντλούνται δύο συλλογικά έργα στα οποία αναλύεται το αρχαιολογικό υπόβαθρο που οδήγησε στην κατασκευή της εφαρμογής. Το ένα είναι το *Classical Archaeology in Context* σε επιμέλεια Donald C. Haggis and Carla M. Antonaccio, όπου παρουσιάζεται η θεωρητική δομή της βάσης δεδομένων Digit²⁷. Το δεύτερο είναι το *KYMISSALA: Archaeology-Education- Sustainability*, σε επιμέλεια Manolis I. Stefanakis, όπου παρουσιάζεται η αρχαιολογική θεωρία στην ανασκαφή της Κυμισάλας, άμεσα συνδεδεμένη με την κατασκευή της εφαρμογής Digit²⁸.

4. Ημερολόγια ανασκαφής

Όσον αφορά στο ημερολόγιο ανασκαφής, ως καταγραφή της εργασίας πεδίου, κατά τη βιβλιογραφική έρευνα παρατηρήθηκε προβληματισμός ως προς την αποτελεσματικότητα των διαφόρων μορφών αρχαιολογικής καταγραφής. Για παράδειγμα, η ημερολογιακή καταγραφή γενικεύσεων ή ρομαντικών στοιχείων δε σχετίζονται άμεσα με τα ανασκαφικά δεδομένα. Ο εκάστοτε τύπος ημερολογίου είναι ως ένα σημείο προβληματικός εξαιτίας της εγγενούς δυνατότητάς του τόσο να αποκρύπτει όσο και να διευκρινίζει τα αποτελέσματα της αρχαιολογικής έρευνας την οποία καταγράφει.

Το αρχαιολογικό ημερολόγιο είναι κείμενο που περιλαμβάνει παρατηρήσεις, σκέψεις, δεδομένα και ενίοτε συναισθήματα. Είτε είναι κείμενα για προσωπική θεώρηση και χρήση ή για δημόσια πληροφόρηση και άσχετα από το συγγραφικό τους ύφος ανήκουν σε μια δεδομένη «στιγμή». Συλλαμβάνουν έναν ιδιαίτερο χρόνο και χώρο και αυτό τα κάνει αυθεντικά και αξιόπιστα, συνήθως,

²⁵ Huvila 2012.

²⁶ Sonnenwald 1999, 176–190.

²⁷ Manolis I. Stefanakis, Konstantinos Kalogeropoulos, Andreas Georgopoulos, and Chryssi Bourbou 2015.

²⁸ Manolis I. Stefanakis 2017.

αρχεία της ιδιαίτερης αρχαιολογικής πρακτικής και των πληροφοριών που αντλούνται σε δεδομένο χώρο και χρόνο. Η ημερολογιακή αρχαιολογική πληροφορία απαντάται σε αφηγηματική μορφή σε αρχαιοδιφικά κείμενα και στο έργο αρχαιολόγων εξερευνητών του 18ου και του 19ου αιώνα²⁹. Για τον Ian Hodder η μεταστροφή κατά τον 20ό αιώνα από την ατομική στην ιδρυματική έρευνα (πανεπιστήμια, οργανισμοί, ιδρύματα), ενθάρρυνε την επιδίωξη αντικειμενικών και αποσπασμένων προσεγγίσεων στην αρχαιολογική αφήγηση. Σε αυτή τη νέα αφήγηση ο ρόλος του «Εγώ» φαίνεται εξαιρετικά μειωμένος³⁰. Ωστόσο, υπό την επίδραση κυρίως της ανθρωπολογικής έρευνας και της μεταδιαδικαστικής θεώρησης της αρχαιολογίας, τις τελευταίες δεκαετίες το αρχαιολογικό ημερολόγιο μεταλλάχθηκε εκ νέου, εξαιτίας της έμφασης στην υποκειμενικότητα της αρχαιολογικής πρακτικής και τον συρμό των φαινομενολογικών προσεγγίσεων³¹.

Ο Hodder στο *The Archaeological Process: an introduction* και ο Tilley στο *A Phenomenology of Landscape: places, paths and monuments* περιγράφουν ότι οι ερμηνείες των μεθόδων της αρχαιολογίας πεδίου στο αρχαιολογικό ημερολόγιο περιέχουν συχνά χρονολογικά στοιχεία που περιγράφουν τη σειριακή φύση της εργασίας πεδίου και δείχνουν την υποκειμενικότητα της εμπειρίας όσων εργάζονται στο πεδίο³². Δείγμα της υποκειμενικότητας στην ημερολογιακή καταγραφή αποτελεί και η τήρηση ημερολογίων πεδίου από φοιτητές της αρχαιολογίας, που εργάζονται για την απόκτηση εμπειρίας στο πεδίο³³.

Σε όλα τα παραπάνω θα πρέπει να αναφερθούν δεκάδες άρθρα και συζητήσεις στον επιστημονικό τύπο που διερευνούν θεωρητικά και πρακτικά ζητήματα της αρχαιολογικής έρευνας και η διερεύνηση της ελληνικής γκριζας βιβλιογραφίας που απέδωσε λιγοστούς καρπούς στο θέμα της αρχαιολογικής θεωρίας και μεθόδου, πλην της διδακτορικής διατριβής *Ανασκαφική Μεθοδολογία και Σχεδιασμός Πληροφοριακού Συστήματος για τη Διαχείριση Αρχαιολογικών Τεκμηρίων*, που ασχολείται με μεθοδολογικά αρχαιολογικά ζητήματα και την

²⁹ Morris 1994, 27. Επίσης, Peltz & Myrone 1999, 5-9.

³⁰ Hodder 1989, 268-74. Επίσης, Joyce 2002, 52-55.

³¹ Βλ. Chadwick 2003, 97-117 και Hodder 1999, 121-123. Επίσης Olsen 2006, 144-150 και Tilley 1994, 32-36.

³² Hamilton and Whitehouse 2006, 31-71.

³³ Hamilakis 1999, 60-79.

ψηφιοποίηση της ανασκαφικής διαδικασίας, αλλά υπό την οπτική γωνία των γεωπληροφορικών συστημάτων³⁴.

³⁴ Κατσιάνης 2009.

Κεφάλαιο 1. Εισαγωγή στην ψηφιοποίηση αρχαιολογικών δεδομένων

1.1. Πολιτισμική τεχνολογία και αρχαιολογία πεδίου

Η πολιτισμική τεχνολογία και η αρχαιολογία πεδίου είναι δύο έννοιες στενά συνυφασμένες την τελευταία 40ετία. Η πειραματική σύνδεση της αρχαιολογικής πρακτικής με την ψηφιακή καταγραφή είναι ουσιαστικά τμήμα της εισβολής της τεχνολογίας των υπολογιστικών συστημάτων σε διάφορους πολιτιστικούς τομείς. Ένα μεγάλο πρόβλημα εν τη γενέσει αυτής της νέας πρακτικής υπήρξε η οριοθέτηση της ταξινομίας και της ακρίβειας της ψηφιακής πληροφορίας. Θέμα μείζον γύρω από το οποίο αναπτύχθηκαν διαφορετικά και συνήθως συμπληρωματικά γνωστικά πεδία. Είναι γεγονός πως έχουν γίνει βήματα στον τομέα της εκπαίδευσης σε υπολογιστικά συστήματα, αλλά προς το παρόν η πολιτισμική τεχνολογία διαμορφώνεται *intra muris* από έναν κλειστό επιστημονικό κύκλο, που φιλοδοξεί στις επερχόμενες δεκαετίες να εκπαιδεύσει ένα σημαντικό αριθμό επιστημόνων –για τη χώρα μας– προς αυτή την κατεύθυνση.

Επί του παρόντος οι προσπάθειες –ιδιαίτερα σε ό,τι αφορά στην αρχαιολογία– είναι ατομικές ή ομαδικά κλειστές στα πλαίσια του επιστημονικού πειράματος. Η προσπάθειά μας, εν προκειμένω, είναι να προσπαθήσουμε να καταγράψουμε και να συγκρίνουμε υφιστάμενους σχεδιασμούς και εν τέλει να πειραματιστούμε δια ζώσης με μία βάση δεδομένων, που μπορεί να εφαρμοστεί με τις κατάλληλες τροποποιήσεις σε ποικίλα πεδία της πολιτισμικής τεχνολογίας. Το ενδιαφέρον μας στρέφεται αρχικά στην αρχαιολογία πεδίου και στους καλύτερους δυνατούς τρόπους με τους οποίους μπορεί μια τέτοια βάση δεδομένων να υπηρετήσει τη διαδικασία καταγραφής, αποθήκευσης και επαναχρησιμοποίησης του αρχαιολογικού αρχείου για ερευνητικούς, ερμηνευτικούς λόγους ή λόγους δημοσίευσης και παρουσίασης στο ευρύ κοινό.

Σε πληθώρα ερευνών εισάγονται και αναπτύσσονται οι έννοιες και οι μέθοδοι που χρησιμοποιούν οι αρχαιολόγοι στο πεδίο και ειδικότερα στην ανασκαφή. Το εννοιολογικό πλαίσιο της συγκεκριμένης έρευνας αφορά στο

σύνολο των θεωρητικών τάσεων στο επίπεδο της ανάλυσης και καταγραφής των αποτελεσμάτων που συνδέονται με την ανασκαφή και την αρχαιολογική έρευνα. Η έμφαση τοποθετείται στην ψηφιοποίηση των δεδομένων του αρχαιολογικού αρχείου, και στην επαναλαμβανόμενη χρησιμοποίηση των μεταβλητών του για την εξαγωγή συμπερασμάτων. Είναι σημαντικό να επισημανθεί ότι η εργασία πεδίου και η ανάλυση συνδέονται αξεδιάλυτα, αλλά η πρώτη συνήθως απολαμβάνει μεγαλύτερης προσοχής από τη δεύτερη. Η ψηφιοποίηση του αρχαιολογικού αρχείου, εφόσον λαμβάνονται υπ' όψιν ζητήματα καλής πρακτικής και ταξινόμησης, είναι δυνατόν να αποκαταστήσει σε μεγάλο βαθμό την ανισορροπία.

1.2. Ιστορικό της έρευνας

Οι υπολογιστές χρησιμοποιούνται στην αρχαιολογία επί σειρά ετών. Από την απλή χρήση επεξεργαστών κειμένων έγιναν σε παγκόσμιο επίπεδο εργαλείο για τους αρχαιολόγους. Μέσω των υπολογιστών αυτοματοποιήθηκαν αρκετές εργασίες, ανάμεσα στις οποίες είναι πλέον κοινός τόπος η στατιστική, η κατασκευή καταλόγων και η κατασκευή σχεδίων σε περιπτώσεις επιφανειακής επιθεώρησης.

Σύνθετες χρήσεις των υπολογιστών περιλαμβάνουν την αποθήκευση όλης της πληροφορίας για την ίδια την ανασκαφή, δηλαδή πληροφορίες για τα σκάμματα, τη στρωματογραφία, το προσωπικό, τις φωτογραφίες, τα σχέδια, την κατασκευή τρισδιάστατων μοντέλων των αρχαιολογικών θέσεων ή κατασκευών, πληροφορίες για την τοπογραφία, το ετήσιο επίπεδο βροχόπτωσης, τη βλάστηση, τη θερμοκρασία και τους υδάτινους πόρους. Άλλες μείζονες χρήσεις περιλαμβάνουν:

A. συστήματα διαχείρισης βάσεων δεδομένων, ερευνητικές και επίσημες βάσεις δεδομένων για αρχεία μνημείων.

B. ποσοτική ανάλυση με χρήση στατιστικής.

Γ. συστήματα γεωγραφικών πληροφοριών (GIS), σύνδεση δεδομένων με χάρτες.

Δ. γραφική επίδειξη της ανασκαφής με τη χρήση τοπογραφικών και γεωφυσικών δεδομένων.

Ε. επιτόπια καταγραφή της ανασκαφής και ανάλυση δεδομένων μετά την ανασκαφή.

Στ. γραφικά, ηλεκτρονικές ανακατασκευές κτηρίων και προσομοιώσεις του παρελθόντος

Ζ. μάθηση μέσω υπολογιστή.

Η. πρόσβαση διαδικτύου για ηλεκτρονική αλληλογραφία, πηγές, ηλεκτρονικά αρχεία και λογισμικό.

Οι υπολογιστές φαίνεται πως χρησιμοποιήθηκαν για πρώτη φορά από τους αρχαιολόγους στα τέλη της δεκαετίας του 1950, κυρίως για χρονοβόρες στατιστικές μετρήσεις³⁵. Η εξέλιξη της τεχνολογίας είχε ως αποτέλεσμα να μειωθούν οι δυσκολίες χρήσης από μη ειδικούς και να γίνουν σημαντικότερα τα οφέλη από τη χρήση των υπολογιστών. Ωστόσο, μόνο μετά από την εμφάνιση και την εμπορική χρήση του αποκαλούμενου προσωπικού υπολογιστή, που εισήγαγε η IBM το 1981³⁶, στάθηκε δυνατόν να χρησιμοποιήσουν οι αρχαιολόγοι την υπολογιστική τεχνολογία χωρίς τη βοήθεια ειδικών.

Το 1967 ο George Cowgill διατύπωσε την άποψη ότι η πρωιμότερη χρήση υπολογιστών στην ευρωπαϊκή αρχαιολογία έγινε από τους Peter Ihm και Jean-Claude Gardin το 1958/1959 και στις ΗΠΑ από τον James Deetz το 1960³⁷. Ήταν ο Roberto Busa εκείνος που χρησιμοποίησε υπολογιστή για ανθρωπιστική επιστήμη, σε ένα έργο που σήμερα θεωρείται μνημειώδες, τη δημιουργία ενός καταλόγου λέξεων (*index verborum*) όλων των λέξεων που απαντώνται σε έργα του Θωμά του Ακινάτη και σχετικών συγγραφέων, δηλαδή 11 περίπου εκατομμυρίων λέξεων μεσαιωνικής Λατινικής³⁸. Από τότε η δραστηριότητα στη χρήση των υπολογιστών στην αρχαιολογία αυξήθηκε ουσιαστικά σε όγκο και ποιότητα, ενώ έκανε την πρώτη εμφάνισή του το ετήσιο συνέδριο «Εφαρμογές Υπολογιστών στην Αρχαιολογία» (Computer Applications in Archaeology) - CAA το 1973.

³⁵ Απαιτούσαν ιδιαίτερη υπομονή, αφού η εισαγωγή και η ανάλυση των δεδομένων γινόταν με κωδικοποιημένες κάρτες που εισήγαγαν τα δεδομένα και άλλες κάρτες που εισήγαγαν οδηγίες στον υπολογιστή. Το παραμικρό λάθος θα μπορούσε να αποβεί μοιραίο για τη διαδικασία και κάθε διαδικασία κωδικοποιείτο ανεξάρτητα για τον υπολογιστή.

³⁶ Ουσιαστικά πρόκειται για τον IBM 5100 Portable Computer, σύντομο ιστορικό του οποίου βρίσκεται στο διαδικτυακό αρχείο της IBM. Βλ. *IBM Archives, The birth of the IBM PC*, διαθέσιμο online.

³⁷ Cowgill 1967, 17.

³⁸ Winter 1999, 3-20.

Έκτοτε, οι υπολογιστές έπαιξαν σημαντικό ρόλο στην αρχαιολογική έρευνα επί σειρά ετών και η χρήση τους επεκτάθηκε ουσιαστικά πέρα από τη στατιστική ανάλυση και τις βάσεις δεδομένων στις αρχές της δεκαετίας του '80. Παρόλο που η επέκταση του ρόλου τους πρόσφερε πολλά πλεονεκτήματα, απαίτησε και σημαντικές αλλαγές σε βασικές μεθόδους οργάνωσης των αρχαιολογικών εγχειρημάτων³⁹. Η διαρκής εξέλιξη των υπολογιστών έφερε νέα προγράμματα, που σχεδόν εξάλειψαν την ανάγκη συγκεκριμένης κωδικοποίησης για κάθε διακριτό αρχαιολογικό εγχείρημα. Δημιουργήθηκαν συστήματα διαχείρισης μεγάλων ποσοτήτων πληροφοριών, όπως τα συστήματα διαχείρισης βάσεων δεδομένων (DBMS) ανοικτού κώδικα ή εμπορικά κλειστού κώδικα, γενόμενα σταδιακά ευκολότερα στη χρήση. Ταυτόχρονα αναπτύχθηκαν στατιστικά πακέτα για την εκτέλεση κάθε σχετικής λειτουργίας με το αρχαιολογικό έργο.

Στο μεταξύ χρησιμοποιήθηκαν και τα προγράμματα σχεδιασμού (CAD) για τη διαμόρφωση τρισδιάστατων μοντέλων και τον σχεδιασμό σύνθετων αντικειμένων, όπως και συστήματα γεωγραφικών πληροφοριών (GIS) για την ανάλυση και την αποτελεσματική διαχείριση γεωγραφικών δεδομένων⁴⁰. Οι πρώτοι προσωπικοί υπολογιστές δε διέθεταν την κατάλληλη υπολογιστική δύναμη να χειριστούν αυτά τα σύνθετα προγράμματα. Οι σύγχρονοι υπολογιστές, ωστόσο, και όλα τα υφιστάμενα προγράμματα ήταν δυνατόν να χρησιμοποιηθούν από αρχαιολόγους με μικρή σχετικά εκπαίδευση στους υπολογιστές.

Οι εξελίξεις στη χρήση των υπολογιστών ήταν εν γένει το αποτέλεσμα εμπορικής δραστηριότητας. Με λιγοστές εξαιρέσεις οι αρχαιολόγοι δεν έγραψαν προγράμματα για τον εαυτό τους. Αντίθετα, υιοθέτησαν και προσάρμοσαν εμπορικά προϊόντα για δική τους χρήση⁴¹. Οι υπολογιστές και τα προγράμματα που έχουν διαμορφωθεί για αρχαιολογική χρήση είναι πλέον τόσο εξελιγμένα,

³⁹ Dibble and McPherron 1988, 431-440.

⁴⁰ Τα πρώτα συστήματα έκαναν εκτεταμένη χρήση αεροφωτογραφιών και δορυφορικών φωτογραφιών, για τη λήψη πληροφοριών σχετικών με τις γεωγραφικές, τοπογραφικές και κλιματικές συνθήκες.

⁴¹ Εξαίρεση αποτελεί το σύστημα Ibycus του David Packard, ο οποίος σχεδίασε υπολογιστή και προγράμματα ειδικά για ελληνικά κείμενα. Αν και ήταν πολύ εξειδικευμένα, χρησιμοποιήθηκαν με επιτυχία από πολλά επιστημονικά ιδρύματα. Εντέλει σχεδιάστηκαν άλλα προγράμματα, βασισμένα σε κοινές εμπορικές εκδόσεις υπολογιστών και το Ibycus τερματίστηκε. Βλ. σχετικά Schreibman et al 2008, 48.

ώστε ο σημερινός αρχαιολόγος μπορεί να χρησιμοποιήσει έναν υπολογιστή για την παραγωγή τρισδιάστατων μοντέλων, συλλέγοντας τοπογραφικές πληροφορίες και δεδομένα επιφανειακών επιθεωρήσεων της περιβάλλουσας περιοχής, αποθηκεύοντας όλες τις πληροφορίες της ανασκαφής και εκτελώντας δυναμικά οποιαδήποτε ανάλυση επιτρέπουν οι πληροφορίες που συνέλεξε. Λίγες αρχαιολογικές θέσεις χρησιμοποιούν τους υπολογιστές τόσο εκτεταμένα, αλλά πλέον η δυνατότητα υφίσταται.

Η αυξανόμενη διασύνδεση με τους υπολογιστές συνέπεσε με την επινόηση και τη χρήση σημαντικών αρχαιολογικών εργαλείων, όπως η ραδιοχρονολόγηση C^{14} , η παλυνολογία και η δενδροχρονολόγηση. Στην πραγματικότητα η ποσότητα των δεδομένων που χρησιμοποιούν αυτά τα εργαλεία απαιτούσε μεγάλη υπολογιστική ισχύ. Αυτό δεν ήταν πρόβλημα, δεδομένης της εξέλιξης των μικροεπεξεργαστών. Το αποτέλεσμα, όμως, όλων αυτών των δεδομένων μπορούσε να προκαλέσει σύγχυση στον μέσο αρχαιολόγο χειριστή. Συνεπώς, απαιτήθηκαν νέοι τρόποι χειρισμού των δεδομένων. Για παράδειγμα, τα τρισδιάστατα μοντέλα έπρεπε να συνδεθούν με δεδομένα για τα τέχνηρα από τις αρχαιολογικές θέσεις και τα μοντέλα των αρχαιολογικών θέσεων να συσχετιστούν με γεωγραφικές και τοπογραφικές πληροφορίες από την περιβάλλουσα ύπαιθρο ή το αντίστοιχο αστικό περιβάλλον. Οι τεχνολογίες διασύνδεσης και ταξινόμησης δεδομένων σε αυτό το πλαίσιο είναι διαθέσιμες, αλλά δεν χρησιμοποιούνται ευρέως.

Συνοψίζοντας, θα μπορούσαμε να πούμε ότι σήμερα ο ρόλος του υπολογιστή στην αρχαιολογία υπήρξε και συνεχίζει να είναι ευρύς και πολυεπίπεδος. Από το 1980, οπότε χρησιμοποιήθηκαν οι μικροϋπολογιστές, χρησιμοποιούνται σε κάθε φάση της ανασκαφής και του αρχαιολογικού έργου γενικότερα για να εκτελέσουν τρεις μείζονες λειτουργίες: την απόκτηση δεδομένων, την ανάλυση και την παρουσίαση τους.

Η ηλεκτρονική λήψη δεδομένων περιλαμβάνει τη χρήση λογισμικού και υλικού για τη μέτρηση αντικειμένων, συστήματα απεικόνισης βίντεο για τη μέτρηση της επιφάνειας και των μορφών, και άλλα μέσα που μπορούν να συνδεθούν με δεδομένα πεδίου σε μορφή αναγνώσιμη από υπολογιστική μηχανή για μετέπειτα ανάλυση. Σημαντική εξέλιξη στον τομέα αυτό, που δημοσιεύτηκε για πρώτη φορά από τον Dibble, ήταν η προσαρμογή ενός τοπογραφικού

συστήματος που ονομάζεται συνολικός σταθμός που περιελάμβανε και ηλεκτρονικό θεοδόλιχο για τον καθορισμό γωνιών σε συνδυασμό με ένα ηλεκτρονικό μετρητή απόστασης για την τρισδιάστατη καταγραφή θέσεων τεχνέργων σε μία αρχαιολογική θέση⁴². Η χρήση των συνολικών σταθμών έγινε σε διαφορετικές συνθήκες, ιδιαίτερα αντιθετικές στο πεδίο όπως αναφέρεται⁴³. Ο συνολικός σταθμός παίρνει πολύ ακριβείς μετρήσεις αποστάσεων και η σχετική κατεύθυνση καταγράφεται αντικατοπτρίζοντας μια ακτίνα λέιζερ από ένα μικρό πρίσμα, που τοποθετείται στο μακρινό σημείο που πρέπει να μετρηθεί. Συνολικοί σταθμοί ήταν ήδη σε χρήση για μεγάλης κλίμακας τοπογραφική αποτύπωση των χώρων. «Η καινοτομία στη συγκεκριμένη περίπτωση», λέει ο McPherron, «ήταν να πάρει κανείς κάτι που χρησιμοποιείτο για τη μέτρηση χιλιομέτρων και να το εφαρμόσει για τη μέτρηση εκατοστών»⁴⁴.

Η ανάλυση δεδομένων εξαρτάται από την κατασκευή και τη διαχείριση σύνθετων βάσεων δεδομένων με αρχαιολογικές πληροφορίες. Τα δεδομένα που αντλούνται από τα διαφορετικά ανασκαφικά στρώματα, συντίθενται προκειμένου να αντληθούν συμπεράσματα ειδικά και γενικά. Για παράδειγμα, οι πληροφορίες για τις θέσεις τεχνέργων μπορεί να συνδυαστούν σε μία σχεσιακή βάση δεδομένων, αποκαλύπτοντας πρότυπα διασποράς παρόμοιων τεχνέργων. Η δυνατότητα εισαγωγής και διαχείρισης περισσότερων κατηγοριών πληροφορίας επιτρέπει την ανάπτυξη και δοκιμή πολύπλοκων υποθέσεων. Οι φορητοί υπολογιστές με αυξημένη υπολογιστική ισχύ μπορούν να χρησιμοποιηθούν στο πεδίο για την εισαγωγή δεδομένων *in situ* και η ανάλυσή τους στο εργαστήριο να καθοδηγεί τους αρχαιολόγους για την επόμενη ημέρα της ανασκαφής.

Η παρουσίαση των δεδομένων *per se* είναι δυνατόν να παρέχει σημαντική βοήθεια στην απαιτούμενη ανάλυση με τη μορφή γραφημάτων. Προγράμματα γραφικών στρωματογραφικής ανάλυσης με ηλεκτρονικές μετρήσεις στο πεδίο της ανασκαφής είναι δυνατόν να παράγουν τρισδιάστατα μοντέλα, κατάλληλα

⁴² Οι Dibble και McPherron ανέπτυξαν αρκετές freeware εφαρμογές για Windows OS προς διευκόλυνση του αρχαιολογικού έργου στο πεδίο, ανάμεσα στις οποίες ξεχωρίζουν το NewPlot (αρχαιολογικά προσανατολισμένο σύστημα GIS), το EDM-CE και EDM Windows (προγράμματα απόκτησης δεδομένων για χρήση με Total Stations για Windows Mobile και Windows OS αντίστοιχα) και το E4 (πρόγραμμα συλλογής δεδομένων για ανάλυση τεχνέργων). Το σύστημά τους υιοθετήθηκε από πολλούς αρχαιολόγους σε άλλες ανασκαφές στην Β. Αμερική, την Ευρώπη και την Αφρική και έχουν γίνει εκτεταμένες δημοσιεύσεις πάνω στη μέθοδό τους.

⁴³ McPherron 2002, 28-32.

⁴⁴ McPherron 2005, 1003-1014.

διαμορφωμένα χρωματικά, ώστε να αποδίδουν ιδιαίτερα χαρακτηριστικά. Τα πρότυπα διασποράς σε τέτοιες περιπτώσεις στρωματογραφικής ανάλυσης καθίστανται εμφανή με άμεσο τρόπο.

Σε θεωρητικό επίπεδο οι Bimber και Chang Marcos Llobera επιχειρηματολόγησαν για τη δημιουργία μίας νέας θεωρητικής εστίασης στη αρχαιολογία, την Αρχαιολογική Επιστήμη Πληροφορικής, «σχετική με τη δημιουργία, αναπαράσταση και διαχείριση της αρχαιολογικής πληροφορίας στο πλαίσιο των συστημάτων πληροφορικής»⁴⁵.

Το ερευνητικό εργαστήριο του πανεπιστημίου της Ουάσινγκτον για την ψηφιακή αρχαιολογία δημιουργήθηκε για την προώθηση και την ανάπτυξη της Αρχαιολογικής Επιστήμης της Πληροφορικής. Είναι «αναδυόμενος τομέας, που προκύπτει από τη διασύνδεση της αρχαιολογίας με την επιστήμη των υπολογιστών και της πληροφορικής, όπως επίσης με τα μαθηματικά»⁴⁶ και θεωρείται εν γένει νέο γνωστικό πεδίο.

Η αρχαιολογική πληροφορική δεν είναι κάτι πλήρες, αλλά μια διαδικασία εν εξελίξει. Τα άτομα ή οι οργανισμοί που τη χρησιμοποιούν, δε φθάνουν στην ολοκληρωτική εφαρμογή της πληροφορικής στην αρχαιολογία, καθώς επεξεργάζονται διαρκώς νέες χρήσεις των υπολογιστών και των ιδιαίτερων εφαρμογών, των σχετικών με την αρχαιολογία, τρέχοντας κυριολεκτικά πίσω από τις εξελίξεις της τεχνολογίας υπολογιστών. Ωστόσο, υπάρχει ένα σταθερό θεμέλιο και η απαραίτητη γνώση για τη διαμόρφωση και χρήση βάσεων δεδομένων, σταθεροποιημένο πλαίσιο λειτουργίας εξυπηρετητών (servers) και σύγχρονες αντικειμενοστρεφείς γλώσσες προγραμματισμού που μπορούν να παράγουν το περίγραμμα λειτουργίας της αρχαιολογικής πληροφορικής, έτσι όπως προκύπτει από την αρχαιολογική θεωρία και τις υφιστάμενες μεθόδους ανασκαφής.

Οπωσδήποτε, είναι αναγκαία η ιστορική προσέγγιση σε διαφορετικές περιοχές της χρήσης υπολογιστών, καθώς οι τεχνολογικές εξελίξεις ανατρέπουν συχνά κάθε προσπάθεια λεπτομερούς καταγραφής συγκεκριμένων και εξειδικευμένων εφαρμογών. Σε άλλες περιπτώσεις, πειραματικές εφαρμογές του παρελθόντος όπως φαίνεται από παρελθούσες δημοσιεύσεις των συνεδρίων της

⁴⁵ Llobera 2011, 218.

⁴⁶ DigAR Lab. (2011). «Welcome to the DigAR Lab!», διαθέσιμο online.

CAA (Computer Applications in Archaeology) αποτελούν πλέον συνήθη πρακτική για πολλούς αρχαιολόγους. Για παράδειγμα, η καταγραφή της ανασκαφής και τα συστήματα διαχείρισης της πολιτιστικής κληρονομιάς είναι δύο περιοχές στις οποίες οι περιορισμοί και το δυναμικό των υπολογιστών, οι ανάγκες και οι στόχοι των προγραμματιστών και των απλών χρηστών κατέληξαν σε συζήτηση που διαρκεί ήδη πολλά χρόνια⁴⁷. Ωστόσο, η χρήση των ηλεκτρονικών υπολογιστών στην αρχαιολογία δεν είναι εξειδίκευση ούτε θεωρητική σχολή σκέψης, αλλά μια προσέγγιση, ένας τρόπος χρήσης των υπολογιστών με τον καλύτερο δυνατό τρόπο, κατανοώντας τις δυνατότητες και τις αδυναμίες και την πληροφορική τεχνολογία ως σύνολο.

Υπάρχουν διανοητικές και τεχνολογικές επαναστάσεις, που ενίοτε συμβαίνουν ταυτόχρονα. Ο Thomas Kuhn θεωρεί ότι οι ανακαλύψεις οδηγούν ή συνεισφέρουν σε αλλαγή παραδείγματος στην έρευνα και την εφαρμογή⁴⁸. Η τεχνολογική επανάσταση, στην οποία συμμετέχει η αρχαιολογία, διαθέτει αρκετές όψεις που επηρεάζουν το παρόν παράδειγμα. Αρχικά, οι ψηφιακές τεχνολογίες προσφέρουν έναν τρόπο απεικόνισης του πραγματικού κόσμου. Κατόπιν, επιτρέπουν τις μετρήσεις, τις στατιστικές αναλύσεις, τη διαχείριση και την αξιολόγηση των μετρήσεων με ποικίλους αναλυτικούς τρόπους. Επίσης, επιτρέπουν τη σχεδίαση ικανοποιητικών μοντέλων και προσομοιώνουν τις λειτουργίες του πραγματικού κόσμου με στόχο την κατανόηση περίπλοκων και συσχετισμένων λειτουργιών επιτελούμενων από τους ανθρώπους στο περιβάλλον τους. Έτσι γίνεται δυνατή η δημιουργία εικονικών κόσμων. Τέλος, οι ψηφιακές τεχνολογίες επιτρέπουν τη μεταφορά όλων αυτών των μοντέλων και των εικονικών κόσμων σε ένα διαρκώς αυξανόμενο παγκόσμιο κοινό⁴⁹.

Έχει ήδη διαμορφωθεί μια παγκόσμια ψηφιακή κοινότητα, ομάδες της οποίας αποτελούνται από αρχαιολόγους και επιστήμονες άλλων ειδικοτήτων, ανάμεσα στους οποίους περιλαμβάνονται ανθρωπολόγοι, γεωλόγοι, αρχιτέκτονες, τοπογράφοι παλυνολόγοι, γεωγράφοι, οικονομολόγοι και άλλοι, όπως επίσης το πλατύ, γενικό κοινό. Ο ρόλος της ταυτότητας σε μια τέτοια ψηφιακή κοινότητα, ακόμα και σε υποδιαιρέσεις της, μειώνεται σημαντικά ή αλλάζει, παρά τις σημαντικές προσπάθειες διαμόρφωσης μιας ψηφιακής

⁴⁷ Lock 2003, xiii.

⁴⁸ Kuhn 1970, 66.

⁴⁹ Zubrow 2006, 12.

απαραβίαστης ταυτότητας. Τα όρια ανάμεσα στους επαγγελματίες και τους ερασιτέχνες αρχαιολόγους, ακόμη και το πλατύ κοινό θολώνουν με αυξανόμενο ρυθμό.

Δεν μπορεί κανείς να υποθέσει το υπόβαθρο και τις γνώσεις κάποιου αρχαιολόγου ή μέλους που στέλνει μήνυμα ηλεκτρονικού ταχυδρομείου ή είναι υπεύθυνο για κάποιον αρχαιολογικό ιστότοπο ή ηλεκτρονικό περιοδικό. Δεδομένου ότι η ψηφιακή κοινότητα είναι ακόμη ανοικτή σε ευρύ κοινό, δεν είναι δυνατόν να ελεγχθεί η πρόσβαση σε αρχαιολογικές ιστοσελίδες ή ηλεκτρονικά περιοδικά. Το περιεχόμενο, όπως και οι ακριβείς τοποθεσίες αρχαιολογικών χώρων ή αντικειμένων, χρειάζονται μεγαλύτερη προστασία στον ψηφιακό κόσμο. Στον κυβερνοχώρο η ευκολία πώλησης νόμιμων ή παράνομων αρχαιοτήτων ενισχύεται μέσω συναλλαγών, στις οποίες οι αρχαιότητες πωλούνται μεταξύ ατόμων που ποτέ δε χρειάζεται να αποκαλύψουν το πραγματικό τους όνομα ή ακόμα και την τοποθεσία τους⁵⁰. Την ίδια στιγμή, γίνεται προσπάθεια παρεμπόδισης της παράνομης διακίνησης αρχαιοτήτων με νέες ψηφιακές μεθόδους για τον εντοπισμό και την παρακολούθηση των κλεμμένων τεχνέργων⁵¹.

Οι ψηφιακές κοινότητες αλλάζουν με ταχύτατους ρυθμούς. Υφίσταται διαδικασία μετάλλαξης στον κυβερνοχώρο που σημαίνει ότι η τεχνολογία της πληροφορίας αλλάζει πολύ γρήγορα και δεν μπορεί να συνδεθεί με καμία ιδιαίτερη τεχνολογία ή μέσο. Με τον ίδιο τρόπο που εξαφανίστηκε ουσιαστικά το μελάνι και οι δισκέτες, με τα CD και τα DVD να ακολουθούν αμέσως μετά, δημιουργούνται νέες ψηφιακές και ηλεκτρονικές τεχνολογίες, οι οποίες με τη σειρά τους αναπλάθουν την ψηφιακή κοινότητα με πολλούς τρόπους. Το Linux αργά αλλά σταθερά κερδίζει μεγαλύτερο μερίδιο στην αγορά⁵², τα ανοιχτά συστήματα αντικαθιστούν τα κλειστά και τα γραφικά αντικαθιστούν το κείμενο. Ο ίδιος ο υπολογιστής ως εργαλείο αλλάζει διαρκώς. Ωστόσο, στην προκειμένη περίπτωση δεν ενδιαφέρει η τροχιά της τεχνολογίας και οι πιθανές εξελίξεις της, αλλά ο πολυδιάστατος χώρος στον οποίο κινούνται οι αρχαιολόγοι. Ορισμένοι εργάζονται στον κυβερνοχώρο, άλλοι είναι μελετητές της ψηφιακής πληροφορίας, άλλοι διαμαρτύρονται για την εγκατάλειψη των στυλό και του

⁵⁰ Anglim-Kreder and Nintrup 2014, 143.

⁵¹ Zubrow 2006, 13.

⁵² Operating System Market Share, στο *NET MARKETSHARE*, διαθέσιμο online.

χαρτιού. Είναι λογικό ότι κάθε αρχαιολόγος επιλέγει τον χώρο στον οποίο νιώθει και εργάζεται άνετα στην κλίμακα της καινοτομίας που διαμορφώνεται με ένα άκρο τον κυβερνοχώρο και άλλο άκρο την παραδοσιακή τεχνολογία.

Ωστόσο, η νέα τεχνολογία όχι μόνο καταργεί την παλαιότερη, αλλά γίνεται το νέο περιβάλλον. Σε αυτή την περίπτωση δε σχετίζεται απλώς με τη μεθοδολογία, αλλά είναι καθοριστική για ορισμένες πτυχές της θεωρίας. Για ορισμένους αρχαιολόγους η τεχνολογία καθίσταται όψη ενός «τεχνολογικού ντετερμινισμού» και ο αρχαιολόγοι της ψηφιακής κοινότητας συμμετέχουν σε ένα νέο σύστημα πεποιθήσεων. Η διασύνδεση με αυτό το νέο σύστημα πεποιθήσεων αποκτά τεχνολογικό νόημα και είναι δυνατόν να επεκταθεί σε ντετερμινισμό, ιδιαίτερα στην περίπτωση της λογικής του αρχαιολογικού αρχείου⁵³. Οι διαδικασίες γίνονται μηχανικές, αλλά όλος αυτός ο μηχανισμός καθίσταται αυτοαναφορικός. Από τη φύση της, η ψηφιακή αρχαιολογία εξελίσσεται μέσω της αυξανόμενης χρήσης της ψηφιακής τεχνολογίας. Δημιουργώντας και αναδημιουργώντας την τεχνολογία, καθορίζουμε το μέλλον και βελτιώνουμε τις λύσεις στα προβλήματα που μάς ενδιαφέρουν. Με άλλα λόγια, οι αρχαιολόγοι της ψηφιακής κοινότητας δημιουργούν προβλήματα που απαιτούν ψηφιακές λύσεις.

Μελετώντας την εικόνα της ιστορίας της αρχαιολογίας και την ιστορία της ψηφιακής καινοτομίας στην αρχαιολογία, ορισμένες τάσεις είναι σαφείς. Η μία είναι ότι οι υπολογιστές αναπτύσσουν ορισμένες πτυχές της αρχαιολογίας, αλλά όχι όλες. Επίσης, υφίστανται συσχετισμοί μεταξύ των εξελίξεων στην αρχαιολογική θεωρία και των διαφόρων χρήσεων των υπολογιστών στην αρχαιολογία. Σε κάθε περίπτωση η εισβολή της πολιτισμικής τεχνολογίας ακόμα και στο επίπεδο διαμόρφωσης της αρχαιολογικής θεωρίας δεν είναι δυνατόν να αγνοηθεί, όχι μόνο στο επίπεδο της αρχαιολογικής, αλλά εν γένει της πολιτισμικής έρευνας.

1.3. Κατηγορίες αρχαιολογικής δραστηριότητας

Η πολιτισμική τεχνολογία στο σύνολό της είναι στενά συνδεδεμένη με ό,τι ορίζουμε αρχαιολογικό έργο. Ο Jean-Claude Gardin (1980) διακρίνει μεταξύ των

⁵³ Burmeister 2017, 23.

διάφορων κατηγοριών αρχαιολογικής δραστηριότητας και της παραγωγής διανοητικού έργου⁵⁴. Αυτές οι διαδικασίες συνδέονται για να διαμορφώσουν μια αλυσίδα από τη βασική απόκτηση των δεδομένων, μέσω της κατασκευής προτάσεων ή θεωριών, έως την τελική δημοσίευση. Ο τελευταίος κρίκος της αλυσίδας αναπόφευκτα οδηγεί στην απόκτηση νέων στοιχείων. Αυτό συμβαίνει επειδή η συλλογή στοιχείων δεν είναι ποτέ τυχαία ή «αντικειμενική», αλλά κατευθύνεται πάντα από τις θεωρίες που υιοθετούμε, την πρότερη εμπειρία μας και φυσικά τις προκαταλήψεις που διεισδύουν με τον ένα ή τον άλλο τρόπο στον αυστηρά επιστημονικό προσανατολισμό. Υπό αυτή την οπτική γωνία, η επιστροφή στην απόκτηση νέων στοιχείων και η περαιτέρω τεκμηρίωση της εκάστοτε προτεινόμενης υπόθεσης είναι μια διαδικασία αναπόφευκτη, αλλά χρήσιμη⁵⁵.

Για τον Gardin οι τρεις βασικές διαδικασίες σε αυτή την αλυσίδα είναι η απόκτηση στοιχείων, η ανάλυση (από τα αρχαιολογικά ευρήματα έως τη δημοσίευση) και οι στρατηγικές παρατήρησης. Κατόπιν υποδιαιρεί κάθε μια από αυτές τις διαδικασίες, προκειμένου να εξετάσει τις λεπτομέρειες του αρχαιολογικού έργου. Οι στρατηγικές παρατήρησης σχετίζονται με το πώς αντιλαμβανόμαστε τα φαινόμενα και πώς επιλέγουμε τα φαινόμενα, ή «δεδομένα», για να τα παρατηρήσουμε. Ο ίδιος προτείνει ότι η αρχαιολογική ανάλυση περιλαμβάνει βασικές δραστηριότητες όπως η καταχώριση και περιγραφή αρχαιολογικών δεδομένων, π.χ. τέχνηρα.

Η έρευνα, ωστόσο, δεν ικανοποιείται με απλούς καταλόγους. Εγγενώς απαιτεί δραστηριότητες, στις οποίες εμπλέκεται η διευθέτηση της αρχαιολογικής μαρτυρίας σε κάποιο είδος δομής: ομάδες, κατηγοριοποιήσεις ή τυπολογίες⁵⁶. Οι συγκεκριμένες δομές θεωρούνται συνήθως ότι απευθύνονται μόνον σε κινητά ευρήματα. Στην πραγματικότητα εφαρμόζονται στο ευρύτερο πλαίσιο μιας αρχαιολογικής θέσης, σε χρονολογίες, σε οικοδεδομένα, σε οικονομικά συστήματα ή τους ίδιους τους αρχαιολογικούς πολιτισμούς, σε ό,τι εν γένει μπορεί να οριστεί ως «αντικείμενο». Οι μέθοδοι που χρησιμοποιούμε για να δημιουργήσουμε τέτοιες δομές, αποτελούν σημαντικό τμήμα του αρχαιολογικού έργου εκτός πεδίου και συνιστούν προϋπόθεση για ενδιαφέροντα είδη ανάλυσης.

⁵⁴ Gardin 1980, 13-18.

⁵⁵ Wallace 2012, 11.

⁵⁶ Banning 2006, 52.

Αυτές οι δομές περιγράφονται ως «αναγνώριση προτύπων» και «ιστορικά συμπεράσματα».

Οι ομάδες, οι κατηγορίες ή τύποι φαινόμενων διαμορφώνουν το δομικό υλικό των αρχαιολογικών αναλύσεων. Ακολουθεί η σύγκριση, η αναζήτηση ιδιαίτερων συσχετίσεων ή αντιφάσεων ανάμεσα σε διάφορους τύπους ή μεταβλητές, ασυνήθεις ομαδοποιήσεις στον χώρο και τον χρόνο ή άλλα πρότυπα. Τα πρότυπα με τη σειρά τους οδηγούν σε συμπεράσματα, υποθέσεις για αλλαγές της ανθρώπινης συμπεριφοράς, για ομοιότητες και διαφορές ανάμεσα σε ομάδες ανθρώπων στο εγγύς ή απώτερο παρελθόν, για τους παράγοντες που επηρέασαν τις ανθρώπινες συμπεριφορές, ή τον τρόπο με τον οποίο συνεισέφεραν στη σύγχρονη ζωή. Αυτό που ο Gardin (1980) αποκαλεί «ιστορικό συμπέρασμα», είναι ουσιαστικά ερμηνεία και ανάλυση γεγονότων, αλλαγών, λειτουργιών και διαφορών στους παρελθόντες ανθρώπινους πολιτισμούς.

Ο Coombs περιγράφει την ανάλυση ως ανίχνευση σχέσεων, τάξης και δομής στα δεδομένα⁵⁷. Για παράδειγμα, μπορούμε να ανιχνεύσουμε τη σχέση μεταξύ ενός οικιστικού προτύπου και της τροφοπαρασκευαστικής στρατηγικής που ακολουθήθηκε ή μεταξύ της έλλειψης των πόρων και της μείωσης λίθινων κατασκευών. Τα οικιστικά πρότυπα ή οι τροφοπαρασκευαστικές διαδικασίες παράγουν προϊόντα, και οι αρχαιολόγοι ταξινομούν τα προϊόντα του έργου τους στο πεδίο είτε ως «συνθέσεις» είτε ως «ερμηνείες». Η σύνθεση είναι μια απλή συλλογή και παρουσίαση δεδομένων. Η ερμηνεία απαιτεί επεξεργασία της πληροφορίας, έτσι ώστε να αντληθούν συμπεράσματα για την τεχνολογία, τα σύμβολα, την κοινωνική οργάνωση και άλλες όψεις του πολιτισμού, μη εγγενείς στα ακατέργαστα στοιχεία ή τα ίδια τα τέχνηρα.

Οι συνθέσεις είναι αναμφισβήτητα το πλέον ορατό αποτέλεσμα του αρχαιολογικού έργου. Ο Gardin καθορίζει τη σύνθεση ως συστηματοποιημένο σύνολο αλληλοσχετιζόμενων προτάσεων που περιγράφουν τα υλικά κατάλοιπα, προκειμένου να διευκολυνθεί η μελέτη των αρχαίων λαών⁵⁸. Τέτοιου είδους συνθέσεις είναι συχνά συμβολικές κατασκευές και χρησιμοποιούν μια ιδιαίτερη μεταγλώσσα, προκειμένου να απεικονίσουν το περιεχόμενό τους. Τα σχέδια, οι χάρτες, οι ψηφιοποιήσεις, ακόμα και ο κώδικας ενός υπολογιστικού

⁵⁷ Coombs 1964, 5.

⁵⁸ Gardin 1980, 28.

προγράμματος, δεν αντιπροσωπεύουν τίποτε άλλο από απλοποιήσεις της πραγματικότητας και συμβολικές απεικονίσεις της. Προκειμένου να γίνουν χρήσιμα τα νοήματα που προκύπτουν από τις συνθέσεις της πληροφορίας, η επιστήμη της αρχαιολογίας ακολουθεί τη δική της περιγραφική γλώσσα σε ό,τι αφορά στην αρχαιολογική μαρτυρία, μια γλώσσα με συγκεκριμένους κανόνες συστηματικής περιγραφής και μετρήσεων. Ένα από τα πλεονεκτήματα αυτής της γλώσσας είναι ότι επιτρέπει τη χρήση υπολογιστών για τη διαχείριση μεγάλων ποσοτήτων πληροφοριών.

Για παράδειγμα, ένα είδος γλώσσας, αν και συνήθως δεν αναγνωρίζεται ως τέτοια από τους αρχαιολόγους, είναι το σύνολο των γραφικών που απαιτούνται για την παρουσίαση και ερμηνεία της αρχαιολογικής μαρτυρίας. Τα σχέδια και οι χάρτες που συνοδεύουν την αρχαιολογική δημοσίευση δεν είναι πραγματικές απεικονίσεις τεχνέργων ή αρχαιολογικών τόπων. Είναι κωδικοποιημένες απεικονίσεις, που μεταδίδουν συγκεκριμένο είδος πληροφορίας και παραλείπουν στις περισσότερες περιπτώσεις τον «θόρυβο», δηλαδή οποιαδήποτε άλλη πληροφορία παρεμποδίζει τη μετάδοση του νοήματος που επιθυμεί ο αρχαιολόγος.

Η αρχαιολογική μαρτυρία, όντας καταρχήν αποσπασματική, δε φαίνεται να διαθέτει φανερό νόημα. Προκειμένου να δημιουργήσει κανείς από το φαινομενικό χάος νοήμονες συνθέσεις που παράγουν λογική πληροφορία, χρειάζεται να αποφασίσει τι είδους μετρήσεις και αναλύσεις θα χρησιμοποιήσει. Θα μετρήσει για παράδειγμα τους πύθους που θα αποδώσει η ανασκαφή ενός ομαδικού τάφου, θα επιλέξει να μετρήσει πλήρεις σκελετούς ή απλώς θραύσματα οστών;

Αν το παράδειγμα μετατοπιστεί στην αρχαιολογική μαρτυρία μιας κατοικίας, θα επιχειρήσει να εκτιμήσει πόσα αγγεία βρήκε, αγνοώντας τα υπολείμματα τροφής, ή θα προσπαθήσει να κατανοήσει πόσο κρέας για παράδειγμα παρείχαν τα ζώα για τη βασική διατροφή των ενοίκων του κτηρίου; Θα επιχειρήσει να συνδυάσει τις δύο απόλυτες μετρήσεις ή θα παράγει συνδυασμούς απόλυτων μετρήσεων, περνώντας τα σύνορα της βασικής θεωρίας μετρήσεων στις άγνωστες περιοχές των προβλημάτων ποσοτικοποίησης που είναι ενίοτε μοναδικές στην αρχαιολογία; Θα μείνει στην ειδική και απόλυτη μέτρηση που τού υποδεικνύει το «δέντρο» ή θα επιχειρήσει να περάσει στη γενική εικόνα

του «δάσους»; Δεδομένου ότι ο στόχος της αρχαιολογίας σήμερα είναι περισσότερο η κατανόηση του γενικού περιγράμματος του δάσους, χωρίς να απεμπολείται η ανάγκη της ακριβούς ανάλυσης του δέντρου, μέσω της υπολογιστικής διαχείρισης της πληροφορίας είναι δυνατόν να αντιλαμβανόμαστε επαναλαμβανόμενα πρότυπα σε μεγάλες ποσότητες πληροφοριών και να τις χρησιμοποιήσουμε ως αποτελεσματικά εργαλεία ανάλυσης. Η συστηματική ανάλυση στο αρχαιολογικό έργο εκτός από την οικονομία νοήματος που παράγει, είναι διαχειρίσιμη στη μεταγενέστερη προσπάθειά μας να ερμηνεύσουμε τα μοτίβα της πληροφορίας που αντιλαμβανόμαστε.

Σε ένα επίπεδο η αρχαιολογική ερμηνεία έχει ως στόχο της την ανακατασκευή παρελθόντων γεγονότων ή τρόπων ζωής μέσω της αρχαιολογικής μαρτυρίας και οποιασδήποτε άλλης διαθέσιμης πληροφορίας. Στην ευρύτερη αντίληψη του νοήματος που αποδίδει η λέξη «ερμηνεία», περιλαμβάνονται ενίοτε και υπολογισμοί για διαδικασίες περιόδων αλλαγών μακρού χρόνου και περιόδων σταθερότητας, για ζητήματα εξέλιξης και προσαρμογής, για ομοιότητες και διαφορές ανάμεσα σε ομάδες ανθρώπων του παρελθόντος, για στρατηγικές επιβίωσης, επικοινωνίας και ικανοποίησης των φυσικών και ψυχολογικών αναγκών, γενικότερα όλων εκείνων των στοιχείων που διαμορφώνουν τον αρχαιολογικό πολιτισμό.

Το σημαντικότερο στην περίπτωση μας είναι ότι οι νοηματικές κατασκευές που χρησιμοποιούνται στην ερμηνεία, αντίθετα από τις απλές συνθέσεις πληροφοριών, ξεπερνούν τα σύνορα των εγγενών ιδιοτήτων των τεχνέργων, των κτισμάτων των κατασκευών και των οικοδεδομένων, του συνόλου δηλαδή της αρχαιολογικής μαρτυρίας. Για παράδειγμα, είναι δυνατόν να περιγράψει ο αρχαιολόγος με τη βοήθεια ειδικευμένων εργαστηρίων τη χημική σύνθεση του πηλού που χρησιμοποιήθηκε για κάποιο τέχνηργο, χωρίς αυτό να προσφέρει κάποια ερμηνεία στην προσπάθεια ανακατασκευής του παρελθόντος. Είναι, επίσης, δυνατόν να χρησιμοποιήσει αυτή τη μέτρηση συνδυαστικά με άλλες πληροφορίες για την πηγή της πρώτης ύλης ή για τον εντοπισμό εργαστηρίων που χρησιμοποιούσαν ακριβώς αυτό το είδος πηλού, ακόμα και για τη χάραξη εμπορικών δρόμων, σε μια ακολουθία τόπων στους οποίους έχει γίνει η ίδια μέτρηση της χημικής σύνθεσης σε ανάλογα τέχνηρα. Με αυτόν τον τρόπο παράγονται ερμηνευτικές συνθέσεις νοήματος, νοηματικές κατασκευές, στις

οποιές η τελική πρόταση περιέχει πληροφορία που δεν είναι εγγενής στην αρχική πρόταση συλλογισμού⁵⁹.

Με επίκεντρο την ερμηνεία, δηλαδή την κατανόηση, η ανάλυση χρειάζεται εργαλεία καταγραφής τέτοια, ώστε να θεμελιώνεται η αναγνώριση προτύπων, να δοκιμάζονται οι υποθέσεις και κυρίως να εκτιμώνται οι ερμηνείες μέσα από τη βάση της πληροφορίας. Η ερμηνεία ή κατανόηση είναι ο στόχος της ανάλυσης και η προσέγγιση της ερμηνείας απαιτεί στατιστικές μεθόδους, οι οποίες παρέχουν μια εικόνα της σχέσης που αναπτύσσεται ανάμεσα στην εγγενή πληροφορία που παρέχει ένα τέχνηργο για παράδειγμα και τη σχέση του με τις εξωτερικές πληροφορίες που σχετίζονται με το πλαίσιο μέσα στο οποίο αποκαλύπτεται αυτό το τέχνηργο και αφορούν κυρίως σε ζητήματα χώρου, χρονολόγησης και λειτουργίας.

Οι απαιτήσεις της ερμηνείας, ωστόσο, δε σταματούν στα όρια της φυσικής λειτουργίας, αλλά διεγείρουν ερωτηματικά ως προς τη σχέση αυτής της καθαρής λειτουργίας με ζητήματα κοινωνικής και ψυχολογικής προσέγγισης του πολιτισμού. Ποιας κοινωνικής προσέγγισης, όμως, όταν κάθε αρχαιολόγος ή αντίστοιχα ο ανθρωπολόγος είναι αφεαυτού κοινωνική και πολιτική μονάδα που σχετίζεται με συγκεκριμένες ιδεολογίες, ενίοτε ιδεοληψίες και εμμονές; Σε αυτό το επίπεδο η αυστηρή μαθηματική προσέγγιση ενός λογάριθμου δεν είναι προς το παρόν ικανή να δώσει απαντήσεις κοινωνικής ερμηνείας, παρά μόνον να διατηρήσει τα δεδομένα ταξινομημένα σε κατηγοριοποιήσεις αυστηρά κατευθυνόμενες από την αρχαιολογική θεωρία.

Σε πολλές επιστήμες οι ερευνητικοί σχεδιασμοί περιλαμβάνουν ή βασίζονται στο πείραμα. Κατά τη διάρκεια του πειράματος είναι δυνατόν να διατηρούνται κάποιοι παράγοντες σταθεροί και να μεταβάλλονται κάποιοι άλλοι, ή έστω μόνον ένας, προκειμένου να παρατηρηθούν οι επιδράσεις αυτών των αλλαγών σε μεταβλητές του πλαισίου μέσα στο οποίο διαμορφώνεται ο πειραματισμός. Το μεγαλύτερο τμήμα της αρχαιολογικής ανάλυσης δεν είναι δυνατόν να κάνει χρήση του πειράματος, γιατί δεν μπορεί να ελέγξει ή να διαχειριστεί συμβάντα του παρελθόντος, εξαιτίας κυρίως του καταστροφικού χαρακτήρα της αρχαιολογικής πρακτικής. Κάτι τέτοιο μπορεί να γίνει μόνο σε αυστηρά ελεγχόμενες συνθήκες, κατά τις οποίες γίνεται προσπάθεια

⁵⁹ Salmon 1982, 33.

αναπαραγωγής ή προσομοίωσης λειτουργικών διαδικασιών, όπως είναι για παράδειγμα η πειραματική παραγωγή λίθινων λεπίδων.

Ένα μεγάλο τμήμα της αρχαιολογικής ανάλυσης εξαρτάται, επίσης, από τη σύγκριση αρχαιολογικών θέσεων διαφορετικού τύπου, ή όμοιων αρχαιολογικών θέσεων σε διαφορετικά φυσικά περιβάλλοντα, με διαφορετικές οικονομίες, ακόμη και σε διαφορετικές χρονολογίες. Τέτοιου είδους αναλύσεις εξαρτώνται κυρίως από μεστές νοήματος και περιεχομένου ομαδοποιήσεις αρχαιολογικών θέσεων ή ταξινομήσεις σε ιδιαίτερους τύπους, προκειμένου να εντοπιστούν οι μεταβλητές που διαφοροποιούν τη μία αρχαιολογική θέση από την άλλη, αφήνοντας όμως ανέπαφο το κεντρικό σώμα της θεωρίας που καλύπτει την ανάλυσή τους και ταυτόχρονα πιστοποιεί το μη-τυχαίο της διαφοροποιημένης μεταβλητής⁶⁰.

Η δημιουργία δεδομένων και η ανάδειξη του νοήματος που προκύπτει από έναν αρχαιολογικό χώρο, σε ένα επίπεδο, επιτυγχάνεται με τη συνεχή αλληλεπίδραση της φυσικής παρατήρησης και των επαναλαμβανόμενων επιπέδων ερμηνευτικών συσχετίσεων και ερμηνειών. Αυτή η διαδικασία παραγωγής δεδομένων που ονόμασε «τρέχουσα ανάλυση» ο Binford⁶¹ περιλαμβάνει τη δυναμική διάδραση παρατηρήσεων και συσχετίσεων σε ένα πλαίσιο στρωματογραφικών πιθανοτήτων. Ο Hirst χαρακτήρισε τη διαδικασία ως «κυκλική διαδικασία ερωτήσεων, υποθέσεων και δοκιμών παρατηρημένων δεδομένων»⁶². Άλλοι θεωρητικοί αρχαιολόγοι την αποκαλούν, επίσης, διαλεκτική διαδικασία⁶³.

Βέβαια, δεν είναι εύκολο να σχεδιάσουμε κάποιο πείραμα, με το οποίο θα ανιχνεύσουμε αλλαγές στη διάρκεια του χρόνου, εξαρτώμενοι από μία μόνο μεταβλητή. Αντίθετα, ψάχνουμε σχέσεις μεταβλητών, χρησιμοποιώντας καλά χρονολογημένα σύνολα. Συνεπώς, η αρχαιολογική ανάλυση αφιερώνει τυπικά ένα μεγάλο μέρος της προσοχής της στη χρονολόγηση. Η σημασία του χρονολογικού ελέγχου υπήρξε άλλωστε το έναυσμα για την εφαρμογή πολλών μεθόδων χρονολόγησης στην αρχαιολογία⁶⁴. Σε ό,τι αφορά στις συγκρίσεις διαφορετικών ομάδων αρχαιολογικών ευρημάτων, χρειαζόμαστε το θεωρητικό

⁶⁰ Banning 2006, 4.

⁶¹ Binford 1972, 159.

⁶² Hirst 1976, 11.

⁶³ Chadwick 2003, 107. Επίσης, Yarrow 2003, 69.

⁶⁴ Dean 1978, 224.

υπόβαθρο που θα κάνει κατανοητές οποιεσδήποτε σχέσεις. Οι κατανοητές συγκρίσεις που διαθέτουν ένα είδος νοήματος, με τη σειρά τους, απαιτούν έγκυρες κλίμακες μέτρησης, αλλά και ένα τρόπο αναγνώρισης του λάθους⁶⁵.

Βάσει των παραπάνω γίνεται κατανοητό πως η αρχαιολογική θεωρία, στο σύνολό της θεωρούμενη, εξαρτάται από ένα πλήθος μεταβλητών, η ανάγνωση των οποίων σχετίζεται άμεσα με το θεωρητικό υπόβαθρο άλλων επιστημών και συνεπώς αναδιατυπώνεται σύμφωνα με τις τρέχουσες εξελίξεις τόσο στο επίπεδο των κοινωνικών, όσο και των φυσικών επιστημών. Η κατασκευή ενός λογαρίθμου ικανού να καταγράφει και να συνδυάζει τα ακριβή αρχεία του αρχαιολογικού πειράματος, δηλαδή της ανασκαφικής διαδικασίας, συναρτάται ουσιαστικά με τις διεθνείς σταθερές και τις νόρμες καλής λειτουργίας, όπως ορίζονται από τους μεγάλους οργανισμούς καταγραφής και συντήρησης πολιτισμικών δεδομένων στο επίπεδο της επιστήμης υπολογιστών.

Η διαμόρφωση ενός συστήματος διαχείρισης περιεχομένου συνοδεύεται, όσον αφορά τουλάχιστον στην αρχαιολογία με νόρμες καλής πρακτικής και συντήρησης των δεδομένων που προκύπτουν από την έρευνα. Η αρχαιολογία από αντικειμενοστρεφές επιστημονικό πεδίο έχει εξελιχθεί σε αναλυτικό πεδίο, το οποίο περιλαμβάνει για τις ανάγκες της ανάλυσης δεδομένα, δηλαδή γραπτά κείμενα, σχέδια, φωτογραφίες και ψηφιακά δεδομένα. Εξαρχής είναι ευθύνη των αρχαιολόγων η διατήρηση των ευρημάτων αλλά και των αρχείων που τηρούν, ώστε να είναι αφενός ακριβείς στη τελική τους δημοσίευση, αφετέρου να είναι βιώσιμα τα αποτελέσματα των ερευνών τους σε βάσεις δεδομένων, που εξασφαλίζουν την επανεξέταση των συμπερασμάτων και ερμηνειών υπό το πρίσμα νέων δεδομένων που συλλέγονται από μελλοντικούς αρχαιολόγους⁶⁶.

1.4. Ζητήματα διαχείρισης της πληροφορίας: Ορίζοντες πληροφόρησης

Οι τεχνικές και οι μέθοδοι καταγραφής και διατήρησης αρχαιολογικών πληροφοριών έχουν συζητηθεί ευρέως στη βιβλιογραφία. Ωστόσο, οι περαιτέρω χρήσεις αυτών των δεδομένων, όπως και οι επιπτώσεις τους στην αρχαιολογική

⁶⁵ Banning 2006, 5.

⁶⁶ Perrin *etal*, 2014, 10-12.

πληροφορία σπάνια συζητούνται ολοκληρωμένα. Οι αρχαιολογικές έρευνες και οι οργανισμοί πολιτισμικής κληρονομιάς συχνά επικεντρώνονται στην παραγωγή δεδομένων για δική τους κυρίως χρήση, χωρίς ιδιαίτερη φροντίδα για τους μελλοντικούς χρήστες των επεξεργασμένων πληροφοριών. Προκειμένου να διασφαλιστεί η λειτουργική και όχι μόνο η φυσική βιωσιμότητα των παλαιότερων και των σημερινών πόρων της αρχαιολογικής πληροφόρησης, απαιτείται ιδιαίτερη έμφαση στη λειτουργική τους βιωσιμότητα.

Ενδιαφέρον παρουσιάζει εδώ η ερευνητική μέθοδος που πρότεινε ο Sonnenwald το 1999 για την ανθρώπινη πληροφορία ως θεωρητικό θεμέλιο⁶⁷. Το συγκεκριμένο θεωρητικό πλαίσιο αναγνωρίζει την ύπαρξη ενός «ορίζοντα πληροφόρησης» με τον οποίο αλληλεπιδρά το άτομο. Για παράδειγμα, ένας αρχαιολόγος πεδίου μπορεί να κάνει χρήση πολλών πηγών πληροφόρησης, όπως είναι τα κοινωνικά δίκτυα, ή τα διαθέσιμα έγγραφα που σχετίζονται με την έρευνά του, να κάνει χρήση εργαλείων για την ανάκτηση πληροφοριών, να πειραματιστεί με τη συσχέτιση και την ταξινόμηση των πληροφοριών, προκειμένου να ενδυναμώσει το ερευνητικό του έργο.

Ο ορίζοντας των αρχαιολογικών πηγών πληροφόρησης είναι δυνατόν να γίνει αντιληπτός ως χώρος ή περιβάλλον⁶⁸, που περικλείει την αρχαιολογική έρευνα και τα πληροφοριακά «εργαλεία» που χρησιμοποιούνται για τη διενέργειά της. Εκτός από τους αναγκαίους πόρους, οι χάρτες που παράγονται για τον συγκεκριμένο ορίζοντα πληροφοριών, παρέχουν μια μέθοδο για την απεικόνιση των αλληλεπιδράσεων που ασκούν μεταξύ τους οι πληροφορίες. Οι ορίζοντες ενημέρωσης ή άντλησης πηγών πληροφόρησης συγκλίνουν με τους διαφορετικούς ρόλους που αναλαμβάνει ο αρχαιολόγος για τη διεκπεραίωση της έρευνας.

Για παράδειγμα, ο ορίζοντας πληροφοριών του συγκεκριμένου ρόλου της αρχαιολογίας πεδίου επικεντρώνεται σε μία θέση ή γεωγραφική περιοχή αρχαιολογικού ενδιαφέροντος. Οι αρχαιολόγοι πεδίου είναι ουσιαστικά χρήστες και δημιουργοί πληροφορίας κατά τη διάρκεια της προετοιμασίας και της εκτέλεσης της ανασκαφικής έρευνας. Είναι δημιουργοί πληροφορίας που καταγράφεται στο αρχαιολογικό ημερολόγιο και χρήστες του ορίζοντα

⁶⁷ Sonnenwald 1999, 176-190.

⁶⁸ Hartel 2017, CoLIS paper 1635.

πληροφοριών, όταν διεξάγουν συγκριτική έρευνα για ταυτοποίηση των δεδομένων τους με γνωστά δεδομένα που έχουν ήδη εκδοθεί ή υπάρχουν ως πληροφορία σε βάσεις δεδομένων. Εδώ, χρειάζεται να επισημανθεί ίσως ότι ο πρωταρχικός ορίζοντας πληροφόρησης του αρχαιολόγου πεδίου είναι το αρχαιολογικό υλικό που προκύπτει από την ανασκαφική έρευνα, γεγονός που παρέχει και την αναγκαία βαρύτητα στην όσο το δυνατόν πληρέστερη και αντικειμενική καταγραφή του αρχαιολογικού ημερολογίου.

Όσον αφορά στη συγκριτική έρευνα, ο ορίζοντας πληροφοριών εξαπλώνεται πέρα από τη συγκεκριμένη γεωγραφική θέση και περίοδο, προκειμένου να περιλάβει πληροφορίες σχετικές με τις αλληλεπιδράσεις αυτής της γεωγραφικής θέσης με άλλες της ίδιας ή άλλης χρονικής περιόδου. Το έργο της ανεύρεσης της πληροφορίας γίνεται με τη χρήση διαφορετικών πηγών και με τη διαρκή πρόσκτηση της πληροφορίας μέσω της παρατήρησης, επιστρέφοντας κυκλικά στο σημείο έναρξης της διαδικασίας. Η διαδικασία είναι ουσιαστικά κυκλική και επαναληπτική. Η διάρκειά της εξαρτάται από τις επαναλήψεις που επιτρέπουν τα διαφορετικής φύσης εμπόδια στην αλληλεπίδραση των πληροφοριών.

Οι αρχαιολόγοι εργάζονται με μια ευρεία ποικιλία πληροφοριών, αλλά ο πυρήνας των πηγών από τις οποίες αντλούνται οι πληροφορίες συγκροτείται από ένα αρκετά περιορισμένο σύνολο υλικών. Ως εκ τούτου, δεν είναι τα ίδια τα υλικά που καθιστούν διακριτούς τους ερευνητικούς ρόλους. Η χρήση της πηγής γίνεται διακριτή εξαιτίας της συγκεκριμένης οργάνωσης του ορίζοντα πληροφόρησης και εξαιτίας της ύπαρξης εστιασμένων πόρων κατά τη διαδικασία αναζήτησης πληροφοριών⁶⁹.

Κατά τον ίδιο τρόπο ο ερευνητικός ρόλος του συλλέκτη αρχαιοτήτων, για παράδειγμα, εστιάζει σε διαφορετικά τμήματα της αρχαιολογικής μαρτυρίας από εκείνα που εστιάζει ο ερευνητικός ρόλος του αρχαιολόγου πεδίου. Οι πηγές πληροφόρησης των συλλεκτών εστιάζουν κυρίως στην οργάνωση και την ευρετηρίαση, και παράγεται έργο που σχετίζεται με την τροποποίηση, την αποθήκευση και την ανάκτηση πληροφοριών. Η έρευνα των πηγών ξεκινά συνήθως από κάποια τοπική βάση δεδομένων συλλογών (πομπό) και εξαπλώνεται για να περιλάβει ένα φάσμα αρχαιολογικής βιβλιογραφίας και

⁶⁹ Sonnenwald and Wildemuth 2001, 13.

άλλων πηγών πληροφοριών (φορέων) που σχετίζονται με κάποιο τέχνηργο. Ο ορίζοντας πηγών πληροφόρησης των συλλεκτών είναι οργανωμένος γύρω από επαναλήψεις, οι οποίες ξεκινούν από ένα μητρώο αρχαιοτήτων ή μία συλλογή δεδομένων. Η διαδικασία, ωστόσο, σπάνια τελειώνει στη βάση δεδομένων. Ο πλέον χαρακτηριστικός δέκτης είναι το τέχνηργο, που λειτουργεί ως ώθηση στη διαδικασία αναζήτησης. Οι αρχαιολόγοι και οι συλλέκτες αρχαιοτήτων έχουν την τάση να συμβουλευούνται πηγές με πρωταρχικό σκοπό την εξεύρεση περιγραφών των αντικειμένων μελέτης τους και των σχετικών συγκριτικών υλικών. Η αρχαιολογία πεδίου σχετίζεται με τις περιγραφές της ανασκαφής ή της περιοχής που ερευνήθηκε και τις αντίστοιχες παρατηρήσεις που έγιναν αλλού, ενώ ο ρόλος του συλλέκτη επικεντρώνεται στην αναζήτηση ειδικών περιγραφικών πληροφοριών για το τέχνηργο ή την ομάδα τεχνέργων.

Με το ίδιο σκεπτικό όσοι εργάζονται στον τομέα της εκλαΐκευσης της αρχαιολογικής γνώσης ενδιαφέρονται κυρίως για ευρύτερα αρχαιολογικά θέματα και όχι για επιμέρους στοιχεία δεδομένων. Ένας τυπικός πομπός είναι η γενική αρχαιολογική και ιστορική λογοτεχνία, η οποία μπορεί να ρίξει φως σε συγκεκριμένα φαινόμενα, για παράδειγμα τη μεσαιωνική ιστορία. Η διαδικασία μεταφέρεται στη λογοτεχνία (φορέας) και τυπικά τελειώνει με την εύρεση κατάλληλων αρχαιολογικών αντικειμένων για απεικόνιση ή δημοσίευση (δέκτης). Ο ρόλος της εκλαΐκευσης σχετίζεται με την τροποποίηση ή την επεξεργασία και την απλούστευση της αρχαιολογικής πληροφορίας, ώστε να έχει νόημα για ένα ευρύτερο και μη εξειδικευμένο ακροατήριο.

Ο ορίζοντας πληροφόρησης της ακαδημαϊκής έρευνας δίνει την ίδια έμφαση στο αρχαιολογικό υλικό ως εύλογο σημείο εισόδου. Ωστόσο, σε αντίθεση με το αρχαιολογικό έργο πεδίου ή του συλλέκτη αρχαιοτήτων, η ευρύτερη επιστημονική έρευνα τείνει να παρουσιάζει κυμαινόμενα σημεία εισόδου (πομπούς). Ο ορίζοντας αποτελείται από μια αρκετά μεγάλη ποικιλία επιστημονικών πηγών πληροφοριών, οι οποίες χρησιμοποιούνται σύμφωνα με τις πραγματικές ανάγκες της έρευνας. Οι ακαδημαϊκοί ερευνητές υπογραμμίζουν συχνά τη σημασία των αρχαιολογικών εκθέσεων ως πηγές πρωτογενούς γνώσης, παρόλο που συχνά έκαναν εξίσου επικριτικές παρατηρήσεις σχετικά με το περιορισμένο πεδίο εφαρμογής τους, το οποίο θεωρήθηκε συνέπεια των περιορισμένων πόρων στην αρχαιολογική εργασία. Η γενική έλλειψη χρήσιμων

και ολοκληρωμένων αρχαιολογικών βάσεων δεδομένων αναγνωρίζεται ευρέως σε διάφορες έρευνες. Η ίδια έννοια ισχύει για όλες τις μορφές ηλεκτρονικών μέσων⁷⁰.

Ο πίνακας που παρατίθεται στη συνέχεια, συνοψίζει τις παρατηρήσεις που έγιναν σχετικά με τους ορίζοντες πληροφόρησης του εκάστοτε ρόλου εργασίας στα πλαίσια του αρχαιολογικού έργου. Οι στήλες του πίνακα ανακεφαλαιώνουν τον ρόλο εργασίας, τον πομπό και τη φύση των πρώτων προσπελάσιμων πληροφοριών.

Πεδίο	Εστίαση	Μετάδοση	Πληροφορία	Ειδικευμένη	Ταξινόμηση	Πρόσβαση
Αρχαιολογία πεδίου	Τόπος	Αναφορά έρευνας	Περιγραφική	Ειδική	Σύνολα πληροφορίας	Αναζήτηση/ Φυλλομέτρηση
Συλλογή αρχαιοσημάτων	Τέχνη/εργο	Βάση δεδομένων συλλογών	Περιγραφική	Ειδική	Βάση δεδομένων	Αναζήτηση/ Φυλλομέτρηση
Εκλαϊκείωση	Υποκείμενο	Γενική λογοτεχνία	Επιτηδευμένη	Γενική	Σύνολα πληροφορίας	Γενικού επιπέδου αναζήτηση
Ακαδημαϊκή έρευνα	Ποικίλλει	Ποικίλλει	Ποικίλλει	Ειδική	Σύνολα πληροφορίας	Αναζήτηση

Πιν. 1. Όψεις των ιδιαίτερων πηγών πληροφορίας με διαφορετικούς εργασιακούς ρόλους.

Η διαδικασία άντλησης πληροφόρησης των επαγγελματιών της αρχαιολογίας αγκαλιάζει ολόκληρο τον κύκλο ζωής των πληροφοριών⁷¹, από τη δημιουργία έως την οργάνωση, τη διάδοση και τη χρήση. Η διαδικασία είναι πολυδιάστατη και πολύπλευρη, επαναληπτική και εντατική. Διαφορετικοί τύποι πληροφοριών δημιουργούνται, οργανώνονται και χρησιμοποιούνται ταυτόχρονα. Παρομοίως, μπορεί να οργανωθεί, να διανεμηθεί και να χρησιμοποιηθεί ταυτόχρονα σε διάφορες περιπτώσεις μία και μόνη πληροφορία. Σε κάθε περίπτωση είναι δυνατή κατασκευή χαρτών πληροφόρησης των αρχαιολόγων

⁷⁰ Lönnqvist, 1988, 75.

⁷¹ Borgman et al 1996, 6-7.

πεδίου όπως και η ενσωμάτωση της ιδέας του ορίζοντα πληροφοριών σε ένα σύστημα διαχείρισης περιεχομένων.

Κεφάλαιο 2: Γενικά και ειδικά ζητήματα αρχαιολογικής θεωρίας

2.1. Η προβληματική των θεωρητικών κλάδων της αρχαιολογίας

Με τον όρο *θεωρία* εννοούμε, συνήθως, ένα ολοκληρωμένο σύστημα ιδεών και προτάσεων που προσεγγίζει ένα ζήτημα από αφηρημένη ή συγκεκριμένη οπτική γωνία. Πρόκειται για τη θεωρία υπό την οπτική γωνία της φιλοσοφίας, δηλαδή ένα σύνολο ισχυρισμών, ουσιαστικά αξιωμάτων, που οδηγούν σε συμπεράσματα ή υποστηρίζουν συμπεράσματα για τη φύση του κόσμου⁷². Σε διαφορετικό εννοιολογικό πλαίσιο η θεωρία ορίζεται ως υπόθεση για την ερμηνεία ενός προβλήματος⁷³. Αναλυτικότερα, είναι μια δομή, που στοχεύει στην ερμηνεία ενός συνόλου παρατηρήσεων, τυποποιώντας το σύνολο των διακριτών παρατηρήσεων ως τάξη φαινομένων και επιβεβαιώνοντας την υποκείμενη πραγματικότητα που επιδρά ή επηρεάζει αυτή την τάξη φαινομένων.

Σε ό,τι αφορά στην επιστήμη, η θεωρία είναι τρόπος εξέτασης ενός πεδίου με ερμηνευτικά και προγνωστικά επαγωγικά συμπεράσματα. Είναι ένας χάρτης που μας βοηθά να εξερευνήσουμε την περιπλοκότητα του κόσμου, να βρούμε διασυνδέσεις και να ερμηνεύσουμε γιατί συμβαίνουν κάποια πράγματα⁷⁴. Στοιχειώδες καθήκον της φιλοσοφίας της επιστήμης θεωρήθηκε συχνά η οριοθέτηση των καλών ή επιστημονικών θεωριών από κακές, μη επιστημονικές. Στο απόγειο του λογικού θετικισμού, η ιδιαίτερα φορμαλιστική προσέγγιση στη θεωρία τη μεταχειρίστηκε με όρους αξιωματικών συστημάτων. Οι θεωρητικοί όροι αυτών των συστημάτων συνδέθηκαν στενά με ένα ιδιόμορφο λεξιλόγιο, που υποτίθεται ότι θα παρείχε το θεμέλιο για την εμπειρική έννοια. Μια λιγότερο φορμαλιστική και περισσότερο εννοιολογική προσέγγιση επιχείρησε ο Thomas Khun με το έργο του, δίνοντας έμφαση στην ανοικτή φύση της επιστημονικής δραστηριότητας, στην ευριστική αξία αναλογιών και μοντέλων, καθώς επίσης στην ελαστικότητα και τον ολισμό του συστήματος, θεωρώντας ότι η αυστηρά φορμαλιστική προσέγγιση οδηγούσε σε διαστρέβλωση το υποκείμενο⁷⁵.

⁷² Abend 2008, 173-199.

⁷³ Sparkes 1991, 114.

⁷⁴ Neuman 2014, 9.

⁷⁵ Rigney 2001, 205.

Στη λογική και τα μαθηματικά η θεωρία είναι ένα συνεκτικό σύστημα των πρωτογενών εννοιών, αξιωμάτων και κανόνων, από τους οποίους είναι δυνατόν να κατασκευαστούν θεωρήματα. Είναι επίσης πρόταση ή σύνολο προτάσεων που προσφέρονται ως υποθετική ερμηνεία για ένα παρατηρημένο φαινόμενο, μια κατάσταση πραγμάτων ή ένα γεγονός⁷⁶. Από μια διαφορετική οπτική γωνία, λοιπόν, η θεωρία είναι μια αφήγηση του κόσμου πέρα από τα μετρήσιμα μεγέθη που προσφέρει η επιστήμη. Αγκαλιάζει ένα σύνολο σχετιζόμενων ορισμών, ένα σύνολο σχέσεων που οργανώνουν την κατανόησή μας για τον εμπειρικό κόσμο με συστηματικό τρόπο. Για παράδειγμα, είναι δυνατόν να διατυπώσουμε κάποια στατιστική σχέση ανάμεσα στη φτώχεια και την αύξηση της εγκληματικότητας, αλλά η ερμηνεία αυτής της σχέσης πιθανώς απαιτεί τη χρησιμοποίηση θεωριών για τα κίνητρα των ανθρώπων ή διαφορετικών νοημάτων που προσαρτώνται στις αρχικές έννοιες *φτώχεια* και *έγκλημα*, ακόμη και των κοινωνικών περιορισμών που κρατούν τμήματα του παγκόσμιου πληθυσμού στη φτώχεια.

Σε μια απλούστερη εκδοχή της η θεωρία είναι μια δεδομένη σειρά, μια τάξη στην οποία επιλέγουμε να στερεώσουμε τα γεγονότα⁷⁷. Όσο περισσότερο τείνει να γίνει κανείς ακριβέστερος ή λεπτομερέστερος σε αυτή τη σειρά γεγονότων, τόσο περισσότερο τείνει να αγνοεί θεωρητικές εκτιμήσεις που απομακρύνονται από την αποδεκτή γνώση, έστω και αν αυτή προκύπτει από πιθανή αναδιατύπωση του θεωρητικού θεμέλιου των επιστημών ή τη ροή των τεχνολογικών εξελίξεων. Ουσιαστικά, το ερώτημα τι είναι «θεωρία», στην προκειμένη περίπτωση συνδέεται άμεσα με το ερώτημα «τι είναι αρχαιολογία».

Οι όποιες συζητήσεις για τον ορισμό της αρχαιολογίας ως ανθρωπιστική σπουδή ή θετική επιστημονική σπουδή είναι μάλλον στείρες. Επιπλέον, τέτοιου είδους συζητήσεις συναρτώνται σε θετικιστικούς ορισμούς, που κατασκευάζονται ανάλογα με το πεδίο έρευνας και το αντικείμενο κάθε διακριτής επιστήμης και όχι με τη συνολική γνώση που αποκτάται από τη διερεύνηση της ιστορίας των επιστημών και τις επιμέρους σχέσεις τους με οποιεσδήποτε κοινωνικές αλλαγές⁷⁸.

⁷⁶ Hazewinkel 2001, διαθέσιμο online.

⁷⁷ Johnson 2010, 118.

⁷⁸ Με τον όρο θετικισμός αναφερόμαστε σε ένα σύνολο επιστημολογικών αντιλήψεων που ορίζουν την επιστημονική μέθοδο ως βέλτιστη προσέγγιση για την αποκάλυψη των διαδικασιών μέσω των οποίων «συμβαίνουν» τόσο τα φυσικά όσο και τα ανθρώπινα συμβάντα. Ουσιαστικά, η θετικιστική προσέγγιση ως τάση επαναλαμβάνεται συχνά στην ιστορία των επιστημών, αλλά στη σύγχρονη μορφή του

Στη σύγχρονη κοινωνική επιστήμη έχουν αμφισβητηθεί οι ισχυρές επιδράσεις του θετικισμού. Οι ίδιοι οι υποστηρικτές αυτού του ρεύματος σκέψης αποδέχονται τις προκαταλήψεις και τους δομικούς περιορισμούς του παρατηρητή, όταν εισάγει ως μεθοδολογικά εργαλεία στις ερμηνείες του μη θετικιστικούς αξιολογικούς χαρακτηρισμούς που σχετίζονται με τη «διαύγεια», την «αξιοπιστία» και την «εγκυρότητα»⁷⁹. Αυτού του είδους ο θετικισμός εξισώνεται γενικά με την «ποσοτική έρευνα» και συνεπώς δε φέρει εγγενώς θεωρητικές ή φιλοσοφικές δεσμεύσεις. Το νέο είδος κοινωνιολογίας πιστώνεται συχνά στον Paul Lazarsfeld⁸⁰, πρωτοπόρο των δημοσκοπήσεων μεγάλης κλίμακας και εισηγητή στατιστικών μεθόδων για την ερμηνεία τους. Η συγκεκριμένη προσέγγιση συνδέεται με τη μέση θεωρία του Merton, δηλαδή την παραγωγή αφηρημένων εννοιών που διαμορφώνεται από αποσπασματικές υποθέσεις και εμπειρικές κανονικότητες, παρά από την αφηρημένη ιδέα ενός κοινωνικού συνόλου⁸¹.

Ως ποσοτική ορίζεται εκείνη η μέθοδος έρευνας που στηρίζεται περισσότερο στη συλλογή και ανάλυση αριθμητικών και στατιστικών δεδομένων παρά σε παρατηρήσεις, υποκειμενικές αναφορές, και μικρά στοχευμένα ερωτηματολόγια για την εξαγωγή συμπερασμάτων. Ειδικότερα στις κοινωνικές επιστήμες η ποσοτική έρευνα σχετίζεται με τη συστηματική, εμπειρική έρευνα των ποσοτικών ιδιοτήτων⁸² και φαινομένων και των σχέσεών τους. Αντικειμενικός στόχος της ποσοτικής έρευνας είναι η ανάπτυξη και χρήση μαθηματικών μοντέλων, θεωριών και/ή υποθέσεων σχετικών με τα φαινόμενα. Η διαδικασία της μέτρησης συνιστά τον πυρήνα της ποσοτικής έρευνας καθώς παρέχει τη θεμελιώδη σχέση μεταξύ εμπειρικής παρατήρησης και μαθηματικής έκφρασης των ποσοτικών σχέσεων.

Η συγκεκριμένη μέθοδος εξ ορισμού ποσοτική στις φυσικές επιστήμες, χρησιμοποιείται ευρέως σε κοινωνικές επιστήμες όπως η ψυχολογία, η

αναπτύχθηκε από τον August Comte στις αρχές του 19ου αι. Βλ. Επίσης Cohen and Maldonado 2007, 9.

⁷⁹ Gartell and Gartell 1996, 36.

⁸⁰ Wacquant 1992, 507-510

⁸¹ Boudon 1991, 519-522.

⁸² Η ποσοτική ιδιότητα είναι εγγενής σε ό,τι έχει μέγεθος και μπορεί να μετρηθεί αριθμητικά. Οι μετρήσεις οποιασδήποτε ιδιαίτερης ποσοτικής μεθόδου εκφράζονται ως ιδιαίτερη ποσότητα, εξαρτώμενη από μια μονάδα, πολλαπλασιασμένη με έναν αριθμό. Παραδείγματα φυσικών ποσοτήτων είναι η απόσταση, η μάζα και ο χρόνος. Πολλές ιδιότητες της κοινωνικής επιστήμης, ανάμεσά τους οι ικανότητες και τα χαρακτηριστικά της προσωπικότητας, διερευνώνται με ποσοτικές μεθόδους.

κοινωνιολογία, οι πολιτικές επιστήμες, η ανθρωπολογία και η αρχαιολογία, αν και σε διαφορετικό εννοιολογικό πλαίσιο⁸³. Οι ποιοτικές μέθοδοι με τη σειρά τους παράγουν πληροφορία μέσω της μελέτης συγκεκριμένων περιπτώσεων διαμορφώνοντας σχετικές υποθέσεις, που συνήθως επαληθεύονται με ποσοτικές μεθόδους⁸⁴.

Ακόμα και η ήπια εκδοχή του κοινωνικού δομισμού της σκέψης του Foucault⁸⁵ υποστηρίζει επί της ουσίας μια τέτοια θέση. Αντί να υπάρχουν με έναν *a priori* τρόπο, δημιουργούνται φαινομενικά ανεξάρτητες υποκειμενικές επικράτειες μέσω μιας διαδικασίας που δε συνδέεται με τη διανοητική και πολιτιστική ιστορία. Σε ένα βαθύτερο επίπεδο η «θεωρία» τείνει να γίνει διακριτό πεδίο, που κατευθύνει τα ερωτηματικά της αρχαιολογίας και οι μέθοδοι που χρησιμοποιούνται για να απαντηθούν αυτά τα ερωτηματικά είναι εν μέρει παραπλανητικές. Πρόκειται για μια εκβιαστική απάντηση, στην απαίτηση να βάλουν οι αρχαιολόγοι τη «θεωρία σε πράξη», στην προσδοκία ότι μπορεί η αρχαιολογική κοινότητα να δείξει πως η ενασχόληση με τη θεωρία έχει θετικιστική επίδραση στην αρχαιολογική ερμηνεία. Οπωσδήποτε, αυτή η προσδοκία δεν είναι παράλογη, αλλά είναι και δυνητικά παραπλανητική, γιατί διαχωρίζει τη θεωρητική αρχαιολογία από άλλες μορφές αρχαιολογικής έκφρασης, στις οποίες η «πρακτική» κρίνεται «ελεύθερη θεωρίας» και σπάνια ευθυγραμμίζεται με τρόπο ώστε να επιδεικνύει μια απλή σχέση αιτίου-αποτελέσματος ανάμεσα στο θεωρητικό και το πρακτικό αρχαιολογικό έργο.

Ουσιαστικά δεν υφίσταται διαχωρισμός ανάμεσα στην αρχαιολογική θεωρία και την αρχαιολογική πράξη. Οποιαδήποτε μορφή αναντιστοιχίας σε τέτοιες περιπτώσεις οδηγεί σε μακροσκελείς διαμάχες και συζητήσεις που δεν παρέχουν απαντήσεις στο βασικό ερώτημα. Σε ποιο βαθμό οι συλλογισμοί εκείνων που αυτοαποκαλούνται θεωρητικοί της αρχαιολογίας αποτελούν ακριβή απεικόνιση των όσων πραγματικά κάνουν στο πεδίο οι αρχαιολόγοι;

⁸³ Στις κοινωνικές επιστήμες ο όρος σχετίζεται με εμπειρικές μεθόδους, απόρροια του φιλοσοφικού θετικισμού και της ιστορίας της στατιστικής επιστήμης και συνιστά αντιθετικό πόλο της ποιοτικής έρευνας.

⁸⁴ Η ανάλυση 1274 άρθρων των μεγαλύτερων περιοδικών εκδόσεων για τις κοινωνικές επιστήμες στις Η.Π.Α. από το 1935 έως το 2005 υπέδειξε ότι περίπου το 2/3 των ερευνών χρησιμοποίησαν ποσοτικές μεθόδους. Βλ. Hunter and Leahey 2008, 290–306.

⁸⁵ Foucault 2002, 102-110.

Σε γενικές γραμμές αναγνωρίζεται ότι υφίσταται έλλειψη αντιστοιχίας, που εξαρτάται σε ένα βαθμό από την «ιστορική» τάση συντόμευσης και περιορισμού της αρχαιολογικής θεωρίας σε ζητήματα πρακτικά και κατά τη γνώμη μας από τη σχετική απομάκρυνση της αρχαιολογίας από ζητήματα πολιτισμικής διαχείρισης. Η συσχέτιση θεωρίας και πράξης ήταν και παραμένει ζητούμενο στην αρχαιολογία, παρόλο που η θεωρία τείνει να προσαρμόζεται σε σύνολα επαρκώς ορισμένων θεμάτων. Η «νέα αρχαιολογία» διαμόρφωσε το δικό της θεωρητικό πλαίσιο, σε σχέση όμως με δύο περιοχές σκέψης. Η μία περιοχή σχετίζεται με την επιστημολογία και η άλλη με ζητήματα ιστορικών περιόδων και διαδικασιών κοινωνικού μετασχηματισμού. Το μαρξιστικό, το μεταδιαδικαστικό και άλλα συναφή ρεύματα της αρχαιολογικής σκέψης ανέδειξαν και μια περιοχή σκέψης πολιτικής φύσης, που ενυπήρχε στην αρχαιολογία από τις πρώτες στιγμές της θεωρητικής της διαμόρφωσης.

Ένα δεύτερο ζήτημα σχετικό με τη φύση της αρχαιολογίας είναι η έλλειψη αντιστοιχίας ανάμεσα στα τυπικά δόγματα της θεωρίας και τις θεμελιώδεις λογικές υποθέσεις της αρχαιολογικής πρακτικής. Για να το θέσουμε απλούστερα, πολύ συχνά περιπτώσεις μελέτης που κατασκευάζονται για την υποστήριξη μιας ιδιαίτερης θεωρητικής θέσης, δε συμφωνούν με την ίδια τη θέση. Ωστόσο, γενικά θεωρούμενη, η αρχαιολογική θεωρία επιχειρεί να συστηματοποιήσει και να δώσει συνοχή σε ένα σύνολο γνώσεων. Με αυτόν τον τρόπο διαμορφώνεται σε επιστημονικό κλάδο, ορίζοντας το επιστημολογικό της πλαίσιο και θεμελιώνοντας παράλληλα το πρακτικό αρχαιολογικό έργο.

Με τον γενικό όρο αρχαιολογική θεωρία αναφερόμαστε σε διάφορα διανοητικά σχήματα, μέσω των οποίων οι αρχαιολόγοι ερμηνεύουν αρχαιολογικά δεδομένα. Η αρχαιολογική θεωρία λειτουργεί ως εφαρμογή της φιλοσοφίας της επιστήμης στην αρχαιολογία, και περιγράφεται ενίοτε ως φιλοσοφία της αρχαιολογίας⁸⁶. Δεν υπάρχει, ωστόσο, μία και μοναδική θεωρία της αρχαιολογίας, αλλά πολλές, καθώς διαφορετικοί αρχαιολόγοι θεωρούν ότι οι πληροφορίες πρέπει να ερμηνεύονται με διαφορετικούς τρόπους⁸⁷. Στην ιστορία της ανάπτυξης της αρχαιολογίας, εμφανίστηκαν διάφορες τάσεις, που έφθασαν σε ένα επίπεδο κορύφωσης πριν την τελική τους εξαφάνιση. Οι διαφορετικές

⁸⁶ Krieger 2006, 3.

⁸⁷ Trigger 2007, 7.

αρχαιολογικές θεωρίες διαφέρουν ως προς τον στόχο που θέτουν και τον τρόπο με τον οποίο προσπαθούν να τον επιτύχουν. Ορισμένες σχολές σκέψης, όπως η διαδικαστική, υποστηρίζουν ότι οι αρχαιολόγοι μπορούν να αναπτύξουν ακριβείς και αντικειμενικές πληροφορίες για τις κοινωνίες του παρελθόντος, εφαρμόζοντας την επιστημονική μέθοδο στις έρευνές τους. Στον αντίποδα, η μεταδιαδικαστική αρχαιολογία, αμφισβητεί αυτή την αντικειμενικότητα και θεωρεί ότι τα αρχαιολογικά δεδομένα αλλοιώνονται από τον υποκειμενικό παράγοντα που υπεισέρχεται στην ανθρώπινη ερμηνεία για τις κοινωνίες του παρελθόντος, καθιστώντας την υποκειμενική⁸⁸.

Άλλες αρχαιολογικές θεωρίες, όπως η μαρξιστική, ερμηνεύουν τα αρχαιολογικά δεδομένα στο κοινωνικό και οικονομικό πλαίσιο. Εκπρόσωπος της μαρξιστικής αρχαιολογίας, ο Vere Gordon Childe άρχισε να διερευνά τις αρχαίες κοινωνίες από την προοπτική της οικονομίας που ανέπτυξαν, απορρίπτοντας ταυτόχρονα την πολιτισμική-ιστορική αρχαιολογία στα τέλη της δεκαετίας του 1920⁸⁹. Σύμφωνα με τους Colin Renfrew και Paul Bahn, «μετά την έξαρση της θεωρητικής συζήτησης που ακολούθησε τον αρχικό αντίκτυπο της νέας αρχαιολογίας, αναζωπυρώθηκε το ενδιαφέρον να εφαρμοστούν στην αρχαιολογία ορισμένες από τις αντιλήψεις του προηγούμενου έργου του Karl Marx, που επανεξετάστηκε από Γάλλους ανθρωπολόγους στη δεκαετία του 1960 και του 1970»⁹⁰. Οι μαρξιστές αρχαιολόγοι γενικά πιστεύουν ότι ο υφιστάμενος διπολισμός μεταξύ της διαδικαστικής και μεταδιαδικαστικής αρχαιολογίας είναι αντίθεση εγγενής στην παραγωγή γνώσης και είναι σύμφωνη με μια διαλεκτική κατανόηση του κόσμου. Επίσης, είναι αρκετοί εκείνοι που πιστεύουν πως η πολικότητα στο πλαίσιο της ανθρωπολογικής επιστήμης και των συναφών κλάδων τροφοδοτεί τα ερωτήματα που προωθούν την πρόοδο στην αρχαιολογική θεωρία και τη γνώση. Αυτή η συνεχής διασύνδεση και σύγκρουση μεταξύ των άκρων υποκειμενικό και αντικειμενικό πιστεύεται ότι οδηγεί σε συνεχή οικοδόμηση μιας εικόνας του παρελθόντος από τους ερευνητές⁹¹.

Η αρχαιολογική θεωρία έχει αυξανόμενη απήχηση, ιδιαίτερα στη Β. Αμερική και τη Μ. Βρετανία. Ωστόσο, σε αρκετές χώρες, ανάμεσά τους και η

⁸⁸ Trigger 2007, 1.

⁸⁹ Trigger 2007, 322.

⁹⁰ Renfrew and Bahn 2004, 179.

⁹¹ McGuire 2008, 18.

Ελλάδα, υφίσταται προς το παρόν χάσμα μεταξύ αρχαιολογικής θεωρίας και πρακτικής και οι αρχαιολόγοι πεδίου ενίοτε αντιμετωπίζουν με δυσπιστία θεωρητικά ζητήματα, στρεφόμενοι αποκλειστικά στα αρχαιολογικά δεδομένα, χωρίς όμως να αντιλαμβάνονται ότι ακόμη και η αποχή από τη θεωρία είναι θεωρητική θέση *per se*. Ωστόσο, η θεωρία είναι τρόπος για την κατανόηση και την ερμηνεία του παρελθόντος. Οι θεωρητικές προσεγγίσεις που αναπτύχθηκαν τα τελευταία εκατόν πενήντα χρόνια, ιδίως μετά τη δεκαετία του '60 διαμόρφωσαν σχολές αρχαιολογικές σκέψης επηρεασμένες σαφώς από τα διανοητικά ρεύματα της εποχής τους⁹².

Η αρχαιολογία, μετά το τέλος της αρχαιοδιφίας στα τέλη του 19ου αι., ανέπτυξε μεθόδους ταξινόμησης της αρχαιολογικής μαρτυρίας, όπως είναι το Σύστημα των Τριών Εποχών του C.S. Thomsen, και η λεπτομερής καταγραφή των δεδομένων του αρχαιολογικού αρχείου που εφήρμοσε ο Pitt-Rivers⁹³. Συνδυασμένες με την ανάπτυξη της επιστήμης της γεωλογίας, ιδιαίτερα της γεωλογικής στρωματογραφίας και της καθιέρωσης της θεωρίας του ομοιομορφισμού⁹⁴, οι νέες μέθοδοι έδωσαν στο παρελθόν μεγαλύτερο χρονικό βάθος από εκείνο που περιέγραφε η Βίβλος. Η πρώτη δημοσίευση του έργου του Δαρβίνου *Περί της Καταγωγής των Ειδών*, το 1859 συνέβαλε, σε συνδυασμό με τις νέες μεθόδους, στη διαμόρφωση ενός επιστημονικού ρεύματος πάνω στο οποίο θεμελιώθηκαν οι πεποιθήσεις της αρχαιολογίας για το απώτατο παρελθόν της ιστορίας της ανθρωπότητας.

2.2. Πολιτισμική εξελικτική αρχαιολογία

Κατά την περίοδο 1860-70, η αρχαιολογία τελούσε υπό την άμεση επίδραση της εθνολογίας. Στην προσπάθειά της να ερμηνεύσει τους αρχαίους πολιτισμούς και την πολιτισμική αλλαγή, υιοθέτησε την προσέγγιση του πολιτισμικού εξελικτισμού. Ο πολιτισμικός εξελικτισμός, ιστορικά γνωστός και ως κοινωνικός πολιτισμικός εξελικτισμός, αναπτύχθηκε αρχικά τον 19ο αιώνα από

⁹² Κουκουζέλη κ.ά. 2003, 246.

⁹³ Renfrew & Bahn 2001, 32.

⁹⁴ Αναζητώντας καθόλα φυσικές ερμηνείες της γεωλογικής ιστορίας της γης οι James Hutton στα τέλη του 18ου αι. και Charles Lyell κατά τον 19ο αι. εντόπισαν παρατηρήσιμες διαδικασίες, χρήσιμες για τη διαχρονική ερμηνεία γεωλογικών φαινομένων. Βλ. Cameron 1993, 42-49.

ανθρωπολόγους που επηρεάστηκαν από την έρευνα του Δαρβίνου για την εξέλιξη. Σήμερα, η πολιτισμική εξέλιξη έχει γίνει η βάση για ένα αυξανόμενο πεδίο επιστημονικής έρευνας στις κοινωνικές επιστήμες, συμπεριλαμβανομένης της ανθρωπολογίας, της οικονομίας και της ψυχολογίας. Στο παρελθόν θεωρείτο ότι η κοινωνική αλλαγή προέκυψε από βιολογικές προσαρμογές, αλλά σήμερα οι ανθρωπολόγοι δέχονται ότι οι κοινωνικές αλλαγές προκύπτουν ως αποτέλεσμα ενός συνδυασμού κοινωνικών, εξελικτικών και βιολογικών επιδράσεων. Τα τελευταία χρόνια υπήρξε σύγκλιση του συνόλου των συναφών θεωριών, με στόχο την πολιτισμική εξέλιξη ως ενιαίο επιστημονικό κλάδο⁹⁵.

Με την αντίληψη του κοινωνικού εξελικτισμού του 19ου αιώνα, σύμφωνα με τον οποίο θεωρούνταν ως ιδιαίτερα πρωτόγονοι οι νομαδικοί λαοί, διαφώνησαν οι θεωρητικοί της αρχαιολογίας, απορρίπτοντας εντέλει τα μοντέλα που πρότεινε η εθνολογία⁹⁶. Σύμφωνα με την εξελικτική προσέγγιση οι ανθρώπινες κοινωνίες εξελίσσονται γραμμικά προς τον πολιτισμό, περνώντας μια σειρά σταδίων, από το απλό και κατώτερο στο σύνθετο και ανώτερο. Πρόκειται για την εξελικτική θεωρία της μονογραμμικής πολιτισμικής εξέλιξης⁹⁷, που προτάθηκε στα τέλη του 19ου αιώνα και προέβλεπε πως όλες τις ανθρώπινες κοινωνίες εξελίσσονται κατά μήκος μιας κοινής τροχιάς από απλές κυνηγετικές και τροφосуλλεκτικές κοινότητες σε εγγράμματος πολιτισμούς. Όλες οι κοινωνίες σύμφωνα με τη συγκεκριμένη θεώρηση διατρέχουν την ίδια βασική ακολουθία σταδίων, με πιθανές διαφορές ως προς την ταχύτητα μετάβασής τους⁹⁸.

Σύμφωνα με αυτή θεώρηση, οι αρχαϊκές κοινωνίες αντιπροσωπεύουν διαφορετικά βήματα σε μια εξελικτική κλίμακα, που κυμαίνεται από τεχνολογικά και οικονομικά απλούς ή πρωτόγονους έως πολύπλοκους εξελιγμένους πολιτισμούς⁹⁹. Το αυξανόμενο επίπεδο πολυπλοκότητας θεωρήθηκε εγγενές στην πολιτισμική εξέλιξη και προτάθηκαν διαφορετικά συστήματα ταξινόμησης, για παράδειγμα, από τους Morgan και Taylor και άλλους¹⁰⁰. Αυτά τα συστήματα ταξινόμησης ενώθηκαν υπό την τελεολογική έννοια του καθορισμένου

⁹⁵ Mesoudi et al 2006, 329–347.

⁹⁶ McGranaghan 2017, 2.

⁹⁷ Unilinear cultural evolution. *Oxford Reference*, διαθέσιμο online.

⁹⁸ Κουκουζέλη κ.ά. 2003, 251.

⁹⁹ Tehrani and Collard 2002, 443-463.

¹⁰⁰ Tehrani 2010, 169-182.

εξελικτικού στόχου, καθιστώντας δυνατή την εξελικτική κατάταξη των κοινωνιών. Επίσης, υποστηρίχθηκε ότι μια τέτοια κατάταξη είχε ηθικές διαστάσεις, καθώς οι «προηγμένες» κοινωνίες θεωρούνταν καλύτερες από τις «πρωτόγονες» κοινωνίες¹⁰¹.

Όπως είναι φυσικό τέτοιου είδους πολιτισμικά εξελικτικά σχήματα αρθρώνονταν καλά με την προσπάθεια εκδυτισμού των λαών των αποικιών. Η σχολή της πολιτισμικής εξέλιξης του 19ου αιώνα ήταν μια προσπάθεια επαγωγικής γενίκευσης του εμπειρικού αρχείου, προκειμένου να γίνουν κατανοητές οι ουσιαστικές διαφορές μεταξύ των πολιτισμών. Βέβαια, η επιστημονική έρευνα στην προκειμένη περίπτωση μπορούσε να συνδυαστεί με πολιτικές αποικιακές φιλοδοξίες. Είναι σαφές ότι το σημείο εκκίνησης δεν ήταν ένα συνεκτικό θεωρητικό εξελικτικό πλαίσιο και ακόμη και μετά τη δημοσίευση του έργου του Δαρβίνου σχετικά με την προέλευση των ειδών, οι υποστηρικτές του πολιτισμικού εξελικτισμού του 19ου και του 20ού αιώνα διατήρησαν μια εντελώς μη δαρβίνεια άποψη για την εξέλιξη¹⁰².

Η πολιτισμική εξέλιξη δέχτηκε ισχυρή κριτική τους τελευταίους δύο αιώνες, καθώς η θεωρία της εξέλιξης του Morgan υποδηλώνει ότι όλοι οι πολιτισμοί ακολουθούν το ίδιο βασικό μοτίβο. Ωστόσο, ο ανθρώπινος πολιτισμός δεν είναι γραμμικός και οι διαφορετικοί πολιτισμοί αναπτύσσονται σε διαφορετικές κατευθύνσεις και με διαφορετικούς ρυθμούς και δεν είναι ικανοποιητική η υπόθεση ότι οι πολιτισμοί αναπτύσσονται με τον ίδιο τρόπο. Μια περαιτέρω κριτική της πολιτισμικής εξέλιξης είναι γνωστή ως «ανθρωπολογία της πολυθρόνας». Το όνομα προκύπτει από το γεγονός ότι πολλοί από τους ανθρωπολόγους που προωθούσαν τέτοιες θεωρίες, δεν είχαν δει από πρώτο χέρι τους πολιτισμούς που μελετούσαν. Η έρευνα και τα στοιχεία που συλλέχθηκαν, έγιναν από εξερευνητές και ιεραπόστολους και όχι από τους ίδιους τους ανθρωπολόγους¹⁰³. Η πολιτισμική εξέλιξη επικρίνεται, επίσης, επειδή είναι εθνοκεντρική και θεωρεί ότι οι πολιτισμοί μιμούνται τον δυτικό πολιτισμό. Υπό την επήρεια του εθνοκεντρισμού και δη του ευρωπαϊκού εθνοκεντρισμού, οι

¹⁰¹ Riede et al 2012, 100.

¹⁰² Bettinger 1991, 23-28.

¹⁰³ Sera-Shriar 2013, 26-40.

πρωτόγονες κοινωνίες θεωρούνται κατώτερες του πολιτισμικού επιπέδου των δυτικών κοινωνιών¹⁰⁴

Η πολιτισμική εξελικτική αρχαιολογία περιθωριοποιήθηκε με την εμφάνιση ενός νέου θεωρητικού πλαισίου στα τέλη του 19ου και τις αρχές του 20ού αιώνα. Το νέο ρεύμα της πολιτισμικής ιστορικής αρχαιολογίας αντικατέστησε τον εξελικτικό με τον ιστορικό προσανατολισμό. Εκκινώντας από την πολιτισμική ιστορική προσέγγιση, ουσιαστικά αντιπαρέταξε στην ιδέα της πολιτισμικής παγκοσμιότητας την ιδέα της πολιτισμικής ιδιαιτερότητας. Ο Webster παρατηρεί ότι το καθοριστικό χαρακτηριστικό της πολιτισμικής ιστορικής αρχαιολογίας ήταν οι «δηλώσεις της, που αποκαλύπτουν κοινές αντιλήψεις για τη φύση των αρχαίων πολιτισμών, για την ποιότητά τους, για το πώς σχετίζονται με το υλικό αρχείο και για τον τρόπο με τον οποίο οι αρχαιολόγοι θα μπορούσαν πραγματικά να τις μελετήσουν»¹⁰⁵. Ο ίδιος σημειώνει ότι το δεύτερο χαρακτηριστικό γνώρισμα της πολιτισμικής ιστορικής σκέψης ήταν η έμφαση που δόθηκε στην ταξινόμηση και τις τυπολογίες¹⁰⁶.

2.3. Πολιτισμική ιστορική αρχαιολογία

Η πολιτισμική-ιστορική ιδιαιτερότητα ως σχολή σκέψης στη Β. Αμερική, σε στενή σχέση με τον Franz Boas και την προσέγγισή του στην ανθρωπολογία, απέρριψε το πολιτιστικό εξελικτικό μοντέλο, θεωρώντας ότι κάθε κοινωνία είναι μια συλλογική εκπροσώπηση του μοναδικού ιστορικού παρελθόντος της. Ο Boas απέρριψε τον παράλληλο εξελικτισμό, την ιδέα δηλαδή ότι όλες οι κοινωνίες βρίσκονται στον ίδιο δρόμο και ότι φθάνουν στο συγκεκριμένο επίπεδο ανάπτυξης τους με τον ίδιο τρόπο. Η πολιτισμική-ιστορική ιδιαιτερότητα έδειξε ότι οι κοινωνίες θα μπορούσαν να φτάσουν στο ίδιο επίπεδο πολιτισμικής ανάπτυξης μέσω διαφορετικών εξελικτικών δρόμων¹⁰⁷.

Ο ιστορικός της αρχαιολογίας Bruce Trigger θεωρεί την ανάπτυξη της πολιτισμικής ιστορικής αρχαιολογίας ως «απάντηση στην αυξανόμενη συνειδητοποίηση της γεωγραφικής μεταβλητότητας του αρχαιολογικού αρχείου»

¹⁰⁴ Hales and Edmonds 2018, 1-27.

¹⁰⁵ Webster 2008, 12.

¹⁰⁶ Webster 2008, 13.

¹⁰⁷ Boas 1920, 311-321.

σε μια εποχή μάλιστα που η πίστη στην πολιτισμική εξελικτική αρχαιολογία άρχισε να φθίνει στη δυτική και κεντρική Ευρώπη¹⁰⁸. Κατά τον 19ο αιώνα συλλέχθηκε μεγάλη ποσότητα αρχαιολογικού υλικού στην ευρωπαϊκή ήπειρο. Αυτό είχε ως αποτέλεσμα την αυξανόμενη συνειδητοποίηση μιας μεγάλης μεταβλητότητας στην τυπολογία των τεχνέργων, των κατασκευών και των κτισμάτων που αποκαλύφθηκαν και που δεν μπορούσε βάσει γεωγραφικής κατανομής να ερμηνευθεί από τα προϋπάρχοντα εξελικτικά παραδείγματα¹⁰⁹.

Κατ' αυτόν τον τρόπο ο πολιτισμός ως έννοια στην αρχαιολογία καθορίστηκε ως σύνολο χαρακτηριστικών ιδιαίτερων πολιτισμικών ή εθνικών ομάδων του παρελθόντος, που ακολουθούσαν τον δικό τους τρόπο ζωής και τις δικές τους ιδέες, μεταλαμπαδεύοντάς τις από γενεά σε γενεά μέσω της παράδοσης. Παράλληλα με την ιδέα του αρχαιολογικού πολιτισμού, προβλήθηκε στη δεκαετία του 1880 η αντίληψη ότι η συμπεριφορά ενός λαού εξαρτάται άμεσα από τη βιολογική του σύνθεση. Θεωρώντας ότι η πολιτισμική αλλαγή είναι αντίθετη προς την ανθρώπινη φύση δεν είναι δυνατόν προέλθει από τα ίδια τα άτομα, αλλά αποκλειστικά από εξωτερικούς παράγοντες, όπως η διάδοση νέων ιδεών από μια εθνική ομάδα σε μια άλλη και η μετανάστευση, που οδηγεί σε αντικαταστάσεις λαών και πολιτισμών¹¹⁰.

Αυτή η διάδοση έχει οριστεί ως «μετάδοση μέσω επαφής, και ως εκ τούτου διάδοση των κανόνων από το ένα άτομο στο άλλο και ευρύτερα από τους ανθρώπους μιας τοπικής παράδοσης στην άλλη». Το ενδιαφέρον σημείο όσον αφορά στη διάχυση, είναι η αλληλεπίδραση, η επαφή μεταξύ ατόμων και κοινωνικών ομάδων και ο μηχανισμός με τον οποίο τα πολιτισμικά χαρακτηριστικά διαχέονται τόσο μέσω εσωτερικών όσο και εξωτερικών επαφών¹¹¹.

Σημαντική προσωπικότητα της μετάβασης από το εξελικτικό στο ιστορικό πλαίσιο θεωρείται ο Oscar Montelius. Ανέπτυξε τυπολογική μέθοδο, βάσει της οποίας συνέκρινε και ταξινόμησε σε ομάδες διαφόρους τύπους τεχνέργων, Χρησιμοποιώντας ως βάση τις διαμορφωμένες ομάδες, κατασκεύασε πολιτισμικές

¹⁰⁸ Trigger 2007, 211.

¹⁰⁹ Trigger 2007, 215.

¹¹⁰ Κουκουζέλη κ.ά. 2003, 255.

¹¹¹ Storey 2011, 7-24.

χρονολογικές σειρές ή ακολουθίες¹¹² για το σύνολο του ευρωπαϊκού χώρου, αναπτύσσοντας με αυτόν τον τρόπο την ευρωπαϊκή προϊστορία. Ερμηνεύοντας την πολιτισμική εξέλιξη της Ευρώπης, θεώρησε πως υπήρξε πολιτισμική διέγερση από την Εγγύς Ανατολής προς δυσμάς με διαδοχικά κύματα διάδοσης και μετανάστευσης¹¹³. Άλλη σημαντική προσωπικότητα στο πλαίσιο της πολιτισμικής ιστορικής αρχαιολογίας υπήρξε ο Vere Gordon Childe, ο πρώτος που απέρριψε το ιστορικό εξελικτικό πλαίσιο και υιοθέτησε το πολιτισμικό ιστορικό ως μόνο κατάλληλο για τη μελέτη και ερμηνεία της ποικιλομορφίας των αρχαίων πολιτισμών. Ήδη από το 1925 όρισε και εφάρμοσε συστηματικά σε μια γεωγραφική περιοχή, που κάλυπτε το σύνολο της ευρωπαϊκής ηπείρου την έννοια του αρχαιολογικού πολιτισμού. Ο Childe άσκησε κριτική στην εξελικτική αρχαιολογία, θεωρώντας ότι οι αρχαιολόγοι που την ασπάζονταν, έδιναν μεγαλύτερη έμφαση στα τέχνηρα από ό,τι στους ανθρώπους που τα κατασκεύασαν¹¹⁴.

Το κύριο μέλημα της πολιτισμικής ιστορικής ή κανονιστικής αρχαιολογίας ήταν η συστηματική περιγραφή και ταξινόμηση βάσει κανόνων στον χώρο και τον χρόνο ενός πλήθους αρχαιολογικών δεδομένων με βάση τη μορφή τους και η κατάρτιση μιας μακράς χρονολογικής ακολουθίας πολιτισμών σε μια δεδομένη γεωγραφική περιοχή. Στο πλαίσιο της συμπεριφοράς η κανονιστική έννοια του πολιτισμού σε μια δεδομένη κοινωνία, υποστηρίζει πως τα πρότυπα συμπεριφοράς είναι το αποτέλεσμα της τήρησης ενός συνόλου. Οι κανόνες προδιαγράφουν τα όρια των ανεκτών συμπεριφορών και τα κατάλοιπα αρχαιολογικών πολιτισμών αντιπροσωπεύουν προγενέστερα πρότυπα συμπεριφοράς¹¹⁵. Η κεραμική, για παράδειγμα, μπορεί να θεωρηθεί αντανάκλαση των κανόνων που διέπουν την τεχνολογική συμπεριφορά μιας αρχαίας κοινωνίας¹¹⁶.

¹¹² Ο Montelius βελτίωσε την έννοια της τυπολογίας εισάγοντας μέθοδο σχετικής χρονολόγησης. Έτσι η τυπολογία έγινε διαδικασία αναγνώρισης μιας χρονολογίας με την οργάνωση υλικών υπολειμμάτων μιας πολιτιστικής παράδοσης με εκείνη τη σειρά που παράγει τον συνεπέστερο σχεδιασμό των πολιτισμικών τους χαρακτηριστικών. Βλ. Gräslund 1987, 679.

¹¹³ Trigger 2007a, 80.

¹¹⁴ Trigger 2007, 247.

¹¹⁵ Sharer and Ashmore 2003, 67.

¹¹⁶ Fagan 1997, 91.

Ωστόσο, η κανονιστική αρχαιολογία προσέφερε περιγραφή και όχι ερμηνεία των πολιτισμών και της πολιτισμικής αλλαγής¹¹⁷. Το κανονιστικό ως κυρίαρχο μοντέλο της πολιτισμικής ιστορικής αρχαιολογίας ορίζει τον πολιτισμό ως σύνολο κανόνων. Αυτοί οι κανόνες είναι ιδέες για όλες τις πτυχές μιας κοινωνίας. Στη συνέχεια, υποθέτει ότι οι κανόνες εκφράζονται στα υλικά κατάλοιπα¹¹⁸ και θεωρεί τον πολιτισμό αμετάβλητο εξαιτίας της ύπαρξης πολιτισμικών κανόνων. Συνεπώς, ερμηνεύει την αλλαγή ως αποτέλεσμα εξωτερικών αιτιών, όπως η διάδοση νέων ιδεών και πολιτισμικών χαρακτηριστικών από κάποιο πολιτισμό σε λιγότερο ανεπτυγμένους περιφερειακούς πολιτισμούς μέσω των εμπορικών δρόμων ή άλλων διεπαφών. Επιπλέον, ερμηνεύει την αλλαγή με μετακινήσεις πληθυσμών εξαιτίας κάποιας μετανάστευσης ή εισβολής, μέσω της οποίας εισάγονται νέα στοιχεία από έναν πολιτισμό σε κάποιον άλλο. Μια τέτοια θεώρηση, όμως, παραβλέπει τις εγγενείς δυνατότητες ενός πολιτισμού να παράγει πολιτισμική αλλαγή και μετατοπίζει την ερμηνεία της πολιτισμικής αλλαγής σε ένα διαφορετικό επίπεδο αιτιών, που αφεαυτού χρειάζεται εξίσου κάποια ερμηνεία¹¹⁹.

2.4. Διαδικαστική αρχαιολογία

Το ζήτημα της ερμηνείας της πολιτισμικής αλλαγής αντιμετωπίζεται διαφορετικά με την ανάδυση ενός νέου ρεύματος αρχαιολογικής σκέψης, αυτό της διαδικαστικής ή νέας αρχαιολογίας. Εκκινώντας στη δεκαετία του 1960, ο Binford και άλλοι Αμερικανοί αρχαιολόγοι ανέπτυξαν τη θεωρητική προσέγγιση που ονομάζεται «διαδικαστική». Σύμφωνα με αυτή την προσέγγιση, τα αρχαιολογικά κατάλοιπα χρειάζεται να διερευνώνται υπό το πρίσμα δυναμικών πολιτισμικών και φυσικών διεργασιών μέσω των οποίων έφθασαν στην κατάσταση, στην οποία βρίσκονται από τους αρχαιολόγους και όχι ως στατικά τελικά προϊόντα, όπως συμβαίνει όταν χρησιμοποιείται η τυπολογική προσέγγιση¹²⁰. Σκοπός της διαδικαστικής αρχαιολογίας ήταν η ανάπτυξη μεθόδων για την κατανόηση του σχηματισμού του αρχαιολογικού αρχείου και η

¹¹⁷ Johnson 2010, 19.

¹¹⁸ Johnson 2010, 16.

¹¹⁹ Κωτσάκης 1999, 11-23.

¹²⁰ Pelegrin 2015, 260-264.

βελτίωση των προσεγγίσεων για τη συμπερίληψη της ιδέας της συμπεριφοράς από τις παρατηρήσεις του αρχείου. Αυτό οδήγησε σε αυξανόμενο ενδιαφέρον για τις διαδικασίες σχηματισμού των αρχαιολογικών τόπων (γεωαρχαιολογία), την εθνοαρχαιολογία και τη χρήση της αναλογίας. Σε αυτή τη διαδικαστική προοπτική, ο πολιτισμός θεωρήθηκε σύστημα αλληλένδετων συστατικών¹²¹.

Το θεωρητικό πλαίσιο στην καρδιά της διαδικαστικής αρχαιολογίας είναι η πολιτισμική εξέλιξη, υπό την οπτική γωνία της νεοεξελικτικής αντίληψης για την πολυγραμμική και όχι τη μονογραμμική πολιτισμική εξέλιξη, που σχετίζεται με τη γενική θεωρία της εξέλιξης. Οι διαδικαστικοί αρχαιολόγοι θεωρούν ότι μπορούν να κατανοήσουν τα αρχαιότερα πολιτισμικά συστήματα μέσω των καταλοίπων που άφησαν πίσω τους. Με άλλα λόγια υιοθετούν τη θεωρία του Leslie White ότι ο πολιτισμός μπορεί να οριστεί ως το εξωσωματικό μέσο περιβαλλοντικής προσαρμογής για τον άνθρωπο¹²². Διερευνούν, δηλαδή, την πολιτισμική προσαρμογή στην περιβαλλοντική αλλαγή και όχι τη σωματική προσαρμογή, την οποία αντιμετωπίζουν οι εξελικτικοί βιολόγοι. Αυτή η εστίαση στην περιβαλλοντική προσαρμογή βασίζεται στην πολιτισμική οικολογία και τις ιδέες της πολυγραμμικής εξέλιξης ανθρωπολόγων όπως ο Julian Steward. Η πολυγραμμική εξέλιξη αφορά στην ιστορική ανασυγκρότηση. Ενδιαφέρεται για συγκεκριμένους πολιτισμούς και ασχολείται μόνο με τις περιορισμένες παραλλαγές της μορφής, της λειτουργίας και της χρονολογικής ακολουθίας που έχουν εμπειρική ισχύ¹²³.

Ως εξωσωματική προσαρμογή, ο πολιτισμός καθορίζεται από περιβαλλοντικούς περιορισμούς¹²⁴. Βάσει αυτού οι διαδικαστικοί αρχαιολόγοι υποστηρίζουν ότι η πολιτισμική αλλαγή συμβαίνει μέσα σε ένα προβλέψιμο πλαίσιο και επιδιώκουν να την κατανοήσουν με την ανάλυση των συνιστωσών της. Επιπλέον, θεωρούν ότι δεδομένου ότι το πλαίσιο αυτό είναι προβλέψιμο, τότε η επιστήμη είναι το κλειδί για την κατανόηση του τρόπου με τον οποίο αλληλεπιδρούν οι συνιστώσες με το πολιτισμικό σύνολο¹²⁵. Αυτό σημαίνει ότι οι πολιτισμικές αλλαγές καθοδηγούνται από εξελικτικές διαδικασίες στην πολιτισμική ανάπτυξη, οι οποίες είναι προσαρμοστικές σε σχέση με το περιβάλλον

¹²¹ Binford 1968, σελ. 16-23

¹²² White 1959, 8.

¹²³ Steward 1955, 18-19

¹²⁴ Flatman 2015, 17.

¹²⁵ Trigger 1989, 289.

και επομένως όχι μόνο κατανοητές αλλά και επιστημονικά προβλέψιμες, όταν κατανοηθεί η αλληλεπίδραση των μεταβλητών. Δυνητικά, λοιπόν, είναι κανείς σε θέση να ανακατασκευάσει πλήρως αυτές τις «πολιτισμικές διαδικασίες», εξ ου και ο προσδιορισμός διαδικαστική αρχαιολογία¹²⁶.

Όπως φαίνεται, η διαδικαστική προσέγγιση δε γεννήθηκε *ex nihilo*, αλλά συνέχισε και ενέτεινε δύο θεωρητικές τάσεις, οι οποίες είχαν αρχίσει να αναπτύσσονται στη βρετανική και τη βορειοαμερικανική αρχαιολογία, παράλληλα με το ρεύμα της πολιτισμικής ιστορικής προσέγγισης, ήδη από τη δεκαετία του 1930. Πρόκειται για τη λειτουργιστική και τη διαδικαστική προσέγγιση που προαναφέρθηκε, οι οποίες μελετούν τη λειτουργία και την αλλαγή των αρχαίων πολιτισμών. Ειδικότερα, η λειτουργιστική προσέγγιση επηρεάστηκε από:

α) την οικολογική προσέγγιση της σκανδιναβικής αρχαιολογίας που συνέδεσε τον πολιτισμό με το περιβάλλον του. Η αρχαιολογία της οικολογίας και των οικισμών ενθάρρυνε τη μελέτη τόσο των προϊστορικών πολιτισμών σε συγκεκριμένα χρονικά σημεία όσο και του τρόπου με τον οποίο άλλαξαν οι πολιτισμοί στο πέρασμα του χρόνου¹²⁷. Σύμφωνα με τον Butzer, η αρχαιολογία της οικολογίας σχετίζεται με τη μελέτη των σχέσεων πολιτισμού και περιβάλλοντος και ο τελικός στόχος της είναι η κατανόηση της ανθρώπινης οικολογίας προϊστορικών κοινοτήτων¹²⁸.

β) τη σοβιετική αρχαιολογία, που εστίασε στην κοινωνιολογική ερμηνεία των αρχαιολογικών δεδομένων, στη μελέτη των οικισμών και την εσωτερική λειτουργία των πολιτισμών. Οι λειτουργιστικές και διαδικαστικές ερμηνείες των αρχαιολογικών δεδομένων είχαν από μακρού συνδεθεί με μελέτες των σχέσεων μεταξύ των πολιτισμών και του περιβάλλοντός τους και με το πώς κατασκευάστηκαν και χρησιμοποιήθηκαν τα τέχνηρα του αρχαιολογικού αρχείου. Εκκινώντας στα τέλη της δεκαετίας του 1920, οι σοβιετικοί αρχαιολόγοι προσπάθησαν να επινοήσουν τρόπους για να κατανοήσουν την κοινωνική, οικονομική και πολιτική συμπεριφορά από το αρχαιολογικό αρχείο. Αυτό που ήταν εντελώς νέο ήταν η αποφασιστικότητά τους να καταλάβουν πώς άλλαξαν τα κοινωνικά και πολιτισμικά συστήματα από την άποψη της δικής τους εσωτερικής δυναμικής. Ωστόσο, η ανάπτυξη της σοβιετικής αρχαιολογίας επλήγη

¹²⁶ Trigger 1989, 289.

¹²⁷ Trigger 207, 385.

¹²⁸ Butzer 1980, 418.

από ακραία ιδεολογική συσσώρευση, η οποία περιόρισε το εύρος των αναλυτικών τεχνικών που θα μπορούσαν να χρησιμοποιηθούν για την οικοδόμηση πολιτιστικών χρονολογιών και την ερμηνεία της αλλαγής στο αρχαιολογικό αρχείο. Στη Δύση, ο πολλαπλασιασμός και η αυξανόμενη πολυπλοκότητα λειτουργιστικών απόψεων, που αντιπροσώπευαν σημαντική τάση στη βρετανική και αμερικανική ανθρωπολογία ήδη από τη δεκαετία του 1920, ενθάρρυναν τους αρχαιολόγους να θεωρούν τους προϊστορικούς πολιτισμούς ως εσωτερικά διαφοροποιημένους και σε κάποιο βαθμό ως ολοκληρωμένους τρόπους ζωής¹²⁹.

γ) τη θεωρία του λειτουργισμού (φονξιοναλισμού) της κοινωνικής ανθρωπολογίας¹³⁰.

Η ανάπτυξη λειτουργιστικών και διαδικαστικών προσεγγίσεων στα αρχαιολογικά δεδομένα οδήγησε στη σταδιακή αντικατάσταση του στείρου προβληματισμού της πολιτισμικής ιστορικής αρχαιολογίας για την εθνότητα από ένα νέο ζωτικό ενδιαφέρον για τον τρόπο λειτουργίας και αλλαγής των προϊστορικών πολιτισμών. Το μειούμενο ενδιαφέρον για τη μελέτη του πολιτισμού και το αυξανόμενο ενδιαφέρον για τη συμπεριφορά δεν ήταν μόνο συμβατό με τις τάσεις άλλων κοινωνικών επιστημών, αλλά και προκάλεσε διαρκείς αλλαγές στην αρχαιολογία¹³¹.

Ο δομικός λειτουργισμός, ή απλά λειτουργισμός, είναι «ένα πλαίσιο για την οικοδόμηση της θεωρίας, που βλέπει την κοινωνία ως σύνθετο σύστημα, του οποίου τα μέρη συνεργάζονται για την προώθηση της αλληλεγγύης και της σταθερότητας. Η συγκεκριμένη προσέγγιση εξετάζει την κοινωνία μακροσκοπικά, επικεντρωμένη σε μεγάλο βαθμό στις κοινωνικές δομές που τη διαμορφώνουν στο σύνολό της, βάσει της πεποίθησης ότι εξελίσσεται ως οργανισμός. Προς τούτους διερευνά τόσο την κοινωνική διάρθρωση όσο και τις κοινωνικές λειτουργίες. Ο λειτουργισμός διερευνά συνολικά την κοινωνία ως προς τη λειτουργία των συστατικών της στοιχείων, δηλαδή των κανόνων, των εθίμων, των παραδόσεων και των θεσμών. Μια κοινή αναλογία, γνωστή από τον Herbert Spencer, παρουσιάζει τα τμήματα της κοινωνίας ως «όργανα» που συνεργάζονται για την καλή λειτουργία του «σώματος» στο σύνολό του¹³².

¹²⁹ Trigger 2007, 384.

¹³⁰ Κουκουζέλη κ.ά. 2003, 260.

¹³¹ Trigger 2007, 385.

¹³² Urry 2000, 23.

Ωστόσο, για τον Talcott Parsons ο λειτουργισμός περιγράφει μάλλον ένα συγκεκριμένο στάδιο στη μεθοδολογική ανάπτυξη της κοινωνικής επιστήμης, παρά μια συγκεκριμένη σχολή σκέψης¹³³.

Ο λειτουργισμός στο αρχαιολογικό πλαίσιο προωθεί την κατανόηση των κοινωνικών και πολιτισμικών συστημάτων εκ των έσω, καθορίζοντας το πώς αλληλοσυνδέονται τα διάφορα μέρη αυτών των συστημάτων και πώς αλληλεπιδρούν μεταξύ τους¹³⁴.

Η νέα αυτή προσέγγιση στην Αρχαιολογία επιδίωκε να ξεπεράσει τα όρια της απλής τυπολογικής μελέτης των τεχνέργων και να προχωρήσει στην ερμηνεία της λειτουργίας τόσο αυτών όσο και όλων των άλλων αρχαιολογικών δεδομένων (ιδίως αυτών που προέρχονταν από οικισμούς και οικιστικά σχήματα), ώστε να αποκτήσει μια γενικότερη εικόνα του πολιτισμού. Έβλεπε τον πολιτισμό ως οντότητα, ως ολοκληρωμένο σύστημα, που αποτελείται από αλληλεξαρτώμενα στοιχεία, όπως η κοινωνική, η πολιτική και η οικονομική οργάνωση, η τεχνολογία, το εμπόριο, η διατροφή, η θρησκεία, και λειτουργεί μέσα σε ένα φυσικό περιβάλλον, διατηρώντας με αυτό μια αρμονική σχέση αλληλεξάρτησης και αλληλεπίδρασης.

Οι αρχαιολόγοι που υιοθέτησαν τη λειτουργιστική προσέγγιση όπως ο G. Childe, ο G. Clark, ο W. Taylor και άλλοι, προσπάθησαν να ανασυνθέσουν το περιβάλλον και τα διάφορα συστατικά στοιχεία των αρχαίων πολιτισμών, και ειδικά την οικονομική και κοινωνική ζωή και τη διατροφή, σε συνεργασία με διάφορους ειδικούς επιστήμονες, κυρίως με βοτανολόγους, ζωολόγους και παλυνολόγους. Στόχος της προσέγγισής τους ήταν η ανασύνθεση της λειτουργίας του αρχαίου υλικού πολιτισμού και η εξέταση τόσο των εσωτερικών όσο και των εξωτερικών αιτιών της αλλαγής. Η εξέταση των εσωτερικών αιτιών αφορούσε κυρίως σε οικολογικούς και οικονομικούς παράγοντες. Ωστόσο, παρόλο που ο Taylor και ο Clark συνέβαλαν σημαντικά στην ενθάρρυνση της χρήσης αρχαιολογικών δεδομένων για την ανακατασκευή των προϊστορικών προτύπων της ζωής, συνέβαλαν ελάχιστα στην ερμηνεία των αλλαγών στο αρχαιολογικό αρχείο. Αντίθετα, ο Childe, παρά το γεγονός ότι ανέπτυξε μερικά πολύ ενδιαφέροντα μοντέλα κοινωνικής αλλαγής, απέτυχε να εξηγήσει πώς αυτά

¹³³ Bourricaud 1981, 8-11.

¹³⁴ Trigger 2007, 314.

τα μοντέλα θα μπορούσαν να εφαρμοστούν επακριβώς στη μελέτη της αρχαιολογικής μαρτυρίας¹³⁵.

Το ενδιαφέρον για τη μελέτη της εσωτερικής δομής των πολιτισμών με τη φονξιοναλιστική προσέγγιση προκάλεσε ενδιαφέρον για τη μελέτη της πολιτισμικής αλλαγής από την άποψη των εσωτερικών διαδικασιών που έλαβαν χώρα μακροπρόθεσμα στους πολιτισμούς. Η συγκεκριμένη προσέγγιση διερευνούσε τις γενικές εξελικτικές διαδικασίες που κρύβονταν πίσω από την ποικιλία των πολιτισμών και προσπαθούσε να διατυπώσει γενικεύσεις με παγκόσμια ερμηνευτική ισχύ, υπό τη μορφή γραμμικών εξελικτικών σχημάτων.

Οι πολιτισμικές αλλαγές αποδόθηκαν κυρίως σε περιβαλλοντικούς, οικονομικούς και τεχνολογικούς και όχι κοινωνικούς παράγοντες, και κατασκευάστηκαν γενικευτικά ερμηνευτικά σχήματα πολιτισμικής εξέλιξης. Αρχαιολόγοι όπως ο Childe, που ήταν ακόμη προσκολλημένος στη θεωρία περί της διάδοσης του πολιτισμού¹³⁶, θεωρούσαν ότι παράγοντες τέτοιου είδους απλώς ευνοούν την υιοθέτηση νέων ιδεών αντί να τις προκαλούν.

Η έμφαση στην επιστήμη σήμαινε την απομάκρυνση από τις ιστοριογραφικές και αντεπιστημονικές προσεγγίσεις του παρελθόντος και από τις απλοϊκές θεωρίες περί διάδοσης του πολιτισμού, ώστε η Αρχαιολογία να γίνει μια ακριβής και ώριμη επιστήμη, η οποία να χρησιμοποιεί όχι μόνον εκσυγχρονισμένες επιστημονικές τεχνικές, αλλά επίσης, και κυρίως, μια αυστηρή επιστημονικό μέθοδο για την ερμηνεία του παρελθόντος, ουδέτερη και αντικειμενική, απαλλαγμένη από ηθικές ή πολιτικές προκαταλήψεις. Πράγματι, η νέα μέθοδος, που συνιστούσε η διαδικαστική αρχαιολογία, και που δεν είναι άλλη από την υποθετική-παραγωγική-νομολογική μέθοδο των θετικών επιστημών, έκανε έναν απόλυτο διαχωρισμό ανάμεσα στη θεωρία και τα δεδομένα και, κατ' επέκταση, ανάμεσα στο παρόν και το παρελθόν.

Η νέα προσέγγιση είχε ως αποτέλεσμα την ξαφνική εγκατάλειψη των προηγούμενων προσπαθειών των αρχαιολόγων να ερμηνεύσουν την πολιτισμική αλλαγή από την άποψη των εξωτερικών επιρροών ή της ανθρώπινης επινοητικότητας, χωρίς αναφορά σε κοινωνικές και οικονομικές συνθήκες. Η νέα προσέγγιση χαρακτηρίστηκε, επίσης, από ρητή απόρριψη του ρατσισμού και της

¹³⁵ Trigger 2007, 385.

¹³⁶ Childe 1946, 24.

απαισιοδοξίας για την ανθρώπινη δημιουργικότητα, που χαρακτήριζε την αρχαιολογία στην κεντρική και δυτική Ευρώπη κατά τη δεκαετία του 1920¹³⁷.

Μέση θεωρία



Εικ. 2.1. Διαγραμματική παράσταση του μοντέλου της μέσης θεωρίας¹³⁸.

Όταν παρουσιάστηκε η έννοια της μέσης θεωρίας στην αρχαιολογία, αρκετές δεκαετίες πριν¹³⁹, θεωρήθηκε ότι μπορεί να συμβάλει στην οικοδόμηση της θεωρίας. Υπάρχουν λίγες ενδείξεις, όμως, ότι οι αρχαιολόγοι κατανοούν την έννοια της μέσης θεωρίας όπως αρχικά σχεδιάστηκε από τους θεωρητικούς της κοινωνικής επιστήμης¹⁴⁰. Αυτοί οι θεωρητικοί προωθούσαν την έννοια, όχι ως ουσιαστική θεωρία οποιουδήποτε συγκεκριμένου φαινομένου, αλλά ως γενική μέση θεωρία. Το βασικό σημείο ήταν να αναπτυχθεί μια στρατηγική για την ενσωμάτωση ερευνητικών προβλημάτων και δεδομένων σε σωρευτικά όργανα επιστημονικής γνώσης, στα οποία θα μπορούσαν να υπαχθούν θεωρίες διατεταγμένες σε διαφορετικά επίπεδα γενίκευσης. Συνεπώς, η έννοια της μέσης θεωρίας ήταν τμήμα μιας ευρύτερης προσπάθειας οικοδόμησης θεωρίας.

Ο Binford θεώρησε πως έπρεπε η θεωρία να στηρίζεται σε δύο τουλάχιστον επίπεδα. Το ένα επίπεδο είναι αυτό της μέσης θεωρίας. Αν δεχθεί κανείς τις παρατηρήσεις που γίνονται βάσει του αρχαιολογικού αρχείου ως

¹³⁷ Trigger 2007, 384-385.

¹³⁸ Johnson 1999, εικ. 4.1.

¹³⁹ Schiffer 1980, 377.

¹⁴⁰ Raab and Goodyear 1984, 255-268.

σύγχρονα γεγονότα, μαζί με την ιδέα ότι τα γεγονότα αυτά είναι στατικά, τότε ξεκάθαρα η βασική προβληματική για τον αρχαιολόγο περιλαμβάνει:

α) Το πώς φθάνει από τα σύγχρονα γεγονότα στις δηλώσεις για το παρελθόν και

(β) Το πώς μετατρέπει τα στατικά δεδομένα που προκύπτουν από την παρατήρηση του αρχαιολογικού αρχείου σε δηλώσεις κατανόησης της δυναμικής του παρελθόντος¹⁴¹.

Ο Binford καταλήγει στο συμπέρασμα ότι χρειάζεται να αναπτυχθούν ιδέες και θεωρίες (μέση θεωρία) σχετικά με τις διαδικασίες σχηματισμού του αρχαιολογικού αρχείου. Μόνο μέσω της ακριβούς κατανόησης τέτοιων διαδικασιών μπορεί να αποδώσει αξιόπιστα ο αρχαιολόγος νόημα σε γεγονότα που εμφανίζονται τόσο στο παρελθόν, όσο και στη σύγχρονη εποχή¹⁴². Η έννοια της αρχαιολογικής μέσης θεωρίας αναπτύχθηκε περαιτέρω από τον Binford, όταν ασχολήθηκε συγκεκριμένα με τα προβλήματα της γενικής μεθοδολογίας της ανάλυσης της πανίδας. Εκεί, αναφέρεται σε αυτή τη μορφή ανάλυσης ως «μέση έρευνα» και «οικοδόμηση μέσης θεωρίας». Υποδεικνύει, επίσης¹⁴³, ότι οι όροι *μέση έρευνα* και *οικοδόμηση μέσης θεωρίας* αντιπροσωπεύουν, ουσιαστικά, αυτό που ο David Clarke αναφέρει ως «ερμηνευτική θεωρία», και προφανώς ισοδυναμεί και με τη «συμπεριφορική αρχαιολογία» όπως προτάθηκε από τον Michael Schiffer το 1976¹⁴⁴. Προσδιορίζοντας ο Binford τι είναι επιθυμητό από τη διεξαγωγή της «μέσης έρευνας» θεωρεί ότι χρειάζονται ακριβή μέσα αναγνώρισης και καλά όργανα μέτρησης συγκεκριμένων ιδιοτήτων των παλαιότερων πολιτιστικών συστημάτων. Χρειάζονται αξιόπιστες γνωστικές συσκευές, «στήλες της Ροζέτας» που επιτρέπουν την ακριβή μετατροπή της παρατήρησης στατικών δεδομένων σε δηλώσεις για τη δυναμική του παρελθόντος. Με αυτόν τον τρόπο οι αρχαιολόγοι επιδιώκουν να οικοδομήσουν ένα παραδειγματικό πλαίσιο αναφοράς για να δίνουν νόημα σε επιλεγμένα χαρακτηριστικά του αρχαιολογικού αρχείου μέσω ενός θεωρητικά τεκμηριωμένου

¹⁴¹ Binford 1977, 6.

¹⁴² Binford 1977, 7.

¹⁴³ Binford 1981, 25.

¹⁴⁴ Clarke 1973, 8.

ερευνητικού σώματος, αντί να αποδέχονται άκριτα τη λαογραφική γνώση ως βάση για την περιγραφή του παρελθόντος¹⁴⁵.

Στο αρχικό κοινωνιολογικό πλαίσιο, η μέση θεωρία προωθήθηκε ως βάση για τη θεωρητικοποίηση των αιτίων της ανθρώπινης κοινωνικής συμπεριφοράς. Η παραγωγική έρευνα σχεδιάστηκε ως εμπειρική, αλλά με μια ιεραρχία προτάσεων σε μέσου εύρους αφαίρεση και παρείχε μια κρίσιμη σχέση μεταξύ συλλογής δεδομένων και θεωριών υψηλότερης τάξης. Στον αντίποδα, οι περισσότερες χρήσεις της μέσης θεωρίας στην αρχαιολογία είναι περισσότερο εστιασμένες στο μεθοδολογικό ζήτημα των διαδικασιών σχηματισμού της αρχαιολογικής θέσης. Αυτή η έμφαση στην πραγματικότητα συνεχίζει την ανάπτυξη μιας υλιστικής επιστημολογίας για την αρχαιολογία. Ο θεμελιώδης στόχος μιας τέτοιας επιστημολογίας είναι να εδραιώσει συμπεράσματα σχετικά με τις προηγούμενες ανθρώπινες συμπεριφορές, αναπτύσσοντας μια αξιόπιστη μεθοδολογία για τη διαφοροποίηση των επιπτώσεων της συμπεριφοράς από τις πολλές άλλες αιτίες του υλικού αρχείου¹⁴⁶.

Εφαρμόζοντας τη μέθοδο της μέσης θεωρίας, οι αρχαιολόγοι κατασκεύαζαν αρχικά ένα θεωρητικό πρότυπο ή μοντέλο με διάφορες εναλλακτικές προτάσεις για την ερμηνεία κάποιου πολιτισμικού φαινομένου. Οι εναλλακτικές εκδοχές αντλούνταν από μια «δεξαμενή» υποθέσεων ή θεωριών, γνωστή ως «μέση θεωρία». Στη συνέχεια, το «πρότυπο» ελεγχόταν έναντι των στατικών δεδομένων που είχαν συλλεχθεί εν είδει εργαστηριακού πειράματος. Εντέλει, τα συμπεράσματα από τον έλεγχο του προτύπου διατυπώνονταν σε επιστημονικούς καθολικούς νόμους, βάσει των οποίων αναπτύσσονταν γενικευτικές, καθολικές και επιστημονικές θεωρίες για την ερμηνεία του παρελθόντος.

Πολιτισμικά συστήματα

Η έμφαση στην ανθρωπολογία οδηγούσε στην απόρριψη δύο βασικών πρακτικών της πολιτισμικής ιστορικής αρχαιολογίας:

¹⁴⁵ Binford 1981, 25.

¹⁴⁶ Raab and Goodyear 1984, 265.

α) της απλής ταξινόμησης τεχνέργων σε πολιτισμούς, η οποία οδηγούσε συχνά στο να αγνοείται ο ανθρώπινος παράγοντας που κρύβεται πίσω από τα αντικείμενα και

β) της απλοϊκής ταύτισης των «αρχαιολογικών πολιτισμών» με αληθινές ανθρώπινες ομάδες του παρελθόντος.

Ο Clarke (1968) τόνισε ότι οι πολιτισμοί, τους οποίους ορίζουν οι αρχαιολόγοι, είναι καθαρά αρχαιολογικές παρά κοινωνικές ή εθνικές οντότητες. Έχουμε δηλαδή να κάνουμε απλώς με αρχαιολογικούς πολιτισμούς. Ο Binford, με τη σειρά του, πίστευε ότι οι αρχαιολόγοι θα έπρεπε να έχουν τις γνώσεις και τις εμπειρίες κοινωνικών ανθρωπολόγων, διότι μόνον αν μελετούσαν τους γενικούς συσχετισμούς ανάμεσα στον υλικό πολιτισμό και την ανθρώπινη συμπεριφορά σε ζωντανές κοινωνίες ανά τον κόσμο θα ήταν σε θέση να ερμηνεύσουν την αρχαία ανθρώπινη συμπεριφορά απ' όλες τις πλευρές της και να αποκαλύψουν την κανονικότητα που, κατά τη γνώμη του, τη χαρακτήριζε. Έτσι, ξεκίνησε ένα πρόγραμμα εθνοαρχαιολογικής έρευνας και διεξήγαγε σειρά διαπολιτισμικών μελετών, χρησιμοποιώντας εθνογραφικά δεδομένα από πολλές ζωντανές κοινωνίες, προκειμένου να παράγει καθολικές γενικεύσεις για τη συμπεριφορά των κυνηγετικών και τροφοσυλλεκτικών κοινωνιών και των σχετικών ταφικών πρακτικών τους¹⁴⁷.

Στοχεύοντας να ερμηνεύσει την πολιτισμική αλλαγή, η διαδικαστική αρχαιολογία του Binford και των οπαδών του υιοθέτησε ένα εντελώς νέο μεθοδολογικό και θεωρητικό πλαίσιο, βασισμένο στην ιδέα του πολιτισμικού συστήματος εμπνευσμένο από τη γενική θεωρία συστημάτων στις φυσικές επιστήμες. Πρότεινε ως εναλλακτική λύση τη θεώρηση του πολιτισμού ως ένα σύστημα αποτελούμενο από υποσυστήματα, σημειώνοντας ότι οι διαφορές και οι ομοιότητες μεταξύ των διαφόρων κατηγοριών αρχαιολογικών καταλοίπων αντανakλούν διαφορετικά υποσυστήματα. Επομένως, αναμένεται να λειτουργούν ανεξάρτητα το ένα από το άλλο στην κανονική λειτουργία του συστήματος ή κατά την αλλαγή του συστήματος¹⁴⁸.

Βέβαια, χρειάζεται να αναφερθεί εδώ ότι ο όρος θεωρία συστημάτων συχνά εφαρμόζεται αδιακρίτως σε δύο διαφορετικούς τομείς σπουδών, τη θεωρία

¹⁴⁷ Trigger 2007, 406.

¹⁴⁸ Binford 1965, 203-210.

γενικών συστημάτων και τη θεωρία μαθηματικών συστημάτων. Η θεωρία γενικών συστημάτων δεν είναι πραγματική θεωρία, καθώς δεν έχει θεσπίσει γενικές αρχές που να ισχύουν για όλα τα συστήματα. Η θεωρία των γενικών συστημάτων λειτουργεί περισσότερο ως πρόγραμμα για την ανάπτυξη μιας τέτοιας θεωρίας, ωστόσο εγγενείς δυσκολίες σε ένα τόσο βασικό επίπεδο δεν παρέχουν το έδαφος για ανάπτυξη. Η θεωρία μαθηματικών συστημάτων είναι μια αυθεντική θεωρία, αλλά είναι θεωρία των καθαρών μαθηματικών. Έχει περιορισμούς που την καθιστούν εφαρμόσιμη σε λίγα, και μόνο πολύ απλά, πραγματικά συστήματα. Δεν είναι αρκετά σύνθετη, ώστε να χειριστεί εκείνες τις καταστάσεις που ενδιαφέρουν τους αρχαιολόγους. Οι αρχαιολόγοι δανείστηκαν από τη θεωρία γενικών συστημάτων ορισμένες βασικές έννοιες όπως θετική και αρνητική ανατροφοδότηση, δυναμική ισορροπία ή ομοιόσταση, προκειμένου να χαρακτηρίσουν ενδιαφέροντα αρχαιολογικά φαινόμενα, ωστόσο αυτές οι έννοιες δεν προέρχονταν από τη θεωρία των γενικών συστημάτων, αλλά από επιστήμες όπως φυσική και η μηχανική¹⁴⁹.

Στην αρχαιολογική θεωρία συστημάτων ο πολιτισμός προσδιορίζεται ως ενιαίο και αυτοτελές σύστημα, οργανωμένο ως δίκτυο υποσυστημάτων. Τα διάφορα ανεξάρτητα υποσυστήματα επικοινωνούν μεταξύ τους και συνδέονται λειτουργικά με μηχανισμούς θετικής ή αρνητικής ανατροφοδότησης ανταλλάσσοντας συνεχώς «ύλη, ενέργεια ή πληροφορίες» και βοηθώντας με τον τρόπο αυτό στην επιβίωση του πολιτισμού, όπως ακριβώς τα όργανα ενός σύνθετου οργανισμού λειτουργούν συλλογικά για να βοηθήσουν τον οργανισμό στο σύνολό του να επιζήσει¹⁵⁰.

Βάσει της συγκεκριμένης βιολογικής αναλογίας, τα διάφορα πολιτισμικά υποσυστήματα θα μπορούσαν να ερμηνευθούν με τον ίδιο τρόπο με τον οποίο μπορούν να ερμηνευθούν τα διάφορα μέρη ενός οργανισμού, δηλαδή με βάση το πόσο καλά απέδιδαν ή πόσο εύρυθμα επιτελούσαν τις λειτουργίες τους για την εξασφάλιση της επιβίωσης του όλου πολιτισμικού συστήματος, καθώς και τον τρόπο που συνδέονταν μεταξύ τους στο πλαίσιο αυτού του συστήματος.

Η διαδικαστική αρχαιολογία θεώρησε ότι το πολιτισμικό σύστημα, ο πολιτισμός, αποτελεί αναπόσπαστο μέρος του ευρύτερου οικοσυστήματος και ότι

¹⁴⁹ Salmon 1978, 174-183.

¹⁵⁰ Renfrew and Bahn 2001, 499.

είναι «προσαρμοστικός», δηλαδή είναι ένα μέσο με τη βοήθεια του οποίου η κοινωνία εξασφαλίζει την ισορροπία της, την προσαρμογή της και, τελικά, την επιβίωσή της μέσα σε ένα περιβάλλον. Είναι δηλαδή το εξωσωματικό μέσο προσαρμογής για τον ανθρώπινο οργανισμό¹⁵¹.

Η θεωρητική έμφαση στο φυσικό περιβάλλον οδήγησε τη διαδικαστική αρχαιολογία σε μια προσπάθεια αποκατάστασης του οικολογικού υποβάθρου και της διατροφικής οικονομίας των κοινωνιών του παρελθόντος. Στο ίδιο πνεύμα, ασχολήθηκε συστηματικά με την ανασύσταση των αρχαίων τρόπων ζωής. Πέρα, όμως, από αυτή την περιγραφική δραστηριότητα, προσπάθησε επίσης να ερμηνεύσει την πολιτισμική αλλαγή δίνοντας έμφαση στην έννοια της ενδογενούς πολιτισμικής αλλαγής, δηλαδή της αλλαγής που γεννάται στο εσωτερικό ενός πολιτισμικού συστήματος. Δεδομένου, όμως, ότι η φυσική κατάσταση του κάθε πολιτισμικού συστήματος είναι η διατήρηση μιας δυναμικής ισορροπίας με το περιβάλλον, οι όποιες αλλαγές δε θα μπορούσε να είναι παρά το αποτέλεσμα μη πολιτισμικών, αιτίων ή ερεθισμάτων (άποψη και της πολιτισμικής ιστορικής αρχαιολογίας).

Η διαφορά είναι ότι στη διαδικαστική αρχαιολογία η πολιτισμική αλλαγή ερμηνεύεται βάσει των αντιδράσεων που προκαλούν στο σύστημα οι εξωτερικοί παράγοντες, άποψη που συνάδει με τη θεώρηση του Binford, ότι η πολιτισμική αλλαγή είναι το αποτέλεσμα «συνθηκών έλλειψης ισορροπίας στο τοπικό οικοσύστημα»¹⁵², διαφορετικά τα αίτια είναι εξωγενείς παράγοντες, που σχετίζονται με το εμπόριο, τον αποικισμό ή την εχθρική εισβολή. Όταν δεχθεί το σύστημα τέτοιου είδους εξωτερικά ερεθίσματα, η αλλαγή συμβαίνει με τη μορφή μιας πολιτισμικής διαδικασίας, μιας σειράς διαταράξεων και αλληλεπιδράσεων ανάμεσα στα υποσυστήματα. Η αλλαγή σε ένα υποσύστημα προξενεί αλλαγή σε άλλο ή άλλα υποσυστήματα, γεγονός που με τη σειρά του επιφέρει νέα αλλαγή στο αρχικό υποσύστημα, επηρεάζοντας καθολικά τη δομή του συστήματος¹⁵³, και τη σχέση του συστήματος με το περιβάλλον. Το φαινόμενο του πολλαπλασιασμού της αλλαγής του πολιτισμικού συστήματος είναι γνωστό ως «φαινόμενο πολλαπλασιαστή». Κατά τον Renfrew οι αλλαγές ή οι καινοτομίες που συμβαίνουν σε ένα πεδίο ανθρώπινης δραστηριότητας (σε ένα πολιτισμικό

¹⁵¹ Binford 1968, 22.

¹⁵² Binford 1968, 431.

¹⁵³ Binford 1968, 21.

υποσύστημα), ενίοτε δρουν, έτσι ώστε να ευνοούν αλλαγές σε άλλους τομείς (σε άλλα υποσυστήματα). Το φαινόμενο πολλαπλασιαστή θεωρείται ότι λειτουργεί όταν αυτές οι αλλαγές σε ένα ή περισσότερα υποσυστήματα αυτενεργούν, έτσι ώστε να ενισχύσουν τις αρχικές αλλαγές στο πρώτο υποσύστημα¹⁵⁴.

Διατυπώνοντας την ερμηνεία της πολιτισμικής αλλαγής η διαδικαστική αρχαιολογία έδωσε νέα έμφαση στην ιδέα της πολιτισμικής εξέλιξης και στη γενίκευση. Ακολουθώντας το ρεύμα του νεοεξελικτισμού στην κοινωνική ανθρωπολογία οι διαδικαστικοί αρχαιολόγοι πίστευαν ότι δεν υπάρχουν πολλές παραλλαγές πολιτισμών. Απλώς ο πολιτισμός εξελίσσεται περνώντας από προκαθορισμένα στάδια κατά μήκος μιας γραμμικής πορείας από τις απλούστερες στις πλέον σύνθετες μορφές κοινωνικής οργάνωσης. Η συγκεκριμένη αντίληψη αποτυπώνεται στο εξελικτικό σχήμα του Service και περιλαμβάνει τα στάδια ομάδα, φυλή, φυλαρχία και κράτος¹⁵⁵. Αν και το συγκεκριμένο σχήμα δεν περιλαμβάνει όλο το εύρος των συστημάτων κοινωνικής οργάνωσης, είναι ακόμη χρήσιμο για τη μελέτη της ανάπτυξης των κοινωνιών του παρελθόντος¹⁵⁶.

Βάσει αυτού του σκεπτικού οι διαδικαστικοί αρχαιολόγοι ερμήνευσαν τα αρχαία πολιτισμικά συστήματα, εντάσσοντάς τα στα στάδια που πρότειναν τα εξελικτικά σχήματα των ανθρωπολόγων της εποχής. Επιπλέον, θεωρούσαν ότι τα διάφορα πολιτισμικά συστήματα και η ανθρώπινη συμπεριφορά χαρακτηρίζονταν από μια κανονικότητα και ότι παρουσίαζαν βασικές ομοιότητες. Έτσι, αποκτώντας χάρη στη διατύπωση επιστημονικών «νόμων» ή γενικεύσεων λεπτομερή εικόνα των διαδικασιών που οδηγούν σταδιακά από τους απλούστερους στους πολυπλοκότερους πολιτισμούς, θα μπορούσαν στη συνέχεια να συμπληρώνουν την αποσπασματική εικόνα των πολιτισμών που μελετούν, να κατανοούν και να ερμηνεύουν πολλά από τα χαρακτηριστικά τους. Συνεπώς, η μελέτη ενός πολιτισμικού συστήματος μπορούσε να βοηθήσει τους αρχαιολόγους στη μελέτη ενός άλλου συστήματος σε ένα άλλο σημείο του κόσμου¹⁵⁷.

Εν κατακλείδι, η ανάπτυξη λειτουργικών και διαδικαστικών προσεγγίσεων στα αρχαιολογικά δεδομένα οδήγησε στη σταδιακή αντικατάσταση του στείρου προβληματισμού της πολιτισμικής ιστορικής αρχαιολογίας για την εθνότητα από

¹⁵⁴ Renfrew 2001, 37.

¹⁵⁵ Service 1971, 8-12.

¹⁵⁶ Kang 2005, 22-35.

¹⁵⁷ Κουκουζέλη κ.ά. 2003, 266.

ένα νέο ζωτικό ενδιαφέρον για τον τρόπο λειτουργίας και αλλαγής των προϊστορικών πολιτισμών. Το μειούμενο ενδιαφέρον για τη μελέτη του πολιτισμού και το αυξανόμενο ενδιαφέρον για τη συμπεριφορά δεν ήταν μόνο συμβατό με τις τάσεις άλλων κοινωνικών επιστημών, αλλά και προκάλεσε διαρκείς αλλαγές στην αρχαιολογία¹⁵⁸.

Είναι κοινός τόπος η αντίληψη ότι η διαδικαστική αρχαιολογία ήταν ακριβής σε πολλά από τα ερωτήματα που έθεσε, αλλά φαίνεται πως προκλήθηκε σύγχυση στις προσπάθειες απάντησής τους¹⁵⁹. Προφανώς, η «νέα αρχαιολογία» έθεσε το ζήτημα της ανάγκης για μια επιστημονική μέθοδο. Η στροφή προς την επιστήμη φαίνεται πως ήταν βάσιμη, δεδομένου ότι προγενέστερες αρχαιολογικές μέθοδοι δεν βοηθούσαν στη διαμόρφωση ενός κοινού τόπου όσον αφορά στις γνώσεις μας για το παρελθόν¹⁶⁰. Ωστόσο, η διαδικαστική αρχαιολογία δεν καθόρισε ποιες επιστημολογικές φιλοσοφίες είναι κατάλληλες ως μοντέλα, ούτε παρουσίασε πειστικές μελέτες περίπτωσης για τα αρχαιολογικά δεδομένα.

2.5. Μεταδιαδικαστική αρχαιολογία

Στα τέλη της δεκαετίας του 1970 και στις αρχές της δεκαετίας του 1980 το ρεύμα της διαδικαστικής Αρχαιολογίας έχασε έδαφος, όταν βρετανοί αρχαιολόγοι, με επικεφαλής τους Ian Hodder, Michael Shanks και Christopher Tilley, εξέφρασαν αμφιβολίες για τη δυνατότητά της να ερμηνεύσει τους αρχαίους πολιτισμούς και την πολιτισμική αλλαγή, συνδέοντας έτσι το παρελθόν με το παρόν. Ασκώντας κριτική στη διαδικαστική αρχαιολογία, οι μεταδιαδικαστικοί αρχαιολόγοι πρότειναν νέες θεωρητικές απόψεις και προσεγγίσεις υπό την επίδραση σύγχρονων θεωριών από τις κοινωνικές και γενικότερα τις ανθρωπιστικές επιστήμες. Οι διάφορες θεωρητικές προσεγγίσεις ονομάζονται μεταδιαδικαστικές προσεγγίσεις και εκπροσωπούν μια χαλαρά συνδεδεμένη ομάδα θεωρητικών σχολών, γνωστή ως μεταδιαδικαστική Αρχαιολογία¹⁶¹.

¹⁵⁸ Trigger 2007, 385,

¹⁵⁹ Johnson 2010, 32-33.

¹⁶⁰ Clarke 1972, 3.

¹⁶¹ Johnson 2010, 101.

Το νέο ρεύμα σκέψης αγκάλιασε ποικίλες θεωρητικές απόψεις όπως ο δομισμός και ο νεομαρξισμός, καθώς και ποικιλία διαφορετικών αρχαιολογικών τεχνικών, όπως η φαινομενολογία¹⁶². Στις Ηνωμένες Πολιτείες, οι αρχαιολόγοι θεωρούν ευρύτερα το μεταδιαδικαστικό ρεύμα συμπλήρωμα του διαδικαστικού κινήματος, ενώ στο Ηνωμένο Βασίλειο, τα δύο ρεύματα θεωρούνται σε μεγάλο βαθμό διακριτά και αντίθετα θεωρητικά κινήματα. Σε άλλα μέρη του κόσμου, το μεταδιαδικαστικό ρεύμα έχει επηρεάσει λιγότερο την αρχαιολογική σκέψη¹⁶³.

Αν και το μεταδιαδικαστικό ρεύμα έχει σημαντικές καταβολές στην ιστορία της τέχνης και την ιστορική αρχαιολογία, κωδικοποιήθηκε στις αρχές της δεκαετίας του 1980 κυρίως μέσα από τα κείμενα του Ian Hodder και των φοιτητών του στο Πανεπιστήμιο του Κέιμπριτζ. Επηρεασμένη από τη συμβολική και στρουκτουραλιστική ανθρωπολογία, η ομάδα του Κέιμπριτζ ανέπτυξε την πρώτη κριτική της διαδικαστικής αρχαιολογίας. Σύμφωνα με την αφήγηση των μεταδιαδικαστικών, η διαδικαστική αρχαιολογία υπονομεύθηκε από την εξάρτησή της από την επιστημονική μέθοδο, τις λειτουργικές εξηγήσεις, τη θεωρία συστημάτων και τους καθολικούς νόμους¹⁶⁴. Επιπλέον, οι μεταδιαδικαστικοί υποστήριξαν ότι απέδιδε ελάχιστη ή καθόλου προσοχή στην κοινωνική κατασκευή του νοήματος και τη δυναμική των σχέσεων εξουσίας στην κοινωνική αρένα. Οι κοινωνικές συγκρούσεις που προκύπτουν από αντιθετικά συμφέροντα αναγνωρίζονται ως ζωτικά χαρακτηριστικά όλων των ανθρώπινων κοινωνιών και αποτελούν σημαντική πηγή αλλαγών. Αυτή η κριτική οδήγησε στον πειραματισμό με τον μεταδομισμό, τον νεομαρξισμό, τη θεωρία των φύλων και την κριτική θεωρία¹⁶⁵.

Οι Miller και Tilley απαριθμώντας τα σημεία κριτικής κατά της διαδικαστικής αρχαιολογίας, θεωρούν ως αρνητικά σημεία: την άκριτη αποδοχή του θετικισμού, την έμφαση στον λειτουργισμό και την περιβαλλοντική προσαρμογή, την έμφαση στις βιολογικές οδηγίες, την περιφρόνηση για τις κοινωνικές σχέσεις τη γνώση ή την ιδεολογία, την έλλειψη ενδιαφέροντος για τη σημερινή κοινωνική παραγωγή της γνώσης, την υπερβολική έμφαση στη σταθερότητα και όχι στις συγκρούσεις, τη μείωση του ρόλου της κοινωνικής

¹⁶² Κουκουζέλη κ.ά. 2003, 269.

¹⁶³ Trigger 2007, 477-478.

¹⁶⁴ Hodder 1982, 1-16.

¹⁶⁵ Preucel 1995, 147-175.

αλλαγής στις επιδράσεις των εξωτερικών παραγόντων και την πίστη στην ποσοτικοποίηση ως στόχο της αρχαιολογίας¹⁶⁶. Για τους μεταδιαδικαστικούς αρχαιολόγους «η αρχαιολογία έχει γίνει τόσο ορθολογική, ώστε έχει αποβάλλει τον ανθρώπινο παράγοντα και οι καλύτερες όψεις της αρχαιολογίας όχι μόνο έχουν γίνει μηχανικές, αλλά στερούνται σχεδόν πολιτιστικού πλαισίου»¹⁶⁷.

Η διαδικαστική θεώρηση του πολιτισμού ως λειτουργικού και προσαρμοστικού είναι για τους μεταδιαδικαστικούς ελλιπής. Θεωρούν ότι ο πολιτισμός είναι μεστός συμβολικών νοημάτων, που δεν είναι άμεσα αντιληπτά. Ασκούν κριτική, επίσης, στο γεγονός ότι διαδικαστική αρχαιολογία, βασισμένη στον λειτουργισμό, θεωρεί ότι ο πολιτισμός σχετίζεται απλώς με τη λειτουργιστική σχέση που μπορεί να παρατηρηθεί ανάμεσα στα υποσυστήματα ενός συστήματος και ότι αποσκοπεί να προσαρμόσει την κοινωνία στο εξωτερικό περιβάλλον. Η μεταδιαδικαστική προσέγγιση για την αρχαιολογική έρευνα είναι διαμετρικά αντίθετη με εκείνη διαδικαστικού ρεύματος. Οι διαδικαστικοί, ως υπέρμαχοι του θετικισμού, πίστευαν ότι η επιστημονική μέθοδος θα έπρεπε και θα μπορούσε να εφαρμοστεί στην αρχαιολογική έρευνα, επιτρέποντας έτσι στους αρχαιολόγους να παρουσιάσουν αντικειμενικές δηλώσεις σχετικά με τις κοινωνίες του παρελθόντος με βάση τα αποδεικτικά στοιχεία. Η μεταδιαδικαστική αρχαιολογία, ωστόσο, αμφισβήτησε αυτή τη θέση υπογραμμίζοντας ότι η αρχαιολογία είναι υποκειμενική και όχι αντικειμενική και ότι η αλήθεια που μπορεί να εξαχθεί από το αρχαιολογικό αρχείο σχετίζεται συχνά προσωπικές απόψεις των αρχαιολόγων αρχαιολόγου που είναι υπεύθυνος για την ανίχνευση και την παρουσίαση των δεδομένων¹⁶⁸. Όπως επεσήμανε ο αρχαιολόγος Matthew Johnson, οι μεταδιαδικαστικοί προτείνουν ότι δεν είναι δυνατόν να αντιμετωπιστεί η θεωρία και τα δεδομένα και ότι χρειάζεται να βλέπει ο αρχαιολόγος τα δεδομένα μέσω ενός νέφους θεωρίας¹⁶⁹.

Πίσω από τη λειτουργική, υλική επιφάνεια των αντικειμένων του πολιτισμού υπάρχει, πάνω απ' όλα, μια εσωτερική λογική, μια συμβολική δομή, ένας κώδικας που αποτελείται από μια σειρά από νοήματα ή, διαφορετικά, μια σειρά από κρυμμένους (γνωστικούς) κανόνες, τους οποίους μπορεί κανείς να

¹⁶⁶ Miller and Tilley 1984, 2.

¹⁶⁷ Leone 1986, 432.

¹⁶⁸ Trigger 2007, 451–452.

¹⁶⁹ Johnson 2010, 102.

παρομοιάσει με τους γραμματικούς κανόνες της γλώσσας. Αυτό κάνει τον πολιτισμό να προσομοιάζει με «κείμενο», το οποίο χρειάζεται να διαβαστεί και να ερμηνευτεί.

Η συμβολική δομή ενός πολιτισμού είναι το μέσο καθορισμού της ανθρώπινης συμπεριφοράς αλλά, ταυτόχρονα, είναι το προϊόν αυτής της συμπεριφοράς. Κατασκευάζεται και επιβάλλεται από τις επικρατέστερες κοινωνικές ομάδες. Οι ομάδες αυτές αποτελούνται από δρώντα άτομα (*social actors*), τα οποία προσπαθούν να αλλάξουν τον συμβολικά δομημένο κόσμο μέσα στον οποίο ζουν ενεργώντας στο πλαίσιο μιας συγκεκριμένης κοινωνικής και πολιτικής στρατηγικής (κοινωνική δράση)¹⁷⁰.

Κριτική, επίσης, ασκείται στη διαδικαστική θεώρηση του πολιτισμού από μεταδιαδικαστικούς αρχαιολόγους που τη θεωρούν εντελώς συντηρητική και ολοκληρωτική στη φύση της ως πρόγραμμα μιας τέλεια καθορισμένης κοινωνίας με τέλεια διακυβέρνηση¹⁷¹. Η συγκεκριμένη άποψη για την κοινωνία θεωρείται ανάλογη με αυτή ενός οικουμενικού, απολυταρχικού κράτους συντηρητικού και καταπιεστικού στη φύση τους για τους Shanks and Tilley.

Σε κάθε περίπτωση στο μεταδιαδικαστικό ρεύμα φαίνεται να υπάρχει αντίδραση κατά των ερμηνειών που είναι μηχανικές ή προκατειλημμένες διανοητικά. Ούτε ο οικολογικός ντετερμινισμός ούτε ο δομισμός ανταποκρίνονται επαρκώς στις ανησυχίες των μεταδιαδικαστικών αρχαιολόγων. Υπό την οπτική γωνία του μεταδιαδικαστικού ρεύματος είναι απογοητευτικό το γεγονός ότι σε μεγάλο βαθμό οι αρχαιολόγοι πρέπει να συνειδητοποιήσουν και να κατανοήσουν ότι η ουσία της ανθρώπινης και επομένως της πολιτισμικής ζωής είναι ταυτόχρονα υλική και συμβολική¹⁷². Η ανθρώπινη συμπεριφορά μπορεί να ορίζεται από τη συμβολική δομή του πολιτισμού, όμως συγχρόνως η συμβολική δομή μπορεί να χρησιμοποιηθεί από δρώντα άτομα με στόχο την αλλαγή. Με αυτόν τον τρόπο, ο πολιτισμός, χάρη στη συμβολική δομή του, μπορεί αφεαυτού να αποτελέσει ένα ισχυρό μέσο κοινωνικής αλλαγής¹⁷³.

Επειδή πιστεύουν ότι η αρχαιολογία είναι εγγενώς υποκειμενική, οι μεταδιαδικαστικοί αρχαιολόγοι υποστηρίζουν ότι όλοι οι αρχαιολόγοι, ανεξάρτητα

¹⁷⁰ Κουκουζέλη κ.ά. 2003, 270.

¹⁷¹ Shanks and Tilley, 1992, 121-122.

¹⁷² Conkey and Spector 1984:24

¹⁷³ Shanks and Tilley 1992, 129-130.

από το αν το αναγνωρίζουν ή όχι, επιβάλλουν πάντοτε τις δικές τους απόψεις και προκαταλήψεις στις ερμηνείες των αρχαιολογικών δεδομένων. Σε πολλές περιπτώσεις, θεωρούν ότι αυτή η προκατάληψη έχει πολιτικό χαρακτήρα¹⁷⁴. Ο μεταδιαδικαστικός Daniel Miller πίστευε ότι η θετικιστική προσέγγιση των διαδικαστικών αρχαιολόγων επεδίωκε μόνο να παράγει τεχνικές γνώσεις, που θα διευκόλυναν την καταπίεση των απλών ανθρώπων από τις ελίτ¹⁷⁵. Σε μια παρόμοια κριτική, οι Miller και Chris Tilley πίστευαν πως, προβάλλοντας την ιδέα ότι οι ανθρώπινες κοινωνίες διαμορφώνονταν από εξωτερικές επιρροές και πιέσεις, οι αρχαιολόγοι δέχθηκαν σιωπηρά την κοινωνική αδικία¹⁷⁶. Πολλοί μεταδιαδικαστικοί προχώρησαν ακόμη περισσότερο και επέκριναν το γεγονός ότι αρχαιολόγοι από πλούσιες, δυτικές χώρες μελετούσαν και έγραφαν την ιστορία φτωχότερων εθνών στον δεύτερο και στον τρίτο κόσμο. Ο Ian Hodder θεωρούσε ότι οι αρχαιολόγοι δεν είχαν δικαίωμα να ερμηνεύουν τις προϊστορίες άλλων εθνοτικών ή πολιτισμικών ομάδων και ότι απλά θα έπρεπε να παρέχουν σε άτομα από αυτές τις ομάδες τη δυνατότητα να κατασκευάσουν τις δικές τους απόψεις για το παρελθόν¹⁷⁷. Ενώ η άποψη του Hodder δεν ήταν καθολικά αποδεκτή μεταξύ των μεταδιαδικαστικών αρχαιολόγων, υπήρχε αρκετή υποστήριξη για την αντίθεσή του στον ρατσισμό, την αποικιοκρατία και τον επαγγελματικό ελιτισμό, στην αρχαιολογία¹⁷⁸. Αρκετοί αρχαιολόγοι του μεταδιαδικαστικού κινήματος, όπως ο Michael Shanks, ο Christopher Tilley και ο Peter Ucko, θεωρούνται πως υπονόμισαν τους ισχυρισμούς της αρχαιολογίας ότι είναι έγκυρη πηγή γνώσης για το παρελθόν, ενθαρρύνοντας έτσι τους ανθρώπους να αμφισβητούν και να αντιστέκονται σε όλες τις μορφές εξουσίας. Η θέση αυτή χαιρετίστηκε από τους υποστηρικτές της ως εκδημοκρατισμός της αρχαιολογίας¹⁷⁹.

Οι διαδικαστικοί ήταν και είναι σταθερά υλιστές στην ερμηνεία τους για τον πολιτισμό, ενώ οι πολιτισμικοί ιστορικοί αρχαιολόγοι είναι ιδεαλιστές. Ενώ οι μεταδιαδικαστικοί υποστηρίζουν ότι οι κοινωνίες του παρελθόντος πρέπει να ερμηνεύονται τόσο με υλιστικές όσο και με ιδεαλιστικές απόψεις. Όπως επεσήμανε ο Johnson, πολλοί ισχυρίζονται ότι θα πρέπει να απορριφθεί η

¹⁷⁴ Trigger 2007, 467.

¹⁷⁵ Miller 1984, 38.

¹⁷⁶ Miller and Tilley 1984, 2.

¹⁷⁷ Hodder 1984, 25–32.

¹⁷⁸ Trigger 2007, 452.

¹⁷⁹ Trigger 2007, 467–468.

αντίθεση ανάμεσα στο υλικό και το ιδεατό¹⁸⁰. Ενώ αναγνωρίζουν οι μεταδιαδικαστικοί ότι οι κοινωνίες του παρελθόντος ερμηνεύουν το περιβάλλον τους εν μέρει υλιστικά, ωστόσο υποστηρίζουν ότι πολλές ιστορικές κοινωνίες έδωσαν μεγάλη έμφαση στην ιδεολογία (η οποία περιελάμβανε τη θρησκεία) για την ερμηνεία του κόσμου τους και τη χειραγώγηση της συμπεριφοράς τους. Παραδείγματα αυτής της χειραγώγησης δίνει ο Bernard Knapp, ο οποίος εξέτασε τους τρόπους με τους οποίους η κοινωνικές ελίτ χειραγωγούσαν την ιδεολογία, για να διατηρήσουν τον πολιτικό και οικονομικό έλεγχο¹⁸¹ και του Mike Parker Pearson, ο οποίος ισχυρίστηκε ότι τα εργαλεία είναι εξίσου προϊόν ιδεολογίας, όπως είναι ένα στέμμα ή ένας κώδικας δικαίου¹⁸².

Πολλοί, αν και όχι όλοι οι μεταδιαδικαστικοί, χρησιμοποίησαν τη θεωρία του δομισμού για να κατανοήσουν τις αρχαίες κοινωνίες. Ο ίδιος ο δομισμός είναι θεωρία που ανέπτυξε ο Γάλλος ανθρωπολόγος Claude Lévi-Strauss, υποστηρίζοντας την ιδέα ότι τα πολιτισμικά πρότυπα δε χρειάζεται να προκαλούνται από εξωτερικά ερεθίσματα και ότι υποκείμενη στον πολιτισμό βρίσκεται μια βαθιά δομή ή ουσία, που διέπεται από τους δικούς της νόμους, την οποία οι άνθρωποι αγνοούν. Ωστόσο, αυτή η βαθιά δομή εξασφαλίζει την κανονικότητα της πολιτισμικής παραγωγής και αναπαραγωγής που προέρχονται από αυτήν. Στο επίκεντρο της θεωρίας του Lévi-Strauss βρίσκεται η άποψη πως «η ανθρώπινη σκέψη διέπεται από εννοιολογικές διχοτομίες, όπως πολιτισμός-φύση, άνδρας-γυναίκα, μέρα-νύχτα και ζωή-θάνατος. Η αρχή των αντιθέσεων είναι καθολικό χαρακτηριστικό εγγενές στον ανθρώπινο εγκέφαλο και ο πολιτισμός βασίζεται σε μια μοναδική επιλογή αντιθέσεων¹⁸³. Αυτή η προσέγγιση έγινε αρχικά από την ανθρωπολογία και εφαρμόστηκε για αρχαιολογικές ερμηνείες από τον Γάλλο αρχαιολόγο André Leroi-Gourhan (1911-1986), που τη χρησιμοποίησε για να ερμηνεύσει προϊστορικά θρησκευτικά σύμβολα¹⁸⁴.

Οι μεταδιαδικαστικοί υιοθέτησαν, επίσης, αντιλήψεις σχετικές με την ανθρώπινη δράση (human agency), υποστηρίζοντας ότι σε άλλες θεωρητικές προσεγγίσεις της αρχαιολογίας, όπως είναι η πολιτισμική-ιστορική και η διαδικαστική, «το άτομο χάθηκε» και οι άνθρωποι απεικονίζονται ως «παθητικά

¹⁸⁰ Johnson 2010, 102.

¹⁸¹ Knapp 1998, 99, 125, 176 και 192.

¹⁸² Pearson 1984, 69-92.

¹⁸³ Trigger 2007, 463.

¹⁸⁴ Moro Abadía and Palacio-Pérez 2015, 1-16.

υποκείμενα που ακολουθούν τυφλά κοινωνικούς κανόνες¹⁸⁵. Το μεταδιαδικαστικό ρεύμα υποστηρίζει ότι οι άνθρωποι είναι ελεύθερα δρώντα άτομα, που σε πολλές περιπτώσεις δρουν για τα δικά τους συμφέροντα και όχι απλώς ακολουθώντας τους κοινωνικούς κανόνες. Αποδεχόμενοι αυτές τις ιδέες, οι μεταδιαδικαστικοί θεωρούν ότι η κοινωνία είναι καθοδηγούμενη από διάφορες μορφές συγκρούσεων¹⁸⁶. Επηρασμένοι από τον κοινωνιολόγο Anthony Giddens και τη θεωρία της δόμησης, πολλοί μεταδιαδικαστικοί αποδέχτηκαν ότι οι περισσότεροι άνθρωποι, ενώ γνωρίζουν και κατανοούν τους κανόνες της κοινωνίας τους, επιλέγουν να τους χειραγωγούν, αντί να τους ακολουθούν υπάκουα. Τέτοιου είδους χειραγωγήση προκαλεί κάμψη των κοινωνικών κανόνων, έως ότου τελικά αλλάξουν¹⁸⁷. Η θεωρία της δόμησης είναι μια κοινωνική θεωρία δημιουργίας και αναπαραγωγής των κοινωνικών συστημάτων, που βασίζεται στην ανάλυση τόσο της δομής όσο και της ανθρώπινης δράσης, δίχως να δίνει προτεραιότητα σε κάποια από τις δύο συνιστώσες. Η θεωρία της δομής εξετάζει τη φαινομενολογία, την ερμηνευτική και τις κοινωνικές πρακτικές στην αδιάσπαστη σύνθεση δομής και της ανθρώπινης δράσης. Αν και ως θεωρία έχει δεχθεί μεγάλη κριτική, παραμένει πυλώνας της σύγχρονης κοινωνιολογικής προσέγγισης¹⁸⁸.

Άλλοι μεταδιαδικαστικοί στρέφονται στις απόψεις του Karl Marx για την ταξική σύγκρουση ως δύναμη της κοινωνικής αλλαγής¹⁸⁹ και έτσι μοιράζονται ομοιότητες με τις απόψεις μαρξιστών αρχαιολόγων. Ορισμένοι αρχαιολόγοι του μεταδιαδικαστικού ρεύματος, όπως ο Julian Thomas, υποστήριξαν ότι η ανθρώπινη δράση δεν είναι απαραίτητη για την εξέταση των κοινωνιών του παρελθόντος, αποδεχόμενοι έτσι μια πολιτισμικά ντετερμινιστική θέση¹⁹⁰.

Η μεταδιαδικαστική αρχαιολογία δίνει μεγάλη έμφαση στην ενθάρρυνση των περιθωριοποιημένων ομάδων και την αλληλεπίδρασή τους με την αρχαιολογία. Κατά τη δεκαετία του 1960 και του 1970 εμφανίστηκε η φεμινιστική αρχαιολογία και οι οπαδοί της, άρχισαν να υποστηρίζουν πως η παρουσία των γυναικών στο αρχαιολογικό αρχείο έχει αγνοηθεί από τους αρχαιολόγους. Σύμφωνα με την αρχαιολόγο Sam Lucy, «τα ζητήματα που

¹⁸⁵ Johnson 2010, 104.

¹⁸⁶ Johnson 2010, 105.

¹⁸⁷ Johnson 2010, 104.

¹⁸⁸ Bryant and Jary 1991, 1-32

¹⁸⁹ Trigger 2007, 469.

¹⁹⁰ Thomas 2000, 149-150.

διερευνά η φεμινιστική αρχαιολογία και το μεταδιαδικαστικό ρεύμα σκέψης υπογραμμίζουν τη σημασία των κοινωνικών και πολιτικών παραγόντων πίσω από την υποτιθέμενη «αντικειμενική» διερεύνηση¹⁹¹.

Μία άλλη περιθωριοποιημένη ομάδα σε σχέση με την αρχαιολογία είναι οι αυτόχθονες πληθυσμοί. Το μεταδιαδικαστικό ρεύμα αγκαλιάζει τη σχολή της αρχαιολογίας των αυτοχθόνων. Η αρχαιολογία των αυτοχθόνων ως σχολή σκέψης επιδιώκει να εμπλέξει και να ενισχύσει τους αυτόχθονες πληθυσμούς στη διατήρηση της κληρονομιάς τους και να διορθώσει τις αντιληπτές ανισότητες στη σύγχρονη αρχαιολογία. Προσπαθεί, επίσης, να ενσωματώσει μη υλικά στοιχεία των πολιτισμών, όπως οι προφορικές παραδόσεις, στην ευρύτερη ιστορική αφήγηση. Η συγκεκριμένη μεθοδολογία προήλθε από τα παγκόσμια αντιαποικιακά κινήματα της δεκαετίας του 1970 και της δεκαετίας του '80, με επικεφαλής τους αυτόχθονες πληθυσμούς αποικιακών κρατών, όπως οι Ηνωμένες Πολιτείες, ο Καναδάς και η Αυστραλία. Σημαντικά θέματα που επιχειρεί να αντιμετωπίσει η αρχαιολογία των αυτοχθόνων είναι ο επαναπατρισμός των τεχνέργων στους αντίστοιχους λαούς τους και η διατήρηση των πολιτισμών και των τόπων πολιτιστικής κληρονομιάς των ιθαγενών. Οι στόχοι της ενθάρρυναν την ανάπτυξη σχέσεων συνεργασίας μεταξύ αρχαιολόγων και ιθαγενών και αύξησε τη συμμετοχή των αυτοχθόνων στην αρχαιολογία και τις σχετικές πολιτικές της¹⁹².

Βέβαια, η αυστηρή κριτική που άσκησε το μεταδιαδικαστικό ρεύμα στη διαδικαστική αρχαιολογία δεν έμεινε αναπάντητη, καθώς θεωρήθηκε υπερβολικά αλαζονική. Όπως επεσήμαναν οι Colin Renfrew και Paul Bahn, για τους σοβαρότερους επικριτές του, το μεταδιαδικαστικό ρεύμα ανέπτυξε έγκυρη κριτική σε ορισμένα ζητήματα, ωστόσο απλά ανέπτυξε ορισμένες από τις ιδέες και τα θεωρητικά προβλήματα που εισήγαγε η διαδικαστική αρχαιολογία. Επίσης, οι ίδιοι επικριτές του θεωρούν ότι εισήγαγε ποικιλία προσεγγίσεων από άλλες θεωρητικές προσεγγίσεις, έτσι ώστε ο όρος μεταδιαδικαστικός, να φαντάζει ως αλαζονική σκιά, όταν υποθέτει ότι αντικατέστησε αυτό που θα μπορούσε να ισχυριστεί ορθά ότι συμπληρώνει¹⁹³.

¹⁹¹ Lucy 1997, 153.

¹⁹² Murray 2011, 363-378.

¹⁹³ Renfrew and Bahn 2004, 44.

Οι Timothy Earle και Robert Preucel εξετάζοντας τη «ριζοσπαστική κριτική» του μεταδιαδικαστικού κινήματος στη διαδικαστική αρχαιολογία δέχονται μεν ότι είχε σχετική αξία και ότι υπογράμμισε ορισμένα σημαντικά σημεία, αλλά κατέληξαν στο συμπέρασμα πως σε γενικές γραμμές η μεταδιαδικαστική προσέγγιση ήταν λανθασμένη, αφενός γιατί δεν έπεισε ότι είναι ικανή να δεισδύσει στον προϊστορικό νου και αφετέρου επειδή απέρριψε εμφανώς τη θεωρία και αγνόησε τη διαμόρφωση μιας αναπαράξιμης και επαληθεύσιμης μεθοδολογίας¹⁹⁴.

Η ριζοσπαστική προσέγγιση κατά τους επικριτές της φαίνεται να αποκλείει το ενδιαφέρον για την ανάπτυξη θεωριών της ανθρώπινης συμπεριφοράς. Ως παράδειγμα παραθέτουν τη δήλωση του Tilley ότι «δεν έχει νόημα να προσπαθήσουμε να διαμορφώσουμε ένα γενικό μοντέλο αλλαγής, αφού όλες οι αλλαγές πραγματοποιούνται μέσα στα πλαίσια καθορισμένων κοινωνικών σχηματισμών και οι δομές, οι δομικές αρχές και οι συνθήκες για κοινωνική δράση διαφέρουν από μία συγκεκριμένη περίπτωση στην άλλη»¹⁹⁵. Θεωρούν επίσης ότι η θέση του Hodder επί του ζητήματος δεν είναι καθόλου σαφής. Στις θεωρητικές συνεισφορές αναγνωρίζει τη σημασία των θεωριών της γενικής κοινωνικής διαδικασίας, αλλά στις αρχαιολογικές εφαρμογές έχει δείξει μεγαλύτερο ενδιαφέρον για την ανασυγκρότηση συγκεκριμένων πλαισίων, παρά για την ερμηνεία των διαπολιτισμικών κανονικοτήτων¹⁹⁶.

Ερμηνευτική προσέγγιση

Η βασική θεωρητική προσέγγιση της μεταδιαδικαστικής αρχαιολογίας είναι γνωστή ως ερμηνευτική προσέγγιση (hermeneutic approach) και προέρχεται από το χώρο της ερμηνευτικής. Η ερμηνευτική προσέγγιση στο μεταδιαδικαστικό ρεύμα συνδέεται με την προαγωγή της γνώσης και την άσκηση κοινωνικής κριτικής στο παρόν σύστημα. Μέσω της ερμηνευτικής προσέγγισης οι αρχαιολόγοι επιχειρούν να ερμηνεύσουν το νοηματικό πλαίσιο μέσα στο οποίο έδρασαν τα άτομα που δημιούργησαν τον υλικό πολιτισμό του παρελθόντος.

Το πλαίσιο καθορίζεται από τη λογική και τη συνοχή της συμβολικής δομής ενός παρελθόντος πολιτισμού, διαφορετική από την παρούσα. Συνεπώς

¹⁹⁴ Earle and Preucel 1987, 501-538.

¹⁹⁵ Tilley 1982, 26-38.

¹⁹⁶ Hodder 1982b, 176.

χρειάζεται ερμηνεία, δεδομένου ότι ο υλικός πολιτισμός του παρελθόντος αντιμετωπίζεται ως «κείμενο» προς ερμηνεία. Η συγκεκριμένη αρχαιολογική ερμηνευτική προσπάθεια ονομάζεται ερμηνευτική του παρελθόντος. Για να γίνει κατανοητό το νοηματικό πλαίσιο του παρελθόντος χρειάζεται, επίσης, να ερμηνευθεί και το νοηματικό πλαίσιο του παρόντος. Είναι εκείνο που περιβάλλει τους σύγχρονους αρχαιολόγους και τους ακολουθεί στην προσπάθειά τους να ερμηνεύσουν το «κείμενο» του αρχαίου υλικού πολιτισμού. Οι όποιες αρχαιολογικές ερμηνείες εκκινούν ήδη από το στάδιο της έρευνας στο πεδίο και κατασκευάζονται στο παρόν¹⁹⁷. Η κατασκευή γίνεται υπό την επίδραση συμβολικών δομών που ισχύουν στη σύγχρονη κοινωνία και συμπεριλαμβάνοντας τα νοήματα των ίδιων των αρχαιολόγων. Η ερμηνεία είναι μετάφραση. Περιλαμβάνει τον αρχαιολόγο που ενεργεί ως διερμηνέας μεταξύ του παρελθόντος και του παρόντος, μεταξύ διαφορετικών απόψεων για το παρελθόν και μεταξύ του ειδικού και του γενικού. Συνεπώς, η ερμηνεία περιλαμβάνει την ακρόαση, την κατανόηση και τη συνύπαρξη διαφορετικών φωνών και όχι μόνο την εφαρμογή καθολικών οργάνων μέτρησης¹⁹⁸.

Επιπλέον, θα μπορούσε κανείς να πει ότι η αρχαιολογική ερμηνεία, ως ακαδημαϊκή γνώση, αποτελεί πολιτική πράξη. Η ερμηνευτική θεωρία υποστηρίζει ότι οι ερμηνείες και οι χρήσεις του παρελθόντος δεν είναι ελεύθερες αξίας, αλλά επηρεάζονται από τις σύγχρονες πολιτικές ατζέντες διότι κατασκευάζονται μέσα σε ένα πεδίο σχέσεων πολιτικής εξουσίας¹⁹⁹. Η ερμηνεία δεν είναι απλά ένα εργαλείο στην εργαλειοθήκη της μετάφρασης. Η έμφαση που δίνεται στη σημασία του πολύπλοκου κοινωνικού και πολιτισμικού περιβάλλοντος, η έμφαση που δίνεται στη διάσταση της γλώσσας και η χρήση της στην πολιτική έρευνα, υποδηλώνουν πως είναι εναλλακτική λύση στις εμπειρικές και ορθολογικές προσεγγίσεις της πολιτικής έρευνας²⁰⁰ και συνεπώς της αρχαιολογίας ως πολιτικής πράξης.

¹⁹⁷ Hodder 1999, 92

¹⁹⁸ Hodder 1991, 7-18.

¹⁹⁹ Wylie 1985, 133-147.

²⁰⁰ Gibbons 2006, 563-571.



Εικ. 2.2. Η σχέση θεωρίας και δεδομένων στη μεταδιαδικαστική αρχαιολογία. Τα δεδομένα υπάρχουν, αλλά γίνονται αντιληπτά μέσω ενός νέφους θεωρίας²⁰¹.

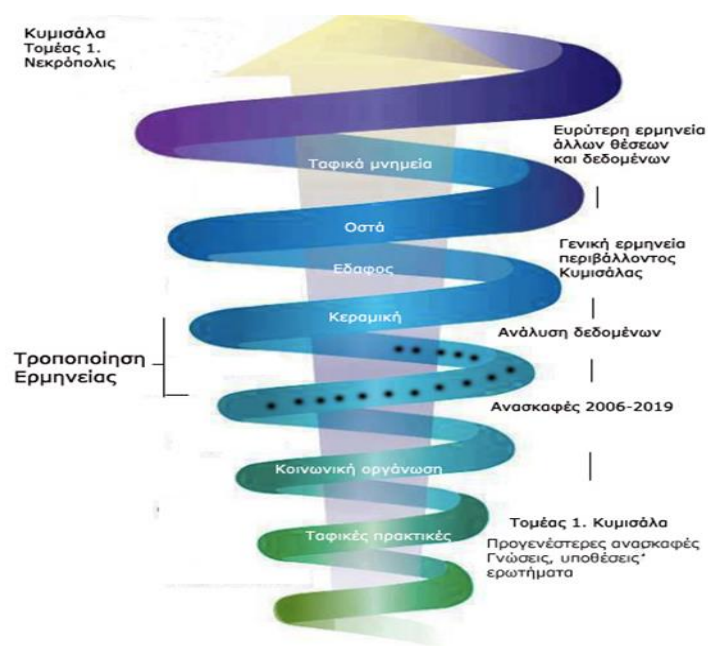
Η αρχαιολογία, σύμφωνα με την άποψη των μεταδιαδικαστικών αρχαιολόγων θα πρέπει να προσπαθεί να ερμηνεύσει το πολιτικό νόημα που κρύβεται πίσω από τις σύγχρονες αρχαιολογικές ερμηνείες και η συγκεκριμένη ερμηνευτική προσπάθεια της αρχαιολογίας αποκαλείται ερμηνευτική του παρόντος. Βλέπουμε, λοιπόν, ότι η αρχαιολογία ασχολείται με μια διπλή ή ακόμα και τετραπλή ερμηνευτική κατά Shanks and Tilley. Οι συγγραφείς περιέγραψαν μια τετραπλή ερμηνευτική στην αρχαιολογία, δηλαδή, τέσσερα επίπεδα ερμηνείας και την ανάγκη κατανόησής τους. Κατά την άποψή τους χρειάζεται κατανόηση της σχέσης μεταξύ του παρελθόντος και του παρόντος, κατανόηση άλλων κοινωνιών και πολιτισμών, κατανόηση της σύγχρονης κοινωνίας, του τύπου της αρχαιολογικής ερμηνείας και κατανόηση των κοινοτήτων των αρχαιολόγων που εκτελούν ερμηνείες. Έτσι, όχι μόνο οι αρχαιολόγοι μεταφράζουν μεταξύ του «κόσμου τους» και του «κόσμου μας», αλλά αντιμετωπίζουν και κόσμους διαχωρισμένους στον χώρο και τον χρόνο²⁰².

Δεδομένου ότι η αρχαιολογική ερμηνευτική διακλαδίζεται και είναι ριζωμένη ταυτόχρονα στο πλαίσιο αναφοράς του παρελθόντος και σε αυτό του παρόντος, έπεται ότι, σε αντίθεση με ό,τι ισχύει στη διαδικαστική αρχαιολογία, τα δεδομένα και η θεωρία, το αντικείμενο και το υποκείμενο, το παρελθόν και το παρόν συνυφαίνονται και δεν είναι δυνατόν να διαχωριστούν και να αντιδιαστέλλονται μεταξύ τους. Συνεπώς τα αρχαιολογικά δεδομένα δεν είναι

²⁰¹ Johnson 2010, 103 fig. 7.2.

²⁰² Shanks and Tilley 1992, 107-108.

ανεξάρτητα από τη θεωρία, αλλά είναι συνυφασμένα σε ένα νέφος θεωρίας (εικ. 2.2.)²⁰³. Η ερμηνευτική προσέγγιση της μεταδιαδικαστικής αρχαιολογίας είναι δυνατόν να απεικονιστεί με σπειροειδές, στο οποίο εμπεριέχεται η ιδέα της τροποποίησης της ερμηνείας, καθώς η ερμηνεία δεν επιστρέφει στο σημείο εκκίνησής της στον ερμηνευτικό κύκλο²⁰⁴.



Εικ. 2.3. Μεταδιαδικαστική ερμηνευτική προσέγγιση της νεκρόπολης της Κυμισάλας. Προσαρμογή από Hodder 1995 εικ. 22²⁰⁵.

Σύμφωνα με τη συγκεκριμένη προσέγγιση οι αρχαιολόγοι αρχίζουν με μια θεωρία που παράγεται από προγενέστερη γνώση και προχωρούν ακολουθώντας το σπειροειδές, μεταβάλλοντας συνεχώς την αρχική τους θεωρία, καθώς έρχονται σε επαφή με τα νέα δεδομένα που προκύπτουν από την ανασκαφική διαδικασία ή τον σχετικό ορίζοντα πληροφόρησης.

Στη διαδρομή του σπειροειδούς ο αρχαιολόγος επιχειρεί διάλογο, ανάμεσα στα δεδομένα και τη θεωρία, το μέρος του συνόλου και το σύνολο, το παρελθόν και το παρόν, έως ότου καταλήξει σε ερμηνεία που παρέχει νόημα στην πλειονότητα των δεδομένων. Η συγκεκριμένη προσέγγιση καθιστά προσωρινή

²⁰³ Κουκουζέλη κ.ά. 2003, 275.

²⁰⁴ Hodder 1995, 204.

²⁰⁵ Hodder 1995, 206.

την αρχαιολογική ερμηνεία και τροποποιήσιμη, έως ότου ανακαλυφθούν περισσότερα δεδομένα²⁰⁶.

Τέλος, σύμφωνα με την ερμηνευτική προσέγγιση που προτείνει η μεταδιαδικαστική αρχαιολογία, η αρχαιολογική ερμηνεία δεν είναι αντικειμενική ή αυθαίρετη, αλλά είναι στοχαστική (*reflexive*). Επιχειρώντας διάλογο ανάμεσα στη θεωρία και τα δεδομένα, οι αρχαιολόγοι διαχειρίζονται παραγωγικά τις αξίες και τις προκατειλημμένες ιδέες τους ως διαμεσολαβητές ανάμεσα στο παρελθόν και το παρόν. Δεν ακολουθούν, όμως, άκριτα τις προκατειλημμένες ιδέες τους, αλλά τις εξετάζουν σε βάθος και τις προσαρμόζουν στην ποικιλία των ερμηνειών που επιδέχονται τα δεδομένα, ανάλογα με το πλαίσιο με το οποίο σχετίζονται.

Η μεταδιαδικαστική κριτική οδήγησε ορισμένους μεταδιαδικαστικούς αρχαιολόγους Shanks, Tilley και Miller στη διατύπωση της κριτικής αρχαιολογίας. Η κριτική αρχαιολογία, σύμφωνα με τους υποστηρικτές της, διατηρεί και επαναβεβαιώνει τη δέσμευση της αρχαιολογίας για την κατανόηση του παρελθόντος. Ωστόσο, η κατασκευή του παρελθόντος είναι ζήτημα συνειδητής επιλογής και ιδεολογίας. Ο όρος ιδεολογία, που χρησιμοποιείται πλέον τόσο στη συμβολική ανθρωπολογία όσο και από τον Hodder (1982, 1984, 1985), δε σημαίνει το ίδιο πράγμα για όλους. Για τον θεωρητικό της μεταδιαδικαστικής κριτικής η ιδεολογία παρέχει ένα μεγάλο μέρος του νοήματος σε μια κοινωνία, αν και το νόημα δεν είναι απαραίτητα ιδεολογία στο σύνολό του. Η έννοια της ιδεολογίας χρησιμοποιήθηκε στην αρχαιολογία από τους Tilley (1982), τον Miller (1982), Shanks and Tilley (1982), Shennan (1982), Miller and Tilley (1984) και άλλους για περιγράψει την ανισότητα. Η ανισότητα συνεπάγεται την αποξένωση από την εργασία και τη χρήση αγαθών ή υπηρεσιών χωρίς πλήρη απόδοση της αξίας τους στους παραγωγούς. Η λέξη που αποδίδει αυτή την ασύμμετρη σχέση είναι η εκμετάλλευση. Η ιδεολογία κρύβει και καλύπτει την εκμετάλλευση, μετατρέποντας τη σε φυσική ή υπερφυσική διαδικασία. Η ιδεολογία ενεργεί μέσα σε μια ιεραρχικά διαστρωματωμένη ή ταξική κοινωνία, για να αναπαραγάγει την ανισότητα χωρίς σοβαρή αντίσταση, βία ή επανάσταση²⁰⁷.

Η κριτική αρχαιολογία, σύμφωνα με τους υποστηρικτές της, παρέχει τη δυνατότητα στους αρχαιολόγους να δραστηριοποιηθούν προς την κατεύθυνση

²⁰⁶ Κουκουζέλη κ.ά. 2003, 276.

²⁰⁷ Leone et al 1987, 283-302.

της ανατροπής του σύγχρονου καπιταλιστικού καθεστώτος, μέσω της ανάπτυξης μιας δημοκρατικής και κριτικής κοινωνικής αρχαιολογίας, η οποία θα ασκεί έντονη κριτική όχι μόνο στο παρελθόν και το παρόν, αλλά και στον ίδιο τον εαυτό της ως μικροπολιτικό πεδίο. Η συγκεκριμένη αντίληψη της αρχαιολογίας ως δύναμη αλλαγής και πολιτικό πρόγραμμα επικρίθηκε από τον Ian Hodder, ο οποίος πιστεύει ότι μια τέτοιου είδους αρχαιολογία είναι καθαρά δογματική και ότι δίνει υπερβολική έμφαση στην κατανόηση του παρόντος και του εαυτού και πολύ λιγότερη στην οικοδόμηση ασφαλούς γνώσης σχετικά με το παρελθόν.

Παρακάτω παρατίθεται συνοπτικός πίνακας των διαφορετικών αρχαιολογικών σχολών και περιόδων που θεμελίωσαν και διαμόρφωσαν τις θεωρητικές απόψεις της σύγχρονης αρχαιολογικής θεωρίας.

Σχολές Αρχαιολογικής Θεωρίας			
Περίοδος	Μέθοδος	Στόχοι	Εκπρόσωποι
Αρχαιοδιφία 16ος έως 19ος αι.	Περισυλλογή αρχαιοτήτων	1. Δημιουργία συλλογών τεχνέργων. 2. Παρατήρηση και περιγραφή μνημείων. 3. Cabinets of Curiosity	<i>Grand Tour</i> <i>Παπική Αυλή</i> Thomas Howard, 21st Earl of Arundel. Robert le Prévôt de Cocherel. Bernard de Montfaucon Anne Claude de Caylus. Nicolas Revett.
Πολιτισμική Εξελικτική Αρχαιολογία Τέλη του 19ου αι.	Πολιτισμική εξελικτική προσέγγιση	Ερμηνεία πολιτισμικής αλλαγής	J. Lubbock E.B. Tylor L.H. Morgan
Πολιτισμική Ιστορική Αρχαιολογία	Πολιτισμική ιστορική προσέγγιση	1. Περιγραφή μορφής της αρχαιολογικής μαρτυρίας. 2. Προσδιορισμός ακολουθίας πολιτισμών σε γεωγραφική περιοχή 3. Ανασύσταση	Oscar Montelius Franz Boas V.G. Childe

		πολιτισμικής ιστορίας γεωγραφικής περιοχής	
Νέα ή Διαδικαστική Αρχαιολογία	Συστημική προσέγγιση	1. Ανασύσταση της λειτουργίας της αρχαιολογικής μαρτυρίας. 2. Ερμηνεία των αρχαίων πολιτισμών και της πολιτισμικής αλλαγής.	Lewis Binford David Clarke
Μεταδιαδικαστική αρχαιολογία	Ερμηνευτική προσέγγιση	1. Ανασύσταση του νοήματος της αρχαιολογικής μαρτυρίας. 2. Ερμηνεία των αρχαίων πολιτισμών και της πολιτισμικής αλλαγής 3. Άρθρωση κριτικού λόγου στο παρελθόν και το παρόν και επίτευξη κοινωνικής αλλαγής.	Ian Hodder Michael Shanks Christopher Tilley

2.6. Ειδικά ζητήματα αρχαιολογικής θεωρίας

Από τις αρχές του 20ού αιώνα, οι περισσότερες αφηγήσεις για την αρχαιολογική μεθοδολογία έχουν αποδεχθεί ότι τα δεδομένα που αποκαλύπτονται από τους αρχαιολόγους ερμηνεύονται στη συνέχεια βάσει θεωρητικής άποψης²⁰⁸. Παρόλα αυτά, η αρχαιολογική κοινότητα φαίνεται διχασμένη ως προς βαθμό με τον οποίο διαπερνά η θεωρία την αρχαιολογική επιστήμη. Από τη μία στέκουν εκείνοι που πιστεύουν ότι ορισμένες αρχαιολογικές τεχνικές -όπως η ανασκαφή ή η καταγραφή- είναι ουδέτερες διαδικασίες, έξω από τα όρια της θεωρίας. Από την άλλη στέκουν εκείνοι που πιστεύουν ότι και οι τεχνικές επηρεάζονται από θεωρητικές εκτιμήσεις²⁰⁹.

²⁰⁸ Hodder 1999, 80.

²⁰⁹ Johnson 2010, 2.

Ο Ian Hodder, διακεκριμένος υποστηρικτής της τελευταίας άποψης, υπογραμμίζει ότι μεθοδολογικές αποφάσεις, όπως το σημείο ανοίγματος ενός σκάμματος, το πόσο επιμελώς ανασκάπτεται ένα στρώμα και αν πρέπει να φυλάσσεται κάθε εύρημα που εντοπίζεται, οι θεωρητικές ερμηνείες του τόπου, ακόμη και οι τεχνικές της ανασκαφής, δεν μπορούν να ξεφύγουν από τη σφαίρα της θεωρίας²¹⁰. Όσοι υιοθετούν την αντίθετη άποψη, προσπαθούν ενίοτε να διαχωρίσουν τα ακατέργαστα δεδομένα από τις θεωρητικές ερμηνείες στις δημοσιεύσεις τους, αλλά αντιμετωπίζονται με σκεπτικισμό από εκείνους που υποστηρίζουν ότι η θεωρητική ερμηνεία διαπερνά ολόκληρη την αρχαιολογική μεθοδολογία και επομένως δεν μπορεί να διαχωριστεί από τα πρωτογενή δεδομένα²¹¹.

Αυτό το σύνθετο και μεταβαλλόμενο καλειδοσκόπιο ενδιαφερόντων, πλαισίων και συζητήσεων διαμορφώνει ένα πολυπλοκότερο θεωρητικό πεδίο από αυτό που υπονοούν τα απλά θεωρητικά διαγράμματα της αντίληψης «θεωρία έναντι των δεδομένων» ή οι συνθέσεις που προτείνουν οι Trigger²¹² και Kristiansen²¹³, αν και κανένας από τους δύο δεν επιχειρεί να χαρτογραφήσει πλήρως το έδαφος της αρχαιολογικής θεωρίας. Σε αυτή την περίπλοκη εικόνα υπεισέρχονται νέες τάσεις στην αρχαιολογική θεωρία, όπως είναι η θεωρία της δράσης (agency theory) και η φαινομενολογία, εισάγοντας νέα ζητήματα και εντάσεις σε αυτό το περίπλοκο και μεταβαλλόμενο έδαφος. Η θεωρία της δράσης και η φαινομενολογία έχουν ποικίλες ιστορικές καταβολές και εφαρμόστηκαν στο αρχαιολογικό αρχείο με διάφορους τρόπους. Ωστόσο, η συσχέτιση και των δύο με την αρχαιολογία προήλθε, εν μέρει, από στοιχεία της μεταδιαδικαστικής κριτικής.

Γενικά, η γέννηση της θεωρίας της δράσης αντικατοπτρίζει την επιθυμία να αντιμετωπιστούν τα ντετερμινιστικά πρότυπα της ανθρώπινης δράσης, αναγνωρίζοντας ότι οι άνθρωποι ενεργούν και αλλάζουν τον εξωτερικό κόσμο με τις ενέργειές τους. Με βάση αυτή την αναγνώριση, οι κοινωνικοί επιστήμονες προσπάθησαν να κατασκευάσουν και να ενσωματώσουν θεωρίες σχετικά με τους τρόπους με τους οποίους οι ανθρώπινες ενέργειες περιορίζονται,

²¹⁰ Hodder 1999, 80–82.

²¹¹ Hodder 1999, 80–81.

²¹² Trigger 1989, 20.

²¹³ Kristiansen 2004, 81.

ενεργοποιούνται, κατασκευάζονται και εκδηλώνονται μέσα σε μεγαλύτερα κοινωνικά συστήματα. Στο επίκεντρο της θεωρίας της δράσης βρίσκεται, επομένως, η βασική συμφωνία ότι οι άνθρωποι δεν είναι ομοιόμορφα αυτόματα που απλώς αντιδρούν στις αλλαγές του εξωτερικού κόσμου αλλά «παίζουν ρόλο στη διαμόρφωση της κοινωνικής πραγματικότητας στην οποία συμμετέχουν»²¹⁴. Η θεωρία της δράσης εστιάζει τόσο στην «επίδραση του συστήματος στην πρακτική όσο και στην επίδραση της πρακτικής στο σύστημα»²¹⁵.

Η φαινομενολογία, ως φιλοσοφική θεωρία και μέθοδος, αφορά στη μελέτη της ανθρώπινης εμπειρίας και συνείδησης στην ανθρώπινη καθημερινότητα. Με αυτόν τον τρόπο είναι δυνατόν να κατανοήσει κανείς τα νοήματα που εμπεριέχονται σε όλα τα πράγματα, εφόσον το ανθρώπινο πνεύμα είναι ικανό να προχωρήσει πέρα από την εμπειρική γνώση στην αληθινή κατανόηση της ουσίας των πραγμάτων, αντίθετα από τον νεοθετικισμό, ο οποίος δίνει έμφαση στη φυσική πλευρά των πραγμάτων. Στην αρχαιολογία, η φαινομενολογία κάνει χρήση της αισθητηριακής εμπειρίας για την ερμηνεία ενός αρχαιολογικού χώρου ή ενός πολιτιστικού τοπίου. Η φαινομενολογία πήρε τη θέση της στην αρχαιολογική θεωρία με τη δημοσίευση του *The Phenomenology of Landscape* του C. Tilley, το 1994, στην οποία πρότεινε την αισθητηριακή εμπειρία ως χρήσιμη τεχνική για την κατανόηση της αλληλεπίδρασης των ιστορικών λαών με τα τοπία που τους περιβάλλουν.

Η φαινομενολογία προκάλεσε έντονες συζητήσεις στο πλαίσιο της αρχαιολογικής θεωρίας²¹⁶ και η σημαντικότερη πιθανώς κριτική που δέχθηκε και δέχεται από την αρχαιολογική κοινότητα επικεντρώνεται στο ότι είναι «μη επιστημονική» και «υποκειμενική»²¹⁷. Στον αντίποδα, αρκετοί αρχαιολόγοι χρησιμοποιούν τη φαινομενολογική θεώρηση στο πεδίο, παράλληλα με άλλες παραδοσιακές μεθόδους για την κατανόηση προϊστορικών θέσεων, όπως η νεολιθική Tavoliere στην Ιταλία²¹⁸, ή το Bodmin Moor της Εποχής του Χαλκού στην Αγγλία²¹⁹.

²¹⁴ Barfield 1997, 4.

²¹⁵ Ortner 1984, 148.

²¹⁶ Brück 2005, 45.

²¹⁷ Hamilton & Whitehouse 2006, 31-32.

²¹⁸ Στο ίδιο.

²¹⁹ Tilley 1996, 161-176.

Η θεωρία της ανθρώπινης δράσης (agency theory)

Η θεωρία της ανθρώπινης δράσης, όσο πολύπλοκη και αν είναι στις επιπτώσεις της, είναι στην ουσία της πολύ απλή²²⁰. Είναι καθολικά αναγνωρισμένη αλήθεια ότι τα κεραμικά παράγονται από ανθρώπους. Με άλλα λόγια, το αρχαιολογικό αρχείο δημιουργείται από ανθρώπους, μέσα από μια σειρά ατομικών συμπεριφορών και ενεργειών. Ωστόσο, τα ανθρώπινα όντα δεν παράγουν κεραμική σε συνθήκες που επιλέγουν. Η κοινωνία δεν είναι ποτέ εντελώς ελεύθερη για όλους. Αν ήταν, δεν θα ήταν κοινωνία. Οι άνθρωποι περιορίζονται από μια ολόκληρη σειρά διαφορετικών δομών και διαρθρωτικών σχέσεων, ορισμένες από τις οποίες είναι τεχνολογικές, ορισμένες φυσικές και ορισμένες κοινωνικοπολιτισμικές.

Ένας τρόπος κατανόησης της θεωρίας της δράσης είναι η «θεωρία της δόμησης», όρος που επινοήθηκε από τον κοινωνιολόγο Anthony Giddens (1984), που βασίστηκε στο έργο του Pierre Bourdieu (1977) και άλλων. Η θεωρία της δόμησης του Giddens προτείνει ότι η δράση και η δομή συνδέονται διαλεκτικά. Τα δρώντα άτομα έχουν τους δικούς τους στόχους και επιλέγουν στρατηγικές για την επίτευξη αυτών των στόχων, αλλά το κάνουν σε μια κοινωνική δομή. Η δομή αυτή, ωστόσο, δεν είναι εξωτερικά και ντετερμινιστικά επιβεβλημένη, αλλά μάλλον βρίσκεται σε κατάσταση διαρκούς αναπαραγωγής και αναδιαπραγμάτευσης μέσω των πολλών μικρών δράσεων ατόμων και ομάδων. Για τους Giddens, Bourdieu και άλλους οι ιδέες της συνήθειας, η καθημερινή δράση και ούτω καθεξής είναι σημαντικές για να καταδείξουν πως υπάρχει τάση να αναπαράγουν τα δρώντα άτομα τη δομή και όχι να τη μετασχηματίζουν²²¹.

Αυτό που είναι απλή παρατήρηση για τον θεωρητικό της αρχαιολογίας οδηγεί σε εννοιολογικά ζητήματα μεγάλης πολυπλοκότητας. Αρχικά, υπάρχει ένα ερώτημα για το πώς αντιλαμβάνεται κανείς τη σχετική ισχύ και τη διαμόρφωση της δράσης και της δομής. Αυτό εξαρτάται από ιστορικά συγκεκριμένα ερωτήματα και πολιτισμικούς παράγοντες που ποικίλλουν από κοινωνία σε

²²⁰ Robb, 2004, 105.

²²¹ Johnson 2006, 117–132.

κοινωνία, όπως η δύναμη της ιδέας του ατόμου ή η φύση και η δομή της εξουσίας.

Ασκώντας κριτική στο μοντέλο Giddens-Bourdieu, ο Adam Smith επισημαίνει πως το «βασικό αρχαιολογικό υποκείμενο», εκτός του ότι δεν είναι ιστορικά ιδιαίτερο και πολιτισμικά μεταβαλλόμενο, όπως υποστηρίζει η μεταδιαδικαστική αφήγηση σε πολεμικές δηλώσεις της, ταξινομείται συνήθως σε άξονες τάξης, φυλής και φύλου, ανεξάρτητα από το ιδιαίτερο πολιτισμικό πλαίσιο²²². Έχει υποστηριχθεί, επίσης, η ιδέα της «ιστορικότητας της δράσης», δηλαδή η πιθανότητα ότι τα δρώντα άτομα διαμορφώνονται διαφορετικά σε διαφορετικούς πολιτισμούς και ιστορικές καταστάσεις και ότι το αρχαιολογικό αρχείο μπορεί να χρησιμοποιηθεί για να διερευνήσει αυτές τις διαμορφώσεις και τον τρόπο με τον οποίο υλοποιούνται, για παράδειγμα η αρχιτεκτονική και το τοπίο²²³. Η φεμινιστική σχολή με τη σειρά της επισημαίνει ότι η κατασκευή ενός μοντέλου ουδέτερου φύλου είναι προβληματική²²⁴. Εντέλει, η ιδέα της δομής σε οποιοδήποτε μοντέλο δράσης χρειάζεται εκτεταμένη ανάλυση και σχόλια. Εάν οι δομές χρειάζονται εκτεταμένη ανάλυση, έτσι και τα δρώντα άτομα. Υπό την οπτική γωνία του «αρχαιολογικού υποκειμένου» ακόμα και τα τέχνηρα μπορούν να θεωρηθούν δρώντα υποκείμενα με κάποια έννοια²²⁵, εξ ου και οι πρόσφατες συζητήσεις για τις βιογραφίες των τεχνέρων²²⁶. Στην εθνοαρχαιολογική μελέτη, οι πρόγονοί του μακρινού παρελθόντος θεωρήθηκαν αφεαυτού ενεργά δρώντα άτομα²²⁷.

Τα παραπάνω είναι σημαντικά και καθιστούν το απλό μοντέλο της δράσης πολύπλοκο και δύσκολο. Ωστόσο, φαίνεται πως αποτελούν μόνο μέρος του προβλήματος. Αυτό που καθιστά τη θεωρία της δράσης ιδιαίτερα δυσνόητη, δεν είναι οι εγγενείς θεωρητικές ιδιότητές της, ούτε η εννοιολογική πολυπλοκότητα εννοιών όπως η συνήθεια, η δομή ή η δράση. Η σοβαρή εξέταση της θεωρίας της δράσης στο αρχαιολογικό πλαίσιο συγκρούεται άμεσα τον οργανωτικό, πρακτικό και εννοιολογικό μηχανισμό της αρχαιολογίας.

²²² Smith 2004, 1-19.

²²³ Johnson 2000, 213-231.

²²⁴ Moore 2000, 259.

²²⁵ Gell 1998, X.

²²⁶ Jones 2003, 178.

²²⁷ Sillar 2004, 153-190.

Η συντριπτική πλειοψηφία των αρχαιολόγων εξακολουθεί να διαιρεί το παρελθόν και το αρχαιολογικό υλικό σε πολιτισμούς, στάδια και τύπους. Οι τρεις αυτές κατηγοριοποιήσεις φαίνονται συνήθεις, στο βαθμό που φαίνονται απλές. Ωστόσο και οι τρεις κατηγοριοποιήσεις είναι οτιδήποτε άλλο εκτός από συνήθεις. Ειδικότερα, και οι τρεις συνηγορούν κατά της προβολής της δράσης στο αρχαιολογικό αρχείο, ιδίως μέσω της κατασκευής του αρχείου με οντότητες που χαρακτηρίζονται από ομοιότητα αντί μεταβλητότητα. Με άλλα λόγια με όρους οντοτήτων στις οποίες η ατομική δράση είναι λιγότερο εμφανής.

Ο αρχαιολογικός «πολιτισμός» ορίζεται ακόμη με όρους ομοιότητας, είτε αυτή η ομοιότητα αφορά σε τέχνηρα ή τάξεις τεχνέρων, ή είναι πολυθετικός²²⁸. Ομοίως, οι αρχαιολόγοι κατασκευάζουν την αντίληψή τους και την ορολογία τους για τις χρονικές περιόδους γύρω την από ομοιότητα και όχι τη μεταβλητότητα. Αυτή η κατασκευή, βασισμένη στις αρχές της στρωματογραφίας, διατρέχει τόσο τις απλές χρονολογικές ακολουθίες μιας αρχαιολογικής ή πολλών αρχαιολογικών θέσεων, όσο και τις μείζονες φάσεις αλλαγής, που χαρακτηρίζονται «ορίζοντες»²²⁹.

Η γλώσσα των τύπων, είτε χρησιμοποιείται σε επίσημες τυπολογίες είτε χρησιμοποιείται με πολύ χαλαρότερη έννοια, είναι η γλώσσα της ομοιότητας και μπορεί να τη δει κανείς ως εκδήλωση ενός κοινού νοητικού προτύπου τόσο στη διαχείριση των αρχαιολογικών δεδομένων, όσο και στην ερμηνεία τους. Βέβαια, δεν είναι μόνο η θεωρία της δράσης που βρίσκεται αντιμέτωπη με αυτή τη στερεά μάζα υποθέσεων και ταξινομήσεων. Οι δαρβινιστές αρχαιολόγοι παρατήρησαν παράλληλα ότι η υιοθέτηση της δαρβίνειας επιλογής ως κατευθυντήρια αρχή για την αρχαιολογική ερμηνεία, θα απαιτήσει την εγκατάλειψη μιας τέτοιας εννοιολογικής γλώσσας και της ταξινόμιας που τη συνοδεύει²³⁰.

Σε ένα άλλο βασικό επίπεδο, η θεωρία της δράσης δεν ταιριάζει με τις υπάρχουσες δομές του αρχαιολογικού λόγου, όπως είναι οι αναφορές για την αρχαιολογική θέση, τον καταμερισμό εργασίας και άλλες πιθανές σχέσεις που διαμορφώνονται πίσω από κάθε αναφορά. Η δομή και η οργάνωση της βασικής

²²⁸ Τα χαρακτηριστικά του G. Childe, που επαναλαμβάνονται ξανά και ξανά σε ομαδοποιήσεις εντός οριοθετημένης περιοχής. Βλ. Banning 2006, 43.

²²⁹ Βλ. Lucas 2001, 47-51, όπου συζητείται η σχέση της στρωματογραφίας και της αμερικανικής πολιτισμικής ιστορίας.

²³⁰ Shennan 2002, 26-28.

αναφοράς καταστέλλουν εξορισμού οποιαδήποτε εφαρμογή της θεωρίας της δράσης. Ο Barrett, επίσης, επισημαίνει ότι αν και η μεταδιαδικαστική αρχαιολογία έχει παρουσιάσει νέες θεωρητικές κατευθύνσεις, εξακολουθούν να επιβιώνουν οι παλαιότερες κατηγορίες της πρακτικής πεδίου και της ταξινόμησης του αρχαιολογικού υλικού²³¹.

Όλοι οι παραπάνω παράγοντες θα έλεγε κανείς ότι συνωμοτούν κατά της αναφοράς της ανθρώπινης δράσης στο παρελθόν. Δεν κάνουν μόνο τις γυναίκες αόρατες, αλλά κάθε έμφυλο ανθρώπινο ον, κάθε πραγματική ανθρώπινη οντότητα με κοινωνική, πολιτική και οικονομική ζωή²³².

Μια ιδιαίτερα αποκαλυπτική πτυχή στη συζήτηση περί θεωρίας της δράσης είναι το ερώτημα εάν τα τέχνηρα έχουν δράση ως «αρχαιολογικά υποκείμενα». Αρχαιολόγοι όπως οι Pollard και Reynolds²³³, Jones²³⁴, Kopytoff²³⁵ και άλλοι με μια σειρά μελετών διερεύνησαν την αντίληψη πως τα τέχνηρα και τα τοπία έχουν ταυτότητα, και συγκεκριμένα δική τους «βιογραφία», που απορρέει από τις αντιλήψεις του Gell για βιογραφίες τεχνέργων και τοπίων.

Η συγκεκριμένη διερεύνηση είναι ενδιαφέρουσα και γόνιμη, κυρίως γιατί οδηγεί τους αρχαιολόγους στην αμφισβήτηση κοινών αντιλήψεων για την έννοια προσωπικότητα (personhood). Σαφώς, όμως, τα τέχνηρα και τα τοπία δεν έχουν δράση με την έννοια που περιγράφεται από τους Giddens και Bourdieu²³⁶. Αυτό που είναι ενδιαφέρον στο συγκεκριμένο πλαίσιο είναι ότι ο αρχαιολόγος οδηγείται σε μια ενδιαφέρουσα και παραγωγική επικράτεια, όπου μπορεί να ασχοληθεί με ευρύτερα ζητήματα της υλικότητας και της ταυτότητας. Οποσδήποτε η έρευνα προς αυτή την κατεύθυνση είναι στραμμένη προς τη διερεύνηση της ταυτότητας του υλικού πολιτισμού και όχι στην έκφραση κάποιας συνεκτικής πνευματικής ή ιδεολογικής θέσης.

Για να συνοψίσουμε, αυτό που οδηγεί και χαρακτηρίζει ταυτόχρονα τις συζητήσεις για τη θεωρία της δράσης είναι μερικώς μόνον ζήτημα θεωρίας. Οι λόγοι για τους οποίους σκέπτονται θετικά ή αρνητικά οι αρχαιολόγοι για το ζήτημα της ανθρώπινης δράσης έχει να κάνει με ευρύτερους παράγοντες. Σε

²³¹ Barrett 2000, 61-68.

²³² Tringham 1991, 93 και 94.

²³³ Pollard and Reynolds 2002, ιδιαίτερα 36-38.

²³⁴ Jones 2003, 83-103.

²³⁵ Kopytoff 1986, 64-91.

²³⁶ Meskell 2004, 39-59.

αυτή την περίπτωση η σχέση ανάμεσα στην ανοιχτή θεωρητική συζήτηση και σε αυτό που πραγματικά κάνουν οι αρχαιολόγοι είναι εμφανές ότι χρειάζεται περισσότερη διερεύνηση σε θεωρητικό και πρακτικό επίπεδο.

Φαινομενολογία

Ένα άλλο ζήτημα που απομόνωσε η μεταδιαδικαστική κριτική ήταν η σημασία του νοήματος, και του υλικού πολιτισμού ως νοήμονος σύνθεσης. Η μεταδιαδικαστική αντίληψη για τον πολιτισμό ως κατασκευής καθαρά ή κυρίαρχα προσαρμοστικής στο περιβάλλον έγινε ευρέως αποδεκτή στους αρχαιολογικούς κύκλους, αν και όχι όλους. Όπως παρατηρεί ο Kristiansen, διαφορετικές κοινότητες συνεχίζουν να «αναπροσαρμόζουν» τις αντιλήψεις άλλων κοινοτήτων²³⁷.

Το βασικό ερώτημα εδώ είναι πώς αντλούν οι αρχαιολόγοι το νόημα. Για να το θέσουμε λεπτομερέστερα, ποιες αυστηρές μέθοδοι θα μπορούσαν να προταθούν για την κατανόηση των πολιτισμικών νοημάτων του παρελθόντος αξιολογήσιμες μέσω της αρχαιολογικής μαρτυρίας; Η κατασκευή μιας ουδέτερης μέσης θεωρίας, ελεύθερης αξιών, μπορεί να είναι ανοιχτή στην κριτική, ακόμη και αποτυχημένη, και είναι εύκολο να αποδειχθεί ότι η τυπική «δοκιμή» δεδομένων στη θεωρία που πρότεινε η νέα αρχαιολογία σχεδόν ποτέ δε συνέβη. Ωστόσο, η αναγνώριση της αποτυχίας δεν απαλλάσσει τους αρχαιολόγους από την ανάγκη ενδελεχούς αξιολόγησης των θεωριών για το παρελθόν. Αυτού του είδους η αξιολόγηση παραμένει απαραίτητο και κεντρικό στοιχείο της αρχαιολογίας.

Η απουσία άμεσης ή προφανούς απάντησης στο συγκεκριμένο μεθοδολογικό ερώτημα άφησε εκτεθειμένους τους μεταδιαδικαστικούς αρχαιολόγους στην κατηγορία του σχετικισμού, κατηγορία που συνεχίζεται μέχρι σήμερα²³⁸. Η απάντηση των μεταδιαδικαστικών ήταν ότι η κατηγορία του σχετικισμού ήταν απόρροια εκλαϊκευσης των ζητημάτων ή ακόμα και εσκεμμένη παρερμηνεία²³⁹. Η δεκαετία του '80 και η δεκαετία του '90 μπορούν να χαρακτηριστούν ιστορικά ως εξερεύνηση των διαφορετικών δυνατών μεθόδων για την εμπειρική μελέτη των πολιτισμικών νόμων.

²³⁷ Kristiansen 2004, 77.

²³⁸ Kohl 2004, 21.

²³⁹ Βλ. Lampeter Archaeology Workshop 1997, 164-198.

Οι αρχαιολόγοι στράφηκαν στα εργαλεία των κοινωνικών επιστημών αναζητώντας κατάλληλες μεθόδους, όπως έκαναν και με τα εργαλεία των φυσικών επιστημών για καθοδήγηση σχετικά με την επιστημολογία και την πειραματική διαδικασία. Μία από τις πρώτες θεωρητικές προσαρτήσεις σε αυτό το πλαίσιο ήταν ο δομισμός και ο μεταδομισμός. Πιθανολόγησαν, λοιπόν, ότι ο υλικός πολιτισμός έμοιαζε με κείμενο και συνεπώς η γλώσσα του υλικού πολιτισμού θα μπορούσε να «διαβαστεί» με τη χρήση μεθόδων που αναπτύσσονται από τους μελετητές της γλώσσας. Μια τέτοια δυνατότητα κατέληξε σε μια σειρά γόνιμων διερευνήσεων²⁴⁰.

Πιθανώς το μεγαλύτερο μέρος της δύναμης της ιδέας του υλικού πολιτισμού ως κείμενο βρίσκεται στην παιδαγωγική και ευριστική του αξία. Τα θεωρητικά σημεία σχετικά με την πολυσημία και τα πολλαπλά νοήματα δεν είναι εύκολο να κατανοηθούν με αφηρημένο τρόπο. Ωστόσο, οποιαδήποτε σκέψη για τη χρήση της γλώσσας παρέχει στον ερευνητή ένα έτοιμο μοντέλο, για να κατανοήσει το πώς μπορούν να δημιουργηθούν διαφορετικές έννοιες από ένα απλό σύνολο κανόνων ή για το πώς μπορούν να αλλάξουν οι έννοιες διαχρονικά με διαφορετικές και επαναλαμβανόμενες αναγνώσεις. Έδωσε, επίσης, στον ερευνητή μια ιδέα για το πώς έμαθαν οι άνθρωποι για τον κόσμο γύρω τους²⁴¹

Με άλλα λόγια, ο υλικός πολιτισμός ως κείμενο λειτούργησε και συνεχίζει να λειτουργεί ως κατανοητή μεταφορά, ένας βολικός τρόπος για να κατανοήσει κανείς ένα ζήτημα ή στοιχείο της αρχαιολογικής μαρτυρίας. Διάφορα, γνωστά πλέον, προβλήματα και ζητήματα αναδύονται με τη χρήση του ως κάτι περισσότερο από απλή παραγωγική μεταφορά, δηλαδή ως συστηματική εφαρμογή στον υλικό πολιτισμό²⁴².

Η στροφή προς τη φαινομενολογία²⁴³ μπορεί ιστορικά να θεωρηθεί ως το επόμενο στάδιο στην προσπάθεια κατανόησης παρελθόντων πολιτισμικών νοημάτων. Αυτή η στροφή φαίνεται σαφέστερα στον Gosden, ο οποίος δημοσίευσε τις απόψεις του για το κοινωνικό ον και το χρόνο ακριβώς αυτή τη

²⁴⁰ Βλ. Για παράδειγμα Tilley 1991, 95-102 και Moore 1996, 79-98.

²⁴¹ Johnson 2002, 69-71.

²⁴² Hodder and Hutson 2003, 59-68.

²⁴³ Η φαινομενολογία, ως φιλοσοφική θεωρία και μέθοδος, τρόπος σκέψης κατ' άλλους, για τη μελέτη της ανθρώπινης εμπειρίας και συνείδησης στην καθημερινή ζωή βρίσκεται σε αντίθεση με τον νεοθετικισμό, που δίνει έμφαση στην υλική πλευρά των πραγμάτων. Βλ. Κουκουζέλη κ.ά. 2003, 292 και Farina 2014, 50-62.

στιγμή της απομάκρυνσης από την πρώιμη μεταδιαδικαστική προτίμηση στα γλωσσολογικά μοντέλα. Ο Gosden εντόπισε ένα πολύπλοκο μονοπάτι ερμηνείας του κοινωνικού όντος μέσω του Heidegger, του Hegel, του Goethe, του Husserl, του Bourdieu, του Nietzsche και άλλων. Με αυτόν τον τρόπο ξεχώρισε τη θέση του από τις παραδόσεις του δορισμού και του μεταδορισμού, που χρησιμοποίησε κατά κόρον η μεταδιαδικαστική αρχαιολογία στις πρώιμες φάσεις της. Σύμφωνα με την άποψη του Gosden, η μεταφορική χρήση του κειμένου μειώνει την «αίσθησή μας για την ουσιαστικότητα της αρχαιολογικής μαρτυρίας»²⁴⁴.

Βέβαια, τα παραπάνω δεν ερμηνεύουν γιατί η φαινομενολογία εμφανίζεται επίμονα στην έρευνα και γιατί έχει γίνει τόσο χαρακτηριστική για μια συγκεκριμένη περιοχή του αρχαιολογικού λόγου, δηλαδή της βρετανικής προϊστορίας. Σήμερα η εφαρμογή της φαινομενολογίας στην αρχαιολογία, και ειδικότερα η μελέτη του τοπίου, έχει γίνει μια εξαιρετικά διακεκριμένη και δημοφιλής δραστηριότητα σε τέτοιο βαθμό, ώστε οι όποιες δημοσιεύσεις επί του θέματος να αναφέρονται στη «βρετανική φαινομενολογική παράδοση»²⁴⁵.

Η φαινομενολογία συσχετίζεται ιδιαίτερα με το έργο των προϊστορικών αρχαιολόγων της βρετανικής Νεολιθικής και Εποχής του Χαλκού και έχουν γίνει προσπάθειες να εφαρμοστεί η φαινομενολογία τυπικά σε ένα ιστορικό τοπίο²⁴⁶. Για να γίνει πλήρως κατανοητός ο λόγος της χρήσης της φαινομενολογίας στον συγκεκριμένο τομέα, πρέπει να κατανοήσει κανείς όχι μόνο το θεωρητικό έδαφος της φαινομενολογικής ερμηνείας, αλλά και τη φύση του βρετανικού αρχαιολογικού αρχείου και την ιστορία της πρακτικής στον συγκεκριμένο τομέα της βρετανικής προϊστορίας. Ανεξάρτητα από τις επίσημες σχέσεις και τις παραδοχές της θεωρίας του, ο υποκείμενος λόγος της βρετανικής φαινομενολογίας σχετίζεται με τον κυρίαρχο λόγο του Ρομαντισμού. Ο φαινομενολογικός λόγος ενσωμάτωσε σε τέτοιο βαθμό τον ρομαντικό λόγο στην ερμηνευτική του, ώστε η παρουσία των ρομαντικών επιδράσεων σπάνια σημειώνεται ή αναγνωρίζεται²⁴⁷.

Ο ρομαντισμός ως κίνημα δεν αφορά μόνο στην ερμηνεία της βρετανικής προϊστορίας. Το ίδιο συνέβη και στην ελληνική αρχαιολογία, δεδομένου ότι η

²⁴⁴ Gosden 1994, 39.

²⁴⁵ Robin 2006, 409-433.

²⁴⁶ Corcos 2001, 22-54.

²⁴⁷ Fleming 2005, 931.

επανάσταση του 1821 έγινε διεθνές ζήτημα με την κινητοποίηση των ιδεών του ευρωπαϊκού ρομαντισμού για την αρχαία Ελλάδα από τους Άγγλους ρομαντικούς ποιητές της εποχής τουλάχιστον, όπως ο P. B. Shelley and Lord Byron. Η προβολή του νεοελληνικού κράτους προς τα πίσω, σε μια ιστορία 3.000 χρόνων είναι ουσιαστικά ρομαντικό εγχείρημα²⁴⁸. Ο ριζοσπαστικός επαναπροσδιορισμός της ελληνικής εθνότητας του ρομαντικού κινήματος κατά τον 19ο αιώνα ακολούθησε τη θεμελίωση της εγχώριας αρχαιολογίας στο ελληνικό πλαίσιο. Στο μεταδιαδικαστικό πλαίσιο η διαμόρφωση της ελληνικής αρχαιολογίας χαρακτηρίζεται «νοήμων ανανέωση των αρχαίων πραγμάτων για τρέχοντα ενδιαφέροντα και σκοπούς». Εισάγεται, επίσης, η έννοια του «αυτόχθονου Ελληνισμού» ως αφήγημα που αναδιατυπώθηκε από τους Έλληνες διανοούμενους σε μια τοπική εκδοχή, για να επιτευχθεί η «χειραφέτηση της εθνικής αφήγησης»²⁴⁹.

Οι παρατηρήσεις των φαινομενολόγων για το προϊστορικό τοπίο «έχουν νόημα» και φαίνεται ότι είναι αποδεκτές στη δομή του αρχαιολογικού λόγου και πρακτικής, χαρακτηριστική του αγγλικού Ρομαντισμού και μάλιστα του αγγλικού εμπειρισμού. Ο εμπειρισμός στην προκειμένη περίπτωση εκδηλώνεται με την πεποίθηση ότι τα δεδομένα μιλούν αφεαυτού, χωρίς τη μεσολάβηση κάποιας θεωρίας. Όλοι αυτοί οι λόγοι δείχνουν γιατί είναι δημοφιλής η φαινομενολογία. Ωστόσο, η συνάφεια ορισμένων θεωρητικών και πρακτικών ανησυχιών στην αρχαιολογία πεδίου δε σημαίνει ότι η προσπάθεια κατανόησης του τοπίου σε ανθρώπινη κλίμακα εξαρτάται από τη βαθιά κατανόηση της θεωρίας της φαινομενολογίας.

Για παράδειγμα, ο Bradley (2000) ή ο Whyte (2013) συζητούν την εμπειρία, το συσχετισμό και μια σειρά από σχετικές έννοιες του τοπίου, χωρίς να χρειάζονται να καταφύγουν σε μια πιο επίσημη ή βαθύτερη ανάγνωση της φαινομενολογικής θεωρίας²⁵⁰. Ο Bradley δανείζεται μερικά στοιχεία από τη φαινομενολογία του τοπίου του Tilley (1994), που επιχειρεί να αναδημιουργήσει αντιλήψεις των ατόμων του παρελθόντος και από τον Richard (1996), που θεωρεί ότι φυσικά υδάτινα χαρακτηριστικά ήταν υπεύθυνα για την παρουσία

²⁴⁸ Pourgouris 2011, 291-295.

²⁴⁹ Hamilakis 2007, 119

²⁵⁰ Bradley 2000 και Whyte 2013, 499-517.

τελετουργικών χώρων, όπως τα ύστερα νεολιθικά χεντζ²⁵¹ στη Βρετανία²⁵². Ο Bradley, εκκινώντας από τέτοιες ιδέες, ωθεί τη συζήτηση σε ένα ακόμη επίπεδο παραπάνω. Προτείνει ότι είναι δυνατόν να εκτιμηθεί η σημασία του τόπου, ανεξαρτήτως μνημείων ή όχι, υπό τον όρο ότι η περιοχή που αναλύεται βρίσκεται κοντά σε περιοχές που έχουν αποκαλύψει αρχαιολογικό υλικό.

Η κριτική που ασκήθηκε στη φαινομενολογία ως θεωρητική προσέγγιση ποικίλει και συζητήθηκε στο *Archaeological Dialogues*²⁵³. Η μαρξιστική και η φεμινιστική σχολή άσκησαν τη δική τους κριτική²⁵⁴. Ο Williams, κυρίως, έδειξε πως ο ρομαντισμός οικοδόμησε τις ιδέες της «υπαίθρου» και της «φύσης» ως διαχρονικές και αμετάβλητες, ενώ στην πραγματικότητα αντικατόπτριζαν ιδέες της αστικής και μητροπολιτικής ανάπτυξης.

Ο ρομαντισμός είναι από αυτήν την άποψη αντίδραση προς τη Βιομηχανική Επανάσταση και δημιουργημά της ταυτόχρονα²⁵⁵. Χρειάζεται να επισημανθεί εδώ, πως όσοι χρησιμοποιούν τη φαινομενολογία δεν είναι πάντα τόσο στοχαστικοί. Βέβαια το ίδιο μπορεί να ειπωθεί και για τους κριτικούς της φαινομενολογίας. Έτσι, ο Fleming (2005), απαντώντας στην κριτική της φαινομενολογίας του καρτεσιανού χώρου, επισημαίνει ότι η παραδοσιακή αρχαιολογία τοπίου δεν είναι απλώς εφαρμογή των θεωριών της ενατένισης²⁵⁶ που αναπτύχθηκαν στην Αναγέννηση, αλλά έχει σφυρηλατηθεί μέσα από τη λεπτομερή παρατήρηση και ερμηνεία στο πεδίο²⁵⁷.

Αρχαιολογική θεωρία;

Από τη γενική και ειδική διερεύνηση της αρχαιολογικής θεωρίας ή των αρχαιολογικών θεωριών, των θέσεων και των αντιθέσεων που περιγράφηκαν προκύπτει πως η αρχαιολογική θεωρία υπάρχει μεν, αλλά όχι ως διακριτό και

²⁵¹ Όρος που χρησιμοποιήθηκε το 1932 από τον Thomas Kendrick, για να περιγράψει κυκλικές ή οβάλ σχήματος περιοχές με διακριτά χαρακτηριστικά τους, εκτός από τους μονόλιθους, το κυκλικό ανάχωμα και την τάφρο που το περιβάλλει. Βλ. Pitts 2011, 26-28.

²⁵² Richard 1996 313-336.

²⁵³ Brück 2005, 45-72.

²⁵⁴ Johnson 2007, 136-137.

²⁵⁵ Williams 1973, 9-13.

²⁵⁶ Η ματιά είναι ανταλλαγή ανάμεσα σε δύο άτομα ή αναζήτηση πληροφορίας, σύμφωνα με τις θεωρίες της ενατένισης ή ματιάς. Η ενατένιση είναι συσσωρευτική διαδικασία, καθώς κάθε ματιά χτίζεται πάνω στην προηγούμενη και απορροφάται αθροιστικά ως γνώση. Βλ. Hawthorn 2006, 508-518.

²⁵⁷ Fleming 2005, 921-932.

στέρεο σώμα γνώσης που αλληλεπιδρά με δεδομένα. Σύμφωνα με την Wylie «το ζήτημα της καταλληλότητας της επιστημολογικής στάσης πρέπει να διευθετείται σε τοπικό επίπεδο, υπό το πρίσμα όσων γνωρίζουμε για συγκεκριμένα θέματα και των πόρων που διαθέτουμε για την έρευνά τους»²⁵⁸. Το ίδιο πιθανώς θα πρέπει να ληφθεί υπόψη σε ό,τι αφορά στην αρχαιολογική θεωρία.

Η κατανόηση των διακυμάνσεων των θεωρητικών απόψεων και των αντιθέσεων ανάμεσα στις διαφορετικές σχολές αρχαιολογικής σκέψης τεκμηριώνεται αρκετά καλά στην ιστορία της αρχαιολογίας και υποδεικνύει ότι υπάρχει αναντιστοιχία ανάμεσα σε αυτό που λένε οι αρχαιολόγοι σε θεωρητικό επίπεδο και σε αυτό που πραγματικά κάνουν. Ο ρόλος της θεωρίας στη συγκέντρωση και το συντονισμό της πληροφορίας που προκύπτει από τα ευρήματα των αρχαιολόγων με τρόπο που αντιμετωπίζει θεωρητικά ζητήματα και ιστορικά δεδομένα είναι εκ των ων ουκ άνευ. Ωστόσο, η κατανόησή μας για τις ακριβείς παραμέτρους της θεωρίας στην αρχαιολογία είναι ελλιπής. Το έργο της κατανόησης των ακριβών παραμέτρων του ρόλου της θεωρίας φαίνεται ότι μόλις ξεκινά και σε μεγάλο βαθμό εξαρτάται από τη σύγκλιση απόψεων από όλους τους κλάδους της θεωρίας. Βήματα προς αυτή την κατεύθυνση γίνονται την τελευταία δεκαετία, αλλά φαίνεται ότι ο δρόμος προς τη διαμόρφωση ενός συνεκτικού σώματος αρχαιολογικής θεωρίας είναι μακρύς.

2.7. Αναλυτικοί χάρτες πληροφόρησης

Σε θεωρητικό και πρακτικό επίπεδο το αρχαιολογικό έργο στο σύνολό του είναι ένας ιδεατός χώρος πληροφορίας. Οι αρχαιολόγοι πεδίου και οι θεωρητικοί της αρχαιολογίας διαχειρίζονται πληροφορίες σε όλα τα στάδια του έργου τους, από την επιλογή μιας αρχαιολογικής θέσης, την ανασκαφή, τις ερμηνευτικές τους προσεγγίσεις έως την τελική δημοσίευση της έρευνάς τους. Συνειδητά ή ασυνειδητά κατασκευάζουν διαρκώς ιδεατούς χάρτες, προκειμένου να διαμορφώσουν δίκτυα πληροφορίας, σχετικά με τη θεωρητική ή πρακτική δραστηριότητά τους, αντλώντας από διαφορετικούς ορίζοντες πληροφόρησης. Οι συγκεκριμένοι χάρτες είναι οδοδείκτες για την επίτευξη αποτελεσματικών

²⁵⁸ Wylie 1992, 15–35.

λύσεων σε ζητήματα που προκύπτουν τόσο από τη θεωρητική, όσο και από την πρακτική έρευνα.

Η Sonnenwald (1999) εισήγαγε τους ορίζοντες πληροφόρησης στα τέλη της δεκαετίας του 1990 ως θεωρητικό πλαίσιο για τη συμπεριφορά της πληροφορίας²⁵⁹. Στη συνέχεια οι Sonnenwald και Wildemuth (2001) πρότειναν ότι σε κάθε εννοιολογικό πλαίσιο υπάρχει ένας «ορίζοντας πληροφόρησης» με τον οποίο είναι δυνατόν να αλληλεπιδράσει ο ερευνητής. Οι ορίζοντες πληροφόρησης περιλαμβάνουν πληροφοριακούς πόρους, που προσεγγίζουν όσοι αναζητούν πληροφορίες²⁶⁰. Οι ορίζοντες πληροφοριών προσδιορίζονται τόσο ατομικά όσο και κοινωνικά και μπορούν να θεωρηθούν ως πυκνοκατοικημένοι χώροι λύσης (Sonnenwald, 1999). Με τον όρο «χώρος λύσης», η Sonnenwald αναφέρεται σε ένα πλαίσιο στο οποίο συνυπάρχουν πολλές λύσεις και το πρόβλημα της ανάκτησης πληροφορίας εξαρτάται από τον προσδιορισμό της αποτελεσματικότερης πορείας προς την καλύτερη λύση και από τον προσδιορισμό του τρόπου με τον οποίο μπορούν να γίνουν ορατές οι πιθανές λύσεις τόσο στον ερευνητή (άτομο), όσο και σε άλλους πόρους πληροφοριών²⁶¹.

Οι ορίζοντες πληροφόρησης χρησιμοποιήθηκαν ως θεωρητικό πλαίσιο για την ερμηνεία και τη συζήτηση ευρημάτων που προκύπτουν από την έρευνα. Οι Sonnenwald, Wildemuth και Harmon (2001) ανέπτυξαν μέθοδο συλλογής δεδομένων βασισμένη στις θεωρητικές γνώσεις των χαρτών οριζόντων πληροφόρησης. Η μέθοδος χρησιμοποιήθηκε για τη συλλογή και στη συνέχεια για τη συζήτηση ερευνητικού υλικού²⁶². Η βασική γραμμή της προσέγγισης συλλογής δεδομένων που βασίζεται σε ορίζοντες πληροφόρησης είναι ότι οι χάρτες ορίζοντα σχεδιάζονται από τους πληροφορητές, έτσι ώστε να εκπροσωπούν τους ορίζοντες της προσωπικής τους πληροφόρησης²⁶³.

Η μέθοδος που βασίζεται στους χάρτες οριζόντων πληροφόρησης είναι καλή βάση για διεξοδικές συζητήσεις και συμπληρώνει τις συνεντεύξεις ως μέθοδο συλλογής δεδομένων²⁶⁴. Ωστόσο, δεν εξετάζεται η δυνατότητα χρησιμοποίησης του χάρτη ενός ορίζοντα πληροφόρησης που σχεδιάζεται από

²⁵⁹ Sonnenwald 1999, 176–190.

²⁶⁰ Sonnenwald and Wildemuth 2001, 1-20.

²⁶¹ Sonnenwald 1999, 187 και Fisher et al 2005, 195.

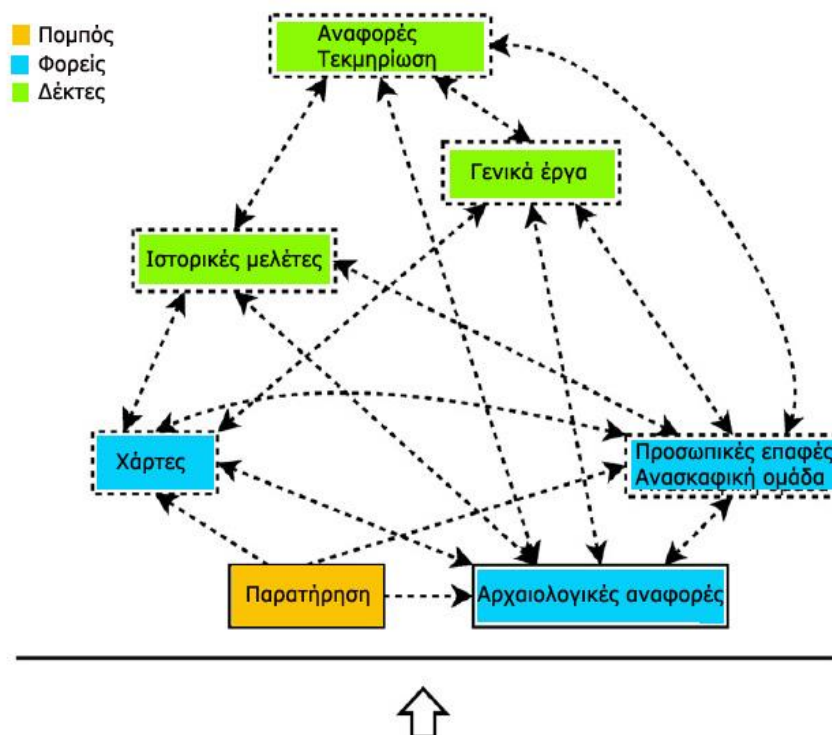
²⁶² Savolainen and Kari 2004, 415-433.

²⁶³ Sonnenwald et al 2001, 65-86.

²⁶⁴ Βλ. για παράδειγμα Serola 2006, 42–45.

ερευνητές, ειδικά ως εργαλείο ανάλυσης δεδομένων. Η χρησιμότητα του ατομικού χάρτη ορίζοντα πληροφόρησης μπορεί να περιοριστεί εξαιτίας της έλλειψης πόρων για μακροχρόνια συλλογή δεδομένων, από το επιστημονικό ενδιαφέρον για τις ομάδες και όχι για την ατομική συμπεριφορά, από την έλλειψη συγκρισιμότητας λόγω ασυνεπούς τεκμηρίωσης μεταξύ μεμονωμένων πληροφορητών, ή από την ανάγκη να αποκτηθεί μια βαθύτερη αναλυτική γνώση του ορίζοντα ενημέρωσης ενός πληροφορητή από ό,τι επιτρέπει ο προσωπικός χάρτης του.

Στο πλαίσιο της αρχαιολογικής θεωρίας και πρακτικής τέτοια θέματα αντιμετωπίζονται με χάρτες οριζόντων αναλυτικής πληροφόρησης, που βοηθούν στην παραγωγή ολοκληρωμένων και συγκρίσιμων παραστάσεων τόσο των ατομικών όσο και των συλλογικών οριζόντων πληροφόρησης. Οι χάρτες που εισηγούνται οι ερευνητές αντιπροσωπεύουν μια αναλυτική προοπτική στους οριζόντες πληροφόρησης μιας κοινής δραστηριότητας, όπως είναι το αρχαιολογικό έργο.



Εικ. 2.4. Ορίζοντας πληροφόρησης του ρόλου του αρχαιολόγου πεδίου, τυπικοί πομποί, δέκτες και φορείς²⁶⁵.

²⁶⁵ Προσαρμογή από Huvila 2009, 23.

Ο ορίζοντας πληροφόρησης ενός αρχαιολόγου πεδίου επικεντρώνεται σε μια γεωγραφική περιοχή αρχαιολογικού ενδιαφέροντος. Συνεπώς, καθορίζεται από τις προσωπικές παρατηρήσεις του αρχαιολόγου για μία γεωγραφική θέση και καθορισμένη χρονική περίοδο (πομποί) και περιλαμβάνει τις σχετικές πληροφορίες που αντλούνται από ποικιλία πηγών. Το έργο πληροφόρησης διεξάγεται με την ταυτόχρονη χρήση διαφόρων πόρων, κυκλικές αποδόσεις στο σημείο εκκίνησης και συνεχή διαδικασία απόκτησης πληροφοριών μέσω της παρατήρησης. Οι κύριοι φορείς της διαδικασίας αναζήτησης πληροφοριών είναι οι αρχαιολογικές αναφορές, η προσωπική επικοινωνία με άλλους αρχαιολόγους ή την ανασκαφική ομάδα και χαρτογραφικές πληροφορίες. Η διαδικασία είναι κυκλική και επαναληπτική. Διαρκεί όσο το δυνατόν περισσότερες επαναλήψεις του κύκλου και όσο επιτρέπουν τα εμπόδια στην αλληλεπίδραση των πληροφοριών. Ένας τυπικός δέκτης εδώ είναι ο πόρος πληροφόρησης, που ικανοποιεί την άμεση ανάγκη πληροφόρησης ή το ενδιαφέρον του αρχαιολόγου ανεξάρτητα από τη μορφή του²⁶⁶.

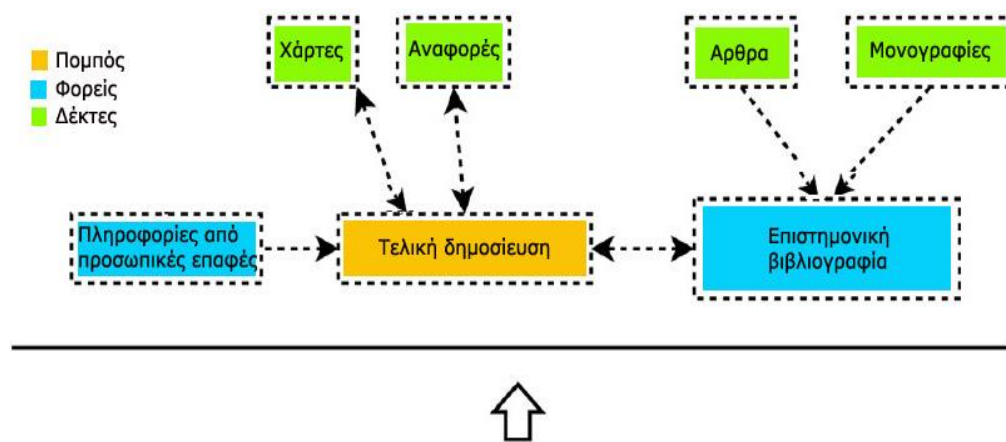
Βάσει των παραπάνω ο αρχαιολόγος πεδίου εκπέμπει μεταφορικά σε μια δεδομένη συχνότητα, με *δέκτες* τους πόρους πληροφόρησης, που είναι συντονισμένοι στην ίδια συχνότητα, προκειμένου να ικανοποιηθούν οι *φορείς* της διαδικασίας αναζήτησης, που είναι κυρίαρχα η κατασκευή των κατάλληλων χαρτών και των αρχαιολογικών αναφορών του και δευτερευόντως οι προσωπικές επαφές που ασκούν από την άποψη της πληροφορίας επίδραση στις αρχαιολογικές αναφορές. Οι πληροφορίες που προσλαμβάνονται μέσω ανάδρασης από τους δέκτες, διαμορφώνουν ένα *χώρο λύσεων*, από τον οποίο ο αρχαιολόγος πεδίου επιλέγει τη βέλτιστη διαδρομή ως το τελικό αποτέλεσμα.

Η ολοκλήρωση του αρχαιολογικού έργου συμβαίνει με την τελική δημοσίευση, γεγονός που προσδίδει έναν άλλο ρόλο στον αρχαιολόγο πεδίου, τον ρόλο του ακαδημαϊκού ερευνητή. Ο ορίζοντας της πληροφόρησης του ακαδημαϊκού ερευνητή δίνει έμφαση στο αρχαιολογικό υλικό και η επιστημονική έρευνα τείνει να παρουσιάζει κυμαινόμενα σημεία εισόδου (πομπούς). Στην προκειμένη περίπτωση, σημείο εισόδου και πομπός είναι η τελική δημοσίευση.

Ο ορίζοντας πληροφόρησης αποτελείται από μια αρκετά μεγάλη ποικιλία ακαδημαϊκών πηγών, οι οποίες χρησιμοποιούνται σύμφωνα με τις πραγματικές

²⁶⁶ Huvila 2009, 19.

ανάγκες πληροφόρησης. Οι ανάγκες και η επακόλουθη διαδικασία αναζήτησης πληροφοριών είναι πιθανό να ξεκινήσουν με μια κυρίαρχη αντίληψη που διαμορφώνεται από τις αρχαιολογικές αναφορές. Οι *φορείς* στην προκειμένη περίπτωση είναι κυρίαρχα η επιστημονική βιβλιογραφία και δευτερευόντως η πληροφορία που αντλεί ο αρχαιολόγος από τις επαφές του. Η επιστημονική διαδικασία πληροφόρησης στον ρόλο της ακαδημαϊκής έρευνας με σκοπό την τελική δημοσίευση τείνει να τερματίζεται, μόνο όταν εξαντληθούν τα πρακτικά όρια των δεκτών, σε επίπεδο χαρτών, αρχαιολογικών αναφορών, επιστημονικών άρθρων και μονογραφιών επί του συγκεκριμένου θέματος που οδηγείται σε δημοσίευση.



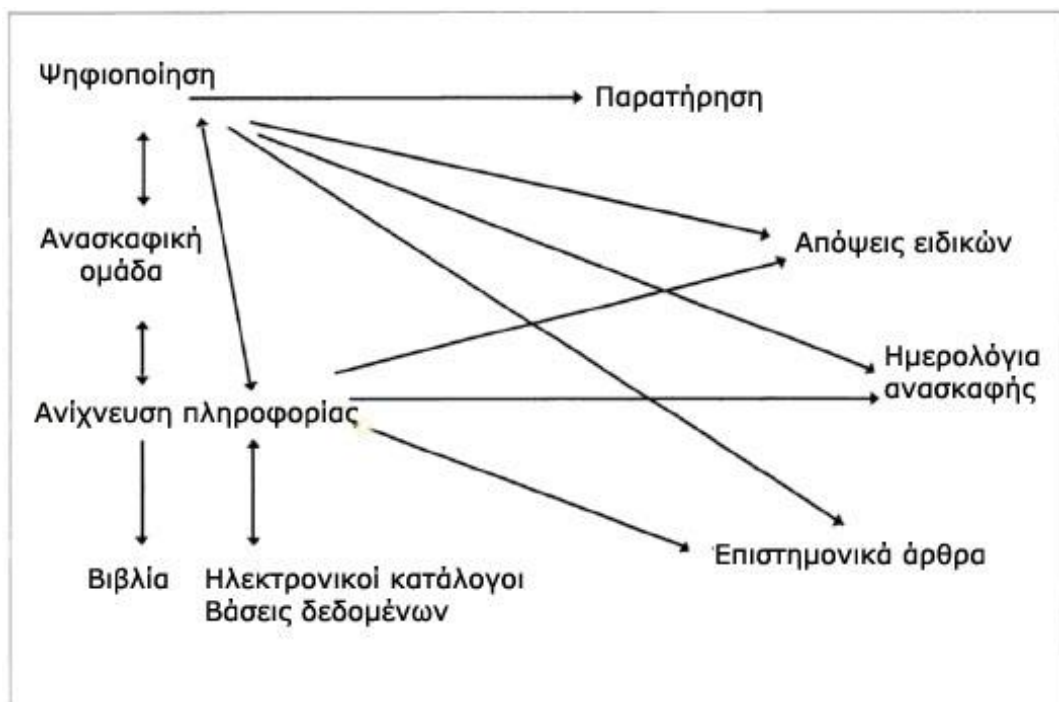
Εικ. 2.5. Αναλυτικός χάρτης πληροφόρησης για την τελική δημοσίευση. Τυπικοί πομποί, δέκτες και φορείς²⁶⁷.

Πρακτικό ζήτημα παραμένει η ενσωμάτωση ενός αναλυτικού χάρτη ορίζοντα πληροφόρησης στη διαδικασία της ψηφιοποίησης του έργου που παράγεται στο πεδίο. Στην προκειμένη περίπτωση πρόκειται για ενσωμάτωση σε μια σχεσιακή βάση δεδομένων, προκειμένου να γίνεται ευκολότερο το έργο της κατασκευής αρχαιολογικών αναφορών, αλλά και της τελικής δημοσίευσης. Εδώ προτείνεται κατασκευή αναλυτικών χαρτών πληροφόρησης για κάθε διακριτό πλαίσιο στην αλληλουχία διαδικασιών, από την επιλογή μιας γεωγραφικής αρχαιολογικής θέσης, την ανάλυσή της σε τομείς έρευνας, τη συστηματική ανασκαφική διαδικασία για κάθε διακριτό τομέα, για κάθε ανασκαφικό τετράγωνο

²⁶⁷ Προσαρμογή από Huvila 2009, 24.

και κάθε διακριτό τέχνηργο ή συσσωρεύσεις τεχνέργων, έως τη στρωματογραφική ανάλυση του αρχαιολογικού αρχείου.

Με αυτόν τον τρόπο είναι δυνατή η συγκριτική έρευνα τεχνέργων στο σύνολο της αρχαιολογικής θέσης και η σύγκρισή της με έρευνα που έχει διεξαχθεί σε άλλες αρχαιολογικές θέσεις για την εξαγωγή συμπερασμάτων μέσω της παρατήρησης, για την ομοιότητα ή ποικιλομορφία των αρχαιολογικών καταλοίπων σε μια ευρύτερη γεωγραφική περιοχή σε δεδομένο χρονικό ορίζοντα.



Εικ. 2.6. Αναλυτικός χάρτης διαδικασίας ψηφιοποίησης ανασκαφικών δεδομένων για τις ανασκαφές στον Τομέα 1 (Νεκρόπολη), Κυμισάλα²⁶⁸.

Τυπικό σημείο εισόδου για την κατασκευή του αναλυτικού χάρτη υπό την οπτική γωνία της μηχανικής λογισμικού είναι η διαδικασία της ψηφιοποίησης (πομπός) με πρωτεύοντα φορέα την ανίχνευση της πληροφορίας και δευτερεύοντες φορείς την παρατήρηση και την ανασκαφική ομάδα. Δέκτες στην προκειμένη περίπτωση είναι οι ηλεκτρονικοί κατάλογοι, οι online βάσεις δεδομένων, οι απόψεις ειδικών, τα ημερολόγια της ανασκαφής, τα επιστημονικά άρθρα και τα βιβλία που περιέχουν σχετικές πληροφορίες.

²⁶⁸ Προσαρμογή από Fisher et al 2005, 194.

Η κατασκευή αναλυτικών χαρτών εξαρτάται σε μεγάλο βαθμό από τις θεωρητικές προκαταλήψεις των αρχαιολόγων και τη σχολή ερμηνείας που επιλέγουν για την παρουσίαση του αρχαιολογικού υλικού. Εδώ προτείνεται η συλλογή όλης της πληροφορίας, για να διαμορφωθεί όσο το δυνατόν πληρέστερος *χώρος λύσεων*, από τον οποίο ο αρχαιολόγος μπορεί να κατασκευάσει την πληρέστερη εικόνα του υλικού και των ερμηνευτικών του δυνατοτήτων. Η στροφή προς τον άκραιο θετικισμό ή την απόλυτη φαινομενολογία αποκλείει εγγενώς τη δυνατότητα της ολοκληρωμένης ερμηνείας, από την άποψη της αρχαιολογικής θεωρίας.

Κεφάλαιο 3: Αρχαιολογική μεθοδολογία

3.1 Αρχαιολογικό αρχείο

Η μεθοδολογία είναι η συστηματική, θεωρητική ανάλυση των μεθόδων που εφαρμόζονται σε ένα πεδίο σπουδών. Περιλαμβάνει τη θεωρητική ανάλυση του συνόλου των μεθόδων και αρχών που συνδέονται με ένα γνωστικό πεδίο. Συνήθως, περιλαμβάνει έννοιες όπως, για παράδειγμα, θεωρητικό μοντέλο, φάσεις και ποσοτικές ή ποιοτικές τεχνικές²⁶⁹. Δεδομένου ότι δεν έχει ως στόχο να παράσχει λύσεις, διαφοροποιείται από τη μέθοδο ως έννοια. Η μεθοδολογία προσφέρει τη θεωρητική βάση για την κατανόηση της μεθόδου, του συνόλου μεθόδων ή των βέλτιστων πρακτικών που μπορούν να εφαρμοστούν σε μια συγκεκριμένη περίπτωση, για παράδειγμα, για τον υπολογισμό ενός συγκεκριμένου αποτελέσματος. Στο αρχαιολογικό πλαίσιο η μεθοδολογία είναι η συστηματική μελέτη των μεθόδων που μπορούν να εφαρμοστούν ή έχουν εφαρμοστεί στην αρχαιολογία και ένας εναλλακτικός ορισμός της είναι η μελέτη ή περιγραφή μεθόδων²⁷⁰.

Οι αρχαιολογικές μέθοδοι σχετίζονται γενικά και ειδικά με το αρχαιολογικό αρχείο ή αρχαιολογική μαρτυρία, δηλαδή ένα σώμα φυσικής (όχι γραπτής) μαρτυρίας για το παρελθόν και είναι μια από τις βασικότερες έννοιες στον πυρήνα της αρχαιολογικής σκέψης²⁷¹, που ασχολείται με την τεκμηρίωση και την ερμηνεία του αρχαιολογικού αρχείου²⁷². Ο όρος «αρχαιολογικό αρχείο» πιθανότατα προήλθε από παράλληλες έννοιες στη γεωλογία (γεωλογικό αρχείο) και την παλαιοντολογία (απολιθωματικό αρχείο). Χρησιμοποιήθηκε ευρέως από τον Gordon Childe στη δεκαετία του 1950 και φαίνεται πως εισήλθε σε κοινή χρήση στη συνέχεια²⁷³. Το αρχαιολογικό αρχείο διαπλάθεται και μεταπλάθεται εξαιτίας ανθρωπογενών και μη δραστηριοτήτων, όπως είναι η γεωργία η εκμετάλλευση γαιών και οι φυσικές καταστροφές.

²⁶⁹ Irny, and Rose 2005, 325-331.

²⁷⁰ Baskerville 1991, 749-764.

²⁷¹ Patrik 1985, 27-62.

²⁷² Hardesty 2008, 1414-1416.

²⁷³ Lucas 2012, 1-17.

Στην πρώτη κριτική επισκόπηση της ιδέας, η Linda Patrik διαπίστωσε ότι κατά τη δεκαετία του 1980 οι αρχαιολόγοι αντιλαμβάνονταν τον όρο με πέντε τουλάχιστον διαφορετικούς τρόπους:

1. Ως βάση για την εναπόθεση υλικού²⁷⁴.
2. Ως φυσικές αποθέσεις²⁷⁵.
3. Ως τέχνηρα και αντικείμενα²⁷⁶.
4. Ως συλλογή δειγμάτων²⁷⁷.
5. Ως αρχαιολογικές αναφορές²⁷⁸.

Η Patrik υποστήριξε ότι οι τρεις πρώτοι ορισμοί αντικατοπτρίζουν ένα «φυσικό μοντέλο» της αρχαιολογικής μαρτυρίας, που θεωρείται άμεσο αποτέλεσμα φυσικών διεργασιών του παρελθόντος (όπως το απολιθωματικό αρχείο). Αντίθετα, οι ορισμοί τέσσερα και πέντε ακολουθούν ένα «κειμενικό μοντέλο», σύμφωνα με το οποίο το αρχαιολογικό αρχείο θεωρείται ότι κωδικοποιεί πολιτιστικές πληροφορίες για το παρελθόν (όπως τα ιστορικά κείμενα). Τόνισε, επίσης, ότι η κατανόηση των αρχαιολόγων για το τι συνιστά αρχαιολογικό αρχείο εξαρτάται από τα ευρύτερα ρεύματα της αρχαιολογικής θεωρίας, δηλαδή ότι οι διαδικαστικοί αρχαιολόγοι ακολουθούν το φυσικό μοντέλο, ενώ οι μεταδιαδικαστικοί αρχαιολόγοι το κειμενικό μοντέλο.

Ο Lucas συμπύκνωσε τον κατάλογο της Patrik σε τρεις διακριτούς ορισμούς του αρχαιολογικού αρχείου:

1. Το αρχαιολογικό αρχείο είναι ο υλικός πολιτισμός.
2. Το αρχαιολογικό αρχείο είναι τα υλικά υπολείμματα του παρελθόντος.
3. Το αρχαιολογικό αρχείο είναι οι πηγές που χρησιμοποιούν οι αρχαιολόγοι²⁷⁹.

Από τα παραπάνω συνάγεται ότι το αρχαιολογικό αρχείο είναι ένας γενικός όρος για το σύνολο των αρχαιολογικών δεδομένων, δηλαδή των καταλοίπων της

²⁷⁴ Clarke 1973, 16.

²⁷⁵ Binford 1964, 425.

²⁷⁶ Renfrew 1972, 441, επίσης Watson et al 1971, 22.

²⁷⁷ Childe 1956, 1, επίσης Cherry et al 1978, 11.

²⁷⁸ de Laet 1957, 13-18.

²⁷⁹ Lucas 2012, 124-168.

αρχαίας ανθρώπινης συμπεριφοράς, που καταγράφονται από τους αρχαιολόγους²⁸⁰.

Το αρχαιολογικό αρχείο ως πλαίσιο περιλαμβάνει τέσσερις κατηγορίες αρχαιολογικών δεδομένων, τα τέχνηρα (artefacts), τις κατασκευές (features), τα κτίσματα (structures) και τα οικοδομημένα (ecofacts). Τα τέχνηρα είναι κινητά ευρήματα, αντικείμενα επεξεργασμένα ή τροποποιημένα από τον άνθρωπο, όπως εργαλεία, όπλα, αγγεία κ.λπ. και αποτελούν βασική πηγή πληροφόρησης στο αρχαιολογικό πλαίσιο. Στην εθνογραφία και την αρχαιολογία προτάθηκε, επιπλέον, η κατηγορία του «προγονικού τεχνέργου», που ορίζεται ως «αντικείμενο φυσικής πρώτης ύλης (κέρας, οψιανός, ξύλο, κόκκαλο, φυσικός χαλκός κ.λπ.) που κατασκευάζεται από έναν λαό βάσει του τρόπου ζωής του για παράδειγμα κυνήγι, τροφосуλλογή, γεωργία και ποιμενικός νομαδικός βίος²⁸¹.

Ορισμένοι ερευνητές διευρύνουν την έννοια του όρου τέχνηρα, ώστε να περιλαμβάνει όλα τα ανθρωπογενή χαρακτηριστικά μιας αρχαιολογικής θέσης, όπως εστίες, κοιλότητες και οπές, δηλαδή «κατασκευές» που ορίζονται ουσιαστικά ως μη φορητά αντικείμενα. Απλά χαρακτηριστικά, όπως οπές, εστίες, δάπεδα, ίχνη θεμελίωσης κ.λπ., παρέχουν στοιχεία για κατασκευές, που κατηγοριοποιούνται στο αρχαιολογικό αρχείο ως «κτίσματα»²⁸². Τα κτίσματα ή οικοδομήματα αποτελούν σημαντική πηγή πληροφόρησης για την αρχαία πολιτισμική συμπεριφορά. Όταν δεν αναγνωρίζονται από υπαρκτές τοιχοποιίες, οι κατοικίες, αποκαλύπτονται συνήθως από περιγράμματα δαπέδων και από ίχνη που αφήνουν στο έδαφος οι τάφροι θεμελίωσης, οι οπές πασσάλων κ.λπ.²⁸³.

Οι αρχαιολογικές θέσεις περιέχουν επίσης οικοδομημένα -φυτικά και ζωικά κατάλοιπα. Ορισμένα από αυτά είναι απορρίμματα τροφής, αλλά ενίοτε τα οικοδομημένα εισέρχονται στις θέσεις μέσω φυσικών διεργασιών. Η ταφονομική μελέτη μιας αρχαιολογικής θέσης υποδεικνύει τον τρόπο με τον οποίο συσσωρεύονται φυτικά και ζωικά κατάλοιπα στον χώρο²⁸⁴. Με αυτόν τον τρόπο τα οικοδομημένα διακρίνονται για τη σχέση τους με την αρχαία πολιτισμική συμπεριφορά σε δεδομένο αρχαιοπεριβάλλον. Περιλαμβάνουν κατάλοιπα αρχαίας χλωρίδας και πανίδας, όπως επίσης και μεταλλεύματα.

²⁸⁰ Κουκουζέλη κ.ά. 2003, 93.

²⁸¹ Hortolà 2017, 213–225.

²⁸² Renfrew and Bahn 2016, 49.

²⁸³ Κουκουζέλη κ.ά. 2003, 94.

²⁸⁴ Kelly and Thomas 2011, 133.

↓ Αρχαιολογικό αρχείο ↓			
Τέχνηρα	Κατασκευές	Κτίσματα	Οικοδομημένα
κινητά ευρήματα, αντικείμενα επεξεργασμένα τροποποιημένα από τον άνθρωπο, εργαλεία, όπλα, αγγεία	οπές, κοιλότητες, εστίες, δάπεδα, τάφροι θεμελίωσης	αρχιτεκτονικά κατάλοιπα κατοικίες, εργαστήρια, ανάκτορα, δημόσια κτήρια, ναοί, ταφικά μνημεία	απορρίμματα τροφής εδάφη και ιζήματα (άμμος ή πηλός) μεταλλεύματα σκωρίες οστά ζώων, κοχύλια, κάρβουνο ή στάχτη, σπόροι καρποί, γύρη, φυτικά κατάλοιπα
↑ Αρχαιολογική θέση ↑ Αρχαιολογική ακολουθία			

Πιν. 1 Αρχαιολογικό αρχείο και αρχαιολογική θέση

Το αρχαιολογικό αρχείο σε μία συγκεκριμένη αρχαιολογική θέση αναφέρεται ενίοτε ως αρχαιολογική ακολουθία. Ωστόσο, οι δύο όροι δεν είναι ακριβώς εναλλάξιμοι, καθώς ο όρος αρχαιολογικό αρχείο είναι σφαιρικότερος ως προς το νόημά του και μπορεί να εφαρμοστεί σε τέχνηρα και άλλες ενδείξεις, όπως οικοδομημένα και τις σχετικές σχέσεις τους, καθώς και στη στρωματογραφία ενός τόπου. Αντίθετα, η ακολουθία σχετίζεται αποκλειστικά με τη χρονολογική σειρά, που προσδιορίζεται με μεθόδους στρωματογραφίας ή/και απόλυτης χρονολόγησης. Έτσι, το αρχαιολογικό αρχείο αποτελείται από γνωστές και άγνωστες αρχαιολογικές θέσεις με υλικό διατηρημένο *in situ*, τέχνηρα σε μουσεία και συλλογές, καθώς και αρχεία αρχαιολογικής έρευνας και ερμηνείας.

Το αρχαιολογικό αρχείο κατανέμεται στην ξηρά και τη θάλασσα σε μεταβλητές πυκνότητες. Εκείνα τα σημεία που διακρίνονται για τις υψηλές πυκνότητες αρχαιολογικών δεδομένων αποκαλούνται «αρχαιολογικές θέσεις». Οι αρχαιολογικές θέσεις είναι σημεία στα οποία απαντώνται τέχνηρα, κατασκευές, κτίσματα και οικοδομημένα²⁸⁵. Εναλλακτικά, ο όρος χρησιμοποιείται για τη συγκέντρωση των κτισμάτων και κατασκευών, που συνεπάγονται προηγούμενη

²⁸⁵ Renfrew and Bahn 2016, 50.

ανθρώπινη εγκατάσταση. Χρησιμοποιείται, επίσης, για να υποδηλώσει κάθε τόπο στον οποίο επικεντρώνουν οι αρχαιολόγοι την έρευνά τους. Το αρχαιολογικό τοπίο (ή ιστορικό τοπίο ή ιστορικό περιβάλλον) συνδυάζεται με τα ανιχνευόμενα τμήματα της αρχαίας χρήσης της γης, ιδιαίτερα με τις τοποθεσίες²⁸⁶. Σύμφωνα με τον Peter Drewett, οι αρχαιολογικές θέσεις αποτελούνται κυρίως από περιοχές δραστηριοτήτων και απορρίμματα. Εκεί οι άνθρωποι έχουν δράσει στο παρελθόν και έχουν κάνει κάποια πράγματα»²⁸⁷. Ο Darvill δίνει έναν πολύ παρόμοιο ορισμό, θεωρώντας ότι αρχαιολογική θέση είναι «κάθε θέση στην οποία εντοπίζονται τέχνηρα, κατασκευές ή κτίσματα κατασκευασμένα ή τροποποιημένα από ανθρώπους»²⁸⁸.

Οι αρχαιολογικές θέσεις ποικίλλουν ως προς το περιεχόμενο, τη λειτουργία, το μέγεθος, τη μορφή και τη διάρκεια χρήσης. Όσον αφορά στο περιεχόμενο, είναι δυνατόν να αποτελούνται μόνον από μία από τις τέσσερις κατηγορίες αρχαιολογικών δεδομένων ή από οποιονδήποτε συνδυασμό τους. Όσον αφορά στη λειτουργία, οι αρχαιολογικές θέσεις μπορούν να διακριθούν σε οικισμούς, σπήλαια και βραχοσκεπές, καταυλισμούς, τελετουργικά κέντρα ή ιερά, ταφικές θέσεις ή νεκροταφεία, και εξειδικευμένες θέσεις, όπως εμπορεία, στρατιωτικές εγκαταστάσεις, λατομεία, μεταλλεία, διαφόρων ειδών εργαστήρια κ.ά.

Όσον αφορά στο μέγεθος και τη μορφή, η αρχαιολογική θέση μπορεί να είναι οικισμός ή μεμονωμένο μνημείο ή άλλο κτίσμα, ακόμη και σύνολο αγγείων ή εργαλείων²⁸⁹. Ως προς τη διάρκεια χρήσης, υπάρχουν αρχαιολογικές θέσεις που χρησιμοποιήθηκαν για πολύ σύντομα χρονικά διαστήματα, άλλες που διήρκεσαν για μία ή περισσότερες γενεές και άλλες που χρησιμοποιήθηκαν επανειλημμένως επί αιώνες ή ακόμη και χιλιετίες²⁹⁰.

²⁸⁶ Carve *et al* 2014, 4

²⁸⁷ Drewett, 1999, 17.

²⁸⁸ Darvill 2002.

²⁸⁹ Αν δεν αναγνωρίζονται εύκολα οι οικισμοί από οχυρωματικά τείχη, τάφρους, περιβόλους, από την εμφανή τοπογραφική τους θέση ή από σημαντικά αρχιτεκτονικά κατάλοιπα, μπορούν να αναγνωριστούν από την κατανομή και την πυκνότητα υλικών καταλοίπων στον χώρο με παρατήρηση και χαρτογράφηση των κατανομών και της τοπογραφίας των τεχνέργων, την ανάλυση των ανθρωπογενών χημικών και γεωχημικών σημάτων στο έδαφος και την επίγεια, εναέρια και υποβρύχια τηλεπισκόπηση και τεχνικές εφαρμογές που εισήχθησαν σχετικά πρόσφατα από τις φυσικές και χημικές επιστήμες. Βλ. David 2001, 521.

²⁹⁰ Renfrew and Bahn 2016, 50.

Η αρχαιολογική θέση είναι βασική μονάδα αρχαιολογικής έρευνας και μεγάλο ποσοστό αρχαιολογικών μελετών εξακολουθεί να αφορά σε μεμονωμένες θέσεις. Ωστόσο, διαπιστώνεται μια αυξανόμενη τάση θεώρησης της αρχαιολογικής θέσης ως μέρος ενός ευρύτερου φυσικού και πολιτισμικού περιβάλλοντος, με αποτέλεσμα να μετατοπίζεται η έμφαση σε μεγαλύτερες και περιεκτικότερες μονάδες έρευνας, τις αρχαιολογικές γεωγραφικές περιοχές. Η αρχαιολογική γεωγραφική περιοχή περιλαμβάνει σειρά αρχαιολογικών θέσεων και είναι σαφώς οριοθετημένη γεωγραφικά βάσει τοπογραφικών στοιχείων. Ωστόσο, στον καθορισμό της υπεισέρχονται πολιτισμικά και οικολογικά στοιχεία, διότι ικανοποιούσε τις ανάγκες διαβίωσης και τις πολιτισμικές ανάγκες ενός συνόλου αλληλένδετων ανθρώπινων κοινωνιών²⁹¹. Η αρχαιολογική γεωγραφική περιοχή ως έννοια συνδέεται με την αρχαιολογία τοπίου, που διερευνά το τοπίο ως φυσική και νοηματική οντότητα. Διερευνά, επίσης, τον τρόπο με τον οποίο οι άνθρωποι απεικόνιζαν τον κόσμο και τον τρόπο με τον οποίο συνεργάζονταν μεταξύ τους μέσω του χώρου, καθώς και τον τρόπο με τον οποίο επέλεξαν να χειραγωγήσουν το περιβάλλον τους ή πώς επηρεάστηκαν στη δράση τους εξαιτίας τοπικών συνθηκών. Η δράση που αναφέρεται εδώ μπορεί να είναι ακούσια ή ασυνείδητη, σωματικά και πνευματικά, ή εκούσια και συνεπώς συνειδητή δράση²⁹².

3.2. Παράγοντες σχηματισμού και μετασχηματισμού του αρχαιολογικού αρχείου

Οι διαδικασίες σχηματισμού και μετασχηματισμού του αρχαιολογικού αρχείου φυσικές και πολιτισμικές παράγουν το σύνολο της μαρτυρίας που μελετούν οι αρχαιολόγοι, όταν διερευνούν κοινωνίες του παρελθόντος²⁹³. Οι παράγοντες σχηματισμού αφορούν στη διαδικασία διαμόρφωσης μιας αρχαιολογικής θέσης και σχετίζονται με την αρχική ή «πρωτογενή» απόθεση των αρχαίων υλικών καταλοίπων. Σχετίζονται αποκλειστικά με τη λειτουργική ή συμβολική ανθρώπινη συμπεριφορά και αναφέρονται ως πολιτισμικοί ή ανθρωπογενείς. Τέτοιοι παράγοντες συνδέονται με δραστηριότητες σχετικές με

²⁹¹ Κουκουζέλη κ.ά. 2003, 95.

²⁹² David and Thomas 2016, 38.

²⁹³ Rathje and Schiffer 1982, 105-153.

την απόκτηση πρώτης ύλης, την κατασκευή, τη χρήση και την επαναχρησιμοποίηση, την κατανομή και την απόρριψη τεχνέργων ή την ανακύκλωσή τους²⁹⁴. Κατά τη χρήση, τα αντικείμενα συμμετέχουν σε δραστηριότητες που μπορεί να έχουν πρακτικές και συμβολικές λειτουργίες. Εδώ χρειάζεται να αναγνωρίσουμε τρεις τύπους σημαντικών λειτουργιών του τεχνέργου. α) Την τεχνολογική λειτουργία, η οποία περιλαμβάνει την εξαγωγή, την επεξεργασία και την αποθήκευση πόρων, τη διατήρηση της τεχνολογίας και την εκπλήρωση των βιολογικών αναγκών των ανθρώπων. β) Την κοινωνική λειτουργία, που επηρεάζει τις κοινωνικές αλληλεπιδράσεις. γ) Την ιδεολογική λειτουργία, που συμβολίζει την ιδεολογία και μεταφέρει άλλες πληροφορίες²⁹⁵. Τα περισσότερα τέχνηρα εκτελούν περισσότερες από μία σημαντικές λειτουργίες. Μετά τη χρήση τους είναι δυνατόν να επαναχρησιμοποιηθούν ή να απορριφθούν. Εάν απορριφθούν εισέρχονται στο αρχαιολογικό πλαίσιο, όπου αλληλεπιδρούν με το φυσικό περιβάλλον²⁹⁶.

Η ανακύκλωση είναι η επιστροφή ενός τεχνέργου μετά από κάποια περίοδο χρήσης στη διαδικασία της κατασκευής. Η ανακύκλωση ορίζεται ως δραστηριότητα με την οποία ένα δευτερογενές υλικό εισάγεται ως πρώτη ύλη σε μια βιομηχανική διαδικασία στην οποία μετατρέπεται σε νέο προϊόν με τέτοιο τρόπο, ώστε να χάνεται η αρχική του ταυτότητα. Δευτερογενή υλικά είναι εκείνα που έχουν εκπληρώσει τη λειτουργία τους και δεν μπορούν να χρησιμοποιηθούν περαιτέρω στη σημερινή μορφή ή σύνθεση τους. Είναι, επίσης, εκείνα τα υλικά που εμφανίζονται ως απόβλητα κατά τη διαδικασία κατασκευής ή τη μετατροπή των τεχνέργων. Οι διαδικασίες συντήρησης μπορούν να οδηγήσουν σε αλλαγές της μορφής του τεχνέργου, αλλά η ανακύκλωση διακρίνεται εύκολα από τη συντήρηση. Οι αλλαγές συντήρησης τείνουν να είναι σχετικά μικρές και το τέχνηργο επαναλαμβάνει την πρότερη λειτουργία του²⁹⁷. Τα τέχνηρα προορίζονται συχνά για νέα χρήση χωρίς να χρειάζονται εκτεταμένες τροποποιήσεις. Αυτός ο τύπος διαδικασίας επαναχρησιμοποίησης ονομάζεται

²⁹⁴ Renfrew and Bahn 2016, 54.

²⁹⁵ Rathje and Schiffer 1982, 65-67.

²⁹⁶ Schiffer 1996, 14.

²⁹⁷ Darnay και Franklin 1972, 2-3.

δευτερεύουσα χρήση²⁹⁸. Για παράδειγμα, οι φθαρμένες μυλόπετρες μπορούν να χρησιμοποιηθούν χωρίς αλλοίωση ως δομικά υλικά²⁹⁹.

Τα τέχνηρα με σημαντικές συμβολικές λειτουργίες καθίστανται ενίοτε απαρχαιωμένα. Τότε πιθανώς απορρίπτονται και μετατρέπονται σε αρχαιολογικό αρχείο. Η επαναχρησιμοποίηση ή η απόρριψη είναι η συνήθης τύχη τεχνέργων που δεν μπορούν να εκτελέσουν την τεχνική ή συμβολική λειτουργία τους. Η έναρξη των διαδικασιών απόρριψης σε περιπτώσεις τεχνικής αστοχίας προκαλείται συχνά από μια ανεπανόρθωτη αλλαγή στη μορφή του τεχνέργου, που μειώνει τη μηχανική του αποτελεσματικότητα, όπως είναι η θραύση και η φθορά. Βέβαια, θα πρέπει να σημειωθεί εδώ ότι η συμβολική λειτουργία ενίοτε παραμένει ακόμα και μετά τη θραύση, άρα την παύση της τεχνικής λειτουργίας του τεχνέργου, όπως συμβαίνει με τη θραύση αγγείων σε ταφικές τελετουργίες³⁰⁰. Οι παράγοντες που επηρεάζουν τους ρυθμούς θραύσης για οποιοδήποτε είδος τεχνέργου είναι πολύπλοκοι και εξαρτώνται από τη φύση της λειτουργίας του και τις ιδιότητες αντοχής των υλικών. Αν και η φθορά θεωρείται συνήθως βραδεία διαδικασία, ορισμένα είδη φθοράς, όπως η σήψη των τροφίμων, οδηγεί γρήγορα στην απόρριψη³⁰¹.

Άλλοι πολιτισμικοί παράγοντες σχετίζονται με ταφονομικές δραστηριότητες. Οι άνθρωποι μπορούν να είναι σημαντικοί ταφονομικοί παράγοντες, καθώς συνδιαμορφώνουν, μαζί με τις φυσικές διεργασίες, την ιστορία μιας απόθεσης με τις ταφικές πρακτικές τους³⁰². Τόσο η θεωρία όσο και η μέθοδος της εθνοαρχαιολογίας και της ταφονομίας συνδέονται εξαιτίας του κοινού ενδιαφέροντος των δύο πεδίων για τις βιολογικές και γεωλογικές διεργασίες που σχηματίζουν αποθέσεις προϊστορικών υλικών. Ωστόσο, τα δύο πεδία διαφέρουν υπό την έννοια ότι η εθνοαρχαιολογία ενδιαφέρεται, επίσης, για τον ρόλο της ανθρώπινης συμπεριφοράς στη διαμόρφωση αρχαιολογικών συσσωρεύσεων³⁰³.

²⁹⁸ Darnay και Franklin 1972, 3.

²⁹⁹ Meighan 1980, 115.

³⁰⁰ Πετρόχειλος 2002-2005, 117-118.

³⁰¹ Green 1961, 51-54.

³⁰² Lyman 1994, 33 και Stutz 2003, 138.

³⁰³ Gifford 1980, 94-106.

3.3. Παράγοντες μετασχηματισμού του αρχαιολογικού αρχείου

Οι παράγοντες μετασχηματισμού καθορίζουν την τελική μορφή του αρχαιολογικού αρχείου. Περιλαμβάνουν οποιαδήποτε διαδικασία, φυσική ή ανθρωπογενή, μετατρέπει μια αρχαία εγκατάσταση σε αρχαιολογική θέση με την πάροδο του χρόνου, δηλαδή τις συνθήκες και τα γεγονότα που επηρεάζουν τα αρχαιολογικά δεδομένα από τη στιγμή της εναπόθεσης μέχρι το χρόνο της ανάκτησής τους. Αρκετοί ερευνητές ισχυρίζονται ότι ως αποτέλεσμα των διαδικασιών σχηματισμού, το αρχαιολογικό αρχείο είναι μια μετασχηματισμένη ή παραμορφωμένη άποψη για τα τέχνηρα, που συμμετείχαν κάποτε σε ένα σύστημα συμπεριφοράς³⁰⁴. Αυτή η σύλληψη αγκάλιασε τη χωρική διάσταση της πολιτισμικής συμπεριφοράς και των αρχαιολογικών καταλοίπων, τονίζοντας τις διάφορες διαδικασίες που μετασχηματίζουν ή στρεβλώνουν τα υλικά μορφικά, χωρικά, ποσοτικά και σχεσιακά³⁰⁵.

Η βασική πρακτική επίπτωση της έννοιας του μετασχηματισμού είναι πολύ απλή. Ανεξάρτητα από το πόσες μαρτυρίες υπάρχουν, ο αρχαιολόγος δεν μπορεί να αναγνώσει τη συμπεριφορά και την οργάνωση απευθείας από τα πρότυπα που ανακαλύπτονται στο αρχαιολογικό αρχείο. Ωστόσο, επειδή οι διαδικασίες σχηματισμού εμπεριέχουν τη λειτουργία της σχηματοποίησης³⁰⁶, οι όποιες στρεβλώσεις μπορούν να αποκατασταθούν με τη χρήση των κατάλληλων αναλυτικών και συσχετιστικών εργαλείων, τα οποία στηρίζονται στη γνώση των νόμων που διέπουν αυτές τις διαδικασίες³⁰⁷.

Πρόσφατες εξελίξεις στην ταφονομία, τη γεωαρχαιολογία, την ιστορική αρχαιολογία, την εθνοαρχαιολογία και την πειραματική αρχαιολογία υποστηρίζουν τα βασικά δόγματα της έννοιας του μετασχηματισμού. Έχει αποδειχθεί ότι οι διαδικασίες σχηματισμού μετασχηματίζουν μορφικά τα τέχνηρα, χωρικά, ποσοτικά και μεταβάλλουν τις σχέσεις τους με το περιβάλλον ή άλλα τέχνηρα. Επίσης, μπορούν να δημιουργήσουν πρότυπα για τα τέχνηρα, τα οποία δε σχετίζονται με προηγούμενες συμπεριφορές που διερευνούνται.

³⁰⁴ Reid *et al* 1975, 209-224.

³⁰⁵ Schiffer 1978, 141-163.

³⁰⁶ Collins 1975, 25-32.

³⁰⁷ Schiffer 1976, 12.

Εντέλει, παρουσιάζουν κανονικότητα, που μπορεί να εκφραστεί συνήθως στατιστικά ως νόμος³⁰⁸.

Οι παράγοντες μετασχηματισμού περιλαμβάνουν δραστηριότητες επαναχρησιμοποίησης των υλικών καταλοίπων στις πρωτογενείς αποθέσεις τους με την ολική ή μερική ανακύκλωση κτισμάτων, τεχνέργων, κατασκευών, οικοδομημένων ή απορριμμάτων σε μια θέση, από τους ίδιους ή άλλους χρήστες, για μικρό ή μεγάλο χρονικό διάστημα, για λειτουργικούς ή συμβολικούς σκοπούς³⁰⁹. Η κεραμική, για παράδειγμα, επαναχρησιμοποιείται ευρέως, για σκοπούς που κυμαίνονται από την ένταξη οστράκων στην αρχιτεκτονική μέχρι τη διατροφή των ζώων³¹⁰. Οι περιπτώσεις επαναχρησιμοποίησης λίθινων τεχνέργων αφθονούν, επίσης, στη βιβλιογραφία, από τις φθαρμένες μυλόπετρες που μετατρέπονται σε δομικά υλικά, έως τα σπασμένα μαχαίρια που επαναχρησιμοποιούνται ως ξύστρες. Ακόμη και τα τρόφιμα που απορρίπτονται, μπορούν να χρησιμοποιηθούν εκ νέου σε ζωοτροφές³¹¹.

Περιλαμβάνουν, επίσης, δραστηριότητες ανθρωπογενούς ή μη διαταραχής των υλικών καταλοίπων στις πρωτογενείς αποθέσεις τους, όπως είναι η επιφανειακή ή πλήρης διαταραχή τεχνέργων, κτισμάτων κ.λπ., η οποία προκαλείται από οικοδομικές δραστηριότητες, άρωση, εκμετάλλευση λατομείων και ορυχείων, βιομηχανικές εργασίες, λαθρανασκαφές, πολεμικές καταστροφές ή άλλες βίαιες ιστορικές δράσεις κ.λπ.

Ανάμεσα στους παράγοντες συντήρησης περιλαμβάνεται το άνυδρο περιβάλλον, που αποτρέπει τη φθορά λόγω της έλλειψης νερού, γεγονός που εξασφαλίζει ότι δεν είναι δυνατόν να δημιουργηθούν οι καταστρεπτικοί μικροοργανισμοί που προκαλούν σήψη στο ανθρώπινο σώμα ή σε παπύρους. Το φαινόμενο παρατηρήθηκε αρχικά στην Αίγυπτο, όπου το ξηρό περιβάλλον διατήρησε σώματα της προδυναστικής περιόδου σχεδόν άθικτα, με δέρμα, τρίχες και νύχια, χωρίς τεχνητή μομιοποίηση³¹².

Περιλαμβάνονται, επίσης, περιπτώσεις ανθρωπογενούς εμπρησμού, ή φυσικών διεργασιών που προκαλούνται από ηφαιστειακή δραστηριότητα, καθώς η φωτιά ψήνει και, επομένως, συντηρεί διάφορες οργανικές ύλες και αντικείμενα

³⁰⁸ Schiffer 1996, 11.

³⁰⁹ Κουκουζέλη κ.ά. 2003, 101.

³¹⁰ Lister and Lister 1981, 66-78.

³¹¹ Horne 1983, 16-21.

³¹² Renfrew and Bahn 2016, 54.

από ωμό πηλό³¹³. Όσον αφορά στην ηφαιστειακή δραστηριότητα, τα πυροκλαστικά υλικά που προκύπτουν από την έκρηξη ενός ηφαιστείου χωρίζονται σε λατυποπαγή και τόφφους. Τα λατυποπαγή είναι συνήθως μεγάλα σώματα, ενώ οι τόφφοι που σχηματίζονται από την ηφαιστειακή τέφρα έχουν το μέγεθος της σκόνης ή της άμμου³¹⁴. Ως μικρά σωματίδια, οι τόφφοι μεταφέρονται εκατοντάδες μίλια από τον άνεμο, καθορίζοντας έναν τεράστιο ορίζοντα για κάποια συγκεκριμένη ηφαιστειακή έκρηξη³¹⁵.

Οι τόφφοι θάβουν και συνεπώς «συντηρούν» ευρήματα για την πρώιμη εξέλιξη των ανθρωποειδών. Ο Hill στην Τανζανία ανακάλυψε ίχνη εξαφανισμένων ζώων και ανθρωποειδών ηλικίας άνω των 3,6 εκατομμυρίων ετών³¹⁶. Η ηφαιστειακή τέφρα γεμίζει τα ίχνη, αρχικά διαμορφωμένα σε μαλακή λάσπη, διατηρώντας τα όταν οι αποθέσεις σκληρύνονται σε βράχο. Το χαρακτηριστικότερο παράδειγμα ηφαιστειακών επιδράσεων στο αρχαιολογικό αρχείο είναι η ρωμαϊκή πόλη Πομπηία, η οποία θάφτηκε από πυροκλάστες από την έκρηξη του Βεζούβιου το 79³¹⁷. Η «συντήρηση» που παρείχε η καταστροφική εγκατάλειψη και ταφή στην Πομπηία έγινε κριτήριο για την αξιολόγηση στοιχείων που έχουν επιβιώσει σε ανάλογες περιπτώσεις³¹⁸.

3.4. Περιβαλλοντικοί παράγοντες μετασχηματισμού

Υφίσταται στενή σχέση αλληλεπίδρασης ανάμεσα στα αρχαιολογικά δεδομένα και το άμεσο περιβάλλον τους, ανθρωπογενές ή φυσικό. Η συγκεκριμένη αλληλεπίδραση είναι υπεύθυνη για τη φθορά ή τη συντήρηση των αρχαιολογικών καταλοίπων. Ο τρόπος και ο ρυθμός φθοράς εξαρτάται σε μεγάλο βαθμό από τις ιδιότητες του αρχαιολογικού υλικού και από τη φύση του περιβάλλοντος ή του μικροπεριβάλλοντός τους³¹⁹.

³¹³ Renfrew and Bahn 2001, 150.

³¹⁴ Butzer 1971, 201.

³¹⁵ Steen-McIntyre 1985, 265-302.

³¹⁶ Βλ. Leakey and Lewin 1978, 70-71, Leakey 1979, 446-457.

³¹⁷ Βλ. Jashemski 1979, 587-622, Richardson 1978, 23-41, Will 1979.

³¹⁸ Binford 1981, 195-208.

³¹⁹ Κουκουζέλη κ.ά. 2003, 101-102.

Οι περιβαλλοντικοί παράγοντες φθοράς ομαδοποιούνται παραδοσιακά, ανάλογα με τον τρόπο που επενεργούν σε υλικά, σε χημικούς, φυσικούς ή βιολογικούς³²⁰.

Οι χημικοί παράγοντες είναι διαδεδομένοι στο αρχαιολογικό πλαίσιο. Η ατμόσφαιρα περιέχει νερό και οξυγόνο, τα οποία επαρκούν για την έναρξη πολλών χημικών αντιδράσεων, συμπεριλαμβανομένης της οξειδωσης των οργανικών υλικών και της διάβρωσης ορισμένων μετάλλων. Οι θερμοκρασίες επηρεάζουν τα ποσοστά των χημικών αντιδράσεων, ιδιαίτερα εκείνων που αφορούν το νερό. Ο κανόνας υποδεικνύει ότι οι ρυθμοί αντίδρασης διπλασιάζονται για κάθε αύξηση της θερμοκρασίας της τάξης των 10°C. Σύμφωνα με αυτή την αρχή, τα ποσοστά χημικής φθοράς γενικά αυξάνονται, όσο αυξάνεται η θερμοκρασία³²¹.

Το περιβάλλον των θαμμένων αντικειμένων ευνοεί τις γρήγορες χημικές αλλαγές. Το έδαφος είναι συχνά υγρό, διευκολύνοντας έτσι τις χημικές αντιδράσεις. Επιπλέον, περιέχει δραστικές ενώσεις, όπως οξέα και βάσεις, που συμμετέχουν σε πολλές διεργασίες φθοράς³²². Τα όξινα εδάφη διαλύουν τα οστά, ενώ τα πολύ βασικά εδάφη καταστρέφουν τη γύρη. Πολλές αρχαιολογικές αποθέσεις περιέχουν, επίσης, υψηλή συγκέντρωση αλάτων, τα οποία καθυστερούν ορισμένους βιολογικούς παράγοντες αποσάθρωσης, αλλά τελικά οδηγούν σε σοβαρή διάβρωση μετάλλων όπως ο σίδηρος, ο άργυρος και ο χαλκός, καθώς και στη φθορά των λίθινων και των κεραμικών τεχνέργων³²³.

Οι φυσικοί παράγοντες φθοράς είναι πανταχού παρόντες στα περισσότερα περιβάλλοντα. Οι διαδικασίες που επηρεάζουν τις αρχαιολογικές θέσεις και τις αποθέσεις συνήθως μετασχηματίζουν τις τυπικές ιδιότητες των αντικειμένων. Έτσι, τα ηφαιστεια, οι τυφώνες, οι σεισμοί, οι κατολισθήσεις και άλλες φυσικές καταστροφές μεταβάλλουν αντικείμενα, ειδικά τα κτίσματα, στο αρχαιολογικό πλαίσιο. Το νερό, ιδιαίτερα το κινούμενο νερό, είναι επίσης ισχυρός φυσικός παράγοντας φθοράς. Σε παραποτάμιες θέσεις και στις ακτογραμμές τα τέχνηρα φθείρονται και αφαιρούνται από το αρχαιολογικό αρχείο, επιπλέον οι λίθοι και τα

³²⁰ Βλ. Dowman 1970, 5, Rathje and Schiffer 1982, 130.

³²¹ Wessel and Thom 1954, 61.

³²² Greathouse *et al* 1954, 109-110.

³²³ Schiffer 1987, 148.

όστρεα καταστρέφονται με τρόπους που μερικές φορές μοιάζουν με πολιτιστική τροποποίηση³²⁴.

Η αποστράγγιση των όμβριων υδάτων βροχής σε οικισμούς και χώρους οδηγεί σε διάβρωση τοίχων και δρόμων. Σε περιοχές με κακή αποστράγγιση η υγρασία προωθεί όλες τις άλλες διεργασίες αποσύνθεσης. Το νερό προκαλεί, επίσης, φυσικές μεταβολές στα πορώδη και υγροσκοπικά υλικά. Για παράδειγμα, η εναλλακτική διαβροχή και ξήρανση του ξύλου προκαλεί ρωγμές κατά μήκος των κόκκων. Το νερό στη στερεά μορφή του πάγου είναι ισχυρός φυσικός παράγοντας φθοράς, καθώς μπορεί να επιμηκύνει ρωγμές σε βράχους και σκυρόδεμα, να διαβρώσει την επιφάνεια πορώδους βράχου και να επιφέρει ευρύ φάσμα λεπτότερων επιδράσεων σε διάφορα υλικά στο αρχαιολογικό πλαίσιο³²⁵. Ο άνεμος με τη σειρά του τροποποιεί τέχνηρα, ειδικά όταν μεταφέρει σωματίδια. Στις ερήμους, για παράδειγμα, τα αιωρούμενα σωματίδια συμβάλλουν στις φυσιολογικές διεργασίες φθοράς των φυσικών υλικών και των πολιτισμικών καταλοίπων³²⁶. Η ίδια φυσική φθορά εξαιτίας του ανέμου παρατηρείται και σε εκτεθειμένα ασβεστοκονιάματα (stucco)³²⁷.

Το ηλιακό φως είναι, επίσης, παράγοντας φυσικής φθοράς, επειδή προκαλεί βραχυπρόθεσμους κύκλους θερμικής διαστολής και συστολής σε αντικείμενα. Σε μεγάλες χρονικές περιόδους, τα αποτελέσματα της θερμικής ανακύκλωσης μπορούν να είναι αισθητά σε ρωγμές που παρατηρούνται σε μεγάλους τοίχους. Εκτός αυτού, δεδομένου ότι τόσο το ορατό όσο και το υπεριώδες φως είναι μορφές ενέργειας, μπορούν να συμμετάσχουν σε χημικές αντιδράσεις, που ενισχύονται συχνά από την παρουσία υγρασίας. Η φωτεινή ενέργεια απορροφάται ευκολότερα στις κίτρινες χρωστικές ουσίες, που απορροφούν το μπλε φως και στις βαφές που περιέχουν σίδηρο και άλλες ακαθαρσίες³²⁸.

³²⁴ Κουκουζέλη κ.ά. 2003, 102.

³²⁵ Shiffer 1987, 149.

³²⁶ Schiffer *et al* 2010, 44.

³²⁷ Shoet *et al* 1999, 1666-1678.

³²⁸ Cronyn 1990, 36.

Παράγοντες φθοράς του αρχαιολογικού αρχείου

Φυσικοί	Χημικοί	Βιολογικοί
<ul style="list-style-type: none"> • Υδροδυναμική ενέργεια και η άνοδος ή πτώση της υδατικής στάθμης, φυσική ή ανθρωπογενής, με φράγματα ή αρδευτικά, υδροηλεκτρικά και αποξηραντικά έργα (προκαλεί διάβρωση θέσεων ή την επικάλυψή τους με ιζηματογενείς αποθέσεις και φθορά τεχνέργων) • Ηλιακή θερμότητα (σήψη οργανικών καταλοίπων και θραύση διαφόρων άλλων υλικών, εξαιτίας διαστολής ή συστολής, καθώς και ρωγμές στη γη, που προκαλούν μετατόπιση θαμμένων αντικειμένων) • Εναλλαγή υγρασίας και ξηρασίας (θραύση υλικών, λόγω διαστολής και συστολής) • Τήξη και πήξη πάγου (θραύση υλικών και μετατόπιση θαμμένων αντικείμενων) • Αιολική ενέργεια και φυσικές καταστροφές (ριζικές μεταβολές στα αρχαία κατάλοιπα, όπως διάβρωση, επικάλυψη με ιζηματογενείς αποθέσεις, κατάρρευση, θραύση και καύση) 	<ul style="list-style-type: none"> • Όμβρια ύδατα ή υγρασία σε συνδυασμό με το οξυγόνο της ατμόσφαιρας (οξειδωση μετάλλων) • Νερό σε συνδυασμό με διάφορες χημικές ενώσεις, που περιέχονται στη μολυσμένη ατμόσφαιρα (χημική καταστροφή και μεταβολή των υλικών, επικάλυψη λίθων και μετάλλων με πατίνα) • Νερό σε συνδυασμό με αλάτι, όπως το θαλάσσιο νερό (οξειδώνει μέταλλα)³²⁹ • Ηλιακή ακτινοβολία (προκαλεί τη χημική φθορά των οργανικών καταλοίπων) • Βασικά περιβάλλοντα (οξειδώνουν μέταλλα και διαβρώνουν λίθους, κεραμική και τοιχογραφίες) • Όξινα περιβάλλοντα (προκαλούν οξείδωση μετάλλων και διάβρωση λίθων, κεραμικής και τοιχογραφιών επιπλέον, διαλύουν τα οστά³³⁰ και αποσυνθέτουν τα οργανικά υλικά) 	<ul style="list-style-type: none"> • Μικροοργανισμοί, δηλαδή μύκητες και βακτηρίδια, έντομα, θαλάσσιοι οργανισμοί, πτωματοφάγα και άλλα ζώα (προκαλούν τη σήψη ξύλου και άλλων οργανικών καταλοίπων, ιδίως μέσα σε υγρά εδάφη, και την καταστροφή άλλων υλικών, ακόμη και λίθων και μετάλλων επιπλέον, τα ζώα μπορούν να προκαλέσουν διαταραχή αρχαίων αποθέσεων)

Πιν. 2 Φυσικοί, χημικοί και βιολογικοί παράγοντες φθοράς αρχαίων καταλοίπων³³¹.

Είναι σύνηθες σήμερα να μην ταξινομούνται οι παράγοντες της φθοράς σε διακριτές φυσικές ή χημικές κατηγορίες, αλλά σε μία σύνθετη κατηγορία φυσικοχημικών παραγόντων αποσύνθεσης. Ορισμένες φυσικές ουσίες όπως το όζον και οι υδρατμοί, εκτός από τους ανθρωπογενείς ατμοσφαιρικούς και υδατικούς ρύπους, συγκαταλέγονται στους πιο δραστικούς φυσικοχημικούς παράγοντες που επηρεάζουν τα αρχαιολογικά κατάλοιπα³³².

³²⁹ Renfrew and Bahn 2001, 56.

³³⁰ Renfrew and Bahn 2001, 440.

³³¹ Προσαρμογή από Κουκουζέλη κ.ά. 2003, 102-103.

³³² Goffer 2007, 423.

3.5. Περιβαλλοντικοί παράγοντες διατήρησης του αρχαιολογικού αρχείου

Η αρχαιολογία συμβάλλει στην κατανόηση των διεργασιών φθοράς και διατήρησης του αρχαιολογικού αρχείου, καθώς οι αρχαιολόγοι επιδεικνύουν σωρεία παραδειγμάτων διατήρησης που επηρεάζουν τη διαμόρφωση των αρχών της φθοράς σε πολλούς κλάδους. Οι μη αρχαιολόγοι εμπλουτίζουν σε αρκετές περιπτώσεις τις έρευνές τους για τους παράγοντες διατήρησης με ενδιαφέρουσες αρχαιολογικές υποθέσεις, που προκύπτουν από το πεδίο³³³.

Δύο ακραίες προοπτικές κυριαρχούν στις συζητήσεις για τη διατήρηση, που παρεμποδίζουν την κατανόηση των διαδικασιών της φθοράς. Η μία άποψη θεωρεί το φυσικό περιβάλλον υπεύθυνο σε μεγάλο βαθμό για την τελική κατάληξη των τεχνέργων. Αυτή η προοπτική, που χαρακτηρίζει τη διατήρηση των τεχνέργων κακή στο τροπικό περιβάλλον και καλή στην έρημο, φαίνεται να αγνοεί όχι μόνο τα χαρακτηριστικά του τεχνέργου, αλλά και μικροπεριβαλλοντικές παραλλαγές πολιτισμικής και φυσικής προέλευσης. Ουσιαστικά, ο τρόπος και ο ρυθμός αλληλεπίδρασης μεταξύ ενός τεχνέργου και του περιβάλλοντος καθορίζονται από τη σύνθεση του τεχνέργου και από τη φύση του άμεσου περιβάλλοντος³³⁴.

Σε μία αρχαιολογική θέση ανακαλύπτονται μικροπεριβάλλοντα με δυνατότητες φθοράς που διαφέρουν πολύ από το ένα το άλλο. Για παράδειγμα, το κλίμα του αρκτικού κύκλου δεν ευνοεί την ανάπτυξη μυκήτων. Ωστόσο, αερομεταφερόμενα σπόρια μύκητα σε θερμότερα, πολιτισμικά δημιουργημένα ενδιαιτήματα και εγκαταλειμμένες κατοικίες στον αρκτικό κύκλο³³⁵. Παρόλο που οι γενικές περιβαλλοντικές παράμετροι διαμορφώνουν γενικά πρότυπα διατήρησης, δεν λαμβάνουν επαρκώς υπόψη τους την ποικιλομορφία στη διατήρηση που ανακαλύπτουν οι αρχαιολόγοι συνήθως, ακόμη και στην ίδια περιβαλλοντική ζώνη ή θέση³³⁶.

Η άλλη ακραία άποψη μεταθέτει το βάρος της διατήρησης του τεχνέργου στο ίδιο το τέχνηρο. Η βιβλιογραφία είναι γεμάτη με αναφορές που δηλώνουν

³³³ Findlay 1975, 102.

³³⁴ Kibblewhite et al 2015, 249-263.

³³⁵ Kirtsideli et al 2017, 300.

³³⁶ Schiffer 1987, 146.

ότι δεν υπήρχαν φθαρτά (ευαίσθητα) τέχνηρα³³⁷ ή ότι επιβίωσαν μόνο τα άφθαρτα (ανθεκτικά)³³⁸. Ωστόσο, οι ιδιότητες των τεχνέργων δεν καθορίζουν αφεαυτού τον τρόπο με τον οποίο θα αλληλεπιδράσουν με το περιβάλλον. Ακόμη και τα πλέον φθαρτά τέχνηρα, όπως το χαρτί ή τα κλωστοϋφαντουργικά προϊόντα, μπορούν να επιβιώσουν χιλιετίες υπό τις κατάλληλες συνθήκες. Κατά τον ίδιο τρόπο τα άφθαρτα τέχνηρα, από σίδηρο ή χάλυβα, φθείρονται σε μια δεκαετία, αν βρεθούν σε ορισμένα μικροπεριβάλλοντα. Συνεπώς, δεν έχει νόημα ο χαρακτηρισμός «φθαρτό» ή «άφθαρτο» και είναι αναγκαίο να εστιαστεί η έρευνα στις ιδιότητες των τεχνέργων και στα χαρακτηριστικά του περιβάλλοντος της απόθεσης, που καθορίζουν από κοινού τον τρόπο και τους ρυθμούς φθοράς του τεχνέργου³³⁹.

Η συγκεκριμένη προοπτική έχει συνέπειες σε ό,τι αφορά στις παραδοσιακές συζητήσεις για «παράδοξα» της διατήρησης³⁴⁰. Το παράδοξο της διατήρησης αφορά στην επιβίωση ενός ευπαθούς τεχνέργου σε δυσμενές περιβάλλον. Η έννοια του παράδοξου, που χρησιμοποιείται για να ερμηνεύσει τις πολλές ανωμαλίες που δημιουργούνται από διατηρημένα τέχνηρα με επίκεντρο το περιβάλλον, έχει ξεπεράσει πλέον τη χρησιμότητά της. Κάθε περίπτωση διατήρησης ή φθοράς εξηγείται από το ίδιο σύνολο γενικών αρχών που διέπουν όλες τις αλληλεπιδράσεις τεχνέργου-περιβάλλοντος³⁴¹.

Μια κοινή κατηγορία παράδοξων συγκροτούν τα άκαυστα οργανικά κατάλοιπα, που επιβιώνουν σε χώρους με γενικά δυσμενείς συνθήκες διατήρησης. Η διατήρηση τέτοιων οικοδομημένων είναι δυνατόν να ερμηνευθεί με πολλούς τρόπους. Για παράδειγμα, αν οι παράγοντες αποσάθρωσης είναι βακτήρια και μύκητες, η βαθιά και ταχεία ταφή ενός αντικειμένου μπορεί να το απομακρύνει αποτελεσματικά από τις επιφανειακές ζώνες, όπου οι παράγοντες αυτοί είναι πιο ενεργοί. Τα πηγάδια και οι βόθροι που γεμίζουν γρήγορα με απορρίμματα, παρέχουν ενίοτε τέτοια μικροπεριβάλλοντα διατήρησης³⁴².

³³⁷ Για την αντίληψη του «φθαρτού» βλ. Hyland et al 2003, 297-416.

³³⁸ Για τη διάκριση μεταξύ φθαρτού και άφθαρτου βλ. Stothers and Tucker 2006, 181.

³³⁹ Schiffer 1987, 147.

³⁴⁰ Schiffer et al 2010, 44.

³⁴¹ Schiffer and Rathje 1973, 169-179.

³⁴² Knorzer 1984, 331-338.

Το χημικό περιβάλλον, επίσης, μπορεί να εμποδίσει την ανάπτυξη οργανισμών αποσύνθεσης. Για παράδειγμα, έχουν βρεθεί σε αρχαία ορυχεία άλατος διατηρημένα σώματα³⁴³. Συνοπτικά, σε όλες τις περιπτώσεις διατήρησης «παράδοξων» λείπουν οι κατάλληλες ερμηνευτικές υποθέσεις, βασισμένες σε γενικές αρχές που αναφέρονται σε συγκεκριμένες συνθήκες του περιβάλλοντος εναπόθεσης. Δεν υπάρχουν παράδοξα διατήρησης, μόνο ελλείψεις εξηγήσεις για τα διατηρημένα τέχνηρα ή οικοδομημένα. Οι περιβαλλοντικοί παράγοντες διατήρησης συνεισφέρουν στη διατήρηση των οργανικών καταλοίπων. Οι περισσότεροι από αυτούς το επιτυγχάνουν αναχαιτίζοντας, με χημικό ή φυσικό τρόπο, την ανάπτυξη των μικροοργανισμών, οι οποίοι ζουν επάνω και μέσα στις οργανικές ύλες και προκαλούν την αποσύνθεσή τους.

Το *permafrost*, ιδιαίτερα αλλά όχι αποκλειστικά στον αρκτικό κύκλο³⁴⁴, ως φυσικός παράγοντας συμβάλλει αποφασιστικά στη διατήρηση καταλοίπων ξύλου. Η φθορά του αρχαιολογικού ξύλου σε θερμοκρασίες κατάψυξης και απόψυξης μελετάται στη Δυτική Γροιλανδία με συνδυασμό περιβαλλοντικής παρακολούθησης, μέτρησης της κατανάλωσης οξυγόνου και μικροσκοπίας των δειγμάτων ξύλου. Το μόνιμα κατεψυγμένο ξύλο διατηρείται σε πολύ καλή κατάσταση μετά από 2-4.000 χρόνια³⁴⁵. Ωστόσο, η έρευνα υποδεικνύει ότι το 70% της υποδομής των αρκτικών περιοχών του υπερπληθυσμού κινδυνεύει, καθώς το κλίμα θερμαίνεται. Η απόψυξη του *permafrost*, εκτός του ότι δυνητικά αποτελεί απειλή για την αρκτική περιβαλλοντική δομή³⁴⁶, αποτελεί απειλή και για το ίδιο το αρχαιολογικό αρχείο³⁴⁷.

³⁴³ Aali *et al* 2012, 61–81.

³⁴⁴ Tarnocai 2009, 3-17.

³⁴⁵ Matthiesen *et al* 2014, 479-495.

³⁴⁶ Hjort *et al* 2018, 1-11.

³⁴⁷ Hollesen *et al* 2017, 1175–1189.

Περιβαλλοντικοί παράγοντες διατήρησης του αρχαιολογικού αρχείου	
Χημικοί	Φυσικοί
<ul style="list-style-type: none"> • οξειδωμένος χαλκός και σίδηρος (είναι τοξικά για τους μικροοργανισμούς και ευνοούν τη διατήρηση των οργανικών καταλοίπων) • ασβεστολιθικά περιβάλλοντα, όπως ασβεστολιθικά εδάφη και πολλά σπήλαια (διατηρούν οστά και μέταλλα) • αλατούχα εδάφη, αλάτι σε μεγάλες πυκνότητες, συνδυασμός άλατος και πετρελαίου, φυσική άσφαλτος και άλλες ασυνήθιστες γεωλογικές συνθήκες (διατηρούν οργανικά κατάλοιπα) 	<ul style="list-style-type: none"> • φωτιά (διατηρεί φυτικές ύλες, σπόρους και ξύλο, καθώς και αντικείμενα από ωμό πηλό) • μικροπεριβάλλοντα σπηλαίων η βραχοσκεπών. • ξηρά και άνυδρα περιβάλλοντα, έρημοι. • ψυχρά περιβάλλοντα, <i>permafrost</i> • αναερόβια περιβάλλοντα, όπως εκτάσεις γης με στάσιμα ή λασπώδη νερά - έλη, βάλτοι, τέλματα ή βούρκοι, ωκεάνια ή λιμναία ιζήματα. • Σημεία καλυμμένα από ιζηματογενείς αποθέσεις, μη διαπερατά από το οξυγόνο και ερμητικά κλεισμένοι χώροι

Πιν. 4 Περιβαλλοντικοί παράγοντες διατήρησης του αρχαιολογικού αρχείου³⁴⁸.

3.6. Το αρχαιολογικό πλαίσιο

Τα αρχαιολογικά ευρήματα δεν υφίστανται ως μεμονωμένες, ανεξάρτητες μονάδες, εκτός τόπου και χρόνου, αλλά το καθένα από αυτά εντάσσεται σε ένα αρχαιολογικό πλαίσιο. Το αρχαιολογικό πλαίσιο είναι η ακριβής θέση ενός αρχαιολογικού ευρήματος στον χώρο και στον χρόνο, μαζί με το άμεσο περιβάλλον του. Δημιουργείται με την επίδραση των παραγόντων διαμόρφωσης ή μετασχηματισμού του αρχαιολογικού αρχείου σε μία συγκεκριμένη θέση απόθεσης του ευρήματος.

Πολλά από τα αρχαία αντικείμενα, που εκτίθενται στην αγορά έργων τέχνης και στα μουσεία είναι προϊόντα λαθρανασκαφών και βρίσκονται, επομένως, «εκτός αρχαιολογικού πλαισίου». Ο Cannon-Brookes αναφέρεται στα ληλατημένα αντικείμενα ως «πολιτιστικά ορφανά, τα οποία, αποκομμένα από το πλαίσιο τους, παραμένουν για πάντα βουβά και είναι ουσιαστικά άχρηστα για ακαδημαϊκούς σκοπούς»³⁴⁹. Ορισμένοι αρχαιολόγοι εκτιμούν ότι τα συλημένα τέχνηρα έχουν χάσει το 95% της ερευνητικής αξίας τους, γιατί δεν μπορούν να αφηγηθούν τι συνέβη στο παρελθόν³⁵⁰. Έτσι, γίνεται πιθανώς κατανοητό ότι, από επιστημονική άποψη, τέχνηρα εκτός πλαισίου δεν είναι εξίσου χρήσιμα με εκείνα που ανακτώνται μέσω μιας ορθής ανασκαφικής μεθόδου. Η συγκεκριμένη

³⁴⁸ Προσαρμογή από Κουκουζέλη κ.ά. 2003, 103-104.

³⁴⁹ Cannon-Brookes 1994, 350.

³⁵⁰ Monastersky 1990, 393.

άποψη δεν είναι νέα. Σύμφωνα με τον Howard Carter, ανασκαφέα του τάφου του βασιλιά Τουταγχαμών στην Αίγυπτο, «αν το έργο στο πεδίο διεξαγόταν με συστηματικό και επιστημονικό τρόπο η γνώση μας για την αιγυπτιακή αρχαιολογία θα ήταν τουλάχιστον 50% μεγαλύτερη. Υπάρχουν αμέτρητα εγκαταλελειμμένα αντικείμενα στις αποθήκες των μουσείων που θα μάς έδιναν πολύτιμες πληροφορίες και θα μπορούσαν να μάς πουν από πού ήρθαν»³⁵¹. Δεν αποτελεί απλό ατύχημα το γεγονός ότι οι αρχαιολογικοί χώροι είναι στόχοι όσων εμπλέκονται στην παράνομη διακίνηση τεχνέργων. Δεδομένου ότι πολλές γεωγραφικές θέσεις βρίσκονται σε απομακρυσμένες περιοχές, όπως για παράδειγμα η Κυμισάλα, ο κίνδυνος σύλληψης των λαθρανασκαφών είναι σχετικά μικρός. Σημαντικότερο, ωστόσο, είναι το γεγονός ότι τα τέχνηρα που δεν έχουν τεκμηριωθεί, φωτογραφηθεί ή καταγραφεί είναι αδύνατο να εντοπιστούν και, ως εκ τούτου, μπορούν να διακινηθούν στη διεθνή αγορά ατιμωρητί³⁵².

Το πλαίσιο είναι εξαιρετικά σημαντικό για τον αρχαιολόγο, καθώς είναι το θεμέλιο της αρχαιολογίας και παρέχει στα ευρήματα την αυθεντικότητα και την αρχαιολογική σημασία τους³⁵³. Έτσι, τα αρχαιολογικά ευρήματα έχουν επιστημονική αξία, μόνο όταν είναι γνωστό το πλαίσιο τους. Σύμφωνα με τον Taylor «όσο περισσότερα είναι γνωστά για το πλαίσιο που σχετίζεται με ευρήματα, τόσο περισσότερα είναι δυνατόν να συναχθούν όχι μόνο για τη λειτουργία τους σε μια κοινωνία του παρελθόντος, αλλά και για τον τρόπο και την αιτία κατασκευής τους, όπως και για ευρύτερα ζητήματα που εμπλέκουν την αρχαία οικονομία, το εμπόριο ή τη θρησκεία»³⁵⁴. Η μελέτη του αρχαιολογικού πλαισίου των ευρημάτων μπορεί να μας δώσει λεπτομερείς ενδείξεις σχετικά με το νόημά τους, συμβάλλοντας έτσι στην κατανόηση και ερμηνεία της αρχαίας ανθρώπινης συμπεριφοράς και του αρχαίου υλικού πολιτισμού³⁵⁵.

Το αρχαιολογικό πλαίσιο ενός ευρήματος είναι το αποτέλεσμα των παραγόντων σχηματισμού ή των παραγόντων μετασχηματισμού του αρχαιολογικού αρχείου, που ευθύνονται για την απόθεση του ευρήματος. Διακρίνεται σε πρωτογενές και δευτερογενές αρχαιολογικό πλαίσιο. Το

³⁵¹ Carter and Mace 1923, 125.

³⁵² Platthy 1993, 45.

³⁵³ Ford 1977, 14.

³⁵⁴ Taylor 1948, 154.

³⁵⁵ Κουκουζέλη κ.ά. 2003, 110.

πρωτογενές πλαίσιο είναι το πλαίσιο ενός αρχαιολογικού ευρήματος όπως αυτό δημιουργήθηκε από τους παράγοντες σχηματισμού της αρχαιολογικής μαρτυρίας. Με άλλα λόγια είναι το προϊόν της πρωτογενούς απόθεσης του ευρήματος. Διακρίνεται για την αδιατάρακτη συσχέτιση, μήτρα και προέλευση, κατάσταση, δηλαδή, στην οποία δεν έχει διαταραχθεί η αρχική εναπόθεση των αρχαιολογικών δεδομένων³⁵⁶. Το δευτερογενές πλαίσιο είναι το πλαίσιο ενός αρχαιολογικού ευρήματος, που έχει διαταραχθεί από ανθρώπινη δραστηριότητα ή από φυσικά φαινόμενα. Η προέλευση, η συσχέτιση και η μήτρα τέτοιων αρχαιολογικών δεδομένων έχουν μεταβληθεί πλήρως ή μερικώς από μετασχηματιστικές διαδικασίες μετά την αρχική εναπόθεση³⁵⁷. Η διάκριση ανάμεσα σε πρωτογενές και δευτερογενές πλαίσιο φαίνεται ότι προστέθηκε στις συζητήσεις για το αρχαιολογικό πλαίσιο και τη σχετική βιβλιογραφία μετά τη δεκαετία του '60³⁵⁸.

Έχει προταθεί και ένα άλλο πλαίσιο, το συστημικό. Πρόκειται για συμπεριφορικό σύστημα, στο οποίο τα τέχνηρα γίνονται μέρος ενός ανακυκλούμενου συστήματος κατασκευής, χρήσης, επαναχρησιμοποίησης και απόρριψης. Μόλις τα αρχαιολογικά δεδομένα εισέλθουν στο έδαφος, αποτελούν μέρος του αρχαιολογικού πλαισίου, όπου μπορούν να συνεχίσουν να επηρεάζονται από την ανθρώπινη δράση και από φυσικές διαδικασίες³⁵⁹.

Το αρχαιολογικό πλαίσιο ενός ευρήματος προσδιορίζεται με το περίβλημα (μήτρα), την προέλευση (οριζόντια και κάθετη θέση μέσα στη μήτρα) του ευρήματος, καθώς και τη συσχέτισή του με άλλα ευρήματα. Το περίβλημα (μήτρα) είναι, το υλικό (απόθεση ή στρώμα) που περικλείει και συγκροτεί τα διάφορα αρχαιολογικά ευρήματα. Οι αποθέσεις και το έδαφος του αρχαιολογικού χώρου είναι η βασική μήτρα των υλικών υπολειμμάτων που παρουσιάζουν αρχαιολογικό ενδιαφέρον³⁶⁰. Στις περισσότερες περιπτώσεις το περίβλημα του ευρήματος είναι φυσικό, εφόσον τα υλικά που περιβάλλουν οφείλονται σε φυσικά φαινόμενα. Ενδέχεται, όμως, να είναι και πολιτισμικό, δηλαδή προϊόν ανθρώπινης δραστηριότητας. Οι φυσικές και χημικές ιδιότητες του περιβλήματος των αρχαιολογικών καταλοίπων, εφόσον καταγραφούν ορθά, είναι ενδείξεις για

³⁵⁶ Hardesty 2010, 10.

³⁵⁷ Renfrew and Bahn 2016, 50.




³⁵⁸ Lyman 2012, 225.

³⁵⁹ Schiffer 1972, 156-165.

³⁶⁰ Garrison 2016, 55.

να κατανοήσουν οι αρχαιολόγοι τα αίτια της πρωτογενούς ή δευτερογενούς απόθεσής τους³⁶¹.

Η προέλευση (provenience) είναι η ακριβής θέση, μετρήσιμη οριζόντια και κάθετα, ενός αρχαιολογικού ευρήματος στο περίβλημα όπου ανακαλύφθηκε και συνεπώς στην αρχαιολογική θέση συνολικά. Η προέλευση είναι ουσιαστική για την καταγραφή του πλαισίου του ευρήματος, τη συσχέτισή του με άλλα τέχνηρα ή κατασκευές και το γεωλογικό στρώμα μιας αρχαιολογικής θέσης³⁶². Θα πρέπει να ληφθεί υπόψη ότι η λέξη προέλευση, όσον αφορά στην αρχαιολογία, έχει δύο νοήματα, ως τόπος ανακάλυψης και ως τόπος προέλευσης. Στην προκειμένη περίπτωση γίνεται αναφορά στον τόπο ανακάλυψης³⁶³. Η καταγραφή της προέλευσης ενός ευρήματος γίνεται με δύο οριζόντιες και μία κάθετη μέτρηση και επιτρέπει τον καθορισμό της συσχέτισης (association) του ευρήματος με άλλα ευρήματα, συνήθως μέσα στο ίδιο περίβλημα (χωρικό οριζόντιο πλαίσιο) ή τη σχέση του ευρήματος με άλλα ευρήματα, που βρίσκονται σε διαφορετικά γεωλογικά στρώματα (χρονικό κάθετο πλαίσιο). Στρώματα χωρίς ευρήματα, αντιπροσωπεύουν το αποκαλούμενο χρονολογικό ή στρωματογραφικό κενό (hiatus)³⁶⁴.

Humus			
Στρώμα 1	Στρώση 1 Χώμα – ρίζες - αργοί λίθοι – τέχνηρα ή ομάδες τεχνέργων (μήτρα, συσχέτιση, επαλληλία) 		
	Στρώση 2 Χώμα – αργοί λίθοι – τέχνηρα ή ομάδες τεχνέργων (μήτρα, συσχέτιση, επαλληλία) 		
Στρώμα 2	Αμμόλιθοι (επαλληλία)	Στρωματογραφικό κενό (hiatus)	
Στρώμα 3	Απόθεση καταστροφής (επαλληλία)		
Στρώμα 4	Στρώση 1 Χώμα – αργοί λίθοι – καύσεις		Ανθρωπογενής διαταραχή (λάξευση)
	Στρώση 2 Ταφικός θάλαμος, ομάδα, terminus ante quem (επαλληλία, μήτρα, συσχέτιση) 		
Στρώμα 5	Κλίνη (επαλληλία) γεωλογική απόθεση. Ασβεστόλιθος		
Στρώμα 6	Γεωλογική απόθεση (επαλληλία). Σχιστόλιθος		

Πιν. 5. Προσαρμογή από Macleod 2005, 1.

³⁶¹ Κουκουζέλη κ.ά. 2003, 111.

³⁶² Kelly and Thomas 2014, 58.

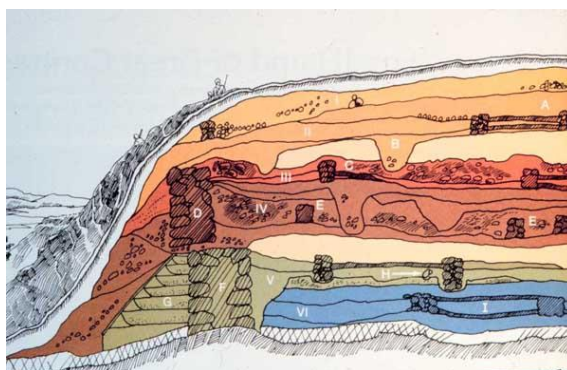
³⁶³ Price and Burton 2011, 213.

³⁶⁴ Macleod 2005, 11.

3.7. Στρωματογραφία, συσχέτιση και επαλληλία

Έως το τέλος του δέκατου ένατου αιώνα η γεωλογία άσκησε μεγάλη επίδραση στην ανάπτυξη αρχαιολογικών εννοιών³⁶⁵. Η αρχαιολογική στρωματογραφία βασίζεται στους θεμελιώδεις γεωλογικούς νόμους της «υπέρθεσης» και των «στρωμάτων» που προσδιορίζονται παλαιοντολογικά από τα απολιθώματα³⁶⁶. Ο νόμος της υπέρθεσης δεν έχει έως τώρα αναθεωρηθεί για αρχαιολογικούς σκοπούς, παρά τις μεγάλες διαφορές μεταξύ των ενοποιημένων ιζηματογενών στρωμάτων που διερευνούνται στη γεωλογία και των χαλαρών ενίοτε στρωμάτων μιας αρχαιολογικής θέσης³⁶⁷

Η στρωματογραφία (stratigraphy) είναι η μελέτη της επάλληλης διάταξης αποθέσεων ή στρωμάτων, δηλαδή της διαστρωμάτωσης (stratification) της αρχαιολογικής θέσης. Τα στρώματα της αρχαιολογικής θέσης είναι συνήθως γεωλογικά και πολιτισμικά, αλλά μπορούν να είναι και καθαρά γεωλογικά σε περιπτώσεις που υφίστανται περίοδοι εγκατάλειψης της θέσης. Τα στρώματα ποικίλλουν όσον αφορά στο βάθος, τη σύσταση, το χρώμα και το περιεχόμενό τους, ανάλογα με τις φυσικές ή πολιτισμικές μεταβολές που συμβαίνουν σε μια θέση και τη διάρκειά τους³⁶⁸. Η πολυπλοκότητα των στρωματογραφικών σχέσεων φαίνεται στην παρακάτω εικόνα. Τα διαφορετικά χρώματα αντιστοιχούν σε διαφορετικές κατοικήσεις, ταξινομημένα χρονολογικά από το τελευταίο (I) έως το αρχαιότερο (VI)



Εικ. 3.1. Διαταραγμένα στρώματα από αλληπάλληλες κατοικήσεις³⁶⁹.

³⁶⁵ Daniel 1975, 25.

³⁶⁶ Rowe 1970, 58-69.

³⁶⁷ Harris 1979, 111-17.

³⁶⁸ Renfrew and Bahn 2001, 104 και 117.

³⁶⁹ Staten Island Museum at CSI: Archaeology Study, διαθέσιμο online.

Τα αρχαιολογικά στρώματα διακρίνονται σε «κλειστά» και «διαταραγμένα». Τα κλειστά στρώματα είναι το αποτέλεσμα αδιατάρακτης πρωτογενούς δραστηριότητας, παραμένουν άθικτα από τη στιγμή της αρχικής τους διάπλασης και είναι σαφώς διαχωρισμένα μεταξύ τους³⁷⁰. Τα διαταραγμένα στρώματα είναι το αποτέλεσμα δευτερογενούς δραστηριότητας, φυσικής ή ανθρωπογενούς, με αποτέλεσμα τη διαταραχή του στρώματος από φυσική διάβρωση, τεκτονική δραστηριότητα, εκσκαφή, άντληση ή άλλη ανθρωπογενή παραβίαση των αδιατάρακτων στρωμάτων. Τα διαταραγμένα στρώματα παρουσιάζονται συνήθως σε σύνθετες αρχαιολογικές θέσεις με βαθιά διαστρωμάτωση.

Η διαταραχή των στρωμάτων μιας αρχαιολογικής θέσης απαιτεί προσεκτική παρατήρηση, καθώς η κατανομή των αρχαιολογικών δεδομένων σε μία θέση δεν μπορεί να αποδοθεί πάντα σε προηγούμενη συμπεριφορά. Ως εκ τούτου φαίνεται αβάσιμη η θέση του Binford ότι το αρχαιολογικό αρχείο είναι ένα «απολιθωμένο» αρχείο της πραγματικής λειτουργίας μιας εξαφανισμένης κοινωνίας³⁷¹. Οι Thompson και Longacre υιοθέτησαν με τη σειρά τους εξίσου ακραία θέση, δηλώνοντας ότι όλα τα κατάλοιπα μιας αρχαιολογικής θέσης είναι διαμορφωμένα ή δομημένα άμεσα από τον τρόπο οργάνωσης μιας αρχαίας κοινωνίας και τον τρόπο της συλλογικής συμπεριφοράς του λαού της³⁷². Οι Krause και Thorne, ωστόσο, έκαναν εξαίρεση σε αυτή τη γενίκευση, θεωρώντας πως δεν είναι δυνατόν να υποθέσουμε ότι όλα τα αρχαιολογικά δεδομένα μπορούν να αποδοθούν άμεσα στη συμπεριφορά των προϊστορικών κατοίκων μιας θέσης³⁷³.

Είναι ιδιαίτερα πιθανό ότι η κατανομή των αρχαιολογικών καταλοίπων είναι αποτέλεσμα της σκόπιμης ανθρώπινης δραστηριότητας. Ωστόσο, αυτό δε σημαίνει ότι η μορφοποίηση των καταλοίπων και η ανθρώπινη συμπεριφορά που την παρήγαγε είναι πανομοιότυπα. Πριν από οποιαδήποτε επίδειξη ισομορφισμού ανάμεσα στα τέχνηρα και την ανθρώπινη συμπεριφορά, πρέπει να είναι γνωστές οι διαδικασίες που επενέργησαν στη μήτρα, στην οποία ενσωματώθηκαν τα κατάλοιπα και να αξιολογηθούν τα αποτελέσματά τους. Αυτή η μήτρα είναι

³⁷⁰ Renfrew and Bahn 2001, 146.

³⁷¹ Binford 1964, 424.

³⁷² Thompson and Longacre 1966, 270.

³⁷³ Krause and Thorne 1971, 246.

συνήθως χώμα, αργοί λίθοι και άλλα συστατικά, οργανικά ή ανόργανα, που υπόκεινται σε μετασχηματισμό με πολλές χημικές, βιολογικές και μηχανικές διεργασίες. Αυτός ο συλλογισμός αναπτύχθηκε λεπτομερώς πρώτα από τον Ascher³⁷⁴ και αργότερα από τον Schiffer³⁷⁵. Και οι δύο συγγραφείς ασχολούνται με τις διαδικασίες της διαμόρφωσης του τόπου και με την επακόλουθη αφήγηση τού μετασχηματισμού του. Ο Ascher ιδιαίτερα επεσήμανε ότι η εντροπία λαμβάνει χώρα με διαφορετικούς ρυθμούς σε μια ζωντανή κοινότητα. Αφού εγκαταλειφθεί κάποια θέση, οι φυσικές διεργασίες αρχίζουν να αποδιοργανώνουν τα ανθρωπογενή πρότυπα οργάνωσης της ύλης³⁷⁶. Το έδαφος δεν είναι στατικό σώμα. Είναι ένα δυναμικό, ανοιχτό σύστημα, στο οποίο είναι δυνατόν επιδράσουν διάφορες φυσικές η ανθρωπογενείς διαδικασίες, με αποτέλεσμα όχι απλά τη μετακίνηση του εδάφους, αλλά και αντικειμένων (συμπεριλαμβανομένων των τεχνέργων), από τη μια θέση στην άλλη. Πρέπει, συνεπώς, να συμπεριλαμβάνεται ως ένα από τα σημαντικότερα φυσικά χαρακτηριστικά που διαμορφώνουν την ερμηνεία του αρχαιολογικού αρχείου³⁷⁷.

Το βάθος της διαστρωμάτωσης σε μια θέση εξαρτάται από τη φύση της αρχαιολογικής θέσης και το διάστημα της χρήσης της. Πολλές θέσεις χρησιμοποιήθηκαν επανειλημμένως επί σειρά αιώνων και συνεπώς παρουσιάζουν βαθιά διαστρωμάτωση. Βάσει του γενικού κανόνα, που στηρίζεται στην αρχή της επαλληλίας, τα υψηλότερα στρώματα είναι τα νεότερα και περιέχουν, επομένως, ευρήματα νεότερης χρονολογίας από τα κατώτερα και αρχαιότερα στρώματα. Η χρονική αλληλουχία των στρωμάτων παρέχει σχετική χρονολόγηση για τα στρώματα και τα περιεχόμενά τους. Για την ακριβέστερη, απόλυτη χρονολόγηση του κάθε στρώματος χρησιμοποιούνται είτε δείγματα εδάφους και οικοδομημένα ή κτίσματα, κατασκευές και τέχνηρα με την προϋπόθεση, βέβαια, ότι δεν αποτελούν «παραβιάσεις» από υπερκείμενα στρώματα και το στρώμα είναι «κλειστό»³⁷⁸. Η χρονολογία που προκύπτει είναι εκείνη κατά την οποία, ή μετά από την οποία, σχηματίστηκε το εξεταζόμενο στρώμα (stratum). Με άλλα λόγια, είναι το *terminus post quem* -όριο μετά από το οποίο- του στρώματος.

³⁷⁴ Ascher 1968, 43-52.

³⁷⁵ Schiffer 1972, 156-165.

³⁷⁶ Ascher 1968, 46-47 και 52.

³⁷⁷ Wood and Johnson 1978, 315-381.

³⁷⁸ Harris 1989, 7-13.

Όταν σε ένα αδιατάρακτο στρώμα υπάρχει ένας αριθμός ευρημάτων διαφόρων χρονολογιών, το εύρημα με την πιο πρόσφατη χρονολογία είναι εκείνο που παρέχει τελικά το *terminus post quem* του στρώματος³⁷⁹. Όταν υλικά, στρώματα ή ευρήματα, που μπορούν να χρονολογηθούν, βρίσκονται επάνω από ένα στρώμα ή αποτελούν «παραβιάσεις» μέσα σε αυτό, τότε παρέχουν μια χρονολογία κατά την οποία ή πριν από την οποία το κατώτερο στρώμα πρέπει να είχε αποθεθεί. Πρόκειται για το *terminus ante quem* -όριο πριν από το οποίο- για το συγκεκριμένο στρώμα³⁸⁰.

Η κατανόηση μιας διαστρωμάτωσης με διαταραγμένα στρώματα είναι ιδιαίτερα δύσκολη και απαιτεί μεγάλη προσοχή εκ μέρους των αρχαιολόγων. Ακόμα και σε διαταραγμένα στρώματα, ωστόσο, υπάρχουν αποθέσεις οι οποίες αποτελούν αυτοτελή στρωματογραφική ενότητα. Αυτές οι αποθέσεις είναι δυνατόν να αποχωριστούν από το στρώμα στο οποίο βρέθηκαν ή από το αμέσως υπερκείμενο ή υποκείμενο στρώμα και να μελετηθούν ως διακριτό σύνολο ή ομάδα (*assemblage*), αποδίδοντας τα περιεχόμενά τους στο ίδιο αρχαιολογικό πλαίσιο³⁸¹. Ο John Charman φέρεται ότι ήταν από τους πρώτους που συνειδητοποίησε τη σημασία της ομάδας. Παρόλο που ο κύριος στόχος του ήταν η μελέτη του κατακερματισμού, ασχολήθηκε με διάφορους τρόπους συσχετισμού των τεχνέργων σε σύνολα³⁸². Δύο συγγραφείς έχουν σαφώς διατυπώσει τη σημασία των ομάδων για την κατανόηση της αρχαιολογικής πρακτικής, ο Gavin Lucas και ο Chris Fowler. Ο Fowler υποστηρίζει την κατανόηση της αρχαιολογικής πρακτικής ως ομαδοποίηση, που διαμορφώνεται από υλικά, τέχνηρα, τόπους, ανθρώπους, φυτά, ζώα, τεχνικές, τεχνολογίες και ιδέες. Θεωρεί, επίσης, ότι ο αρχαιολόγος αποτελεί τμήμα αυτής της ομαδοποίησης. Ο Fowler αντλεί από το έργο του Gavin Lucas, ο οποίος θεωρεί ότι προκειμένου να υπερβεί τις προβληματικές διακρίσεις ανάμεσα στο κοινωνικό και το υλικό, η αρχαιολογική πρακτική θα πρέπει να στρέψει το ενδιαφέρον της στις «οντότητες και τις σχέσεις τους». Με αυτόν τον τρόπο η αποκαλούμενη υλικότητα είναι θεμελιωδώς σχεσιακή διαδικασία στην περίπτωση της ομάδας, και ότι αυτό που

³⁷⁹ Barker 1993, 224.

³⁸⁰ Grant et al 2005, 90.

³⁸¹ Κουκουζέλη κ.ά. 2003, 114.

³⁸² Hamilakis and Jones, 2017, 81

ενδιαφέρει πραγματικά είναι οι σχέσεις ανάμεσα στις οντότητες της ομάδας³⁸³. Η συγκεκριμένη άποψη παρουσιάζει ενδιαφέρον όχι μόνο για την ανασκαφική διαδικασία, αλλά και για τη μεταανασκαφική ανάλυση και ερμηνεία, ακόμη και τη δημοσίευση του ανασκαφικού έργου.

Η γενική ιδέα της *ομάδας* έγινε το θεμέλιο για μια σειρά σύγχρονων τάσεων στην αρχαιολογία. Έχει προταθεί, για παράδειγμα, ότι η νεωτεριστική αντίληψη της αρχαιολογίας ως ανάκτηση ή απλή αποχωμάτωση, θα πρέπει να αντικατασταθεί με την αρχαιολογία ως «ομαδοποίηση και ανα-ομαδοποίηση», μια δυναμική και δημιουργική δράση στο παρόν³⁸⁴. Μια τέτοια αντίληψη αποτελεί περαιτέρω εξέλιξη της κριτικής της έννοιας του «αρχαιολογικού αρχείου» ως δεδομένη και στατική οντότητα, η οποία ανακαλύπτεται και γίνεται αντικείμενο μελέτης από αρχαιολόγους³⁸⁵. Επιπλέον, οι πρόσφατες συζητήσεις για την παιδαγωγική της αρχαιολογίας αναδεικνύουν το σκεπτικό της ομαδοποίησης ως παραγωγικό τρόπο αναδιάρθρωσης της διδασκαλίας και της μάθησης³⁸⁶. Ακόμα και στο πλαίσιο της φωτογράφισης του αρχαιολογικού αρχείου προτάθηκε ότι η φωτογράφιση ιδωμένη στο πλαίσιο της ομαδοποίησης είναι *ομάδα*, στην οποία συμμετέχει ο αρχαιολόγος ή ο φωτογράφος, η κάμερα, το τοπίο, το τέχνηργο, το φως και οι σκιάσεις, ακόμη και οι εκάστοτε παρατηρητές της φωτογράφισης³⁸⁷.

Βάσει των παραπάνω το στρώμα (*stratum*) και η στρώση (*substratum*) αποτελούν κλειδιά σε μια σχεσιακή βάση δεδομένων (*relational database*), προκειμένου να καταγραφεί η θέση τεχνέργων ή ομάδων τεχνέργων³⁸⁸. Η εξάρτηση ενός τεχνέργου ή ομάδας τεχνέργων ως δεδομένο σε ένα στρώμα ή στρώση είναι δυνατόν να οδηγήσει σε οπτικοποιημένη απόδοση της στρωματογραφίας και συνεπώς των αναγκαίων στρωματογραφικών συσχετίσεων για την ερμηνεία των αρχαιολογικών δεδομένων³⁸⁹.

³⁸³ Lucas 2012, 167-168.

³⁸⁴ Harrison 2011, 141-161.

³⁸⁵ Hamilakis 1999, 60-79.

³⁸⁶ Cobb and Croucher 2014, 197-216.

³⁸⁷ Carabott, Hamilakis and Papargyriou 2015, 11.

³⁸⁸ Harris et al 1993, 232.

³⁸⁹ Green et al 2001, 271-376.

3.8. Αρχαιολογική έρευνα

Η αρχαιολογική έρευνα έχει διεπιστημονική φύση και διέρχεται από σειρά σταδίων που έχουν ως εξής σε ιδανικές συνθήκες:

- α. Διατύπωση ερευνητικού σχεδίου
- β. Προετοιμασία για την εκτέλεση του έργου
- γ. Ανασκαφή και ανεύρεση δεδομένων στο πεδίο
- δ. Ταξινόμηση δεδομένων
- ε. Ανάλυση δεδομένων
- στ. Ερμηνεία δεδομένων
- ζ. Τελική δημοσίευση³⁹⁰.

Η αρχαιολογική έρευνα πεδίου, αναλυτικότερα, πραγματοποιείται συνήθως σε έξι διαδοχικά στάδια, καθένα από τα οποία περιλαμβάνει μια σειρά διαφορετικών μεθόδων.

1. Αναγνώριση (Reconnaissance): Ο στόχος της αναγνώρισης είναι η διερεύνηση του τοπίου και ο εντοπισμός κάποιας αρχαιολογικής θέσης, δηλαδή περιοχές που ενδέχεται να περιέχουν συγκεντρωμένα κατάλοιπα προηγούμενης ανθρώπινης δραστηριότητας. Σε μια προσέγγιση πολλαπλών σταδίων, η αναγνώριση βοηθά στον σχεδιασμό μιας διεξοδικότερης έρευνας³⁹¹.

2. Η αξιολόγηση (Evaluation): η περιοχή για την οποία υπάρχουν υποψίες ότι περιέχει αρχαιολογικά δεδομένα ορίζεται ως θέση και στη συνέχεια υποβάλλεται σε αξιολόγηση για να καθοριστεί η καλύτερη δυνατή πορεία περαιτέρω δράσης³⁹². Σκοπός της αξιολόγησης είναι η παροχή όσο το δυνατόν περισσότερων πληροφοριών στη σχεδίαση της έρευνας (design), που αποτελεί το επόμενο στάδιο. Στο στάδιο της αξιολόγησης συγκεντρώνονται οι ερευνητικοί στόχοι και οι κοινωνικοί παράγοντες που επηρεάζουν περαιτέρω τις την έρευνα. Στη συγκεκριμένη ερευνητική διαδικασία χρησιμοποιούνται τεχνικές έρευνας επιφανείας εδάφους και υπεδάφους, αν και πολλές τοποθεσίες απαιτούν μικρής κλίμακας ανασκαφή με δοκιμαστικές τομές³⁹³.

³⁹⁰ Κουκουζέλη κ.ά. 2003, 121-122.

³⁹¹ Banning 2002, 25.

³⁹² Chazan 2017, 18.

³⁹³ Krakker *et al* 1983, 469-480.

3. Ο σχεδιασμός (Design): είναι το κύριο στάδιο της έρευνας πεδίου, επειδή στο σημείο αυτό αποφασίζεται το μέλλον της αρχαιολογικής θέσης. Η σημασία του σχεδιασμού είναι τριπλή. Αρχικά, καθορίζει τον σκοπό της έρευνας και τα αποτελέσματά της. Δεύτερον, καθορίζει τα επιθυμητά μέτρα για τη διατήρηση της αρχαιολογικής θέσης. Τρίτον, οι αποφάσεις λαμβάνονται στο στάδιο του σχεδιασμού είναι συχνά μη αναστρέψιμες, ιδίως εκείνες που αφορούν στην ανασκαφή. Ο σχεδιασμός ενός έργου είναι ένα ολοκληρωμένο πλαίσιο, που ενσωματώνει και δικαιολογεί ένα ερευνητικό πρόγραμμα, ένα πρόγραμμα διατήρησης και ένα πρόγραμμα συνεχούς αλληλεπίδρασης με το κοινό και τους ενδιαφερόμενους³⁹⁴.

4. Η εφαρμογή (Implementation): είναι η ενεργή εκτέλεση όλων των προγραμμάτων που συμφωνήθηκαν στη φάση του σχεδιασμού³⁹⁵. Η έρευνα, η ανασκαφή, η καταγραφή της αρχαιολογικής μαρτυρίας υλοποιούνται σε μία ή διαφορετικές ανασκαφικές φάσεις εργασίας πεδίου από μικρές ή μεγάλες ανασκαφικές ομάδες. Το έργο αυτών των ομάδων είναι η μελέτη του τοπίου, η δειγματοληψία, η ανασκαφή και η καταγραφή του αρχαιολογικού αρχείου. Η στρατηγική για την καταγραφή και τη δειγματοληψία περιγράφεται στον σχεδιασμό του έργου και καθορίζεται από τους ερευνητικούς στόχους και το έδαφος³⁹⁶.

5. Ακολουθεί η μεταανασκαφική ανάλυση (analysis): αφορά σε διαδικασίες μελέτης του αρχαιολογικού υλικού μετά την ολοκλήρωση της ανασκαφής. Από την έλευση της Νέας Αρχαιολογίας στη δεκαετία του 1960, η χρήση επιστημονικών τεχνικών στην αρχαιολογία απέκτησε ιδιαίτερη σημασία. Αυτή η τάση αντικατοπτρίζεται άμεσα στην αυξανόμενη εφαρμογή της επιστημονικής μεθόδου στη μεταανασκαφική ανάλυση³⁹⁷. Το πρώτο βήμα είναι να προσδιοριστεί το τι προσπαθεί να βρει ο αρχαιολόγος και ποιες τεχνικές είναι κατάλληλες για την παροχή απαντήσεων. Οι τεχνικές που επιλέγονται, εξαρτώνται άμεσα από το είδος των δεδομένων. Οι αρχαιολόγοι συχνά αλλάζουν

³⁹⁴ Schiffer *et al* 1978, 1-28.

³⁹⁵ Sharer and Ashmore 2003, 151-156.

³⁹⁶ Carver 2015, 6.

³⁹⁷ Rice 1990, 1-2.

ή προσθέτουν τεχνικές στη διαδικασία ανάλυσης, καθώς οι παρατηρήσεις μπορούν να μεταβάλουν τα αρχικά ερευνητικά ερωτήματα³⁹⁸.

Τόσο η έρευνα όσο και η ανασκαφή συνήθως δημιουργούν μεγάλες ομάδες τεχνέργων και λεπτομερείς χωρικές καταγραφές σχετικές με αντικείμενα, πλαίσια και χαρακτηριστικά.

Τα τρία κύρια μέρη του αναλυτικού προγράμματος είναι η ομαδοποίηση, ο χώρος και η χρονική σειρά. Συνδυαζόμενα τα τρία διακριτά μέρη οδηγούν σε μια αφήγηση για το τι συνέβη στην υπό εξέταση αρχαιολογική θέση. Η ομαδοποίηση περιλαμβάνει όλα τα αντικείμενα και τα υλικά που περιλαμβάνονται στη σχεδιασμένη ανάλυση. Η απόδοση της ομαδοποίησης αυξάνεται σημαντικά με τη διαλογή (κοσκίνισμα) του χώματος, που βοηθά στην σύλληψη μικρών αντικειμένων που δεν είναι ορατά με γυμνό οφθαλμό ή με την τεχνική της επίπλευσης για τη μικροσκοπική οργανική ύλη που προέρχεται από φυτά. Η χρήση επιστημονικών μεθόδων αυξάνει σημαντικά το εύρος της γνώσης που αντλείται από το έδαφος, με τη μικροσκοπική εξέταση υπολείμματων όπως οι σπόροι, η γύρη, τα έντομα και τα υπολείμματα σε αγγεία³⁹⁹.

Η συστηματική μελέτη των υλικών από πέτρα, μέταλλο, κεραμικό, ξύλο, δέρμα και κλωστοϋφαντουργικά προϊόντα είναι καλά αναπτυγμένη. Στα βιολογικά κλάσματα ομάδων (biota) περιλαμβάνονται εδαφολογικά, ζωικά και ανθρώπινα υπολείμματα, σπόροι, γύρη, φυτόλιθοι και άλλα φυτικά κατάλοιπα και έντομα, που αποδίδουν χρονολογικές ακολουθίες μέσω της ραδιοχρονολόγησης. Άλλες ομάδες παρέχουν γραπτές ενδείξεις χρονολόγησης, για παράδειγμα τα νομίσματα με τις ημερομηνίες κοπής και κυκλοφορίας⁴⁰⁰.

Η χρήση του χώρου είναι θεμελιώδης για την κατανόηση της ανθρώπινης δραστηριότητας και καταγράφεται σε ένα ευρύ φάσμα χωρικών διαστάσεων, από το επίπεδο της γεωγραφικής αρχαιολογικής θέσης, στο επίπεδο του οικισμού, του νεκροταφείου, της οικίας, του δωματίου ή της μικρής εστίας. Μεγάλο τμήμα του έργου της αρχαιολογίας πεδίου σχετίζεται με τη μέτρηση, συνήθως σε τρεις διαστάσεις. Η χρήση του χώρου αναλύεται με σχεδίαση της ακριβούς θέσης των αντικειμένων, τη σχεδίαση κατασκευών και οικοδομημάτων. Λεπτομέρειες

³⁹⁸ Balme and Paterson 2006, 176.

³⁹⁹ Beck 2015, 11-17.

⁴⁰⁰ Carver 2011, 7.

σχετικά με τη χρήση του χώρου μέσα σε ένα κτίριο καθορίζονται, επίσης, με χημική και γεωφυσική χαρτογράφηση⁴⁰¹.

Σε μια τέτοια έρευνα περιλαμβάνεται ο προσδιορισμός των δραστηριοτήτων, οι διαδρομές και τα μονοπάτια, οι κοινωνικές ιεραρχίες -από τους χώρους και το μέγεθος της οικίας ή την πολυτέλεια των κτερισμάτων ενός τάφου και την πολυπλοκότητα λάξευσης του βράχου- και η χρονολογική ακολουθία. Στις ανεσκαμμένες τοποθεσίες, η ακολουθία καθορίζεται με τη στρωματογραφία. Οι χρονολογικές σχέσεις τους καταγράφονται επί τόπου και καταρτίζεται σε πλήρες διάγραμμα η στρωματογραφία. Η χρονολογική ακολουθία μπορεί επίσης να συμπληρωθεί από τις απόλυτες ημερομηνίες των τεχνέργων και από τις μεταξύ τους σχέσεις στον χώρο⁴⁰².

Η χρονολογική αλληλουχία και ο χώρος παράγουν το μοντέλο ή την αφήγηση της αρχαιολογικής θέσης, δηλαδή τον καλύτερο δυνατό αρχαιολογικό απολογισμό των γεγονότων που συνέβησαν σε ένα τόπο⁴⁰³. Περαιτέρω έρευνα σχετίζεται με τη σύγκριση μιας θέσης με άλλες αρχαιολογικές θέσεις στο στενό ή το ευρύ χωρικό περιβάλλον και στοχεύει στην κατανόησή τους από ανθρώπινη, ιστορική και περιβαλλοντική άποψη. Σε αυτή τη διαδικασία ζωτικός σύμμαχος είναι η εθνοαρχαιολογία, η οποία παρέχει αναλογίες της αρχαίας πρακτικής και της ανθρώπινης συμπεριφοράς, βάσει παρατηρήσεων σε σύγχρονες κοινότητες⁴⁰⁴.

6. Η δημοσίευση (publication). Βάσει των αναλύσεων τα αποτελέσματα της εργασίας πεδίου συντίθενται και στη συνέχεια δημοσιεύονται προς χρήση εκείνων που τις χρειάζονται με τη μορφή μονογραφιών, εκδόσεων δημόσιας αρχαιολογίας, αναφορών και εκθέσεων. Όλα τα αρχεία που καταγράφονται κατατίθενται σε αρχείο (συχνά ψηφιακό) για τη μακροπρόθεσμη διατήρηση πληροφοριών προς χρήση των μελλοντικών γενεών⁴⁰⁵.

3.9. Μέθοδοι ανεύρεσης αρχαιολογικών δεδομένων

Εκτός των τυχαίων ανακαλύψεων που γίνονται εξαιτίας της διάβρωσης του εδάφους φυσικής ή τεχνητής (οικοδομική δραστηριότητα) και της βαθιάς

⁴⁰¹ Carver 2011, Κεφ. 2

⁴⁰² O'Connell 1995, 205-255.

⁴⁰³ Pluciennik 1999, 653-678.

⁴⁰⁴ Ascher 1961, 317-325.

⁴⁰⁵ Morgan 2013, 279-288.

άρωσης, οι αρχαιολόγοι στηρίζονται συχνά στη γραπτή παράδοση και σε τοπωνύμια⁴⁰⁶ ή σε ορατές θέσεις αρχαίων ερειπίων. Ωστόσο, η ανεύρεση αρχαιολογικών δεδομένων είναι συστηματική διαδικασία έρευνας πεδίου. Για τον εντοπισμό και τον καθορισμό θέσεων και άλλων αρχαίων καταλοίπων σε δεδομένη γεωγραφική περιοχή χρησιμοποιούνται τρεις αλληλοσυμπληρούμενες ερευνητικές μέθοδοι, η έρευνα από τον αέρα και το διάστημα, η έρευνα επιφάνειας και η έρευνα υπεδάφους⁴⁰⁷.

Η έρευνα από τον αέρα και το διάστημα περιλαμβάνει κυρίως την αεροφωτογράφιση και την τηλεπισκόπηση από εναέρια ραντάρ και δορυφόρους. Η αρχαιολογική αεροφωτογράφιση έχει συντελέσει σε ορισμένες από τις θεαματικότερες εξελίξεις του περασμένου μισού αιώνα, καθώς γνωρίζουμε τον αριθμό και την ποικιλία των αρχαιολογικών χώρων σε πολλές χώρες. Εκατοντάδες τοποθεσίες έχουν ανασκαφεί επειδή οι δυνατότητές τους ανακαλύφθηκαν από αεροφωτογραφίες και σε πολλές άλλες περιπτώσεις η αεροφωτογράφιση βοήθησε να αποφασιστεί πού, σε μια μεγάλη περιοχή, θα ήταν παραγωγικότερη η ανασκαφή. Συνεπώς, η αεροφωτογράφιση είναι ιδιαίτερα χρήσιμη για τον εντοπισμό και τον καθορισμό αρχαιολογικών θέσεων και συχνά προετοιμάζει το έδαφος για έρευνα επιφανείας⁴⁰⁸. Στην έρευνα από αέρος την τελευταία δεκαετία σημαντικό ρόλο στη διαδικασία της φωτογράφισης παίζουν τα UAV (Unmanned Aerial Vehicles ή drone) με φωτογράφιση και βιντεοσκόπηση υψηλής ευκρίνειας. Προσφέρουν σημαντικές δυνατότητες αεροφωτογράφισης χαμηλής ανύψωσης προς υποστήριξη της αρχαιολογικής έρευνας⁴⁰⁹.

Αεροφωτογράφιση

Ένα από τα μεγάλα πλεονεκτήματα της αρχαιολογικής έρευνας ήταν η έλευση της εναέριας επισκόπησης κατά τον πρώτο παγκόσμιο πόλεμο. Οι αρχαιολόγοι έσπευσαν να εφαρμόσουν αυτή τη νέα τεχνολογία σε περιοχές που κυμαίνονται από την Αγγλία⁴¹⁰ έως τη Συρία⁴¹¹. Όχι μόνο ήταν ευκολότερο να να

⁴⁰⁶ Βλ. Gelling 2011, 988-1002, επίσης Seidl 2008, 33-56 και Humphreys 1993, 43-53 για τη σημασία των τοπωνυμίων σε διαφορετικές χώρες και ηπείρους.

⁴⁰⁷ Κουκουζέλη κ.ά. 2003, 126

⁴⁰⁸ Coles 1972, 21-45.

⁴⁰⁹ Scott 2016, 1-43.

⁴¹⁰ Crawford 1929.

ανιχνευθούν αρχιτεκτονικά κατάλοιπα, που αποκαλύπτονταν στο έδαφος, όπως θαμμένοι τάφροι και ίχνη θεμελίωσης, αλλά ανιχνεύονταν και θαμμένες κατασκευές με τα «σημάδια βλάστησης»⁴¹².

Στη σύγχρονη έρευνα μέσω της αεροφωτογράφισης παρέχεται πανοραμική άποψη μιας ολόκληρης γεωγραφικής αρχαιολογικής περιοχής. Επίσης, είναι δυνατός ο καθορισμός των ορίων της εξεταζόμενης γεωγραφικής περιοχής και του αριθμού των αρχαιολογικών θέσεων που περιλαμβάνει. Με αυτόν τον τρόπο συγκεντρώνονται οι πρώτες πληροφορίες για το μέγεθος, τη μορφή και το σχέδιο, ενίοτε και τη λειτουργία των θέσεων στην υπό εξέταση περιοχή, καθώς εντοπίζονται αρχαίες οδοί μονοπάτια, γέφυρες, αρδευτικά έργα, αναλημματικοί τοίχοι, ακόμη και σημάδια άροσης.

Η αεροφωτογράφιση διεξάγεται υπό γωνία, από χαμηλό ύψος και με το πλάγιο φως του ήλιου. Οι αρχαιολογικές θέσεις αποκαλύπτονται κατά την αεροφωτογράφιση ως σκιές, ως σημάδια εδάφους και ως σημάδια βλάστησης. Ως σκιές εμφανίζονται αρχαιολογικές θέσεις μεγάλου μεγέθους, σχεδόν εξαφανισμένες από την επιφάνεια του εδάφους, εξαιτίας της βαθιάς άροσης ή διάβρωσης, και είναι ορατές ως χαμηλές ανωμαλίες του εδάφους⁴¹³.

Ως σημάδια εδάφους και ως σημάδια βλάστησης αποκαλύπτονται θέσεις σε επίπεδες περιοχές έντονης γεωργικής εκμετάλλευσης, οι οποίες είτε είναι θαμμένες κοντά στην επιφάνεια του εδάφους είτε έχουν ισοπεδωθεί. Τέτοιου είδους θέσεις δεν είναι ορατές από το επίπεδο του εδάφους. Ωστόσο, τα σημάδια εδάφους όσο και τα σημάδια βλάστησης είναι αρκετά εφήμερα και η ορατότητά τους εξαρτάται από μια σειρά παράγοντες, οι οποίοι επηρεάζουν το έδαφος ή τη βλάστηση αντίστοιχα. Ειδικότερα τα σημάδια εδάφους αποκαλύπτονται είτε με την απόξεση της άνω επιφάνειας των αρχαίων τοίχων από βαθιά άροση, είτε σε γυμνό και επίπεδο έδαφος χάρη στην συγκράτηση υγρασίας από θαμμένες κατασκευές που περιέχουν λεπτότερα εδάφη, όπως ο άργιλος, ή χάρη στη διαφορετική τήξη του πάγου ή του χιονιού λόγω της θερμικής ακτινοβολίας που εκπέμπεται από τα θαμμένα αντικείμενα στο έδαφος⁴¹⁴. Τα σημάδια βλάστησης εμφανίζονται όταν η παρουσία αρχιτεκτονικών λειψάνων ή κατασκευών κάτω

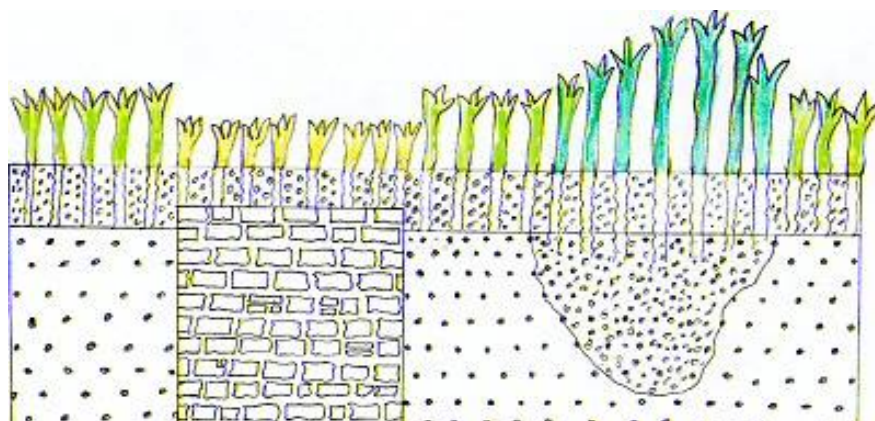
⁴¹¹ Poidebard, 1934.

⁴¹² Bewley and Rackowski 2001, 1-8.

⁴¹³ Renfrew and Bahn 2001, 80.

⁴¹⁴ Renfrew and Bahn 2001, 81.

από την επιφάνεια αγρών προξενεί διαφορετικούς ρυθμούς ανάπτυξης και ωρίμανσης των καλλιεργειών. Οι αποκλίσεις που δημιουργούνται στη θερμοκρασία του εδάφους εξαιτίας της ακτινοβολίας αντικειμένων στο έδαφος μετρούνται από αέρος με τη χρήση ραδιομέτρων ως τμήμα μιας άλλης μεθόδου εντοπισμού αρχαιολογικών θέσεων και άλλων αρχαίων καταλοίπων, γνωστή ως θερμική αναγνωριστική έρευνα⁴¹⁵. Η παρατήρηση των σημαδιών βλάστησης είναι η πιο ευαίσθητη και αποτελεσματική μέθοδος για την εντόπιση θαμμένων κατασκευών μέσω της αεροφωτογράφισης⁴¹⁶. Ωστόσο, η εμφάνιση τέτοιων σημείων δε διαρκεί πολύ χρονικά και εξαρτάται από πολλούς παράγοντες, όπως είναι το είδος της καλλιέργειας και του εδάφους, οι διαφορετικές γεωργικές μέθοδοι, καθώς και οι κλιματικές συνθήκες κατά την εποχή ανάπτυξης της σχετικής χλωρίδας.



Εικ. 3.2. Διάγραμμα υπανάπτυξης σοδειάς πάνω από θαμμένο τοίχο και υπερανάπτυξης πάνω από τάφρο, εξαιτίας της υψηλής συγκράτησης υγρασίας⁴¹⁷.

Λόγω της έλλειψης σταθερότητας των περισσότερων από τα σημάδια εδάφους και βλάστησης, οι αρχαιολογικές θέσεις που παρουσιάζουν κάποια πολυπλοκότητα φωτογραφίζονται κάτω από διαφορετικές συνθήκες για μεγάλο χρονικό διάστημα. Με αυτόν τον τρόπο διαμορφώνεται αρχείο των χαρακτηριστικών τους και σχεδιάζονται οι κατάλληλοι χάρτες των θέσεων αυτών για αρχαιολογικούς σκοπούς⁴¹⁸.

⁴¹⁵ Βλ. σχετικές αναφορές για την εφαρμογή της μεθόδου στο Ben-Dor *et al* 1999, 117-127.

⁴¹⁶ Renfrew and Bahn 2001, 79, 81.

⁴¹⁷ Wikipedia Commons, διαθέσιμο online.

⁴¹⁸ Renfrew and Bahn 2001, 82.

Τηλεπισκόπηση (εναέρια ραντάρ και δορυφόροι)

Η τηλεπισκόπηση από εναέρια ραντάρ και δορυφόρους είναι, προς το παρόν τουλάχιστον, χρήσιμη μόνο για τον εντοπισμό θέσεων και άλλων αρχαίων καταλοίπων⁴¹⁹. Οι μέθοδοι αυτές άρχισαν να χρησιμοποιούνται από την αρχαιολογία σχετικά πρόσφατα με αρκετά ενθαρρυντικά αποτελέσματα για το μέλλον. Εδικότερα, εναέρια ραντάρ τοποθετημένα σε αεροσκάφη τύπου SLAR (Side-looking airborne radar - SLAR)⁴²⁰ ή ραντάρ τύπου SIR ή SAR (Synthetic Aperture Radar)⁴²¹ σε διαστημόπλοια και δορυφόρους είναι δυνατόν να εντοπίσουν την παρουσία σχηματισμών και αντικειμένων στο έδαφος της Γης και να προσδιορίσουν τη θέση τους, εκπέμποντας σύντομους παλμούς μικροκυμάτων και καταγράφοντας τον χρόνο επιστροφής των σημάτων ή των «ανακλάσεων» που δημιουργούνται. Ανωμαλίες του εδάφους, κτίρια και αντικείμενα με υψηλή περιεκτικότητα σε υγρασία προξενούν σήματα μεγάλης έντασης. Οι γραμμές των παλμών συλλαμβάνονται από τα όργανα με τη μορφή ανοιχτόχρωμων ή σκουρόχρωμων αντιθέσεων, παράγοντας σχετικές εικόνες. Τα εναέρια ραντάρ έχουν το επιπρόσθετο πλεονέκτημα ότι μπορούν να διαπεράσουν την ατμόσφαιρα κάτω από σχεδόν οποιοσδήποτε καιρικές συνθήκες, την πυκνή βλάστηση, ακόμη και τα εξαιρετικά ξηρά εδάφη. Μπορούν να αποκαλύψουν διάφορες μεταβολές στο αρχαίο τοπίο, όπως αλλαγές στην τοπογραφία ή διαταραχές στο υπέδαφος μεγάλων θέσεων και είναι χρήσιμα για τον ενοπισμό αρχαίων ναυαγίων⁴²².

Οι δορυφόροι, καταγράφοντας την ένταση της υπέρυθρης ακτινοβολίας που εκπέμπεται σε διαφορετικά μήκη κύματος από τα διάφορα συστατικά της γήινης επιφάνειας, μετατρέπουν με ηλεκτρονικό τρόπο τα δεδομένα που έχουν καταγράψει σε φωτογραφικές εικόνες. Η δορυφορική τηλεπισκόπηση, αν και χρήσιμη για την Αρχαιολογία, παρουσιάζει ορισμένους περιορισμούς ως προς την ακρίβεια της κλίμακας για αρχαιολογικούς σκοπούς και η ευκρίνεια των εικόνων από δορυφόρους τύπου LANDSAT και SPOT, δεν είναι επαρκής επί του παρόντος. Ωστόσο, οι δορυφορικές εικόνες είναι πλέον συμπληρωματική πηγή πληροφοριών για την αεροφωτογράφιση. Για τις περιοχές στις οποίες υπάρχει

⁴¹⁹ Κουκουζέλη κ.ά. 2003, 128.

⁴²⁰ Chan and Koo 2008, 27-60.

⁴²¹ Chan and Lim 2008, 269-290.

⁴²² Guzinski *et al* 2016, 43-263.

περιορισμένη εναέρια φωτογραφική κάλυψη, τα εκτεταμένα αρχεία δορυφορικών εικόνων, που μπορούν να χρονολογούνται από τις αρχές της δεκαετίας του 1960, αποτελούν σημαντικό πόρο για την ανίχνευση αρχαιολογικών χαρακτηριστικών τα οποία έχουν από μακρού καταστραφεί, εξαιτίας της γεωργίας και της αστικής επέκτασης⁴²³.

Το LiDAR (Light Detection and Ranging-ανίχνευση και εντοπισμός φωτός) είναι τεχνική τηλεπισκόπησης, που χρησιμοποιείται για έρευνα υψηλής ευκρινείας τοπίων. Η τεχνολογία βασίζεται στη χρήση ενός σαρωτή λέιζερ, τοποθετημένου σε αεροσκάφος. Οι σαρωτές λέιζερ εκπέμπουν παλμούς φως λέιζερ με ρυθμό πολλών εκατοντάδων παλμών ανά δευτερόλεπτο και μετρούν το χρόνο που χρειάζεται η ανάκλαση αυτού του παλμού για να επιστρέψει στο όργανο. Χρησιμοποιείται GPS για την καταγραφή των συντεταγμένων κάθε μέτρησης. Με αυτή τη μέθοδο, ισχυροί υπολογιστές είναι σε θέση να υπολογίζουν εκατομμύρια μετρήσεις σε ένα σύνολο ή νέφος δεδομένων (cloud point). Με τη σειρά του, το σύννεφο δεδομένων χρησιμοποιείται για να παράγει λεπτομερείς απεικονίσεις της επιφάνειας του εδάφους, που ονομάζονται «ψηφιακά μοντέλα εδάφους» (DTM). Το LiDAR πραγματοποιεί πολύ ακριβείς μετρήσεις της επιφάνειας του εδάφους, καταγράφει ό,τι μπορείτε να δει από τον αέρα, παράγει χάρτες 3D, δηλαδή στις διαστάσεις X, Y και Z. Ωστόσο, δε διεισδύει στη γη ή στο νερό, δηλαδή δε βλέπει θαμμένα χαρακτηριστικά⁴²⁴.

Έρευνα επιφανείας

Η έρευνα επιφανείας (surface survey ή απλώς survey) χρησιμοποιείται από μακρού χρόνου στην αρχαιολογία και σήμερα είναι το κύριο μέσο εντοπισμού και καθορισμού αρχαιολογικών θέσεων και άλλων καταλοίπων σε μια γεωγραφική αρχαιολογική θέση. Ένας από τους πρώτους συγγραφείς που περιέγραψαν τις μεθόδους της έρευνας επιφανείας και ιδιαίτερα μεθόδους για την έρευνα σκορπισμένων τεχνέργων ήταν ο W. G. Clarke⁴²⁵. Η περιφερειακή αρχαιολογική έρευνα είναι ένα όργανο του οποίου η φύση και το καθεστώς

⁴²³ Fowler 2010, 99-110.

⁴²⁴ AOC 2015, 4-5.

⁴²⁵ Clarke 1922, 24-32.

έχουν αλλάξει, ιδιαίτερα από το 1970. Τέσσερεις κύριοι λόγοι προτάθηκαν για την πραγματοποίηση τέτοιων ερευνών⁴²⁶:

- α) προοπτική ανακάλυψης αρχαιολογικών θέσεων,
- β) εκτίμηση (π.χ. της πυκνότητας τεχνέργων),
- γ) χαρακτηρισμός ή διευκρίνιση της κατανομής θέσεων σε μία περιοχή⁴²⁷.
- δ) δοκιμή υπόθεσης⁴²⁸.

Τα στοιχεία της έρευνας επιφανείας ενίοτε υποφέρουν από ατομικές προκαταλήψεις και ανομοιότητες στην κάλυψη της έρευνας, από μεθοδολογική ασυνέπεια και συχνά από έλλειψη επαρκούς δημοσίευσης⁴²⁹. Σημαντικό ρόλο, επίσης, παίζει η καταστροφή των αρχαιολογικών καταλοίπων εξαιτίας της διάβρωσης. Οι συζητήσεις στη βιβλιογραφία σχετικά με την αξία της έρευνας ή σχετικά με τα πλεονεκτήματα των ερευνών σε μια συγκεκριμένη αρχαιολογική θέση ή σε μια ευρύτερη γεωγραφική περιοχή οφείλονται συχνά στην αδυναμία αναγνώρισης από μέρους των αρχαιολόγων ότι έχουν διαφορετικούς στόχους ή αντιλαμβάνονται τα αρχαιολογικά ευρήματα με εντελώς διαφορετικά μοντέλα σκέψης. Μία από τις μακρόχρονες συζητήσεις στην αρχαιολογία ήταν η χρησιμότητα της κατανομής των τεχνέργων στην επιφάνεια σε σχέση με τη θαμμένη κατανομή των τεχνέργων⁴³⁰.

Είναι σύνηθες για τους αρχαιολόγους να υποθέτουν ότι τα επιφανειακά κατάλοιπα είναι απλά διαστρεβλωμένα ίχνη αρχαίων πολιτιστικών καταλοίπων, ευνοώντας έτσι την ανασκαφή έναντι της έρευνας επιφανείας. Άλλοι αρχαιολόγοι, ωστόσο, επισημαίνουν ότι ουσιαστικά όλα τα αρχαιολογικά κατάλοιπα βρισκόταν κάποτε σε επιφάνειες και μόνο ορισμένα από αυτά έχουν θαφτεί εξαιτίας διαφόρων γεωμορφολογικών ή τοπικών διαδικασιών. Επιπλέον, αναγνωρίζεται πλέον ότι δεν είναι απαραίτητη η συσχέτιση ανάμεσα στα επιφανειακά κατάλοιπα και τα κατάλοιπα του υπεδάφους, για να θεωρηθούν τα αρχαιολογικά στοιχεία έγκυρα⁴³¹. Δεν πρέπει πλέον να αναμένεται ότι η χωρική

⁴²⁶ Orton 2002, 67.

⁴²⁷ Wobst 1983, 37.

⁴²⁸ Wobst 1983, 40.

⁴²⁹ Banning 2002, 10.

⁴³⁰ Lewarch and O'Brien 1981, 297-342.

⁴³¹ Synenki 1984, 14-38.

κατανομή ενός υλικού πολιτισμού στην επιφάνεια του θα «αντικατοπτρίζει» αυτό που βρίσκεται σε ένα στρώμα μιας στρωματοποιημένης θέσης⁴³². Μόνο η πειραματική εργασία και η μελέτη των διεργασιών σχηματισμού των τόπων βοηθούν στην κατανόηση της σχέσης μεταξύ επιφανειακών και υπόγειων καταλοίπων⁴³³.

Παρά τις αντιρρήσεις που έχουν κατά καιρούς εκφραστεί, η σύγχρονη συστηματική έρευνα επιφάνειας μπορεί να αποκαλύψει με αξιόπιστο τρόπο διαβαθμίσεις στην ένταση της ανθρώπινης δραστηριότητας στο σύνολο της επιφανείας μιας γεωγραφικής περιοχής, κάτι που δεν είναι εφικτό, ακόμη και με την πλέον εκτεταμένη ανασκαφή. Πιθανά μειονεκτήματα της εκτεταμένης έρευνας επιφάνειας είναι ότι ενίοτε μένουν ανεξερεύνητες οι μικρότερες και θαμμένες θέσεις, οι οποίες καλύπτουν το μεγαλύτερο μέρος της επιφάνειας μιας γεωγραφικής περιοχής. Κατ' αυτόν τον τρόπο παρέχεται ελλιπής εικόνα του τρόπου κατοίκησης του αρχαίου τοπίου⁴³⁴.

Δεδομένου ότι η έρευνα επιφάνειας είναι η άμεση εξέταση της επιφάνειας του εδάφους μιας προκαθορισμένης περιοχής με στόχο τον εντοπισμό ορατών αρχαίων πολιτισμικών καταλοίπων, υφίσταται διάκριση ανάμεσα στην έρευνα επιφάνειας μιας ευρύτερης γεωγραφικής αρχαιολογικής θέσης και την έρευνα επιφάνειας μιας αρχαιολογικής θέσης⁴³⁵.

Οι μέθοδοι που ακολουθούνται στην έρευνα επιφάνειας μιας ευρύτερης γεωγραφικής αρχαιολογικής θέσης, σχετίζονται με αποφάσεις που αφορούν στον σχεδιασμό και την εκτέλεσή της. Η βασικότερη απόφαση που λαμβάνει οποιοσδήποτε αρχαιολόγος για την πραγματοποίηση μιας έρευνας είναι ο τόπος διεξαγωγής της. Στην ιδανική περίπτωση, μια τέτοια επιλογή βασίζεται σε θεωρητικά, μεθοδολογικά και εμπειρικά ζητήματα, στα οποία στηρίζεται το κεντρικό ερευνητικό θέμα. Στην πραγματικότητα, η τοποθεσία ενός συγκεκριμένου ερευνητικού έργου καθορίζεται από μια ποικιλία πολιτικών και οικονομικών προβληματισμών και την ανάγκη για αξιολόγηση πολιτιστικών

⁴³² Dunnell 1988, 29-38.

⁴³³ Ammerman 1985, 33-40.

⁴³⁴ Κουκουζέλη κ.ά. 2003, 130.

⁴³⁵ Κουκουζέλη κ.ά. 2003, 131.

πόρων των περιοχών στα πλαίσια των οικονομικών, πολιτιστικών και νομοθετικών επιλογών μιας χώρας⁴³⁶.

Το πρωταρχικό ζήτημα είναι τα όρια διεξαγωγής της έρευνας. Βασικά, μπορούν να εντοπιστούν τρία είδη ορίων: τα αυθαίρετα, τα πολιτιστικά και τα γεωγραφικά ή φυσικά. Τα φυσικά-γεωγραφικά όρια είναι εκείνα που ορίζονται από κάποιο τοπογραφικό μοτίβο. Τα πολιτισμικά όρια είναι εκείνα που ορίζονται βάσει τεχνητών κριτηρίων. Τα αυθαίρετα όρια είναι εκείνα που ορίζονται πάνω από κριτήρια που δε σχετίζονται με φυσικά ή πολιτισμικά πρότυπα. Μία από τις κρισιμότερες αποφάσεις για την ανάπτυξη των στοιχείων της έρευνας σχετίζεται με το ερώτημα του τι συνιστά και τι δεν συνιστά αρχαιολογική θέση. Με την ανακάλυψη τεχνέργων ο ερευνητής πρέπει να πάρει απόφαση σχετικά με την κατάσταση της αρχαιολογικής θέσης. Ο Chenhall θεωρεί ότι η επάρκεια μιας τέτοιας απόφασης είναι αντανάκλαση της αρχαιολογικής αντίληψης. Θεωρεί ότι σε κάθε στάδιο παίρνονται αποφάσεις για τη διατήρηση και για την απόρριψη δεδομένων⁴³⁷. Επομένως, η επάρκεια μιας έρευνας εξαρτάται εν μέρει από την αντίληψη του ερευνητή σχετικά με τη σημασία ορισμένων κατηγοριών αρχαιολογικών δεδομένων⁴³⁸.

Όλοι προβληματισμοί που εγείρονται, σχετίζονται με τον ορισμό των αρχαιολογικών θέσεων με τις οποίες εργάζονται οι ερευνητές. Ωστόσο οι ορισμοί ποικίλλουν σημαντικά. Για παράδειγμα, η Νοτιοαφρικανική Ανθρωπολογική Ομάδα Έρευνας (SARG) ορίζει ως αρχαιολογική θέση «κάθε πολιτιστική τοποθεσία, με τέχνηρα ή κατασκευές» με πυκνότητα τεχνέργων τουλάχιστον 5/m²⁴³⁹. Με τη σειρά του το Κρατικό Μουσείο της Αριζόνα χρησιμοποιεί, επίσης, το κριτήριο πυκνότητας τεχνέργων για τον ορισμό της αρχαιολογικής θέσης, αλλά επιπλέον διευκρινίζει ότι ο αρχαιολογικός τόπος χώρος χρειάζεται να έχει περισσότερους από έναν καθορισμένους τόπους προηγούμενης ανθρώπινης δραστηριότητας και πρέπει να παρουσιάζει καθορισμένα όρια στον χρόνο και στον χώρο⁴⁴⁰. Είναι εμφανές ότι τέτοιοι ορισμοί παρουσιάζουν αδυναμίες, καθώς σε πολλές έρευνες αρχαιολογικές θέσεις με πυκνότητα 5 τεχνέργων ανά τετραγωνικό μέτρο είναι μάλλον η εξαίρεση παρά ο κανόνας. Εκτός από τα

⁴³⁶ Plog *et al* 1978, 383-421.

⁴³⁷ Chenhall 1975, 8.

⁴³⁸ Plog *et al* 1978, 386.

⁴³⁹ Plog and Hill 1971, 8.

⁴⁴⁰ Fuller *et al* 1976, 68.

προβλήματα με τους ορισμούς που βασίζονται στην πυκνότητα των θέσεων, για ορισμένους τύπους θέσεων είναι εξαιρετικά δύσκολο να προσδιοριστούν καθορισμένα όρια και πολλαπλοί προσδιορισμένοι τόποι δραστηριότητας⁴⁴¹.

Σήμερα, ωστόσο, η εκτεταμένη γεωγραφική έρευνα παραμένει σημαντική μέθοδος συλλογής δεδομένων για αρχαιολογικές έρευνες, ειδικά για μελέτες διαχείρισης πολιτιστικών πόρων (CRM). Το επιφανειακό αρχαιολογικό αρχείο μπορεί να θεωρηθεί «συνεχής κατανομή τεχνέργων στην επιφάνεια της γης με ιδιαίτερα μεταβλητά χαρακτηριστικά πυκνότητας»⁴⁴². Ανάλογα με το πρόβλημα που διερευνάται, η γενική στρατηγική της αρχαιολογικής έρευνας είναι να ενισχύσει ή να μεγιστοποιήσει τις «πιθανότητες ανακάλυψης αρχαιολογικών καταλοίπων»⁴⁴³.

Σε σύγκριση με τα ανασκαφικά δεδομένα, τα δεδομένα της έρευνας επιφανείας είναι εξίσου σημαντικά, αν και λιγότερο δαπανηρά. Στην πραγματικότητα, διάφορα προβλήματα στην αρχαιολογία επιλύονται συμβατικά μόνο μέσω της έρευνας, όπως υποδεικνύουν αρκετές συνεισφορές στο *Surface Archaeology* (Sullivan 1998)⁴⁴⁴.

Βαθμός έντασης της έρευνας επιφανείας είναι ο όρος που χρησιμοποιούν οι αρχαιολόγοι για να αναφερθούν στην πυκνότητα των προσπαθειών κατά την έρευνα ενός δεδομένου χώρου. Μπορεί να μετρηθεί με τον αριθμό των ατόμων και τις ώρες έρευνας ή τον αριθμό των δοκιμαστικών τομών ανά στρέμμα για παράδειγμα. Όσο περισσότερες είναι οι παρατηρήσεις ανά περιοχή, ή όσο μικρότερη είναι η απόσταση μεταξύ διατομών, τόσο μεγαλύτερη είναι η πληρότητα ή η ένταση της έρευνας. Η ένταση συσχετίζεται θετικά τόσο με το κόστος όσο και με τον χρόνο έρευνας, για μια δεδομένη ομάδα ερευνητών. Η επιχειρησιακή έρευνα συνήθως περιγράφει την ένταση ως πυκνότητα της προσπάθειας αναζήτησης⁴⁴⁵. Ο χρόνος αναζήτησης έχει προφανώς σχέση με την πιθανότητα ανίχνευσης ενός στόχου και ένας τρόπος για να μετρηθεί η ένταση της προσπάθειας είναι με το συνολικό ποσό του χρόνου έρευνας που αφιερώνεται ανά μονάδα επιφανείας, για παράδειγμα με εργατώρες ανά

⁴⁴¹ Thomas 1975, 81.

⁴⁴² Dunnell and Dancey 1983, 272.

⁴⁴³ Schiffer *et al* 1978, 2.

⁴⁴⁴ Sullivan 1998, 28-38

⁴⁴⁵ Koopman 1980, 140-142.

στρέμμα. Αυτό που δεν είναι τόσο προφανές, ωστόσο, είναι ότι η σχέση μεταξύ του χρόνου έρευνας και της πιθανότητας ανίχνευσης δεν είναι γραμμική, αλλά εκθετική⁴⁴⁶.

Η **εντατική έρευνα επιφάνειας** στοχεύει στην κάλυψη μιας πολύ μικρότερης περιοχής αλλά λεπτομερέστερα, συγκριτικά με την εκτεταμένη έρευνα επιφάνειας. Είναι συστηματική και διεξάγεται από ομάδες αρχαιολόγων και ειδικών επιστημόνων. Ειδικότερα, ένας αριθμός ατόμων σε τακτές αποστάσεις (5-15 μέτρων) μεταξύ τους περπατούν στη σειρά με τον ίδιο ρυθμό και σε ευθείες γραμμές, ερευνώντας την επιφάνεια του εδάφους για οποιαδήποτε αρχαιολογικά κατάλοιπα (συνήθως, όμως, τέχνηρα και ιδίως κεραμικά όστρακα) εμπίπτουν στο οπτικό τους πεδίο.

Υπό κανονικές συνθήκες η έρευνα επιφάνειας αναμένεται να καλύψει πλήρως το εύρος της ερευνώμενης περιοχής. Ωστόσο, όντας χρονοβόρα και δαπανηρή διαδικασία, η εντατική έρευνα επιφάνειας ενίοτε αντικαθίσταται στην επιλεγμένη περιοχή με δειγματοληπτικές μεθόδους. Η δειγματοληψία μπορεί να είναι στατιστική και μη στατιστική. Στη στατιστική δειγματοληψία χρησιμοποιούνται στατιστικές τεχνικές της θεωρίας των πιθανοτήτων, ίδιες με εκείνες που χρησιμοποιούνται και σε δημοσκοπήσεις. Στη μη στατιστική δειγματοληψία χρησιμοποιούνται γεωγραφικές και περιβαλλοντικές μονάδες, εξαρτώμενες ουσιαστικά από προσωπικές κρίσεις για το τι καθιστά μια περιοχή κατάλληλη για κατοίκηση. Η ιδέα της χρήσης μεθόδων δειγματοληψίας βάσει της θεωρίας των πιθανοτήτων στην αρχαιολογική έρευνα συνήθως αποδίδεται στον Binford⁴⁴⁷. Ο Binford υπογράμμισε την ιδέα της περιοχής (σύνολο θέσεων) ως φυσική μονάδα της αρχαιολογικής έρευνας και, αναγνωρίζοντας την αδυναμία πλήρους κάλυψης σε αυτή την κλίμακα, υποστήριξε τις τεχνικές δειγματοληψίας ως τρόπο επίτευξης αντιπροσωπευτικών και αξιόπιστων δεδομένων με κάλυψη μικρότερη του τοπογραφικού συνόλου⁴⁴⁸.

Οι δειγματοληπτικές μονάδες σε μια εντατική έρευνα επιφάνειας είναι τετράγωνα ή διαδρομές και είτε αποτελούν τμήμα ενός καννάβου που καλύπτει την ερευνώμενη περιοχή ή ορίζονται από υφιστάμενους φυσικούς ή τεχνητούς

⁴⁴⁶ Koopman 1980, 55, 71-74, 329.

⁴⁴⁷ Richardson and Gajewski 2002, 1-9.

⁴⁴⁸ Binford 1964, 425-441.

διαχωρισμούς του τοπίου. Οποσδήποτε τα τετράγωνα παρέχουν ακριβέστερες πληροφορίες σχετικές με την ακριβή προέλευση των ευρημάτων, αλλά οι διαδρομές προσφέρονται για επιφανειακές έρευνες μεγάλης κλίμακας, καθώς επιτρέπουν την κάλυψη μεγάλων αποστάσεων με γοργούς ρυθμούς και ενδεχομένως τον εντοπισμό περισσότερων αρχαιολογικών θέσεων.

Οι χρησιμοποιούμενες στατιστικές μέθοδοι δειγματοληψίας εξαρτώνται από τον αριθμό και τη θέση των δειγματοληπτικών μονάδων στην υπό έρευνα γεωγραφική περιοχή και καθορίζονται βάσει τεσσάρων μεθόδων⁴⁴⁹

1. Η μέθοδος της απλής τυχαίας δειγματοληψίας είναι η βάση όλων των δειγματοληπτικών τεχνικών. Κάθε πιθανό δείγμα από μέγεθος n μονάδων έχει ίσες πιθανότητες επιλογής⁴⁵⁰. Τυχαίο δείγμα είναι αυτό που επιλέγεται τόσο προσεκτικά, ώστε να γνωρίζουμε την πιθανότητα επιλογής κάποιας συγκεκριμένης μονάδας (για αυτό ονομάζεται ενίοτε δειγματοληψία πιθανότητας). Είναι, επίσης, σημαντικό σε συμβατικές δειγματοληψίες οι επιλογές να είναι ανεξάρτητες, δηλαδή η επιλογή μιας συγκεκριμένης μονάδας να μην επηρεάζει τις πιθανότητες επιλογής των άλλων μονάδων. Σημαντική πτυχή της προσαρμοστικής δειγματοληψίας είναι ότι καταργεί αυτή την απαίτηση, υπό ελεγχόμενες συνθήκες⁴⁵¹.

2. Η συστηματική τυχαία δειγματοληψία απαιτεί από τον χρήστη να παραγγείλει τις μονάδες του πληθυσμού με κάποιο τρόπο, να επιλέξει τυχαία μία μονάδα από τις πρώτες που έχει σχεδιάσει, για παράδειγμα την 6η, στη συνέχεια, να επιλέξει τις επόμενες μονάδες λαμβάνοντας κάθε 6η μονάδα στη σειρά. Ο Orton επισημαίνει ότι ένα μειονέκτημα της συστηματικής δειγματοληψίας είναι ότι το διάστημα δειγματοληψίας είναι πιθανό να σχετίζεται συμπτωματικά με κάποια κανονικότητα στον χώρο⁴⁵². Για παράδειγμα, κατά τη δειγματοληψία τετραγώνων πλέγματος σε χάρτη, η συστηματική δειγματοληψία μπορεί να δημιουργήσει διαγώνιες γραμμές τετραγώνων δειγματοληψίας, οι οποίες με τη σειρά τους μπορεί να σχετίζονται με φυσικά τοπογραφικά ή γεωλογικά χαρακτηριστικά. Υπάρχουν, όμως, περιπτώσεις στις οποίες η συστηματική αποδίδει καλύτερα από την απλή τυχαία δειγματοληψία. Οι αρχαιολόγοι συχνά

⁴⁴⁹ Renfrew and Bahn 2001, 77.

⁴⁵⁰ Richardson and Gajewski 2002, 1-9 (3).

⁴⁵¹ Orton 2002, 21.

⁴⁵² Orton 2002, 22.

προτιμούν συστηματικά δείγματα, εν μέρει επειδή φαίνεται να δίνουν «καλύτερη κάλυψη» ή «ομοιόμορφη εξάπλωση» του χώρου και εν μέρει επειδή είναι ευκολότερα και ταχύτερα να επιλεγούν⁴⁵³.

3. Με τη μέθοδο της απλής στρωματοποιημένης δειγματοληψίας απλά σχηματίζονται υποομάδες (στρώματα) των πληθυσμιακών μονάδων και επιλέγεται ένα απλό τυχαίο δείγμα σε κάθε υποομάδα. Σύμφωνα με τον Orton πρέπει πάντα να λαμβάνεται υπόψη η δυνατότητα διαστρωμάτωσης ενός τόπου. Ο ορισμός των στρωμάτων θα μπορούσε να βασίζεται σε οποιαδήποτε ιδιότητα ή συνδυασμό ιδιοτήτων, όπως η γεωμορφολογία ή η ανύψωση, ή σε οποιαδήποτε πτυχή που πιστεύεται ότι ενδέχεται να επηρεάσει τις παραμέτρους που μελετώνται -για παράδειγμα την πυκνότητα αντικειμένων που βρίσκονται σε μια τοποθεσία⁴⁵⁴. Ταυτόχρονα λαμβάνεται υπόψιν οποιοδήποτε ιστορικό ή αρχαιολογικό αρχείο σχετικό με την περιοχή⁴⁵⁵.

4. Η στρωματοποιημένη και συστηματική δειγματοληψία είναι τεχνική δειγματοληψίας βάσει της θεωρίας των πιθανοτήτων που συνδυάζει στοιχεία απλής τυχαίας δειγματοληψίας, στρωματοποιημένης τυχαίας δειγματοληψίας και συστηματικής δειγματοληψίας -σε μια προσπάθεια μείωσης της μεροληψίας κατά τη δειγματοληψία⁴⁵⁶. Η στρωματοποιημένη και συστηματική δειγματοληψία με μη ευθυγραμμισμένες δειγματοληπτικές μονάδες είναι σύνθετη δειγματοληπτική προσέγγιση, η οποία συνδυάζει τις αρχές όλων των προηγούμενων και είναι επίσης ελαστική και ακριβής. Επίσης, απαιτεί οι μονάδες δειγματοληψίας να τοποθετούνται σε υποομάδες. Ωστόσο, οι υποομάδες συνήθως λαμβάνονται από μονάδες κοντά στην τοποθεσία. Στη συνέχεια λαμβάνεται ένα απλό τυχαίο δείγμα των υποομάδων και κάθε μονάδα εντός της επιλεγμένης υποομάδας αποτελεί μέρος του δείγματος⁴⁵⁷.

Γεωφυσικές και γεωχημικές μέθοδοι

Οι γεωφυσικές και γεωχημικές τεχνικές είναι δυνατόν να συμβάλλουν σημαντικά στη γεωαρχαιολογική έρευνα. Επιτρέπουν την κατά το δυνατόν μη επεμβατική και γρήγορη απεικόνιση των αρχαιολογικών τοπίων. Τα δεδομένα

⁴⁵³ Richardson and Gajewski 2002, 4.

⁴⁵⁴ Richardson and Gajewski 2002, 4.

⁴⁵⁵ Κουκουζέλη κ.ά. 2003, 138.

⁴⁵⁶ Mueller 1974, 3.

⁴⁵⁷ Richardson and Gajewski 2002, 4.

που αντλούνται, απαντούν σε επιστημονικά ερωτήματα, που αφορούν στη διερεύνηση μιας περιοχής και του περιβάλλοντός της. Το μεθοδολογικό φάσμα περιλαμβάνει τη διερεύνηση της στρωματογραφίας ενός τόπου, τη χαρτογράφηση των υπολειμμάτων παρελθούσας ανθρώπινης κατοχής και την ανακατασκευή των περιοχών σε παλαιότερες περιόδους. Κάθε μέθοδος έχει τα δικά της πλεονεκτήματα και περιορισμούς και μέσω του συνδυασμού των διαφόρων τεχνικών, μπορούν να αντιμετωπιστούν διάφορες πτυχές μιας αρχαιολογικής θέσης.

Ως **έρευνα υπεδάφους** εννοείται σειρά μεθόδων που ερευνούν το υπέδαφος απευθείας από την επιφάνεια του εδάφους και χρησιμοποιούνται για τον εντοπισμό και τον καθορισμό αρχαιολογικών θέσεων και άλλων χαρακτηριστικών του τοπίου σε μια γεωγραφική περιοχή. Ουσιαστικά συμπληρώνουν και επεκτείνουν τα δεδομένα που αποκτώνται από την αεροφωτογράφιση και την έρευνα επιφάνειας⁴⁵⁸.

Τα δεδομένα της έρευνας υπεδάφους μπορούν να χρησιμοποιηθούν πριν από την ανασκαφή, ενίοτε και αντί της ανασκαφής, καθώς το μεγαλύτερο πλεονέκτημά τους είναι ότι οι μέθοδοι έρευνας δεν είναι καταστρεπτικές –οι περισσότερες τουλάχιστον- για το αρχαιολογικό αρχείο κατά την έρευνα. Η γεωφυσική έρευνα υπεδάφους έχει χρησιμοποιηθεί στο παρελθόν με αρκετή επιτυχία και είναι αποτελεσματική, όταν εφαρμόζεται σωστά και συμπληρωματικά με άλλου είδους έρευνες. Είναι ιδιαίτερα χρήσιμη, σε ολοκληρωμένα ερευνητικά σχέδια, στα οποία οι ερμηνείες είναι δυνατόν να δοκιμαστούν και να εμπλουτιστούν⁴⁵⁹. Οι γεωφυσικές μέθοδοι μετρούν τις διάφορες φυσικές ιδιότητες των υπόγειων εδαφών και των πετρωμάτων. Τέτοιες ιδιότητες όχι μόνο σχηματίζονται από γεωφυσικές διεργασίες αλλά μπορούν επίσης να αντικατοπτρίζουν τις αλλοιώσεις που έχουν προκληθεί από ανθρώπινη δραστηριότητα⁴⁶⁰. Οι μέθοδοι που χρησιμοποιεί η αρχαιολογία για την έρευνα του υπεδάφους είναι σε μεγάλο βαθμό συμπληρωματικές και χωρίζονται γεωφυσικές, γεωχημικές και γεωτρητικές⁴⁶¹.

⁴⁵⁸ Κουκουζέλη κ.ά. 2003, 141.

⁴⁵⁹ Tabor 2004, 41–46

⁴⁶⁰ Weymouth 1986, 313.

⁴⁶¹ Κουκουζέλη κ.ά. 2003, 142.

Οι γεωφυσικές μέθοδοι *παθητικές* ή *ενεργητικές* μπορούν να εντοπίσουν ανωμαλίες που προξενήθηκαν ανθρώπινη δραστηριότητα σε ομοιογενή υπεδάφη ή στον βυθό της θάλασσας. Στην κατηγορία των παθητικών μεθόδων η μόνη τεχνική που συνδυάζεται με τις απαιτήσεις της αρχαιολογίας είναι η μαγνητική διασκόπηση και χρησιμοποιεί συσκευή υψηλής ευαισθησίας για τη μέτρηση του συνολικού μαγνητικού πεδίου της γης. Τα σιδηρούχα μεταλλικά αντικείμενα ή οι ανομοιογένειες στις μαγνητικές ιδιότητες του εδάφους προκαλούν αποκλίσεις στο μετρούμενο πεδίο. Υπάρχει και άλλη παθητική μέθοδος, η βαρυτική διασκόπηση, που χρησιμοποιείται κυρίως από γεωφυσικούς επιστήμονες. Αυτή η μέθοδος μέτρησης εξαρτάται από τη διαφορά μάζας μεταξύ του τεχνέργου, του οικοδομημένου ή της κατασκευής και της περιβάλλουσας μήτρας. Ωστόσο, δεν είναι αρκετά ευαίσθητη για αρχαιολογικούς σκοπούς⁴⁶².

Στην κατηγορία των ενεργητικών τεχνικών περιλαμβάνονται διάφορες μέθοδοι, στις οποίες περιλαμβάνεται η εισαγωγή παλμού στη γη με τη μέθοδο της σεισμικής διάθλασης. Το βάθος ενός ανώτερου στρώματος εδάφους είναι δυνατόν να εκτιμηθεί με τη μέτρηση του χρόνου της κίνησης κύματος μεταξύ ενός δονητή - συνήθως σφυρί που χτυπά μια πλάκα και την επιστροφή του παλμού σε γεώφωνα⁴⁶³. Αυτή η μέθοδος παρέχει μεγαλύτερες δυνατότητες ανάλυσης από τη μέθοδο ανάκλασης του παλμού, αλλά η εισαγωγή του σήματος στη γη δεν είναι πάντα απλή διαδικασία⁴⁶⁴.

Οι επόμενες τεχνικές ομαδοποιούνται υπό τον γενικό όρο ηλεκτρομαγνητικές μέθοδοι. Οι ηλεκτρομαγνητικές (ΗΜ) μέθοδοι διασκόπησης αναφέρονται σε μεθόδους που χρησιμοποιούν εναλλασσόμενα ηλεκτρομαγνητικά ρεύματα χαμηλής συνήθως συχνότητας, τα οποία επάγονται στη γη χωρίς τη χρήση ηλεκτροδίων. Τα ηλεκτρομαγνητικά πεδία των πομπών επάγουν στη γη εναλλασσόμενα ρεύματα, τα οποία παράγουν δευτερογενή ηλεκτρομαγνητικά πεδία. Τα δευτερογενή πεδία έχουν ιδιότητες που εξαρτώνται από τη δομή του υπεδάφους. Ο σκοπός της συγκεκριμένης μεθόδου είναι η ανακάλυψη της γεωηλεκτρικής δομής βάσει των ιδιοτήτων του δευτερογενούς πεδίου και συνεπώς η ανίχνευση και καταγραφή των ανωμαλιών που υποδεικνύουν. Τα

⁴⁶² Για τη χρήση της μεθόδου στο πεδίο βλ. Linington 1966, 37-41.

⁴⁶³ Clark 1996, 121.

⁴⁶⁴ Weymouth 1986, 314.

όργανα που χρησιμοποιούνται, κυμαίνονται από τους ανιχνευτές μετάλλων έως τους προηγμένους μετρητές αγωγιμότητας του εδάφους.

Η κύρια πηγή των μαγνητικών ιδιοτήτων του εδάφους είναι τα ορυκτά του σιδήρου, που εμπεριέχονται στην κεραμική και άλλα τέχνηρα. Τα κεραμικά τέχνηρα περιέχουν ποσότητες μαγνητίτη, αιματίτη και άλλων σιδηρομαγνητικών ορυκτών, σε ποσοστό γύρω στο 5% του πηλού από τον οποίο πλάστηκαν. Μετά την όπτηση οι μαγνητικοί κόκκοι προσανατολίζονται και λαμβάνουν τη διεύθυνση του γήινου μαγνητικού πεδίου κατά τη στιγμή ψύξης του κεραμικού. Έτσι, το τέχνηρο αποκτά θερμοπαραμένουσα μαγνήτιση. Τα ορυκτά που παίζουν σημαντικό ρόλο στη διαμόρφωση της μαγνητικής επιδεκτικότητας των εδαφών είναι τα οξειδία του σιδήρου βουρσίτης, μαγνητίτης, αιματίτης, μαγκεμίτης, τα υδροξείδια του σιδήρου $Fe(OH)_2$, $Fe(OH)_3$ βερναλίτης, το οξείδιο του τιτανίου και σιδήρου ιλμενίτης και το σουλφίδιο του σιδήρου μαγνητοπυρίτης. Η «ισχυρή συνιστώσα» της μαγνητικής επιδεκτικότητας οφείλεται στον μαγνητίτη και τον μαγκεμίτη, ενώ η «ασθενής συνιστώσα» στον αιματίτη και άλλα οξειδία ή σουλφίδια⁴⁶⁵.

Εδάφη τα οποία έχουν υποστεί καύση (π.χ. καύση καλλιεργήσιμων γαιών και αρχαία ανθρώπινη κατοίκηση) ή μηχανισμούς ζύμωσης (άρωση καλλιεργήσιμης γης, ανακύκλωση αρχαιολογικών ερειπίων, αποθέτες) παρουσιάζουν υψηλή μαγνητική επιδεκτικότητα⁴⁶⁶. Η αιτία του φαινομένου είναι κυρίως ο σχηματισμός του μαγκεμίτη μέσω αναγωγικών αντιδράσεων. Έτσι, τα εδάφη αυτά παρουσιάζουν ενισχυμένη επαγόμενη και παραμένουσα μαγνήτιση και προκαλούν ισχυρές μαγνητικές ανωμαλίες⁴⁶⁷.

Για τη μαγνητική διασκόπηση αρχαιολογικών καταλοίπων χρησιμοποιούνται συσκευές όπως το μαγνητόμετρο πρωτονίου. Η βασική δομή του συνίσταται σε μια χάλκινη, συνήθως ηλεκτραγώγιμη σπείρα, που περιβάλλει έναν κύκλινδρο παραφίνης, κηροζίνης, ή ισοπροπυλικής αλκοόλης, υλικά πλούσια σε πρωτόνια⁴⁶⁸. Μετρήσιμο ηλεκτρικό ρεύμα εισάγεται στη σπείρα εξαιτίας της επαναδιευθέτησης των πρωτονίων της παραφίνης, καθώς το

⁴⁶⁵ Maher 1986, 76-92.

⁴⁶⁶ Cole *et al* 1995, 114-162.

⁴⁶⁷ Linford 2005, 127-144.

⁴⁶⁸ Drewett 1999, 53.

μαγνητόμετρο περνά πάνω από εναλασσόμενο μαγνητικό πεδίο⁴⁶⁹. Οι εναλλαγές ή παρεκλίσεις από το ομοιόμορφο μαγνητικό πεδίο της γης υποδεικνύουν διαφοροποιήσεις στη μήτρα του εδάφους. Σύμφωνα με τον Weymouth οι κατασκευές που μπορούν να ανιχνευθούν με μαγνητόμετρο είναι αποθέτες, θεμέλια, τοίχοι, σκαλιά, τάφροι, εστίες, καμένες κατασκευές και κεραμίδες⁴⁷⁰. Από τεχνική άποψη, αυτά τα χαρακτηριστικά έχουν διαφορετική μαγνητική ευαισθησία από το περιβάλλον έδαφος, οδηγώντας σε μικρές μεταβολές του μαγνητικού πεδίου της γης⁴⁷¹. Το μειονέκτημα του μαγνητόμετρου είναι ότι επηρεάζεται από ηλεκτροφόρα καλώδια, ηλεκτρικά τρένα και από γειτονικά μεταλλικά αντικείμενα οποιουδήποτε μεγέθους, όπως μεταλλικούς φράχτες, σιδερένιες γέφυρες, οχήματα κ.ά. και είναι, επομένως, ακατάλληλο για το αστικό περιβάλλον⁴⁷². Για την αποφυγή των προβλημάτων που προκαλούν τα μεταλλικά αντικείμενα ή κατασκευές χρησιμοποιείται ο μετρητής μαγνητικής ροής για την κατασκευή χαρτών, στους οποίους οι περιοχές υψηλής μαγνητικής χωρητικότητας σημειώνονται με γραμμές, στίγματα, «κορυφώσεις» κ.λπ.⁴⁷³.

Η μέθοδος ηλεκτρικής αντίστασης μετρά τις μεταβολές στην ηλεκτρική αντίσταση του εδάφους, οι οποίες οφείλονται στην παρουσία θαμμένων αρχαίων καταλοίπων. Η συσκευή που χρησιμοποιείται για το σκοπό αυτό είναι ο μετρητής ηλεκτρικής αντίστασης (εισάγει στο έδαφος τέσσερα ηλεκτρόδια κάθε φορά σε ίσα διαστήματα ακολουθώντας τις γραμμές ενός καννάβου ή έναν αριθμό παράλληλων γραμμών που καλύπτουν την εξεταζόμενη θέση· τα ηλεκτρόδια διοχετεύουν ηλεκτρικό ρεύμα μέσα στο έδαφος και μετρούν την αντίσταση που προβάλλεται σε αυτό⁴⁷⁴. Το πλεονέκτημα της συγκεκριμένης μεθόδου είναι ότι δεν επηρεάζεται από γειτονικά ηλεκτροφόρα καλώδια ή μεταλλικά αντικείμενα, για αυτό και προτιμάται από τη μαγνητική έρευνα επιφάνειας. Το μειονέκτημά της είναι ότι εξαρτάται από τις καιρικές συνθήκες (σε εύκρατα κλίματα, οι θερινοί μήνες είναι οι πιο πρόσφοροι για την εφαρμογή της μεθόδου ηλεκτρικής αντίστασης).

⁴⁶⁹ Black and Johnston 1962, 199-205.

⁴⁷⁰ Weymouth 1986, 343.

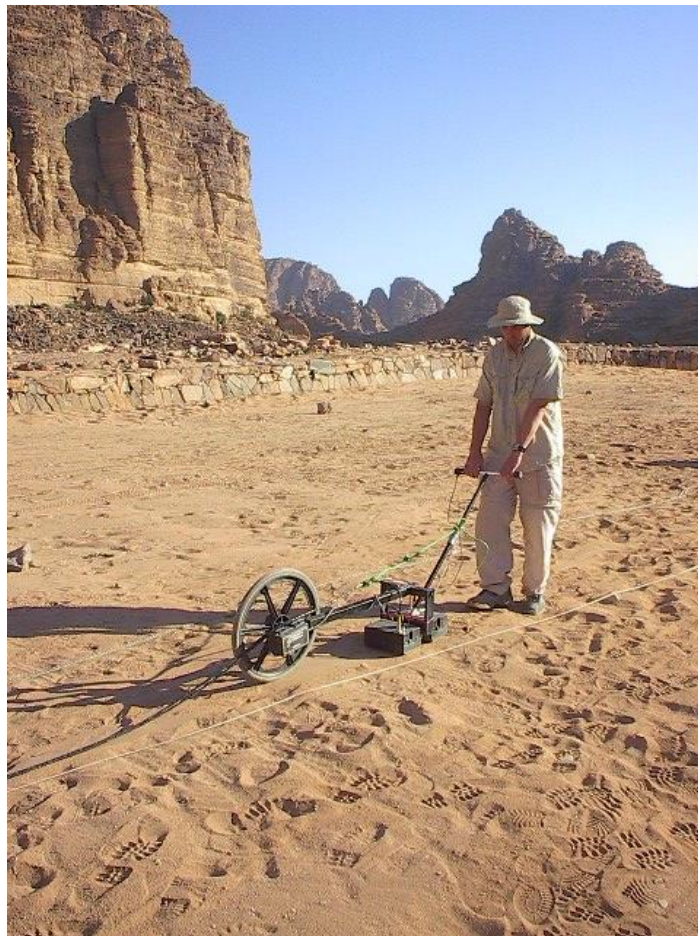
⁴⁷¹ Hester *et al* 2016, 61.

⁴⁷² Κουκουζέλη κ.ά 2003, 142.

⁴⁷³ Renfrew and Bahn 2001, 97.

⁴⁷⁴ Renfrew and Bahn 2001, 96, 101.

Το ραντάρ υπεδάφους (Ground Penetrating Radar) μετρά τις αντιθέσεις που υπάρχουν στις ηλεκτρικές και μαγνητικές ιδιότητες του υπεδάφους, εξαιτίας της παρουσίας αρχαίων καταλοίπων. Η συσκευή ραντάρ τοποθετείται σε τροχήλατη βάση και σύρεται με αργό ρυθμό σε μικρή απόσταση από την επιφάνεια του εδάφους ή απευθείας επάνω σε αυτήν, κατά μήκος μιας διαδρομής ή μιας γραμμής καννάβου, εκπέμποντας ηλεκτρομαγνητικά σήματα ή παλμούς (ραδιοκύματα). Οι ανακλάσεις που προξενούνται στα σήματα ή τους παλμούς από τις ανωμαλίες του υπεδάφους καταγράφονται, παρέχοντας πληροφορίες σχετικές με το βάθος και τη φύση των ανωμαλιών.



Εικ. 3.3. Διασκόπηση με ραντάρ υπεδάφους στο Wadi Ramm, Ιορδανία⁴⁷⁵

Η ερευνητική μέθοδος με ραντάρ υπεδάφους είναι συμπληρωματική της μαγνητικής διασκόπησης και της μεθόδου ηλεκτρικής αντίστασης και μπορεί να φθάσει σε πολύ μεγαλύτερα βάθη. Επίσης, παρέχει καθαρότερη εικόνα του υπεδάφους, εφόσον υπάρχουν σαφή περιγράμματα λίθινων κατασκευών,

⁴⁷⁵ Archaeo-Physics LLC, διαθέσιμο online, επίσης Wikimedia Commons.

κατοικίες με καλά συμπιεσμένα δάπεδα, λάκκοι, μεταλλικά αντικείμενα ή αλλαγές στις ιδιότητες των υλικών, ρωγμές και κενά⁴⁷⁶. Επίσης παρέχει από άκρο σε άκρο της εξεταζόμενης θέσης πολλαπλές εγκάρσιες τομές του υπεδάφους, οι οποίες είναι εξαιρετικά χρήσιμες για την κατανόηση της στρωματογραφίας της θέσης και την τρισδιάστατη απεικόνιση των καταλοίπων της. Στα μειονεκτήματά της μεθόδου συγκαταλέγονται οι καιρικές συνθήκες και η δυσκολία χειρισμού του ραντάρ και της ερμηνείας των αποτελεσμάτων⁴⁷⁷.

Η μέθοδος ηχητικής διερεύνησης, γνωστή και ως μέθοδος ακουστικής ανάκλασης ή μέθοδος ηχοβόλησης, χρησιμοποιείται κυρίως στην υποβρύχια Αρχαιολογία. Μια ανιχνευτική ηχητική συσκευή στέλνει έναν ηχητικό παλμό και μετρά την ταχύτητα με την οποία ανταποκρίνεται ο παλμός σε διάφορες αντιθέσεις που τυχόν υπάρχουν στις μηχανικές (σε αντίθεση με τις μαγνητικές ή ηλεκτρικές) ιδιότητες του υπεδάφους λόγω της παρουσίας αρχαίων καταλοίπων⁴⁷⁸.

Ένας εντελώς διαφορετικός τρόπος εντοπισμού της ανθρώπινης δραστηριότητας στο τοπίο είναι ο εντοπισμός αλλαγών στη χημεία του εδάφους που προκύπτουν από ανθρωπογενείς δραστηριότητες. Έχει υπολογιστεί ότι 100 άνθρωποι, που εκτελούν μια σειρά οικιακών και γεωργικών δραστηριοτήτων, καταθέτουν περίπου 124 κιλά φωσφόρου στο τοπίο ετησίως⁴⁷⁹. Οι γεωχημικές μέθοδοι εντοπισμού συνίστανται στη συλλογή κυλινδρικών δειγμάτων εδάφους, που αποσπώνται με γεώτρηση σε κανονικά διαστήματα καννάβου από το επιφανειακό στρώμα του εδάφους μιας θέσης και της ευρύτερης περιοχής της. Τα δείγματα υποβάλλονται σε χημική ανάλυση για την ανίχνευση στοιχείων που υποδηλώνουν ανθρώπινη κατοίκηση και αγροτικές δραστηριότητες. Με την ανάλυση φωσφορικού άλατος και την ανάλυση ιχνοστοιχείων μετάλλων, οι οποίες αποτελούν τις δύο βασικές γεωχημικές μεθόδους που χρησιμοποιούνται από την αρχαιολογία, είναι δυνατόν να προσδιοριστούν με ακρίβεια τα ευρύτερα όρια μιας θέσης, καθώς επίσης οι περιοχές πυκνής κατοίκησης και η λειτουργία τους -κάτι που δεν είναι δυνατόν να αποκαλυφθεί με άλλες μεθόδους εκτός της ανασκαφής και είναι ιδιαίτερα χρήσιμο για θέσεις χωρίς αρχιτεκτονικά κατάλοιπα.

⁴⁷⁶ Daniels 2004, 1-4.

⁴⁷⁷ Κουκουζέλη κ.ά. 2003, 143.

⁴⁷⁸ Renfrew and Bahn 2001, 93.

⁴⁷⁹ Proudfoot 1976, 93-113.

Έτσι, τα αποτελέσματα της γεωχημικής ανάλυσης μπορούν να συνδυαστούν με τα δεδομένα της αεροφωτογράφισης, της γεωφυσικής έρευνας (ιδίως της έρευνας μαγνητικής χωρητικότητας) και της έρευνας επιφάνειας, για να συνθέσουν μια πληρέστερη εικόνα του ευρύτερου χώρου μιας θέσης, ανεξάρτητα από το αν αυτή θα ανασκαφεί ή όχι. Ένα μειονέκτημα της ανάλυσης φωσφορικού άλατος είναι το γεγονός ότι το φωσφορικό άλας που περιέχεται στο επιφανειακό στρώμα εδάφους μιας θέσης μπορεί να αποστραγγιστεί και να χαθεί όταν το υπέδαφος της θέσης είναι αμμώδες ή τυρφώδες.

Οι γεωτρητικές μέθοδοι εκτελούνται με μεταλλικές ράβδους ή γεωτρύπανα και τρυπάνια και χρησιμοποιούνται για την ανεύρεση θαμμένων αρχαιολογικών καταλοίπων, τον υπολογισμό του βάθους τους, την εξέταση της συνοχής τους. Η συγκεκριμένη μέθοδος διερεύνησης τους υπεδάφους παρέχει στον αρχαιολόγο ζωτικής σημασίας στρωματογραφική πληροφορία και δείγματα εδάφους⁴⁸⁰. Μειονέκτημα της γεωτρητικής ανίχνευσης είναι ότι μπορεί να αποδειχθεί καταστρεπτική για θαμμένες κατασκευές ή εύθραυστα τέχνηρα⁴⁸¹.



Εικ. 3.4. Δείγμα εδάφους σε μεταλλική ράβδο⁴⁸²

Πλεονεκτήματα και μειονεκτήματα μεθόδων έρευνας υπεδάφους

Μέθοδος	Πλεονεκτήματα	Μειονεκτήματα
Ραντάρ υπεδάφους	<ul style="list-style-type: none"> • Εφαρμογή σε ρηχή και μετρίου βάθους έρευνα • Εφαρμογή σε αστικό και 	<ul style="list-style-type: none"> • Απαιτεί καλή επαφή με την επιφάνεια του εδάφους

⁴⁸⁰ Stein 1986, 505-527.

⁴⁸¹ Κουκουζέλη κ.ά. 2003, 145.

⁴⁸² Stein 1986, 515.

Μέθοδος	Πλεονεκτήματα	Μειονεκτήματα
	<p>περιβάλλον</p> <ul style="list-style-type: none"> • Λειτουργία σε λιμναία περιβάλλοντα • Δεδομένα υψηλής ανάλυσης • Γρήγορη συλλογή δεδομένων • Στρωματογραφία και ογκομετρικές εικόνες υπεδάφους • Ικανότητα χαρτογράφησης γεωλογικών ή/και γεωμορφολογικών χαρακτηριστικών • Προσδιορισμός του βάθους της του βραχώδους υποστρώματος • Ικανότητα ανίχνευσης κενών, σπηλαίων, τάφων, τοιχοδομών 	<ul style="list-style-type: none"> • Εξάρτηση από κλιματικές συνθήκες • Δε λειτουργεί σε θαλάσσιο περιβάλλον • Εξειδικευμένος τρόπος επεξεργασίας δεδομένων • Μέτρια δυνατότητα μετακίνησης εξοπλισμού
Μέθοδοι ηλεκτρομαγνητικής επαγωγής	<ul style="list-style-type: none"> • Εφαρμογή τόσο στις ρηχές όσο και σε βαθιές έρευνες • Κατάλληλο για χρήση σε αγροτικές τοποθεσίες • Δεν υπάρχει ανάγκη επαφής με το έδαφος • Ίδανικό για έρευνες μεγάλης κλίμακας • Μεγάλη κινητικότητα οργάνων • Παροχή αγωγιμότητας εδάφους και μαγνητικής ευαισθησίας εδάφους • Παροχή πληροφοριών σχετικά με τη γεωλογία και τη λιθολογία, τις στρωματογραφικές αλλαγές και την καταγραφή του τοπίου • Ικανότητα ανίχνευσης τάφρων, κοιλοτήτων, καμένων αρχιτεκτονημάτων, παλαιοεδαφών, παλαιοκαναλιών που συσσωρεύουν μαγνητικά ορυκτά μέσω πεδογενικών ή διαβρωτικών διεργασιών 	<ul style="list-style-type: none"> • Ακατάλληλη μέθοδος για αστικά περιβάλλοντα • Μέτρια δυνατότητα επεξεργασίας δεδομένων • Δυσκολία συλλογής δεδομένων πολλαπλών αισθητήρων
Τεχνικές ηλεκτρικής αντίστασης (ΤΗΑ)	<ul style="list-style-type: none"> • Εφαρμογή σε μεσαίου βάθους και 	<ul style="list-style-type: none"> • Ανάγκη επαφής με την επιφάνεια του

Μέθοδος	Πλεονεκτήματα	Μειονεκτήματα
	<p>βαθιές έρευνες</p> <ul style="list-style-type: none"> • Κατάλληλο για χρήση τόσο σε αγροτικές όσο και σε αστικές περιοχές και θαλάσσιες έρευνες • Εύκολη επεξεργασία δεδομένων • 2D και 3D μοντελοποίηση των υπόγειων γεωλογικών στρωμάτων • Απλή εφαρμογή σε αρχαιολογικές έρευνες • Ιδανική για την ανίχνευση στόχων βαθιάς εναπόθεσης (π.χ. αρχαίοι λιμένες που καλύπτονται από προσχλωσιγενή ιζήματα, τάφρους και παλαιοκανάλια) 	<p>εδάφους</p> <ul style="list-style-type: none"> • Εξάρτηση από κλιματικές συνθήκες • Οι μετρήσεις ΤΗΑ χρειάζονται επεξεργασία εξειδικευμένων δεδομένων • Ακριβά όργανα • Σχετικά αργή για εκτεταμένη χαρτογράφηση • Δυσκολία ως προς την κινητικότητα και ευελιξία οργάνων
Μαγνητικές μέθοδοι	<ul style="list-style-type: none"> • Εφαρμογή σε έρευνες μικρού βάθους • Κατάλληλη για χρήση σε αγροτικές τοποθεσίες • Δεν υπάρχει ανάγκη επαφής με το έδαφος • Ιδανική για εκτεταμένες μελέτες χαρτογράφησης με υψηλή ανάλυση δειγματοληψίας • Εύκολη επεξεργασία των δεδομένων • Μεγάλη κινητικότητα των οργάνων μέτρησης • Ανίχνευση σιδηρούχων ή καμένων χαρακτηριστικών, μεταλλικών αντικειμένων • Δυνατότητα χαρτογράφησης της έκτασης των πεδογενικών ορυκτών του εδάφους 	<ul style="list-style-type: none"> • Ακατάλληλη για αστικά περιβάλλοντα • Εξάρτηση από τις συνθήκες εδάφους • Υψηλή ευαισθησία σε μεταλλικά αντικείμενα • Ακατάλληλη για μελέτες στρωματογραφίας ιζημάτων
Μαγνητική επιδεκτικότητα	<ul style="list-style-type: none"> • Εφαρμογή σε έρευνες μικρού βάθους 	<ul style="list-style-type: none"> • Ακατάλληλη σε αστικά περιβάλλοντα • Ανάγκη για

Μέθοδος	Πλεονεκτήματα	Μειονεκτήματα
	<ul style="list-style-type: none"> • Κατάλληλο για χρήση σε αγροτικές τοποθεσίες • Δυνατότητα χαρτογράφησης των συνθηκών εναπόθεσης σε αρχαιολογικούς χώρους • Ανίχνευση επέκτασης και έντασης χρήσης • Προέλευση αρχαιολογικών υλικών 	<p>εργαστηριακές μετρήσεις</p> <ul style="list-style-type: none"> • Δεν είναι κατάλληλη για υψηλής ανάλυσης ή εκτεταμένες έρευνες
Ηχοβόλιση	<ul style="list-style-type: none"> • Εφαρμογή σε μεσαίου και μεγάλου βάθους • Κατάλληλη για χρήση τόσο σε αγροτικές όσο και σε αστικές περιοχές και θαλάσσιες έρευνες • Ικανότητα χαρτογράφησης της στρωματογραφίας των ιζημάτων και ανακατασκευή της παλαιο-τοπογραφίας και του βάθους του υπεδάφους • Ικανότητα χαρτογράφησης μεγάλων μνημειακών κατασκευών και αναχωμάτων 	<ul style="list-style-type: none"> • Ανάγκη καλής επαφής με την επιφάνεια του εδάφους • Εξειδικευμένος τρόπος επεξεργασίας δεδομένων • Αργή μέθοδος για τη χαρτογράφηση των ερευνών • Ακριβά όργανα
Γεωτρητικές μέθοδοι	Πλούτος στρωματογραφικών δεδομένων	Εν δυνάμει καταστρεπτική διαδικασία
Γεωχημική ανάλυση	<ul style="list-style-type: none"> • Παροχή πληροφοριών σχετικά με γεωργικές και εργαστηριακές δραστηριότητες, κτηνοτροφία και κατασκευή τάφρων • Ανασύσταση πρακτικών χρήσης γης κατά το παρελθόν 	<ul style="list-style-type: none"> • Απαιτεί εργαστηριακές μετρήσεις • Ακατάλληλη για εκτεταμένες έρευνες

Η αρχαιολογική γεωφυσική είναι σειρά μη ή περιορισμένων επεμβατικών τεχνικών για τη διερεύνηση των φυσικών παραμέτρων του περιβάλλοντος μιας αρχαιολογικής θέσης. Η γεωφυσική έρευνα, όπως και η γεωχημική, συμπληρώνει την αρχαιολογική έρευνα ή την ανασκαφή δεδομένου ότι μπορεί να παρέχει πληροφορίες σχετικά με τη στρωματογραφία και να εντοπίζει ανθρωπογενή ίχνη

του παρελθόντος, να τεκμηριώνει τις χωρικές διαστάσεις τους και, υπό ιδανικές συνθήκες, να διερευνά τις φυσικές ιδιότητες των υλικών θαμμένων υπό την επιφάνεια του εδάφους. Τα υπολείμματα ενός υλικού πολιτισμού, όπως τα θεμέλια κτιρίων και οι έμμεσες ενδείξεις ανθρωπογενούς δραστηριότητας, όπως οι ενδείξεις καύσης, είναι άμεσοι στόχοι της γεωφυσικής έρευνας, αφού μπορούν να διαφοροποιηθούν βάσει των υλικών τους ιδιοτήτων από το ευρύτερο πλαίσιο του εδάφους. Εκτός από τον άμεσο εντοπισμό του αρχαιολογικού υλικού, οι γεωφυσικές και γεωχημικές τεχνικές μπορούν να συμβάλλουν σημαντικά στις γεωαρχαιολογικές έρευνες, διασαφηνίζοντας τη στρωματογραφία του χώρου. Σε ορισμένες περιπτώσεις, όπως ο εντοπισμός προϊστορικού υλικού θαμμένου στην ανοικτή θάλασσα ή σε ανοικτές παλαιολιθικές θέσεις, η ανακατασκευή του αρχαίου τοπίου μπορεί να συμβάλει σημαντικά στις αρχαιολογικές έρευνες με την άμεση γεωφυσική ανίχνευση αρχαιολογικών υλικών.

3.10. Η αρχαιολογική ανασκαφή

Η ανασκαφή θεωρείται εν γένει το πλέον αξιόπιστο μέσο ανάκτησης αρχαιολογικών δεδομένων, καθώς αποκαλύπτει κατάλοιπα κάτω από την επιφάνεια του εδάφους. Έτσι, παρέχει την ευκαιρία της άμεσης επαφής και ελέγχου των δεδομένων που προκύπτουν από τον εντοπισμό και τον καθορισμό των αρχαιολογικών θέσεων⁴⁸³. Μέσω της ανασκαφής ανακτώνται από τη γη αρχαιολογικά δεδομένα μη ανακτήσιμα με διαφορετικό τρόπο και αποκαλύπτονται ακολουθίες δομών ή διαστρωματωμένες και ασφαείς χρονολογικές αλληλουχίες περιβαλλοντικών ή οικονομικών δεδομένων μιας αρχαιολογικής θέσης. Το έδαφος είναι ένα ιστορικό κείμενο, το οποίο, όπως και το γραπτό αρχείο, χρειάζεται αποκρυπτογράφηση, μετάφραση και ερμηνεία, πριν μπορέσει να χρησιμοποιηθεί⁴⁸⁴.

Η συζήτηση για τον καταστροφικό χαρακτήρα της ανασκαφικής πρακτικής είναι μεγάλη, καθώς η αντίληψη πως «η ανασκαφή είναι καταστροφή» επαναλαμβάνεται συχνά στους αρχαιολογικούς κύκλους, ακόμη και ως προτροπή για προσεκτική καταγραφή⁴⁸⁵. Αν δεν προβάλλεται ως καταστροφική

⁴⁸³ Renfrew and Bahn 2001, 101.

⁴⁸⁴ Barker 1993, 1-2.

⁴⁸⁵ Lucas 2001a, 3.

δραστηριότητα, ορίζεται ως «μη επαναλήψιμο πείραμα»⁴⁸⁶. Άλλοι δίνουν έμφαση στις δημιουργικές όψεις της ανασκαφής, βλέποντάς τη όχι ως «καταστροφέα» υλικού, αλλά ως «δημιουργό» δεδομένων⁴⁸⁷ και δίνουν έμφαση στις ευκαιρίες για δημιουργικότητα της ανασκαφικής πρακτικής⁴⁸⁸. Παρόμοια, η ανασκαφή θεωρείται υλική παρέμβαση και μορφή αρχειοθέτησης που ενσωματώνει τις υποκειμενικές προσεγγίσεις του ανασκαφέα, τις ερμηνείες και τις προκαταλήψεις του, όπως και την αντίληψη ότι το αρχαιολογικό αρχείο καθίσταται υποκειμενικό από τη στιγμή που εμπλέκεται μαζί του ο αρχαιολόγος⁴⁸⁹.

Οι παραπάνω θεωρήσεις δεν καλύπτουν, ωστόσο, ένα δεύτερο πρόβλημα που αφορά στην προσεκτική εξέταση της ποιότητας των δεδομένων⁴⁹⁰. Και με την πρόοδο των τεχνολογιών καταγραφής, ορισμένοι φαίνεται να επιστρέφουν συντηρητική άποψη, που συγκρίνει την αυξανόμενη ποιότητα των δεδομένων με την αντικειμενικότητα με την οποία μπορεί να ανακτηθεί⁴⁹¹. Ωστόσο, λόγω της έγχυσης των προκαταλήψεων των ανασκαφέων στη διαδικασία της ανασκαφής, η σημασία της αρχαιολογικής καταγραφής δεν είναι τόσο σημαντική για τη διατήρηση ακριβών αντιγράφων των στρωματογραφικών προβλημάτων, όσο για τη διατήρηση του τρόπου με τον οποίο οι ανασκαφείς αλληλεπιδρούν μαζί τους⁴⁹².

Από τη στιγμή που λαμβάνεται η απόφαση για την ανασκαφή μιας αρχαιολογικής θέσης, συγκροτείται ομάδα εργασίας για την πραγματοποίηση του ανασκαφικού έργου. Την ευθύνη για τη γενική εποπτεία και τον συντονισμό της ανασκαφικής ομάδας φέρει ο διευθυντής της ανασκαφής. Ανάλογα με το μέγεθος της αρχαιολογικής θέσης, τους στόχους και τους διαθέσιμους πόρους, που καθορίζονται από τον ερευνητικό σχεδιασμό, η ανασκαφική ομάδα χρειάζεται εκτός από τους επόπτες τομών και άλλους αρχαιολόγους, όπως και εκείνους τους επιστήμονες που απαιτούνται, ανάλογα με τη φύση της θέσης που πρόκειται να ανασκαφτεί. Επίσης, χρειάζονται άτομα αρκετών ειδικοτήτων, όπως σχεδιαστές, χειριστές ηλεκτρονικών υπολογιστών, φωτογράφοι, γραμματείς-

⁴⁸⁶ Barker 1982, 12.

⁴⁸⁷ Frankel 1993, 875-877. Επίσης Carver 2006, 6-8.

⁴⁸⁸ Yarrow 2003, 65-73.

⁴⁸⁹ Lucas 2001b, 35-46

⁴⁹⁰ Carver 2012, 19.

⁴⁹¹ Bradley 2006, 6.

⁴⁹² Roosevelt et al 2015, 325-346.

λογιστές, συντηρητές και βοηθοί για την επεξεργασία των ευρημάτων, έμπειροι τεχνίτες και εργάτες⁴⁹³.

Η διαδικασία

Όποια και αν είναι η φύση της αρχαιολογικής θέσης και της στρωματογραφίας της, ο σκοπός της αρχαιολογικής ανασκαφής είναι η κατανόηση της σειράς των αποθέσεων και των γεγονότων που τα παρήγαγαν. Με άλλα λόγια, η ανασκαφική έρευνα στοχεύει στην παραγωγή μιας ιστορίας των αποθέσεων για κάθε ανασκαμμένο τετράγωνο ή ομάδα τετραγώνων. Στη συνέχεια, κατά τη μετανασκαφική ανάλυση, καθορίζεται η κάθετη χρονολόγηση και η σύνθεση των πολιτισμικών ιστοριών που παράγει η κάθε απόθεση⁴⁹⁴.

Κατά τη διαδικασία αφαιρούνται τα επάλληλα οριζόντια αρχαιολογικά στρώματα της θέσης και καταγράφονται μαζί με τα ευρήματα που περιέχουν και τα αρχαιολογικά πλαίσια των ευρημάτων στο ημερολόγιο της ανασκαφής. Ως πλαίσιο νοείται οποιαδήποτε αρχαιολογική οντότητα στην αρχαιολογική θέση και θα μπορούσε να είναι ένα στρώμα, ένας αποθέτης ή πασσαλότρυπα⁴⁹⁵. Η ανασύνθεση των πληροφοριών βάσει του ημερολογίου, των φωτογραφιών και των σχεδίων, παρέχει μια τρισδιάστατη εικόνα της θέσης⁴⁹⁶.

Η ιδανική ανασκαφή εξάγει από την αρχαιολογική θέση όλα όσα θα μπορούσαν ενδεχομένως να γίνουν γνωστά, δηλαδή όλα όσα επιβίωσαν από τις φυσικές και χημικές διαδικασίες φθοράς. Συνεπώς οι χρησιμοποιούμενες μέθοδοι στη ιδανική ανασκαφή χρειάζεται να είναι εκλεπτυσμένες, καθώς η απλή και βιαστική εκσκαφή θα ανακτήσει μόνο ένα μικρό ποσοστό αρχαιολογικών δεδομένων. Πολύ συχνά, οι ανασκαφές γίνονται βιαστικά ή πραγματοποιούνται με μηχανικούς εκσκαφείς JCB αντί για τα κλασικά εργαλεία, εξαιτίας παραγόντων που βρίσκονται πέρα από τον έλεγχο του αρχαιολόγου, όπως η επικείμενη οικιστική ανάπτυξη, η τυχαία ανακάλυψη ή απλή έλλειψη πόρων⁴⁹⁷.

Όταν αφαιρούνται τα στρώματα της αρχαιολογικής θέσης, καταγράφονται λεπτομερώς και με τη μεγαλύτερη δυνατή ακρίβεια. Βέβαια, η εξαντλητική καταγραφή όλων των ευρημάτων, αν και τεχνικά εφικτή, είναι ασύμφορη και

⁴⁹³ Κουκουζέλη κ.ά. 2003, 150.

⁴⁹⁴ Hester et al 2016, 238.

⁴⁹⁵ Drewett 1999, 107.

⁴⁹⁶ Κουκουζέλη κ.ά. 2003, 147.

⁴⁹⁷ Barker 1993, 71.

κοστοβόρα. Περαιτέρω, η ολοσχερής αφαίρεση όλων των στρωμάτων είναι εν γένει ανεπιθύμητη, καθώς απογυμνώνει την αρχαιολογική θέση από δεδομένα, πιθανώς χρήσιμα σε μελλοντική έρευνα. Μέση οδός στο πρόβλημα θα μπορούσε να είναι η επιλεκτική συντήρηση ή η μακροπρόθεσμη ανασκαφή θέσεων προσεκτικά επιλεγμένων, όπου μπορούν να εξασφαλιστούν οι βέλτιστες συνθήκες ανασκαφής, μαζί με τη σωστική ανασκαφή όσο το δυνατόν περισσότερων θέσεων. Η συγκεκριμένη λύση έχει αποδώσει ικανοποιητικά αποτελέσματα, τουλάχιστον στη βρετανική αρχαιολογία⁴⁹⁸.

Η ταχύτητα και ο τρόπος διεξαγωγής της ανασκαφής εξαρτάται από το είδος της ανασκαφής. Υπάρχει μια διαρκής σύγκρουση μεταξύ της σωστής ανασκαφής και της ανάγκης για ταχύτητα, καθώς υπάρχει μια βέλτιστη ταχύτητα με την οποία μπορεί να πραγματοποιηθεί η ανασκαφή. Η αρχαιολογική θέση, ανεξαρτήτως είδους, υπαγορεύει την ταχύτητα της ανασκαφής⁴⁹⁹.

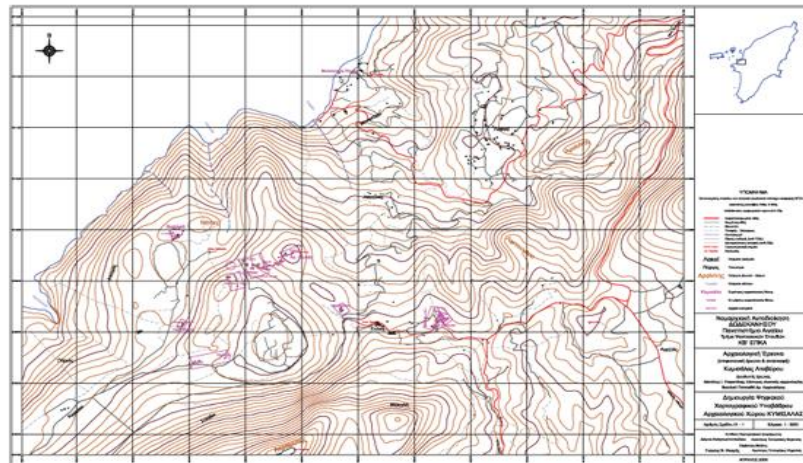
Στις σωστικές ανασκαφές που διεξάγονται σε θέσεις που απειλούνται από ανθρωπογενείς ή φυσικές καταστροφές, η ταχύτητα και τα χρησιμοποιούμενα εργαλεία –συνήθως μηχανικοί εκσκαφείς- απαιτούν αυξημένη προσοχή, ώστε να είναι δυνατή η ανάκτηση και η καταγραφή όσο το δυνατόν περισσότερων δεδομένων πριν από την πλήρη εκσκαφή και συνεπώς καταστροφή της θέσης. Στον αντίποδα, στις συστηματικές ανασκαφές, που διεξάγονται χωρίς χρονικές πιέσεις και σε θέσεις που δεν απειλούνται από φυσικές ή ανθρωπογενείς καταστροφές, η ανάκτηση και καταγραφή των αρχαιολογικών δεδομένων γίνεται με συστηματικό τρόπο⁵⁰⁰.

Πριν από την ανασκαφή μιας θέσης είναι αναγκαία η σχεδίαση ενός συνολικού γενικού καννάβου και του γενικού τοπογραφικού σχεδίου, στο οποίο αποτυπώνονται τα όρια της θέσης, τα φυσικά χαρακτηριστικά της, οι πιθανές ανθρωπογενείς παρεμβάσεις, οι υψομετρικές καμπύλες και τα διαφορετικά υψόμετρα πάνω από τη στάθμη της θάλασσας.

⁴⁹⁸ Barker 1993, 100.

⁴⁹⁹ Barker 1993, 71.

⁵⁰⁰ Κουκουζέλη κ.ά. 2003, 148-149.



Εικ. 3.5. Τοπογραφικό σχέδιο της αρχαιολογικής θέσης της Κυμισάλας.
(Ημερολόγια ανασκαφής 2008)

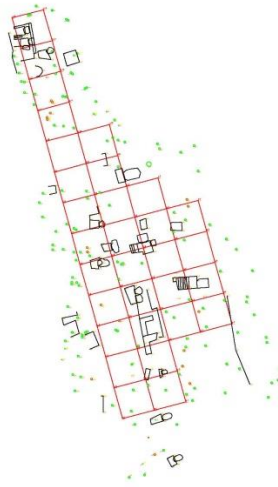
Ο σκελετός οποιουδήποτε συστήματος καταγραφής είναι ο κάρναβος και σχεδιάζεται πριν την εκκίνηση της ανασκαφής. Είναι σημαντικό, επίσης, να συνδέεται η ανασκαφή και ο κάρναβος με μόνιμα χαρακτηριστικά στο τοπίο. Όταν μια ανασκαφή λαμβάνει χώρα σε ένα ριζικά αλλαγμένο τοπίο, χρειάζεται να εντοπίζεται η σχέση του νεότερου με το παλαιότερο τοπίο⁵⁰¹. Ο γενικός κάρναβος μιας θέσης κατασκευάζεται με βάση τη «γραμμή αναφοράς» (datum line), νοητή γραμμή στον άξονα Βοράς-Νότος που ταυτίζεται με ένα μεσημβρινό γεωγραφικού μήκους και τη «γραμμή βάσης» (base line) νοητή γραμμή κάθετη προς τη γραμμή αναφοράς, που ταυτίζεται με έναν παράλληλο γεωγραφικού πλάτους⁵⁰². Ένα σημαντικό φυσικό ή πολιτισμικό χαρακτηριστικό που βρίσκεται σε βολική θέση, μπορεί να χαρακτηριστεί ως σημείο αναφοράς της αρχαιολογικής θέσης, το οποίο μπορεί να είναι μόνιμο ή προσωρινό, ανάλογα με τις απαιτήσεις μιας συγκεκριμένης ανασκαφής⁵⁰³. Ένα επιλεγμένο σημείο του γενικού κάρναβου επί της γραμμής αναφοράς χρησιμεύει ως «σταθερό σημείο 0,00» (datum point) ή σημείο αναφοράς για το σύνολο των μετρήσεων που διεξάγονται κατά την ανασκαφή⁵⁰⁴.

⁵⁰¹ Barker 1993, 71.

⁵⁰² Coles 2014, 112-113.

⁵⁰³ Hester et al 2016, 204.

⁵⁰⁴ Κουκουζέλη κ.ά. 2003, 150.



Εικ. 3.6. Ο κάνναβος της θέσης Σκαλί, Τομέας 1, Κυμισάλα, 2008, στον άξονα Βορράς-Νότος. (Ημερολόγια ανασκαφής, 2008)

Μέθοδοι ανασκαφής

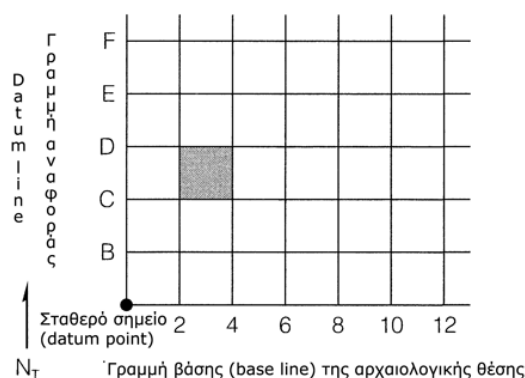
Είναι δύο οι βασικές μέθοδοι ανασκαφής, η κάθετη και η οριζόντια:

Στην κάθετη (Wheeler method, grid system), το ανασκαφικό πεδίο διαιρείται με κάνναβο σε τετράγωνα διαστάσεων 5x5 μ. Μεταξύ των τετραγώνων του καννάβου διατηρούνται κύριοι μάρτυρες (baulks) ομοιόμορφου πλάτους, περίπου 0.50 μ, οπότε τα ανασκαφικά τετράγωνα αποκτούν καθαρές διαστάσεις 4x4 μ. Οι κύριοι μάρτυρες παρέχουν στις τέσσερις όψεις της κάθε τομής τη στρωματογραφική της εικόνα. Στο εσωτερικό των τετραγώνων είναι δυνατόν να χρησιμοποιηθούν δευτερεύοντες μάρτυρες για τον διαρκή έλεγχο της στρωματογραφίας. Επίσης, σε καθορισμένα σημεία των τετραγώνων είναι δυνατόν να ανοιχθούν δοκιμαστικές τομές (trial trenches ή control pits), που παρέχουν ενδείξεις για τη στρωματογραφική εικόνα του τετραγώνου, πριν αφαιρεθούν με την πρόοδο της ανασκαφής⁵⁰⁵.

Τα τετράγωνα του ανασκαφικού καννάβου φέρουν αλφαριθμητικές ενδείξεις (Α,Β,Γ,Δ ή Α,Β,С,Д κ.λπ. 1,2,3,4 κ.λπ. Καταγράφεται, επίσης, το υψόμετρο των πασσάλων από την αφετηρία ή το σταθερό σημείο 0,00, ώστε να μπορούν να χρησιμοποιηθούν για τη μέτρηση της θέσης και του υψομέτρου όλων των αντικειμένων που ανακαλύπτονται στο τετράγωνο. Πλεονέκτημα της

⁵⁰⁵ Κουκουζέλη κ.ά. 2003, 152

συγκεκριμένης μεθόδου θεωρείται η αποκάλυψη της κάθετης σχέσης των στρωμάτων σε μια αρχαιολογική θέση⁵⁰⁶.



Εικ. 3.7. Ο προσανατολισμός του καννάβου είναι ο αληθής Βορράς. Το σκιασμένο τετράγωνο είναι το C2⁵⁰⁷.

Η κάθετη ανασκαφή καθιστά σχεδόν αδύνατη την κατανόηση ακόμη και σχετικά απλών τοποθεσιών, εκτός και αν αφαιρεθούν οι μάρτυρες⁵⁰⁸. Συνεπώς, μειονέκτημα της κάθετης ανασκαφής είναι τα εν δυνάμει ελλιπή ή ακόμη και παραπλανητικά αποτελέσματα, εξαιτίας της παρεμβολής των μαρτύρων, επειδή δεν παρέχεται συνεκτική άποψη των κτισμάτων ή των κατασκευών που αποκαλύπτονται, δεδομένου μάλιστα ότι υφίσταται πιθανότητα οι μάρτυρες να καλύπτουν βασικά χαρακτηριστικά της αρχαιολογικής θέσης. Το πρόβλημα συνήθως επιλύεται με την αφαίρεση διαδρόμων κατά την πρόοδο της ανασκαφής για την αποκάλυψη ιδιαίτερων αρχαιολογικών δεδομένων ή όλων των διαδρόμων στο τελικό στάδιο της ανασκαφής⁵⁰⁹.

Οριζόντια μέθοδος

Η οριζόντια μέθοδος ή μέθοδος ανοικτού χώρου (horizontal or open area method) περιγράφει την ανασκαφή που γίνεται χωρίς μάρτυρες και τομές⁵¹⁰ και χρησιμοποιείται για την αποκάλυψη εκτεταμένων περιοχών μιας αρχαιολογικής θέσης με την ανασκαφή μεγάλων ομοιόμορφων ορθογώνιων τομών, παράλληλων προς τον γενικό κάνναβο της θέσης. Η αρχή πίσω από την

⁵⁰⁶ Hester et al 2016, 124.

⁵⁰⁷ Προσαρμογή από Hester et al 2016, 210.

⁵⁰⁸ Drewett, 1999, 95.

⁵⁰⁹ Κουκουζέλη κ.ά. 2003, 152.

⁵¹⁰ Higginbotham 1985, 8-14.

οριζόντια μέθοδο είναι ιδανικά η αποκάλυψη, καταγραφή και αφαίρεση όλων των αρχαιολογικών αποθέσεων, σε γενικές γραμμές με την αντίστροφη σειρά της εναπόθεσης⁵¹¹. Για τη μελέτη της στρωματογραφίας χρησιμοποιούνται κυρίως οι όψεις της τομής, αλλά σε περίπτωση σύνθετης στρωματογραφίας είναι δυνατόν να χρησιμοποιηθούν δοκιμαστικές τομές ή δευτερεύοντες μάρτυρες⁵¹². Ως μία από τις μεγαλύτερες δυσκολίες σε αυτή τη μέθοδο περιγράφεται η δυσχέρεια συμφιλίωσης της ανάγκης για μια συνεκτική άποψη των ανασκαμμένων επιφανειών με την ανάγκη για την ύπαρξη τομών⁵¹³.

Η υιοθέτηση της μεθόδου σε θέσεις με ρηχή στρωματογραφία είναι φυσική, ωστόσο, για θέσεις με βαθιά διαστρωμάτωση η περίπτωση αλλάζει. Το βασικότερο πλεονέκτημα στη μέθοδο του ανοικτού χώρου είναι η αποκάλυψη της οριζόντιας σχέσης των δεδομένων στον χώρο, χωρίς χωρίς την παρεμβολή μαρτύρων. Ουσιαστικά, εξασφαλίζεται συνεκτικός σχεδιασμός και καταγραφή της περιοχής που ανασκάπτεται. Ωστόσο, και αυτό αφορά ιδιαίτερα σε θέσεις με βαθιά διαστρωμάτωση, αποτελεί μειονέκτημα το γεγονός ότι τα στρώματα δε φαίνονται ποτέ όλα μαζί ταυτοχρόνως και οι κάθετες μεταξύ τους σχέσεις ελέγχονται αργότερα με την ανασύσταση των στρωμάτων στο γραφικά ή ηλεκτρονικά βάσει της τρισδιάστατης αποτύπωσής τους κατά την ανασκαφή⁵¹⁴.

Ειδικές ανασκαφικές τεχνικές

Στην περίπτωση που το πεδίο της ανασκαφής είναι μικροί τεχνητοί λόφοι ή τούμπες, είναι δυνατόν να χρησιμοποιηθεί η μέθοδος των τεταρτημορίων⁵¹⁵. Ο λόφος διαιρείται σε τέσσερα τεταρτημόρια κύκλου με τη χρήση μαρτύρων πλάτους ενός μέτρου ή και περισσότερο. Η ανασκαφή κάθε τεταρτημορίου προχωρά συστηματικά και οι μάρτυρες διατηρούν την εικόνα της διαστρωμάτωσης⁵¹⁶.

⁵¹¹ Drewett 1999, 97.

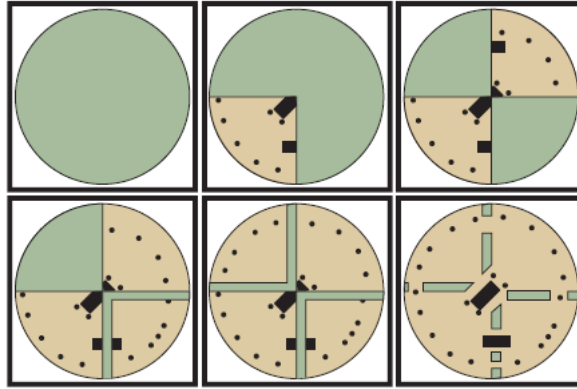
⁵¹² Renfrew and Bahn 2001, 106.

⁵¹³ Barker 1993, 82.

⁵¹⁴ Κουκουζέλη κ.ά. 2003, 155.

⁵¹⁵ Renfrew and Bahn 2001, 108.

⁵¹⁶ Atkinson 1953, 59, Clark 1947, 97 και Wheeler 1954, 95.



Εικ. 3.8. Η μέθοδος τεταρτημορίων σε έξι στάδια⁵¹⁷.

Σε βαθιές ανασκαφές το βάθος των σκαμμάτων είναι δυνατόν να κλιμακώνεται ανά διαστήματα, προκειμένου να διατηρηθεί η σταθερότητα των πλευρικών τοίχων κατά τη διάρκεια της ανασκαφής και να μειωθεί ο κίνδυνος κατάρρευσης. Αναπόφευκτη συνέπεια είναι ότι η ανασκαφή, που ξεκινά σε σχετικά μεγάλη έκταση, περιορίζεται στα βαθύτερα επίπεδα. Η μέθοδος των κλιμακωτών τομών (step trenching), όπως αποκαλείται, προσφέρει τη δυνατότητα αποκάλυψης καθορισμένων πολιτιστικών επιπέδων και ανάδειξη της πλήρους ακολουθίας των γεωλογικών στρωμάτων⁵¹⁸.



Εικ. 3.9. Σκάμμα με τη μέθοδο των κλιμακωτών τομών⁵¹⁹

Η μέθοδος απόξεσης (shovel scraping) εφαρμόζεται όταν αρχαιολογικά δεδομένα –συνήθως μεγάλες κατασκευές- βρίσκονται ακριβώς κάτω από την επιφάνεια του εδάφους. Η απόξεση επιτρέπει την πλήρη ταυτοποίηση των

⁵¹⁷ Renfrew and Bahn 2016, 123.

⁵¹⁸ Step trenching, *Oxford Reference*, διαθέσιμο online.

⁵¹⁹ Morgan, C. 2011, διαθέσιμο online.

χαρακτηριστικών του υπεδάφους⁵²⁰ και συνίσταται στην αφαίρεση του επιφανειακού στρώματος με απόξεση, έως ότου εμφανιστεί η κατασκευή⁵²¹.

Σε μια άλλη προσέγγιση του διαχωρισμού σε τεταρτημόρια, που ονομάζεται μέθοδος της λωρίδας (strip and quartering) και αφορά σε γηλόφους, η ανασκαφή αρχίζει στην άκρη της περιοχής που πρόκειται να ανασκαφεί. Η όψη της απόθεσης αποκαλύπτεται με διαδοχικές παράλληλες λωρίδες, συνήθως πλάτους ενάμισι μέτρου. Με αυτόν τον τρόπο δημιουργείται ένα στρωματογραφικό προφίλ, έτσι ώστε να μπορεί να συντεθεί αργότερα η διαστρωμάτωση ολόκληρης της απόθεσης⁵²².

3.11 Εργαλεία

Τα εργαλεία θεωρούνται γενικά αντικείμενα που χρησιμοποιούνται για να επεκτείνουν την ικανότητα των ανθρώπων να τροποποιούν χαρακτηριστικά του περιβάλλοντός τους⁵²³. Ωστόσο, στη σύγχρονη αρχαιολογία ο όρος εργαλείο χρειάζεται μάλλον να επεκταθεί για να περιλάβει και άλλου είδους διανοητικά εργαλεία, που σχετίζονται με τα συστήματα πληροφορικής και τη χρήση τους στην υπολογιστική αρχαιολογία, αλλά και στο ανασκαφικό πεδίο. Μπορεί, επίσης, να περιλάβει σύγχρονα εργαλεία αποτύπωσης και μετρήσεων. Με άλλα λόγια στη σύγχρονη ανασκαφική έρευνα χρειάζεται πιθανώς μια νέα τυπολογική ταξινόμηση. Σε μια τέτοια ταξινόμηση τα εργαλεία της σύγχρονης ανασκαφικής έρευνας θα μπορούσαν να είναι:

Κλασικά και ηλεκτρονικά εργαλεία μέτρησης: Λαμβάνοντας πάντα υπόψιν ότι οι μονάδες μέτρησης και συνεπώς οι μετρήσεις είναι σχετικές⁵²⁴, οι αρχαιολόγοι για να επιτύχουν τη μέγιστη δυνατή ακρίβεια χρησιμοποιούν κλασικά και ηλεκτρονικά εργαλεία μέτρησης, όπως είναι η μετροταινία, το αλφάδι, το βαρίδι και το νήμα της στάθμης, ή ο χωροβάτης⁵²⁵. Χρησιμοποιούν, επίσης, ηλεκτρονικά όργανα, όπως είναι οι θεοδόλιχοι εξοπλισμένοι με ακτίνες λέιζερ, οι οποίοι συνδέονται με μικρούς ηλεκτρονικούς υπολογιστές στον χώρο

⁵²⁰ Esarey and Pauketat 1992, 15.

⁵²¹ Renfrew and Bahn 2001, 91.

⁵²² Cole 1951, 59.

⁵²³ Feibleman 1967, 329-337.

⁵²⁴ Ramenofsky and Steffen 1998, 3.

⁵²⁵ Renfrew and Bahn 2001, 111

της ανασκαφής. Ωστόσο, έχουν και τη δυνατότητα τριδιάστατων μετρήσεων σε επίπεδο αρχαιολογικής θέσης, σε επίπεδο κτηρίων ή τεχνέργων⁵²⁶.

Μέσης ή μακράς εμβελείας επίγειοι ανιχνευτές λέιζερ (TLS) συνήθως μετρούν τις αποστάσεις υπολογίζοντας τη διάρκεια της πτήσης (ToF) της ανακλώμενης δέσμης λέιζερ⁵²⁷. Σε αντίθεση με το αερομεταφερόμενο LiDAR, οι αισθητήρες TLS είναι σταθεροί κατά τη διάρκεια των μετρήσεων, απαιτώντας συχνή επανατοποθέτηση για να καλύψουν πλήρως μια πολύπλοκη επιφάνεια. Οι επίγειοι ανιχνευτές λέιζερ χρησιμοποιούνται συχνά σε εφαρμογές πολιτιστικής κληρονομιάς, κυρίως όταν εμπλέκονται αρχιτεκτονική ή διατηρημένα έργα τέχνης, όπως ανάγλυφα ή αγάλματα, αλλά και για την τεκμηρίωση εκτεθειμένων επιφανειών σε στρωματογραφικές ανασκαφές⁵²⁸.

Κλασικά και μηχανικά εργαλεία εκσκαφής: Στα κλασικά σκαπτικά εργαλεία και εργαλεία καθαρισμού ή διαλογής, περιλαμβάνονται τα φτυάρια, οι αξίνες, τα σκαλιδάκια, τα μυστριά, τα σκουπάκια, τα τριγωνάκια ακόμη και τα οδοντιατρικά εργαλεία. Όσον αφορά στη διαλογή, χρησιμοποιούνται κόσκινα για την ανακάλυψη μικροσκοπικών τεχνέργων ή ζωικών και φυτικών καταλοίπων⁵²⁹ και μηχανές επίπλευσης (flotation machines)⁵³⁰. Όσον αφορά στα κλασικά σκαπτικά εργαλεία, αναφέρεται ότι ο αριθμός και η ποικιλία των εργαλείων που χρησιμοποιούνται στην αρχαιολογική έρευνα σε όλο τον κόσμο είναι σχεδόν απεριόριστα. Ανάλογα με το είδος και τις συνθήκες της ανασκαφής ακόμη και το ελάχιστο επίπεδο εξοπλισμού, πρέπει απαραίτητως να περιλαμβάνει ποικιλία εργαλείων⁵³¹. Βέβαια, ο τρόπος με τον οποίο γίνεται η ανασκαφή και τα εργαλεία που χρησιμοποιούνται, εξαρτώνται από το είδος της αρχαιολογικής θέσης. Σε θέσεις οι οποίες παρουσιάζουν βαθιά διαστρωμάτωση, τα υψηλότερα στρώματα, που συνήθως στερούνται αρχαιολογικών ευρημάτων, αφαιρούνται με κλασικά σκαπτικά εργαλεία. Στην περίπτωση θέσεων με πολύ βαθιά διαστρωμάτωση, είναι δυνατόν να χρησιμοποιηθούν ακόμη και μικροί εκσκαφείς για την αφαίρεση των επιφανειακών στρωμάτων⁵³².

⁵²⁶ Lambers and Remondino 2008, 27-35.

⁵²⁷ Boehler 2006, 89-99.

⁵²⁸ Doneus and Neubauer 193-203.

⁵²⁹ Renfrew and Bahn 2001, 109

⁵³⁰ Renfrew and Bahn 2001, 249,536

⁵³¹ Hester et al 2016, 69.

⁵³² Harrison 2009, 22.

Κλασικά και ηλεκτρονικά εργαλεία αποτύπωσης και τεκμηρίωσης: Η τήρηση ημερολογίων στο πεδίο παραδοσιακά πραγματοποιείται με χαρτί και μολύβι, αλλά σε κάποιο βαθμό η όλη διαδικασία αντικαθίσταται σταδιακά με τη χρήση υπολογιστών. Οι σημειώσεις πεδίου, οι χάρτες, τα σχέδια πεδίου, αεροφωτογραφίες, τα σκίτσα κ.ο.κ. μπορούν πλέον να αποθηκευτούν σε υπολογιστή πεδίου⁵³³.

Εργαλεία γεωφυσικής και γεωχημικής έρευνας: Το ALS (Airborne Laser Scanning) ή το LiDAR είναι εργαλεία που συνέβαλαν αποφασιστικά στην αρχαιολογική έρευνα. Με τη σειρά τους οι δορυφορικές εικόνες έκαναν δυνατή την εστίαση ακόμη και στις πιο απομακρυσμένες γωνιές του πλανήτη για τον εντοπισμό αρχαιολογικών θέσεων. Η καθολική διαθεσιμότητα του Nasa's World Wind⁵³⁴, ή του Google Earth είναι εξαιρετικό εργαλείο για τους αρχαιολόγους, ακόμα και αν υπάρχουν σημεία επί του παρόντος στην ερμηνεία των δορυφορικών εικόνων που μπορούν να επιλυθούν μόνο στο έδαφος.

Στο επίπεδο της γεωχημικής ανάλυσης είναι πλέον διαθέσιμοι φορητοί μετρητές φασματοσκοπίας φθορισμού ακτίνων Χ (pXRF), που επιτρέπουν την ταχεία καταγραφή δειγμάτων στο πεδίο..

Στατιστικά εργαλεία προανασκαφικής και μεταανασκαφικής ανάλυσης: Τα αρχαιολογικά δεδομένα είναι συχνά αριθμητικά και για αυτό ο αρχαιολόγος χρειάζεται συχνά τα στατιστικά εργαλεία τόσο κατά την προανασκαφική έρευνα, όσο και κατά τη μεταανασκαφική ανάλυση⁵³⁵. Τα στατιστικά εργαλεία είναι χρήσιμα για τη διερεύνηση αρχαιολογικών δεδομένων, καθώς τα δεδομένα προοδευτικά καθίστανται πολυπαραγοντικά και η ανάλυσή τους προσλαμβάνει διαρκώς νέες διαστάσεις, απαιτώντας συνεπώς νέα στατιστικά εργαλεία ποσοτικής ανάλυσης και δειγματοληπτικών μεθόδων.

Σε κάθε περίπτωση η αρχαιολογική έρευνα είναι διεπιστημονική στη φύση της και αυτό σημαίνει πρακτικά ότι χρησιμοποιεί μεγάλο εύρος εργαλείων σε διανοητικό και υλικό επίπεδο, προανασκαφικά και μεταανασκαφικά, Συνεπώς ο κατάλογος των αρχαιολογικών εργαλείων δεν είναι δυνατόν να περιοριστεί στα κλασικά σκαπτικά εργαλεία μόνον, γιατί κάθε εργαλείο εφόσον αξιοποιείται στην

⁵³³ Hester et al 2016, 124.

⁵³⁴ NASA, *WorldWind Java*, διαθέσιμο online.

⁵³⁵ Burger et al 2002, 409-423.

αρχαιολογική έρευνα, είναι «αρχαιολογικό» εργαλείο και πιθανώς χρήζει ιδιαίτερης ανάλυσης στη διεθνή αρχαιολογική βιβλιογραφία.

Κεφάλαιο 4: Μετανασκαφική μεθοδολογία

Συχνά στην αρχαιολογία υφίσταται μια διάκριση μεταξύ εκείνων που ανασκάπτουν και δημοσιεύουν και εκείνων που αναλύουν το υλικό που ανακτήθηκε από τις αρχαιολογικές θέσεις. Οι μετανασκαφικές αναλύσεις πραγματοποιούνται από ειδικευμένους αρχαιολόγους (π.χ. στην κεραμική, στα λίθινα και τη μεταλλουργία), που δεν έχουν εμπλακεί στην αρχική ανασκαφή του τόπου από τον οποίο ανακτήθηκε το υλικό. Σε μια αυστηρά γραμμική άποψη της αρχαιολογικής πρακτικής, το έργο της μετανασκαφικής ανάλυσης θα έπρεπε να είναι κεντρικό. Ωστόσο, λόγω της κατακερματισμένης φύσης της αρχαιολογικής πρακτικής, η μετανασκαφική ανάλυση βρίσκεται συχνά στην περιφέρεια της ερμηνευτικής διαδικασίας⁵³⁶.

Συχνά, ο ειδικευμένος αρχαιολόγος, μετά την ανασκαφή, διαθέτει ελάχιστες ή καθόλου πληροφορίες σχετικά με το πλαίσιο ή τη φύση της αρχαιολογικής θέσης, από την οποία αντλήθηκε το προς ανάλυση υλικό. Μόλις γίνει η ερμηνεία του τεχνέργου ή του οικοδομημένου, βασισμένη στη σύγκριση με άλλα τέχνηρα ή οικοδομημένα, οι πληροφορίες μεταφέρονται στη συνέχεια στον αρχαιολόγο-ανασκαφέα, που τις χρησιμοποιεί για να επιβεβαιώσει ή να διαψεύσει τις αρχικές παρατηρήσεις του. Αυτή η εικόνα της μετανασκαφικής διαδικασίας ευθυγραμμίζεται με την επαγωγική λογική της αντικειμενικής επιστήμης. Η μετανασκαφική ανάλυση, στη συνέχεια, αντιμετωπίζεται ως μια μορφή ανεξάρτητης διαδικασίας κατά την οποία δοκιμάζεται το υλικό και η ίδια η αρχαιολογική θέση. Σύμφωνα με αυτό το φιλοσοφικό μοντέλο, τα τέχνηρα χρησιμοποιούνται απλώς ως μέσα χρονολόγησης της θέσης, ακριβώς με τον ίδιο τρόπο που χρησιμοποιούνται ως χρονολογικοί δείκτες η τεχνική χρονολόγησης ραδιενεργού άνθρακα και η δένδροχρονολόγηση⁵³⁷.

Από την εμφάνιση της Νέας Αρχαιολογίας στη δεκαετία του 1960, έχει αυξηθεί η σημασία της χρήσης επιστημονικών τεχνικών στην αρχαιολογία⁵³⁸. Αυτή η τάση αντικατοπτρίζεται άμεσα στην αυξανόμενη εφαρμογή της επιστημονικής μεθόδου στη μετανασκαφική ανάλυση. Τα βήματα της ανάλυσης προσδιορίζονται από το τι προσπαθεί να βρει ο αρχαιολόγος ή ο εξειδικευμένος

⁵³⁶ Jones, 46.

⁵³⁷ Blinkhorn and Cumberpatch 1997.

⁵³⁸ Rice 1990, 1-2.

επιστήμονας και ποιες τεχνικές είναι δυνατόν να χρησιμοποιηθούν για την παροχή απαντήσεων. Οι τεχνικές που επιλέγονται, εξαρτώνται τελικά από το είδος του προς μελέτη τεχνέργου. Εδώ χρειάζεται να σημειωθεί ότι οι αρχαιολόγοι συχνά αλλάζουν ή προσθέτουν τεχνικές στη διαδικασία ανάλυσης, καθώς οι παρατηρήσεις μπορούν να μεταβάλουν τα αρχικά ερευνητικά ερωτήματα⁵³⁹. Στις περισσότερες περιπτώσεις, τα βασικά βήματα που είναι κρίσιμα για την ανάλυση, όπως ο καθαρισμός και η σήμανση τεχνέργων, εκτελούνται σε γενικό εργαστηριακό περιβάλλον, ενώ πιο εξειδικευμένες τεχνικές εκτελούνται από εξειδικευμένους αρχαιολόγους ή επιστήμονες σε ειδικό εργαστηριακό περιβάλλον⁵⁴⁰.

4.1. Μελέτες κεραμικής

Η κεραμική επιβιώνει σχεδόν σε όλα τα περιβάλλοντα. Παρέχει χρονολογική μαρτυρία, όπως επίσης και μαρτυρία για το ανταλλακτικό εμπόριο και την οικονομία γενικότερα, όπως και για την κοινωνική δυναμική ενός τόπου. Για τη χρωματική κατηγοριοποίηση των οστράκων χρησιμοποιείται το σύστημα της κλίμακας Μάνσελ (Munsell color system). Για την κατηγοριοποίηση των διαφορετικών μεγεθών κόκκου και σκληρότητας χρησιμοποιούνται άλλες βαθμονομημένες κλίμακες, όπως η εμπειρική κλίμακα Μος (Mohs)⁵⁴¹.

Πιθανώς χρειάζεται να επισημανθεί εδώ ότι «η σκληρότητα είναι η φυσική ιδιότητα που συνήθως θεωρείται ότι σχετίζεται με τη στερεότητα της κεραμικής και την ανθεκτικότητα της». Κατά τη δοκιμασία της κλίμακας Mohs το στοιχείο δοκιμής χαράσσεται από μια σειρά ορυκτών αυξανόμενης σκληρότητας. Η τιμή σκληρότητας είναι εκείνη του ορυκτού που επιτυγχάνει πρώτο ευκρινές χάραγμα, διακριτό με μεγεθυντικό φακό⁵⁴². Ο Bronitsky θέτει επιπλέον δύο αντιρρήσεις για τη χρήση της κλίμακας Mohs στην αρχαιολογική κεραμική⁵⁴³. Θεωρεί ότι τα αρχαιολογικά κεραμικά είναι αρκετά ετερογενή ως προς τη σύνθεσή τους και συνεπώς η σκληρότητα ποικίλει ανάλογα με το τμήμα της κεραμικής που διερευνάται. Επίσης, δεν έχουν διεξαχθεί ποτέ δοκιμές

⁵³⁹ Balme and Paterson 2006, 176

⁵⁴⁰ Neumann and Sanford 2001, 186.

⁵⁴¹ Bronitsky 1986, 221.

⁵⁴² Bronitsky 1986, 222-223 και Grimshaw 1971, 866.

⁵⁴³ Bronitsky 1986, 222.

αντικειμενικότητας ανάμεσα στις μετρήσεις διαφορετικών ατόμων, για να προσδιοριστεί ο βαθμός διαφοράς σε μετρήσεις που γίνονται από διαφορετικά άτομα⁵⁴⁴.

Αναμφίβολα, το ζήτημα της αξιοπιστίας πρέπει να αντιμετωπιστεί κατά τη θέσπιση αντικειμενικών διαδικασιών στις δοκιμασίες σκληρότητας της κεραμικής. Ωστόσο, η αρχαιολογική κεραμική που αναφέρεται από τον Bronitsky είναι προϊστορική κεραμική. Η κεραμική της ιστορικής εποχής είναι περισσότερο ομοιογενής ως προς τη σύνθεσή της. Ο Bronitsky θεωρεί ότι οι περισσότερες προϊστορικές κεραμικές κυμαίνονται μεταξύ του 2,5 και του 4 στην κλίμακα Mohs. Συνεπώς, οι τιμές που παράγονται από την κλίμακα είναι μικρής συγκριτικής αξίας. Αυτό, ωστόσο δε συμβαίνει με την κεραμική της ιστορικής εποχής. Οι τιμές της κλίμακας Mohs που λαμβάνονται τόσο από άβαφη κεραμική όσο και από τα υαλώματα αυξάνονται προβλέψιμα. Ωστόσο, όταν λαμβάνονται τιμές σκληρότητας, πρέπει να δίδεται και η ανάλογη προσοχή κατά την ερμηνεία τους, αφού, όπως παρατηρεί ο Grimshaw, πολλοί παράγοντες επηρεάζουν την αντοχή των κεραμικών υλικών⁵⁴⁵. Ανάμεσά τους ξεχωρίζουν οι παρακάτω:

1. η χημική ή ορυκτολογική σύνθεση της πηλοκονίας
2. οι φυσικές ιδιότητες της πηλοκονίας.
3. ο τρόπος παρασκευής της πηλοκονίας.
4. ο τρόπος κατασκευής του τέχνηργου.
5. οι συνθήκες ξήρανσης.
6. οι συνθήκες όπτησης.
7. η θερμοκρασία στην οποία χρησιμοποιείται το αγγείο ή το τέχνηργο.
8. άλλες συνθήκες στις οποίες υποβάλλεται ή έχει υποβληθεί το τέχνηργο, συμπεριλαμβανομένων των καιρικών συνθηκών, ξαφνικών αλλαγών θερμοκρασίας, παρατεταμένης θέρμανσης κ.λπ.⁵⁴⁶

⁵⁴⁴ Rye 1981, 121.

⁵⁴⁵ Grimshaw 1971, 871.

⁵⁴⁶ Majewski and O'Brien 1987, 114.

Σκληρότητα Mohs	Ορυκτό	Χημικός τύπος	Απόλυτη σκληρότητα
1	Τάλκης	$Mg_3Si_4O_{10}(OH)_2$	1
2	Γύψος	$CaSO_4 \cdot 2H_2O$	2
3	Ασβεστίτης	$CaCO_3$	14
4	Φθορίτης	CaF_2	21
5	Απατίτης	$Ca_5(PO_4)_3(OH^-, Cl^-, F^-)$	48
6	Ορθόκλαστο	$KAlSi_3O_8$	72
7	Χαλαζίας	SiO_2	100
8	Τοπάζι	$Al_2SiO_4(OH^-, F^-)_2$	200
9	Κορούνδιο	Al_2O_3	400
10	Διαμάντι	C	1500

Πιν. 1. Η κλίμακα σκληρότητας Mosh⁵⁴⁷.

Από την κεραμεική αντλούνται τόσο πληροφορίες για τη διαδικασία κατασκευής διαφορετικών εργαστηρίων, όσο και χωρικές πληροφορίες για την ευρύτερη γεωγραφική κατανομή εργαστηρίων. Λιθολογικές μελέτες των χαρακτηριστικών των πετρωμάτων που χρησιμοποιούνται συχνά για την υάλωση αποδίδουν πληροφορίες για ιδιαίτερους κατασκευαστές ή γεωγραφικές περιοχές, όπως επίσης και ποσοτικές πληροφορίες για τον αριθμό ή το βάρος των οστράκων, ανάλογα με τη μέθοδο ποσοτικής ανάλυσης της κεραμεικής που ακολουθείται για μια ιδιαίτερη αρχαιολογική θέση⁵⁴⁸.

Σε ορισμένες περιπτώσεις γίνεται χρήση ποσοτικών μεθόδων, που χρησιμοποιούνται στην παλυνολογική ανάλυση. Αντί για μετρήσεις του ελάχιστου αριθμού ατόμων, στην κεραμεική ανάλυση (συμπερίληψη οστράκων και ακέραιων αγγείων) γίνονται μετρήσεις του ελάχιστου αριθμού αγγείων (ακέραια αγγεία και αντιπροσωπευτικά τμήματα κεραμεικής που δείχνουν προς την κατεύθυνση συγκεκριμένου αγγείου). Ο συγκεκριμένος τύπος ανάλυσης χρησιμοποιεί τον αριθμό των αντιπροσωπευτικών τμημάτων κεραμεικής ως

⁵⁴⁷ Mukherjee 2012, 373.

⁵⁴⁸ Grant *et al* 2005, 67.

υποκατάστατο του αριθμού ακέραιων αγγείων σε μια ομάδα. Αν και η συγκεκριμένη μέθοδος είναι ενίοτε προβληματική, παρέχει καλό μέτρο αναλογίας και κατανομής τεχνέργων σε μια δεδομένη αρχαιολογική θέση⁵⁴⁹.

4.2. Ανάλυση λίθινων εργαλείων

Τα λίθινα εργαλεία αποτελούν συχνά αντικείμενο αρχαιολογικής ανάλυσης, καθώς παρουσιάζουν εξαιρετικές δυνατότητες διατήρησης και είναι συχνά τα πλέον πολυάριθμα τέχνηρα σε πρώιμες προϊστορικές θέσεις⁵⁵⁰. Σε ορισμένες αρχαιολογικές θέσεις, μάλιστα, τα λίθινα εργαλεία είναι η μοναδική μαρτυρία ανθρώπινης δραστηριότητας⁵⁵¹. Η κατηγορία των λίθινων εργαλείων πέτρας δεν περιλαμβάνει μόνον τελικά εργαλεία, αλλά και πυρήνες και αποκρούσματα. Η τεχνική που χρησιμοποιείται για την ανάλυση πέτρινων εργαλείων είναι η κατηγοριοποίηση. Η κατηγοριοποίηση οργανώνει τις παρατηρήσεις σε «ένα περιορισμένο σύνολο ομάδων που είναι όμοιες με έναν καθορισμένο τρόπο»⁵⁵². Η κατηγοριοποίηση είναι δυνατόν να επιτευχθεί με:

1. τη χρήση ενός συνόλου μεταβλητών για την κατανομή τεχνέργων σε κατηγορίες ή

2. τη χρήση στατιστικών στοιχείων και δεδομένων των τεχνέργων για την εύρεση φυσικών ομάδων στο σύνολο των τεχνέργων⁵⁵³

Οι παράγοντες που μετρούνται περιλαμβάνουν το μέγεθος, το σχήμα, το χρώμα, την πρώτη ύλη και την τεχνολογική ή τυπολογική κατηγορία⁵⁵⁴. Κατά την ανάλυση οι αρχαιολόγοι χρειάζεται συχνά να λαμβάνουν αποφάσεις σχετικά με τον τρόπο μέτρησης αυτών των παραγόντων, προκειμένου να επιτευχθεί το υψηλότερο δυνατό επίπεδο αντικειμενικότητας.

Οι επιφάνειες των λίθινων εργαλείων γίνονται συχνά αντικείμενο ιδιαίτερης προσοχής. Η εξέταση των επιφανειών παρέχει ενδείξεις για το πώς κατασκευάστηκαν τα εργαλεία. Οι τυπικές τεχνικές σχεδιασμού περιλαμβάνουν

⁵⁴⁹ Balme and Paterson 2006, 380.

⁵⁵⁰ Grant et al. 2005, 68.

⁵⁵¹ Στο ίδιο.

⁵⁵² Balme and Paterson 2006, 176.

⁵⁵³ Balme and Paterson 2006, 179.

⁵⁵⁴ Grant et al. 2005, 68, επίσης Balme and Paterson 2006, 184.

θραύση, άμεση ανελαστική κρούση⁵⁵⁵, έμμεση κρούση ή λείανση. Ενίοτε, τα λίθινα εργαλεία έχουν υποστεί διαδοχικές τροποήσεις και η προσοχή στις επιφάνειες είναι απαραίτητη για την αναγνώριση κάθε σταδίου της διαδικασίας κατασκευής⁵⁵⁶.

4.3. Μεταλλουργική ανάλυση

Πριν αρχίσει η ανάλυση, τα μεταλλικά τέχνηρα απαιτούν εντατικό καθαρισμό με εξειδικευμένες μεθόδους καθαρισμού. Για τον καθαρισμό χρησιμοποιείται η ηλεκτρόλυση για την αποφυγή φθοράς. Για παράδειγμα, τα μέταλλα από ναυάγια εγκλωβίζουν συμπήγματα κάτω από επιστρώσεις. Τα ορυκτά του θαλάσσιου ύδατος αντιδρούν με τα μέταλλα και τα περιβάλλοντα ιζήματα για να σχηματίσουν μια πυκνή κρούστα γύρω από το μέταλλο. Τα ορυκτά, τα αντικείμενα που περιβάλλουν και οι τεχνικές συντήρησης καταγράφονται με φωτογραφίες και ακτίνες Χ⁵⁵⁷.

Μόλις καθαριστούν τα μέταλλα, οι ειδικοί επιστήμονες χρησιμοποιούν μικροσκόπια για να εξετάσουν λεπτομέρειες που αποκαλύπτουν πληροφορίες σχετικά με τη σύνθεση και τις τεχνικές κατασκευής. Για παράδειγμα, είναι δυνατόν να αναγνωρισθεί το σχήμα του τεχνέργου, υφιστάμενες ρωγμές και τα σημεία συναρμολόγησης. Οι κύριοι παράγοντες που επηρεάζουν τη διάβρωση μετάλλων στα ναυάγια είναι η σύνθεση των μετάλλων, η σύνθεση του νερού, η θερμοκρασία, η θαλάσσια ζωή, η σύνθεση του πυθμένα, η θέση των αντικειμένων σε σχέση με άλλα αντικείμενα ή υλικά του ναυαγίου, το βάθος ταφής κάτω από τον πυθμένα και η έκταση της κίνησης του νερού⁵⁵⁸.

Η μεταλλογραφία εξετάζει το μέγεθος και το σχήμα των κόκκων ορυκτών στα υλικά για ίχνη θέρμανσης, επεξεργασίας και κράματος. Εδώ χρησιμοποιούνται ηλεκτρονικά μικροσκόπια σάρωσης για τη διερεύνηση των τεχνικών κατασκευής κοσμημάτων και όπλων, καθώς επιτρέπουν την ταυτοποίηση λεπτομερειών, όπως η σφυρηλάτηση αναδιπλούμενων στρώσεων μετάλλου για τη δημιουργία ενός σπαθιού. Επιπλέον, η αναγνώριση

⁵⁵⁵ Ματζάνας 2001, 60.

⁵⁵⁶ Hamilton 2011, 1.

⁵⁵⁷ Grant et al 2005, 69.

⁵⁵⁸ North and MacLeod 1987, 68-98.

αποτυπωμάτων των εργαλείων που χρησιμοποιήθηκαν για να δημιουργηθεί το τέχνηργο, εμπλουτίζει τις μελέτες για τις τεχνικές κατασκευής. Άλλη τεχνική που χρησιμοποιείται για την αναγνώριση κραμάτων χρυσού, χαλκού και χαλκού είναι η φασματοσκοπία ατομικής απορρόφησης. Η συγκεκριμένη τεχνική δεν είναι ιδιαίτερα αποτελεσματική, αν το τέχνηργο περιέχει πολλούς τύπους μετάλλων⁵⁵⁹.

4.4. Σκελετικά κατάλοιπα

Τα σκελετικά κατάλοιπα αποδίδουν πληροφορίες για το φύλο, την ηλικία του υποκειμένου κατά τον θάνατό του και το ανάστημά του. Ωστόσο, υπάρχουν διαφορετικές διαδικασίες για την ανάλυση αυτών των παραγόντων όταν ασχολούμαστε με ενήλικα και ανήλικα υποκείμενα⁵⁶⁰.

Όσον αφορά στην ταυτοποίηση του φύλου, το φύλο των ανηλίκων μπορεί να μετρηθεί, μέσω της σύγκρισης του σταδίου ασβεστοποίησης στα δόντια με την ωρίμανση του μετακράνιου σκελετού. Οι μετακράνιοι σκελετοί ωριμάζουν πιο αργά στα αρσενικά από ό,τι στα θηλυκά, ενώ ο ρυθμός ασβεστοποίησης στα δόντια είναι περίπου ο ίδιος και για τα δύο φύλα. Εάν η οδοντιατρική και η μετακρανιακή ανάπτυξη είναι παρόμοια, ο σκελετός ανήκει πιθανώς σε αρσενικό, διαφορετικά ανήκει πιθανώς σε θηλυκό υποκείμενο⁵⁶¹. Στους ενήλικες, το φύλο καθορίζεται από την εξέταση της λεκάνης, της ισχιακής εγκοπής, της ωοθυλακικής περιοχής, της κοτύλης, των οστών, των μακρών οστών και του κρανίου⁵⁶².

Η εκτίμηση της ηλικίας θανάτου των ενηλίκων συνεπάγεται την παρατήρηση των μορφολογικών χαρακτηριστικών των σκελετικών καταλοίπων και τη σύγκριση των πληροφοριών με καταγραμμένες αλλαγές για πρόσφατους πληθυσμούς γνωστής ηλικίας⁵⁶³. Για τους ανηλικούς καταγράφεται η οδοντική ανάπτυξη, το μήκος των μακρών οστών και η σύγκλιση των επιφύσεων για την εκτίμηση της ηλικίας. Όσον αφορά στους ενήλικες, η ηλικία μετράται με μακροσκοπικές και μικροσκοπικές μεθόδους. Οι μακροσκοπικές μέθοδοι δεν περιλαμβάνουν καταστροφή του δείγματος, ενώ οι μικροσκοπικές μέθοδοι

⁵⁵⁹ Ubelaker 2008, 52.

⁵⁶⁰ Lundy 1998, 423-427.

⁵⁶¹ Ubelaker 2008, 53.

⁵⁶² Ubelaker 2008, 60.

⁵⁶³ Ubelaker 2008, 63.

απαιτούν περισσότερο χρόνο και απαιτούν εξοπλισμό, σχετική καταστροφή του σκελετικού υλικού και εξειδικευμένες γνώσεις. Παρά τα μειονεκτήματα, οι μικροσκοπικές μέθοδοι δίνουν ακριβέστερα αποτελέσματα⁵⁶⁴.

Η μέτρηση του μήκους σχετικών οστών και η σύγκρισή τους με αριθμούς, που προκύπτουν από ιστορικές μελέτες μπορεί οδηγήσει στην εκτίμηση του ύψους του σκελετού⁵⁶⁵. Το ανάστημα εκτιμάται συνήθως με τη χρήση είτε της ανατομικής είτε της μαθηματικής μεθόδου. Η ανατομική μέθοδος υπολογίζει το συνολικό ύψος του σκελετού και αρχικά εισήχθη από τον Dwight το 1894⁵⁶⁶. Το 1956 ο Fully επανέφερε τη μέθοδο με μικρές διακυμάνσεις και έγινε γνωστή ως διαδικασία Fully⁵⁶⁷. Αυτή η μέθοδος βασίζεται στα αθροιστικά ύψη των σκελετικών στοιχείων που συμβάλλουν στο ανάστημα στους ανθρώπους. Τα σκελετικά στοιχεία που μετρούνται σε αυτή τη μέθοδο είναι το κρανίο, οι σπόνδυλοι, το μηριαίο οστό, η κνήμη, ο αστράγαλος και το πέλμα, και οι μετρήσεις τους αθροίζονται για τον υπολογισμό του συνολικού ύψους σκελετού. Για τον υπολογισμό του αναστήματος ζώντος προσώπου με την ανατομική μέθοδο, πρέπει επίσης να προστεθούν συντελεστές διόρθωσης, που αντισταθμίζουν τις διαφορές που παράγουν οι μαλακοί ιστοί⁵⁶⁸. Το κύριο μειονέκτημα της ανατομικής μεθόδου είναι ότι απαιτείται σχεδόν πλήρης σκελετός για την εκτίμηση του αναστήματος.

Η μαθηματική μέθοδος, από την άλλη, χρησιμοποιεί ένα ή περισσότερα μήκη οστών για να εκτιμήσει το ανάστημα του ατόμου. Αυτή η μέθοδος χρησιμοποιεί πίνακες μήκους και αναστήματος των οστών και μαθηματικούς τύπους παλινδρόμησης, για να υπολογίσει το συνολικό ύψος του σκελετού ή το ζωντανό ανάστημα από τα μακρά μήκη των οστών. Οι τύποι παλινδρόμησης που προέρχονται από τα μεγάλα μακρά οστά θεωρούνται γενικά ακριβέστεροι από εκείνους που χρησιμοποιούν άλλα οστά όπως το κρανίο⁵⁶⁹ ή τα οστά των χεριών και των ποδιών⁵⁷⁰. Τα μακρά οστά που αποτελούν τη μεγαλύτερη αναλογία σώματος, το μηρό και η κνήμη, είναι επίσης ακριβέστερες από το βραχιόνιο και

⁵⁶⁴ Ubelaker 2008, 60.

⁵⁶⁵ Grant, et al. 2005, 80

⁵⁶⁶ Lundy 1985, 73-76.

⁵⁶⁷ Fully 1956, 266-273.

⁵⁶⁸ Lundy 1983, 337-338.

⁵⁶⁹ Introna et al 1993, 153-160. Επίσης, Chiba, and Terazawa 1998, 87-92.

⁵⁷⁰ Musgrave and Harneja 1978, 113-119. Επίσης, Meadows and Jantz 1992, 147-154, Holland 1995, 315-320 και Byers et al 1989, 275-279.

την ωλένη⁵⁷¹. Άλλα σκελετικά στοιχεία που χρησιμοποιήθηκαν για την εκτίμηση του αναστήματος περιλαμβάνουν την κλείδα⁵⁷².

4.5. Αναλύσεις πανίδας

Οι μελέτες πανίδας προσφέρουν πληροφορίες για τους ζωικούς οργανισμούς που αποτελούν μέρος του οικοσυστήματος μιας περιοχής αλλά και έμμεσες ενδείξεις για τη βλάστηση και το κλίμα. Η ανάλυση βασίζεται κυρίως στη μελέτη ασπόνδυλων οργανισμών⁵⁷³. Τα σπονδυλωτά παίζουν μεγαλύτερο ρόλο στην παλαιοοικονομική έρευνα κυρίως λόγω της σημασίας των εξημερωμένων ζώων για τον άνθρωπο. Ωστόσο, κατάλοιπα ψαριών, πτηνών και μικρών ή μεγάλων άγριων θηλαστικών συνιστούν στοιχεία του φυσικού περιβάλλοντος και επίσης αποτελούν αντικείμενα της περιβαλλοντικής έρευνας⁵⁷⁴.

Στα πανιδικά κατάλοιπα συγκαταλέγονται θεωρούνται ότι περιλαμβάνουν τόσο ψάρια, πουλιά όσο και θηλαστικά. Τα κατάλοιπα της πανίδας χρησιμοποιούνται για την ανασυγκρότηση παρελθόντων περιβαλλόντων και για τον προσδιορισμό του τρόπου με τον οποίο επηρέασαν τα ζώα τις ανθρώπινες οικονομίες. Η μελέτη των αρχαίων ζωικών υπολειμμάτων αναφέρεται ως ζωοαρχαιολογία⁵⁷⁵. Μόλις συλλεχθούν τα οστά, καθαριστούν και επισημανθούν, οι ειδικοί αρχίζουν να προσδιορίζουν τον τύπο του οστού και από ποια είδη προήλθε. Υπολογίζεται ο αριθμός των αναγνωρισμένων οστών, καθώς και το βάρος κάθε δείγματος και ο ελάχιστος αριθμός ατόμων. Η ηλικία και το φύλο ενός ζώου μπορούν να χρησιμοποιηθούν για τον προσδιορισμό δεδομένων σχετικά με το κυνήγι και τη γεωργία. Για απολιθωμένα δείγματα χρησιμοποιείται η βιοστρωματογραφία για τη χρονολόγηση των στρωμάτων και κατ' επέκταση των αρχαιολογικών θέσεων. Τα πανιδικά κατάλοιπα παρέχουν επίσης πληροφορίες για την ανθρώπινη συμπεριφορά, το εμπόριο και την ανθρώπινη μετανάστευση⁵⁷⁶.

⁵⁷¹ Lundy and Feldesman 1987, 54–55.

⁵⁷² Jit and Singh 1956, 137–155

⁵⁷³ Renfrew and Bahn 2001, 256

⁵⁷⁴ Renfrew and Bahn 2001, 258–259

⁵⁷⁵ Sutton and Yohe II 2008, 248–264.

⁵⁷⁶ Weiner 2010, 157.

Ειδικευμένοι στην ανάλυση της πανίδας επιστήμονες και ανθρωπολόγοι χρησιμοποιούν συλλογές σκελετών ως συγκριτικό υλικό, ενώ χρησιμοποιούν επίσης και άλλες μορφές πληροφοριών, συμπεριλαμβανομένων των οικολογικών και μορφολογικών δεδομένων για τα ζωικά είδη, καθώς και λεπτομερέστερα οστεολογικά δεδομένα που προέρχεται από την ανάλυση των ζωολογικών δεδομένων⁵⁷⁷.

Η ανάλυση των ζωοαρχαιολογικών δεδομένων, παρέχει επιπλέον πληροφορίες για τα είδη ζώων και τις ανάγκες που κάλυπταν, για τις κυνηγετικές πρακτικές και τις παρκτικές εκτροφής, την ηλικία και τον τρόπο σφαγής, τα ποσοστά αρσενικών-θηλυκών ζώων, καθώς επίσης για την υγεία των ζώων και τα δευτερεύοντα προϊόντα τους⁵⁷⁸. Πληροφορίες για επιλεκτικές πρακτικές ως προς την ηλικία, το είδος ή το φύλο των ζώων που θανατώνονταν με στόχο τη βελτίωση απόδοσης κρέατος, την αύξηση της παραγωγής μαλλιού κ.λπ. καθώς και αλλαγές από περιοχή σε περιοχή και από εποχή σε εποχή προσφέρουν στοιχεία για τη μελέτη των συστημάτων κτηνοτροφίας και τις πρακτικές εκμετάλλευσης των ζώων, για τις χρήσεις γης και το αγροτικό σύστημα, αλλά και ευρύτερες πληροφορίες για τις κοινωνικές σχέσεις και την ανθρώπινη συμπεριφορά⁵⁷⁹

4.6. Αναλύσεις μαλακίων και ασπονδύλων

Τα ασπόνδυλα μπορούν να προσφέρουν ενδείξεις για το τοπικό περιβάλλον και την ανθρώπινη δραστηριότητα. Βρίσκονται στα περισσότερα περιβάλλοντα και συχνά ομαδοποιούνται βάσει ενδαιιτήματος ή τροφής. Μέσω της χρήσης τους αντλούνται πληροφορίες για τις συνθήκες επιφάνειας του εδάφους, τη βλάστηση και το κλίμα, καθώς και για τα πιθανά αποθηκευμένα προϊόντα και τη χρήση των φυτών. Χερσαία όστρεα, όστρεα του γλυκού νερού, τα θαλάσσια όστρεα και τα θαλάσσια μαλάκια μπορούν επίσης να χρησιμεύσουν ως δείκτες για την κατανάλωση τροφίμων, την κατασκευή και την παραγωγή ασβέστου και βαφής⁵⁸⁰. Το μέγεθός τους ποικίλει και ομαδοποιούνται σε τρεις

⁵⁷⁷ Rackham 1987, 51.

⁵⁷⁸ Renfrew and Bahn 2001, 257, 295-296

⁵⁷⁹ Davis, 1987

⁵⁸⁰ Grant et al. 2005, 88.

ευρείες κατηγορίες⁵⁸¹. Η παρουσία χερσαίων οστρέων σε μια περιοχή είναι πιθανό να υποδεικνύει την ανθρώπινη κατανάλωση, τη δραστηριότητα των τρωκτικών, τις περιβαλλοντικές συνθήκες ή την περισυλλογή από ανθρώπους, λόγω των ιδιαίτερων χαρακτηριστικών τους⁵⁸².

Η παραδοσιακή ανασυγκρότηση του παλαιοπεριβάλλοντος και των ανθρώπινων ενδιαυμάτων βασίζεται στις οικολογικές και περιβαλλοντικές απαιτήσεις των διαφόρων ειδών των μαλακίων, που υποστηρίζονται από τη σχετική ή απόλυτη αφθονία κάθε ταξινομικής κατηγορίας, καθώς και την ποικιλομορφία των ταξινομικών κατηγοριών. Οι μεταβολές στη γεωγραφική κατανομή των ειδών με την πάροδο του χρόνου είναι πιθανό να υποδηλώνουν κλιματικές ή άλλες περιβαλλοντικές μεταβολές, ενώ οι αλλαγές στη σχετική αντιπροσώπευση των ειδών σε μία ομάδα, θα μπορούσαν να υποδηλώνουν τοπικές περιβαλλοντικές μεταβολές ή, εάν τα όστρεα ανήκουν σε βρώσιμα είδη, ανθρώπινη μετατόπιση χάριν περισυλλογής⁵⁸³

Ανεξάρτητα από τον τύπο τους, όλα τα όστρεα μαλακίων συλλέγονται με τυποποιημένη στρωματογραφική στρατηγική δειγματοληψίας⁵⁸⁴. Χρησιμοποιώντας αυτόν τον τύπο στρατηγικής αποφεύγονται τα προβλήματα που δημιουργούνται όταν αγνοούνται μικρότερα όστρεα, ζήτημα που μπορεί να προκύψει από τη χειροκίνητη περισυλλογή⁵⁸⁵. Μόλις συλλεχθούν, τα δείγματα, αποστέλλονται στο εργαστήριο για να στεγνώσουν. Στη συνέχεια συλλέγεται τυπικό βάρος για κάθε δείγμα. Κάθε τυποποιημένο δείγμα τοποθετείται κατόπιν σε μικρή λεκάνη (με ετικέτες για τις στρωματογραφικές πληροφορίες) και καλύπτεται με διάλυμα 70% ζεστού νερού και 30% υπεροξειδίου του υδρογόνου. Τα όστρεα επιπλέουν και απομακρύνονται με ένα σύνολο κοσκίνων, τα οποία τα διαχωρίζουν ανά μέγεθος. Τόσο το χώμα όσο και τα κελύφη τοποθετούνται στη συνέχεια στον κλίβανο ξήρανσης. Μετά την ψύξη, τα κελύφη είναι έτοιμα για ανάλυση και μπορούν να εξαχθούν από τα κόσκινα⁵⁸⁶.

Αναλύοντας τα όστρεα, οι αρχαιολόγοι επικεντρώνονται σε πολλούς παράγοντες, όπως: ταξινομία, ανόργανη σύνθεση του κελύφους και οργανική

⁵⁸¹ Weiner 2010, 158.

⁵⁸² Davies 2008, 1.

⁵⁸³ Thomas 2015, 159-167.

⁵⁸⁴ Davies 2008, 1.

⁵⁸⁵ Davies 2008, 5.

⁵⁸⁶ Weiner 2010, 159-162.

ύλη που παραμένει στο κέλυφος. Για παράδειγμα, η παρουσία ανθρακικών ορυκτών δείχνει ότι το pH του ιζήματος σε μια τοποθεσία ήταν πάντα πάνω από 8⁵⁸⁷. Οι μετρήσεις του pH μπορούν στη συνέχεια να χρησιμοποιηθούν για την ερμηνεία των περιβαλλοντικών συνθηκών ενός συγκεκριμένου τόπου πριν, κατά τη διάρκεια και μετά την κατοχή του και φυσικά για τη γνώση των συνθηκών διατήρησης ή διάβρωσης του αρχαιολογικού υλικού⁵⁸⁸.

Πρόσφατες έρευνες πεδίου υποδεικνύουν ότι οι αρχαιολογικές θέσεις με αφθονία οστρέων κατανέμονται ευρέως και άφθονα σε ολόκληρο τον κόσμο και από διαφορετικές χρονικές περιόδους. Προκειται για ένα εντυπωσιακό και διαρκώς αυξανόμενο αρχείο, όχι μόνο των πτυχών της παρελθούσας ανθρώπινης συμπεριφοράς, αλλά και των περιβαλλοντικών αλλαγών του παρελθόντος⁵⁸⁹.

4.7. Ανάλυση χλωρίδας

Οι οικολογικές μελέτες απασχόλησαν ιδιαίτερα τη διαδικαστική αρχαιολογία. Τα επιμέρους αντικείμενά τους είναι η διατροφή, ο ρόλος της γεωργίας και της κτηνοτροφίας, ο ρόλος των οικόσιτων ζώων, οι διαθέσιμοι τροφικοί πόροι, τα συστήματα καλλιέργειας και χρήσης της γης ανάλογα με το έδαφος και το κλίμα, η επίδραση της φύσης στον άνθρωπο και, αντίστροφα, οι συνέπειες της ανθρώπινης δράσης, ως παράγοντα μεταβολής, στο φυσικό περιβάλλον. Αντικείμενο έρευνας αποτελούν επίσης οι απαρχές της γεωργίας και η εξημέρωση φυτών και ζώων, ενώ σε σχέση με τις αγροτικές κοινωνίες έμφαση δίνεται στη μελέτη των σχέσεων των ανθρώπων με τα φυτά και τα ζώα καθώς και στη μελέτη της ανθρώπινης διαίτας διαμέσου της εξέτασης κυρίως ζωοαρχαιολογικών και αρχαιοβοτανικών δεδομένων. Η παλαιοβοτανική, ειδικότερα, όχι μόνο συνέβαλε ουσιαστικά σε ένα ευρύ φάσμα αρχαιολογικών ζητημάτων, αλλά και συνέταξε λεπτομερείς οδηγούς και περιλήψεις σύγχρονων τεχνικών ανάκτησης και εργαστηριακών αναλύσεων⁵⁹⁰.

Ειδικότερα η παλαιοβοτανική ανάλυση έχει αντικείμενο την ανάλυση των αρχαιοβοτανικών δεδομένων, δηλαδή των απανθρακωμένων σπόρων και των

⁵⁸⁷ Weiner 2010, 160.

⁵⁸⁸ Grant et al. 2005, 82.

⁵⁸⁹ Andrus 2011, 2892-2905.

⁵⁹⁰ Wright 2009, 37-64.

άλλων υπολειμμάτων φυτών όπως το ξύλο, που συλλέγονται προσεκτικά κατά την ανασκαφή με διάφορες μεθόδους νεροκοσκινίσματος των χωμάτων⁵⁹¹. Η ανάλυση αυτή δίνει πληροφορίες σχετικά με τη σημασία κάθε είδους φυτού καθώς και την αποθήκευση και κατανάλωσή του, την ανθρώπινη δίαιτα, τις απαρχές της εξημέρωσης και την εισαγωγή διαφόρων ειδών φυτών, το σύστημα καλλιέργειας, τα ζιζάνια, και, σε συνδυασμό με άλλα αρχαιολογικά δεδομένα, πληροφορίες σχετικά με τις τεχνολογίες άρδευσης, τις τεχνικές προστασίας της γης κ.ά

Τα φυτικά υπολείμματα μετατοπίζονται διακρώς από αυτό που ο Schiffer χαρακτήρισε ως συστημικό πλαίσιο⁵⁹² στο αρχαιολογικό πλαίσιο ως αποτέλεσμα της απόρριψης, της εγκατάλειψης ή της απώλειας. Η απόρριψη περιλαμβάνει στοιχεία που απορρίφθηκαν επειδή κρίθηκαν άχρηστα ή δυσάρεστα. Η τέφρα μιας εστίας που απορρίπτεται είναι ένα παράδειγμα. Η εγκατάλειψη, ωστόσο, όπως και η απόρριψη, είναι μια σκόπιμη ενέργεια. Η απώλεια όμως είναι απρόβλεπτο αποτέλεσμα της ανθρώπινης συμπεριφοράς. Τα μακροβοτανικά υπολείμματα μπορούν, επίσης, να ενσωματωθούν στο αρχαιολογικό πλαίσιο από δραστηριότητες ή υποπροϊόντα ανθρώπινων δραστηριοτήτων, όπως η καύση κοπριάς ως μηχανισμός απανθράκωσης σπόρων⁵⁹³.

4.8. Παλαιοοικολογικές μελέτες

Στόχος της παλαιοοικολογικής μελέτης είναι η ανασύσταση της οικολογίας μιας περιοχής κατά το παρελθόν βάσει διαφόρων ειδών, όπως απολιθώματα φυτών ορατά με γυμνό μάτι, υπολείμματα ξύλου και κάρβουνου, φυτικά μικροαπολιθώματα που αναγνωρίζονται στο μικροσκόπιο (φυτόλιθοι, σπόρια, γύρη).

⁵⁹¹ Renfrew and Bahn 2001, 248-249.

⁵⁹² Schiffer 1987, 47.

⁵⁹³ Miller and Smart 1984, 15-28.

Μακροβοτανικά κατάλοιπα

Τα μακροβοτανικά κατάλοιπα παρέχουν πληροφορίες σχετικά με το παρελθόν κλίμα, τις οικονομικές πρακτικές και τις αλλαγές στο περιβάλλον⁵⁹⁴. Είναι ορατά με γυμνό μάτι και διατηρούνται συνήθως με τις ακόλουθες συνθήκες:

1. Υδάτινα περιβάλλοντα
2. Απανθράκωση
3. Επασβέστωση
4. Ψύξη
5. Αποτύπωση σε πλίνθους ή κεραμικά⁵⁹⁵.

Η ανάλυση μιας μακροβοτανικής ομάδας γίνεται σε διάφορα βήματα. Αρχικά, τα υπολείμματα διαχωρίζονται από το έδαφος με μια διαδικασία γνωστή ως επίπλευση⁵⁹⁶. Τα απολιθωμένα υπολείμματα και τα υπολείμματα από εξαιρετικά ξηρά περιβάλλοντα μπορούν συνήθως να διαχωριστούν από το έδαφος και τις ρίζες μόνο με προσεκτικό ξηρό κοσκίνισμα. Παρόλο που τα συστήματα επίπλευσης διαφέρουν ως προς το μέγεθος και τον σχεδιασμό, προκειμένου να ικανοποιήσουν τις ανάγκες μιας συγκεκριμένης ομάδας καταλοίπων, κάθε σύστημα επιτυγχάνει το ίδιο βασικό έργο⁵⁹⁷.

Η έγχυση νερού σε ένα δείγμα εδάφους διαχωρίζει τα κατάλοιπα σε ελαφρά και βαρέα κλάσματα. Το βαρύ κλάσμα βυθίζεται στον πυθμένα και συλλέγεται. Το ελαφρύ κλάσμα επιπλέει και τοποθετείται σε λεπτότερο κόσκινο. Τα υπολείμματα αποστραγγίζονται από τη μονάδα επίπλευσης μέσω βαλβίδας εξόδου. Τα ελαφρά και βαρέα κλάσματα στη συνέχεια ξηραίνονται και προετοιμάζονται για ανάλυση⁵⁹⁸. Τυπικά, τα ελαφρά κλάσματα ταξινομούνται με τη χρήση μικροσκοπίου. Τέλος, η ανάλυση ολοκληρώνεται με οργάνωση δεδομένων και ανάλυση πολλαπλών μεταβλητών⁵⁹⁹.

⁵⁹⁴ Φυτικά υπολείμματα, ορατά με γυμνό μάτι, που ανακτώνται στο αρχαιολογικό πλαίσιο. Στην κατηγορία ανήκουν σπόροι, θραύσματα ξύλου και άλλοι καρποί. Βλ. macrobotanical remains, *Oxford Reference*, διαθέσιμο online.

⁵⁹⁵ Fritz 2005, 779.

⁵⁹⁶ Fritz 2005, 781.

⁵⁹⁷ Fritz 2005, 782-785.

⁵⁹⁸ Fritz 2005, 788-800.

⁵⁹⁹ Weiner 2010, 135.

Φυτόλιθοι

Οι φυτόλιθοι εμφανίζονται σε μίσχους, φύλλα και ταξιανθίες φυτών. Το πυρίτιο που σχηματίζει φυτολίθους μεταφέρεται από τα υπόγεια ύδατα ως μονοπυριτικό οξύ⁶⁰⁰ και αποτίθεται σε επιδερμικά κύτταρα αναπτυσσόμενων φυτών. Σε ορισμένα είδη σχηματίζονται σωματίδια με διακεκριμένα σχήματα, όταν το διοξείδιο του πυριτίου γεμίζει πλήρως το κύτταρο, στερεοποιείται και διατηρεί το κυτταρικό σχήμα, μετά τη φθορά ή την καύση του οργανικού ιστού⁶⁰¹. Οι φυτόλιθοι ιδιαίτερα οι πυριτιούχοι φυτόλιθοι είναι το ανθεκτικότερο βιογενές φυτικό υλικό σε αρχαιολογικές θέσεις⁶⁰². Ο υψηλός βαθμός συντήρησης του φυτολίθου οφείλεται εν μέρει στη δομή του. Κάθε φυτόλιθος αποτελείται σχεδόν εξ ολοκλήρου από διοξείδιο του πυριτίου με λιγότερο από 0,03% οργανικό υλικό⁶⁰³. Η διαδικασία ανάλυσης των φυτολίθων περιλαμβάνει αρκετά, καλά τυποποιημένα βήματα, που σχετίζονται με την αφαίρεση της οργανικής ύλης, τη συγκέντρωση με φυγοκέντρηση πυκνότητας, το πλύσιμο, την ξήρανση και τη μέτρηση των φυτολίθων⁶⁰⁴.

Η ανάλυση των φυτολίθων συμπληρώνει τη ανάλυση της γύρης και τις μακροβοτανικές αναλύσεις και παρέχει εντοπισμένα δεδομένα σχετικά με την εμφάνιση φυτών, βοηθά στη διόρθωση προκαταλήψεων σε ερμηνείες βασισμένες σε ανθρακούχα βοτανικά υπολείμματα και παράγει δεδομένα όπου υπάρχουν αρχαιολογικές θέσεις με ισχυρές συνθήκες διάβρωσης των οργανικών υλικών. Τα καλλιεργημένα φυτά ταυτοποιούνται μέσω της ανάλυσης φυτολιθικών εδαφών και μέσω αυτών μελετάται η εξέλιξη των καλλιεργητικών συστημάτων. Όταν εφαρμοστεί προσεκτικά, η ανάλυση φυτολίθων είναι δυνατόν να βοηθήσει στην ανασυγκρότηση του παλαιοπεριβάλλοντος⁶⁰⁵.

4.9. Ξύλο

Το ξύλο ως μακροβοτανικό κατάλοιπο είναι φυσική μαρτυρία κατασκευών του παρελθόντος. Τα ξύλινα αντικείμενα μπορούν, επίσης, να υποδεικνύουν

⁶⁰⁰ Jones and Handreck 1967, 108.

⁶⁰¹ Blackman 1971, 769-781.

⁶⁰² Weiner 2010, 136.

⁶⁰³ Weiner 2010, 142-143.

⁶⁰⁴ Grant et al. 2005, 82.

⁶⁰⁵ Pearsall 1982, 862-871.

άλλους τρόπους χρήσης του ξύλου κατά το παρελθόν. Καθώς τείνει να είναι το πανταχού παρόν υλικό των μακροσκοπικών υπολειμμάτων, ιδιαίτερα στην απανθρακωμένη μορφή του, συμβάλλει όλο και περισσότερο στην αρχαιολογική ανακατασκευή του φυσικού περιβάλλοντος, στις γνώσεις μας για τις μακροκλιματικές και μικροκλιματικές αλλαγές του παρελθόντος, για την ανθρώπινη χρήση ξυλείας και στη χρονολόγηση των αρχαιολογικών θέσεων⁶⁰⁶.

Όταν εξάγεται το ξύλο από το πρωταρχικό του πλαίσιο, είναι σημαντικό να αποθηκεύεται σε συνθήκες παρόμοιες με εκείνες του πλαισίου. Αν αποθηκεύεται σε διαφορετικές συνθήκες, είναι δυνατόν να προκληθούν παραμορφώσεις που θα μπορούσαν να μεταβάλλουν τα αποτελέσματα της ανάλυσης. Ενίοτε, το υγρό ξύλο που απαντάται σε υδάτινα περιβάλλοντα συντηρείται σε υδάτινο περιβάλλον με διάφορες τεχνικές, καθώς οποιαδήποτε διαδικασία ξήρανσης, χωρίς συντήρηση, είναι δυνατόν να οδηγήσει όχι μόνο σε παραμορφώσεις αλλά και γοργή φθορά του υλικού⁶⁰⁷.

4.10. Συσσωρεύσεις απολιθωμάτων

Η ανασύσταση του παλαιοπεριβάλλοντος είναι δυνατόν να γίνει με ανάλυση γύρης, σπορίων και διατόμων. Η παλυνολογία, εκτός της γύρης, ασχολείται με τη μελέτη διαφόρων παλυνομόρφων, δηλαδή όλων των μικροοργανισμών που απαντώνται σε σχετικό δείγμα. Τα κυριότερα παλυνόμορφα είναι οι γυρεόκοκκοι και τα σπόρια⁶⁰⁸.

Η ανάλυση της γύρης ή παλυνολογική ανάλυση είναι μια άλλη τεχνική σχετικής χρονολόγησης, καθώς και μέθοδος περιβαλλοντικής ανασυγκρότησης. Αναπτύχθηκε το 1916, όταν βρέθηκαν γυρεόκοκοι σε λιμναία ιζήματα και πυρήνες πάγου. Τα διαγράμματα γύρης μπορούν να χρησιμοποιηθούν για συγκρίσεις μεταξύ δύο αρχαιολογικών θέσεων και για τη δημιουργία χρονοσειρών. Όταν συγκρίνουμε δύο θέσεις, αν τα στρώματα στο διάγραμμα γύρης είναι παρόμοια, τότε μπορούμε να συμπεράνουμε ότι οι θέσεις ανήκουν στην ίδια εποχή. Παράλληλα, μπορούμε να αναδημιουργήσουμε ποια είδη φυτών

⁶⁰⁶ Asouti 2003, 471–484, Dolby 2008, 451–456, επίσης Kreuz 1992, 383–394 και άλλοι.

⁶⁰⁷ Grant et al. 2005, 82

⁶⁰⁸ Punt et al 1974, 5-29.

ήταν διαθέσιμα στο τοπικό περιβάλλον, καθώς και να αναπτύξουμε μια εικόνα του κλίματος. Για παράδειγμα, αν υπάρχει πληθώρα γύρης δέντρων, τότε το κλίμα ήταν ζεστό. Η ανάλυση γύρης είναι στην πραγματικότητα μορφή βιοστρωματογραφίας⁶⁰⁹.

Οι γυρεόκοκκοι είναι εύκολα αναγνωρίσιμοι και διακριτοί, καθώς κάθε δέντρο ή φυτό έχει διαφορετικό σχήμα κόκκου, ενώ το εξωτερικό τους κέλυφος μπορεί να διασωθεί για δεκάδες χιλιάδες χρόνια⁶¹⁰. Η γύρη μεταφέρεται από τον αέρα και εναποτίθεται σε στρώματα. Η λήψη δείγματος χώματος σε μια περιοχή και η ανάλυσή του στο μικροσκόπιο για αναγνώριση των διαφόρων ειδών φυτών που υπάρχουν σε κάθε στρώμα μπορούν να μας δώσουν στοιχεία για το είδος της βλάστησης που υπήρχε σε μια περιοχή μια συγκεκριμένη περίοδο. Η σύνθεση των στοιχείων όλων των στρωμάτων αναπαρίσταται στο διάγραμμα γύρης, που απεικονίζει τις διακυμάνσεις των τύπων της βλάστησης στην πορεία του χρόνου⁶¹¹. Οι διακυμάνσεις αυτές είναι δυνατόν να οφείλονται σε ανθρωπογενή δραστηριότητα, σε κλιματικές αλλαγές, σε μεταβολές της θαλάσσιας στάθμης και άλλους παράγοντες.

Όσον αφορά στα παλυνόμορφα, περιλαμβάνονται σπόρια από μύκητες, κοινόβια και σπόρια από φύκη (διάτομα), υπολείμματα ζωικών μονοκύτταρων ή πολυκύτταρων οργανισμών και υπολείμματα φυτικής ύλης. Τα φυτοπλαγκτικά διάτομα είναι μικροσκοπικά μονοκύτταρα φύκη και βρίσκονται μέσα ή κοντά στο νερό. Εμφανίζονται είτε ως μοναχικά κύτταρα είτε σε αποικίες με διάφορες μορφές. Τα επιμέρους κύτταρα κυμαίνονται σε μέγεθος από 2 έως 200 μικρόμετρα⁶¹². Με τη μελέτη των διατόμων είναι δυνατόν να προσδιοριστούν περιβαλλοντικές αλλαγές, όπως η αποδάσωση και η ρύπανση⁶¹³.

4.11. Γεωαρχειολογία

Τα ιζήματα παρέχουν ενδείξεις για την ανακατασκευή προηγούμενων φυσικών και πολιτισμικών διεργασιών με παρόμοιο τρόπο όπως τα τέχνηρα. Τα ιζήματα είναι δυναμικά στρώματα. Μπορούν να επεκταθούν, να συσσωρευτούν,

⁶⁰⁹ Birx 2010, 355.

⁶¹⁰ Renfrew and Bahn 2001, 244

⁶¹¹ Renfrew and Bahn 2001, 245.

⁶¹² Hasle et al 1996, 5–385.

⁶¹³ Fritz 2005, 773.

να συγκεντρωθούν και να συγχωνευθούν. Για παράδειγμα, ο Edwards (1979) παρατήρησε ότι ένα δείγμα γύρης πάχους 1 cm μπορεί να αντιπροσωπεύει έως και 25-30 χρόνια απόθεσης⁶¹⁴. Οι ειδικοί της γεωαρχαιολογίας χρησιμοποιούν τις αλλαγές που συμβαίνουν στο έδαφος και τη γεωμορφολογία για να ερμηνεύουν την ανθρώπινη συμπεριφορά.

Η γεωαρχαιολογία είναι εφαρμογή των γεωεπιστημών για την επίλυση ερευνητικών προβλημάτων στην αρχαιολογία. Συμβατικά, στον όρο «γεωαρχαιολογία» περιλαμβάνεται ένα υποσύνολο των γεωεπιστημών, ιδιαίτερα γεωμορφολογία, ιζηματολογία, παλυνολογία και η στρωματογραφία⁶¹⁵. Η προτεραιότητα της γεωαρχαιολογίας στη μετανασκαφική ανάλυση, αντικατοπτρίζεται πιθανώς στην παρατήρηση που έκανε ο Renfrew το 1976, ότι «αφού η αρχαιολογία ή τουλάχιστον η προϊστορική αρχαιολογία ανακτά σχεδόν όλα τα βασικά δεδομένα με ανασκαφή, κάθε αρχαιολογικό πρόβλημα ξεκινά ως πρόβλημα στη γεωαρχαιολογία»⁶¹⁶.

Γεωμορφολογικές μελέτες

Οι γεωμορφολογικές μελέτες αποτελούν τη βάση πολλών μελετών περιβαλλοντικής ανασύνθεσης. Οι μεθοδολογικοί δεσμοί της γεωμορφολογίας και της αρχαιολογίας προέρχονται από τους στόχους της σύγχρονης ή τουλάχιστον της διαδικαστικής αρχαιολογίας, που περιλαμβάνουν την ανάλυση των συνθηκών και των διαδικασιών διαμόρφωσης και καταστροφής των αρχαιολογικών θέσεων⁶¹⁷ (Schiffer, 1987). Τόσο η αρχαιολογία όσο και η γεωμορφολογία μπορούν να περιγραφούν ως ιστορικές επιστήμες και η ανάδειξη της αρχαιολογίας τοπίου παρείχε ένα πρόσθετο και ολοκληρωμένο σκέλος για την κατανόηση της εξέλιξης του ανθρώπινου περιβάλλοντος.

Οι συνεισφορές της γεωμορφολογίας στην αρχαιολογία περιλαμβάνουν τη χαρτογράφηση και την κατανόηση του παρόντος, του παρελθόντος και των αλλαγών σε σχηματισμούς εδαφών, ιζηματογενείς αποθέσεις και τα περιβάλλοντα, ειδικότερα εκείνες που προκαλούνται από τη ροή ποτάμιων

⁶¹⁴ Edwards 1979, 255–270.

⁶¹⁵ Pollard 1999, 7–14.

⁶¹⁶ Renfrew 1976, 1–12.

⁶¹⁷ Schiffer 1996, 13–18.

υδάτων⁶¹⁸. Η χαρτογράφηση στοχεύει στον εντοπισμό τόσο των περιγραφικών όσο και των αναλυτικών πληροφοριών⁶¹⁹ για τη δημιουργία αλουβίων ή χερσαίων αποθέσεων που έχουν μεταφερθεί σε χαμηλότερες τοπογραφικά περιοχές, όπως κοιλάδες, παράκτιες πεδιάδες κ.λπ.)⁶²⁰.

Μέσω της γεωμορφολογικής ανάλυσης επιτυγχάνεται, επίσης, η κατανόηση της ανθρώπινης επίδρασης του ανθρώπου στο τοπίο. Οι ανθρώπινες δραστηριότητες είναι στενά συνυφασμένες με τις ποτάμιες διαδικασίες. Σε πρώιμα πολιτισμικά συστήματα και στις αγροτικές κοινωνίες οι ανθρώπινες επιδράσεις στα ποτάμια συστήματα σχετίζονται κυρίως με επιπτώσεις στη χρήση γης εξαιτίας της αυξημένης παραγωγής ύδατος και ιζημάτων. Αργότερα και σε ανεπτυγμένες περιοχές, όπου οι υδραυλικοί δίαυλοι λειτουργούν για άρδευση, έλεγχο πλημμυρών, αποστράγγιση υγροτόπων, πλοήγηση και άμυνα είναι κοινές και οι μορφολογικές μεταβολές των καναλιών αναδεικνύουν τις επιπτώσεις τοπικών μηχανικών αλλαγών στο περιβάλλον⁶²¹.

Οι περιγραφές του φυσικού περιβάλλοντος μιας πολιτιστικής θέσης έχουν τη μεγαλύτερη ιστορία στη γεωαρχαιολογία εξαιτίας της ανάγκης να καθοριστεί το χρονικό και φυσικό πλαίσιο στο οποίο ζούσαν οι ανθρώπινες κοινωνίες του παρελθόντος. Έτσι, η στρωματογραφική ανάλυση και η ανάλυση ιζημάτων που επιδρούν σε πολιτισμούς με το πέρασμα του χρόνου, διερευνά και αναδεικνύει το περιβάλλον στο οποίο αναπτύχθηκαν⁶²². Ωστόσο δείχθηκε ότι έχει ιδιαίτερη σημασία και η ανάλυση του τρόπου με τον οποίο οι περιβαλλοντικές αλλαγές επηρέασαν τους πολιτισμούς και το αντίστροφο⁶²³.

Με την ανάλυση των ιζημάτων, οι αρχαιολόγοι μπορούν να συγκεντρώσουν πληροφορίες σχετικά με τη χρονολογική σειρά των αρχαιολογικών θέσεων, τη δημιουργία και τη λειτουργία μιας θέσης. Τα εργαστήρια για την ανάλυση ιζημάτων τείνουν να επικεντρώνονται στη μελέτη σε ορυκτολογικές, μικρομορφολογικές και κοκκομετρικές μελέτες, σε μελέτες pH, οργανικής ύλης, ανθρακικού ασβεστίου και φωσφόρου. Όπως συμβαίνει με οποιοδήποτε υλικό, οι συγκεκριμένες μέθοδοι που χρησιμοποιούνται εξαρτώνται

⁶¹⁸ Macklin 1995, 205-220.

⁶¹⁹ Davidson 1985, 25-55.

⁶²⁰ Renfrew and Bahn 2001, 239.

⁶²¹ Vischer 1989, 19-40.

⁶²² Brown et al 2006, 40-56.

⁶²³ Hassan 1979, 267-270.

από τις απαντήσεις που θέλει να αντλήσει κανείς από το υλικό. Για παράδειγμα, τόσο η πετρογραφία όσο και η περίθλαση ακτίνων Χ μπορεί να χρησιμοποιηθεί σε μία ορυκτολογική εξέταση, αλλά η επιλογή της μεθόδου εξαρτάται από τα συγκεκριμένα προς ανάλυση ορυκτά⁶²⁴.

4.12. Μέθοδοι Χρονολόγησης

Οι μέθοδοι χρονολόγησης, όχι μόνο για την επιστήμη της αρχαιολογίας, είναι δύο, η σχετική και η απόλυτη χρονολόγηση.

Σχετική χρονολόγηση

Πριν από την ανακάλυψη της ραδιομετρικής χρονολόγησης στις αρχές του 20ού αιώνα, η οποία παρείχε ένα μέσο απόλυτης χρονολόγησης, οι αρχαιολόγοι και οι γεωλόγοι χρησιμοποίησαν τη σχετική χρονολόγηση για τον προσδιορισμό της ηλικίας των αρχαιολογικών ή γεωλογικών υλικών. Αν και η σχετική χρονολόγηση μπορεί να καθορίσει μόνο τη διαδοχή με την οποία συνέβη μια σειρά γεγονότων και όχι το πότε συνέβη, παραμένει χρήσιμη ως τεχνική. Η σχετική χρονολόγηση από βιοστρωματογραφικά δεδομένα είναι η προτιμώμενη μέθοδος στην παλαιοντολογία και είναι, από ορισμένες απόψεις, ακριβέστερη⁶²⁵.

Στη σύγχρονη αρχαιολογική πρακτική η σχετική χρονολόγηση (χρονοσειρά, υπέρθεση και στρωματογραφία) είναι η ραχοκοκαλιά της αρχαιολογικής έρευνας⁶²⁶. Βασίζεται στην αρχή της επαλληλίας και συνίσταται στη δημιουργία μιας χρονοσειράς μεταξύ των διαφόρων αρχαιολογικών δεδομένων, με την ταξινόμησή τους σε μια χρονολογική ακολουθία. Αυτή η χρονολογική ακολουθία αποτελεί το βασικό πλαίσιο, για τον προσδιορισμό της σχετικής ηλικίας νεότερων δεδομένων στη διαδικασία ανασύνθεσης μιας ακολουθίας φάσεων και γεγονότων σε μια αρχαιολογική θέση⁶²⁷.

Στις αρχαιολογικές μεθόδους σχετικής χρονολόγησης περιλαμβάνεται η στρωματογραφία και η τυπολογία, μέσω των οποίων παράγονται αντίστοιχες ακολουθίες. Η στρωματογραφία παρέχει τη δυνατότητα κατασκευής σχετικών

⁶²⁴ Balme and Paterson 2006, 338-339.

⁶²⁵ Stanley 1999, 167-169.

⁶²⁶ O'Brien and Lyman 1999, 6.

⁶²⁷ Κουκουζέλη κ.ά. 2003, 167,

χρονοσειρών των στρωμάτων και των περιεχομένων τους σε μια αρχαιολογική θέση⁶²⁸.

Οι αρχαιολόγοι χειρίζονται τη μεταβλητότητα των τεχνέργων μέσω της τυπολογίας, της ταξινόμησης δηλαδή των τεχνέργων σε τύπους⁶²⁹. Το ερώτημα τι είναι η τυπολογία σχετίζεται άμεσα με το ερώτημα τι είναι τύπος. Ορισμένοι θεωρούν ότι είναι ομάδα παρόμοιων πραγμάτων, άλλοι ότι είναι ιδέα ή ομάδα ιδεών περί της ομοιότητας των πραγμάτων. Άλλοι θεωρούν ότι είναι λέξεις που περιγράφουν τα πράγματα και την ομοιότητά τους⁶³⁰. Και οι τρεις ορισμοί είναι αποδεκτοί στην επιστημονική συζήτηση για την τυπολογία. Ο τύπος με την πληρέστερη έννοιά του αποτελείται από πράγματα, μαζί με τις ιδέες μας για τα πράγματα, όπως επίσης και από τις λέξεις ή τις εικόνες με τις οποίες εκφράζουμε αυτές τις ιδέες⁶³¹.

Με την ανάπτυξη των στατιστικών εργαλείων και της αριθμητικής ταξινόμησης χρησιμοποιήθηκαν μαθηματικές μέθοδοι για την κατασκευή τυπολογιών. Αυτές οι τεχνικές παρέχουν τη δυνατότητα σύγκρισης των βαθμών συνέπειας μεταξύ των ιδιαίτερων χαρακτηριστικών των τεχνέργων. Οι συντελεστές συσχέτισης που δημιουργούνται από αυτές τις μεθόδους βοηθούν τους αρχαιολόγους να διακρίνουν μεταξύ σημαντικών και ασήμαντων ομοιοτήτων μεταξύ των τεχνέργων⁶³². Συνειδητά ή ασυνείδητα οι ονομασίες που αποδίδουν οι αρχαιολόγοι σε κάθε τύπο είναι δυνατόν να επηρεάσουν την επιλογή των οντοτήτων (π.χ. τεχνέργων) που αποδίδονται σε κάθε τύπο. Για αυτό οι ονομασίες ανόμοιων τύπων θα πρέπει να επιλέγονται προσεκτικά⁶³³.

Η μέθοδος σχετικής χρονολόγησης με βάση την τυπολογία στηρίζεται στην αρχή ότι τα τέχνηρα ταξινομούνται σύμφωνα με την αντιληπτή ή μετρήσιμη ομοιότητα μεταξύ των παρατηρούμενων δεδομένων και των συγκεκριμένων αναλυτικών μονάδων, που ονομάζονται τύποι. Εδώ οι τύποι είναι συσσωρεύσεις διαγνωστικών χαρακτηριστικών που καταγράφονται σε ένα συγκεκριμένο σημείο δειγματοληψίας, σε μια συγκεκριμένη στιγμή. Επομένως, η ομοιότητα βασίζεται

⁶²⁸ Birx 2010, 352.

⁶²⁹ Kelly and Hurst 2012, 98.

⁶³⁰ Dunnell 1986, 191-193.

⁶³¹ Leach 1976, 17-22.

⁶³² Spaulding 1953, 305-313.

⁶³³ Foucault 1973, 139.

στην ποιότητα ή την ποσότητα διαγνωστικών ιδιοτήτων που μοιράζονται οι τύποι και τα δεδομένα⁶³⁴.

Στηρίζεται, επίσης, σε μια δεύτερη αρχή της βαθμιαίας αλλαγής των χαρακτηριστικών των τεχνέργων διαμέσου του χρόνου. Επομένως, βάσει των χαρακτηριστικών τους ταξινομούνται σε μια εξελικτική σειρά, για παράδειγμα από τον απλούστερο προς τον συνθετότερο τύπο⁶³⁵. Προκύπτει, έτσι, μια τυπολογική ακολουθία ή χρονοσειρά, η οποία μπορεί να χρησιμοποιηθεί για τη σχετική χρονολόγηση οποιουδήποτε τεχνέργου παρουσιάζει ομοιότητες με κάποιο άλλο, που αποτελεί ήδη τμήμα της τυπολογικής ακολουθίας. Βέβαια, ο χρονολογικός προσδιορισμός της τυπολογικής ακολουθίας πρέπει να προσδιοριστεί με ανεξάρτητα μέσα, για παράδειγμα, με στρωματογραφικές παρατηρήσεις⁶³⁶.

Οι σχετικά απλές μέθοδοι της κατασκευής τυπολογιών εξελίχθηκαν σε πολύπλοκες τεχνικές δημιουργίας εξελικτικών σειρών συχνότητας FST (frequency seriation techniques), οι οποίες βασίζονται στην αρχή ότι κάθε τύπος τεχνέργου φθάνει στο αποκορύφωμα της δημοτικότητάς του σε μια ορισμένη χρονική στιγμή και συνεπώς εμφανίζεται συχνότερα σε μία ομάδα (assemblage). Οι τεχνικές αυτές παριστούν γραφικώς τις συχνότητες του κάθε τύπου τεχνέργου διαμέσου του χρόνου⁶³⁷. Οι εξελικτικές σειρές συχνότητας χρησιμοποιούν σχετικές συχνότητες ή ποσοστά εμφάνισης τύπων τεχνέργων σε μια ομάδα για διαμορφώσουν μια σειρά ή αλληλουχία μεταξύ πολλών ομάδων⁶³⁸.

Στο βαθμό που οι αρχαιολογικές θέσεις μιας γεωγραφικής περιοχής περιέχουν τύπους τεχνέργων με ισοδύναμη συχνότητα δημοτικότητας, τότε θεωρούνται ότι έχουν περίπου την ίδια ηλικία. Οι τεχνικές δημιουργίας εξελικτικών σειρών συχνότητας είναι χρήσιμες στη διασταυρωμένη χρονολόγηση (cross dating), όταν τύποι τεχνέργων μπορούν να χρονολογηθούν βάσει αντικειμένων που χρονολογούνται ασφαλώς, όπως τα νομίσματα. Η διασταυρωμένη χρονολόγηση επιτρέπει στους αρχαιολόγους να συγκρίνουν τα χαρακτηριστικά του εδάφους και τα τέχνηρα σε κάθε στρώμα για να καθορίσουν

⁶³⁴ Bortolini 2016, 652.

⁶³⁵ Renfrew and Bahn 2001, 120

⁶³⁶ Κουκουζέλη κ.ά. 2003, 167-168.

⁶³⁷ Renfrew and Bahn 2001, 122-123

⁶³⁸ Dunnell 1970, 308, επίσης Lyman et al 1997, 57 και O'Brien and Lyman 1999, 121-125.

τη σχέση τους⁶³⁹. Μπορούν, επίσης, να χρησιμοποιήσουν τις εξαγωγές και εισαγωγές τεχνέργων για να επεκτείνουν τις χρονολογικές συνδέσεις μέσω διασταυρωμένης χρονολόγησης με άλλες περιοχές⁶⁴⁰

Προκειμένου να είναι ισχυρή η σειρά συχνότητας, πρέπει να πληροί τρεις προϋποθέσεις: Αρχικά όλες οι ομάδες ή οι χώροι δειγματοληψίας πρέπει να έχουν την ίδια ή συγκρίσιμη χρονική διάρκεια, έτσι ώστε η κατανομή συχνοτήτων να μην επηρεάζεται πέραν των αποτελεσμάτων μεγέθους δείγματος. Κατόπιν, όλες οι ομάδες σε μια σειρά πρέπει να είναι ομόλογες, με άλλα λόγια, να προέρχονται από έναν κοινό πρόγονο και να παρουσιάζουν κληρονομική συνέχεια, εκτός της ιστορικής συνέχειας⁶⁴¹. Τέλος, η συσχέτιση πρέπει να διεξάγεται σε κατάλληλη χωρική κλίμακα, έτσι ώστε η κατανομή συχνότητας χαρακτηριστικών ή τύπων να μπορεί να θεωρηθεί χρονολογικά και όχι γεωγραφικά προσδιορισμένη⁶⁴².

Η **σχετική χρονολόγηση** χρησιμοποιεί και μεθόδους φυσικών επιστημών, στις οποίες περιλαμβάνονται η παλαιοκλιματολογία και η ανάλυση πανίδας:

Η παλαιοκλιματολογία είναι η μελέτη των κλιματικών διακυμάνσεων, για τις οποίες δεν έχουν γίνει συστηματικές μετρήσεις⁶⁴³. Χρησιμοποιεί μια ποικιλία μεθόδων για την απόκτηση δεδομένων που διατηρούνται σε βράχους, ιζήματα, φύλλα πάγου, δακτυλίους δένδρων, κοράλλια, κελύφη και μικροαπολιθώματα. Στη συνέχεια χρησιμοποιεί τα δεδομένα για τον προσδιορισμό της παρελθούσας κατάστασης των διαφόρων κλιματικών περιοχών της Γης και του ατμοσφαιρικού της συστήματος. Ο επιστημονικός τομέας της παλαιοκλιματολογίας έφθασε στην ωριμότητά του στον 20ό αιώνα. Οι μελέτες των παρελθουσών αλλαγών στο περιβάλλον και τη βιοποικιλότητα συχνά αντικατοπτρίζουν τις επιπτώσεις του κλίματος στις μαζικές εξαφανίσεις ειδών και τη βιοτική ανάκαμψη⁶⁴⁴.

Ειδικότερα, η παλαιοκλιματική ανάλυση βασίζεται στην ανάλυση ισοτόπων οξυγόνου για τη χρονολόγηση ωκεάνιων ιζημάτων (που περιέχουν τα

⁶³⁹ Giddings 1966, 127-135.

⁶⁴⁰ Renfrew and Bahn 2012, 132.

⁶⁴¹ Cochrane, 2009, 183-203.

⁶⁴² Dunnell, 1970, 316, επίσης Lipo et al 1997, 301-334.

⁶⁴³ Bradley 2015, 1-2.

⁶⁴⁴ Sahney and Benton 2008, 759-765.

ασβεστούχα κελύφη των τρηματοφόρων μικροοργανισμών) και ιζηματογενών οριζόντων τήξης πάγου και στρωμάτων πάγου και στην ανάλυση κόκκων γύρεως ή παλυνολογία. Το μεγαλύτερο μέρος του σήματος $\delta^{18}\text{O}$ στα βενθικά τρηματοφόρα (foraminifera) σχετίζεται με αλλαγές στην ισοτοπική σύνθεση των ωκεανών υπό την επίδραση του μεταβαλλόμενου όγκου του ηπειρωτικού πάγου⁶⁴⁵. Το $\delta^{18}\text{O}$ είναι μέτρο της αναλογίας σταθερών ισοτόπων οξυγόνου-18 (^{18}O) και οξυγόνου-16 (^{16}O) και είναι σταθερή σύνθεση ισοτόπων⁶⁴⁶.

Στην παλαιοκλιματική έρευνα συνεισφέρει επίσης η ανάλυση μικροπανίδας απολιθωμένων εντόμων και οστρακοειδών χερσαίων εδαφών και γλυκών νερών. Το φάσμα των μη-θαλάσσιων γεωλογικών μελετών που παρέχουν πληροφορίες σχετικές με την παλαιοκλιματολογία είναι σημαντικό, καθώς όλες σχεδόν οι ηπειρωτικές ιζηματογενείς αποθέσεις μεταφέρουν σε κάποιο βαθμό παλαιοκλιματικά σήματα, καθώς οι κατακρημνιστικές, παγετώδεις, λιμναίες και ποτάμιες κοιλάδες είναι συνάρτηση του κλίματος⁶⁴⁷.

Από παλαιοκλιματική άποψη μία από τις σημαντικότερες ιδιότητες των εντόμων είναι η ικανότητά τους να καταλαμβάνουν σύντομα νέα εδάφη μετά από κλιματική βελτίωση. Παρέχουν έτσι έναν πιο ευαίσθητο δείκτη κλιματικής διακύμανσης από ό, τι τα φυτά, τα οποία έχουν πολύ πιο αργά ποσοστά μετανάστευσης. Τα κολεόπτερα, για παράδειγμα μπορούν να καταλάβουν και να εγκαταλείψουν μια νέα επικράτεια ανταποκρινόμενα σε ένα αξιοσημείωτο αλλά σύντομο θερμό διάστημα, ενώ δεν υπάρχουν αποδείξεις για τέτοια συμβάντα στα αρχαία γύρεως λόγω της υστέρησης στο χρόνο απόκρισης της βλάστησης⁶⁴⁸.

Από τις διάφορες αυτές επιμέρους αναλύσεις προκύπτουν, με μορφή διαγραμμάτων, οι ανάλογες κλιματολογικές και περιβαλλοντικές χρονοσειρές, οι οποίες χρησιμοποιούνται από την αρχαιολογία για τη σχετική χρονολόγηση των αρχαίων πολιτισμών σε όλο τον πλανήτη⁶⁴⁹.

Απόλυτη χρονολόγηση είναι η διαδικασία προσδιορισμού της ηλικίας ενός αρχαιολογικού δεδομένου στην αρχαιολογία και τη γεωλογία. Ορισμένοι επιστήμονες προτιμούν τους όρους χρονομετρική ή ημερολογιακή χρονολόγηση,

⁶⁴⁵ Bradley 2015, 222.

⁶⁴⁶ Tian et al 2018, 6712.

⁶⁴⁷ Flint 1976, 519–528.

⁶⁴⁸ Coope and Brophy 1972, 97-142, επίσης Morgan, 1973 173-212.

⁶⁴⁹ Renfrew and Bahn 2001, 127.

καθώς η χρήση της λέξης «απόλυτη» συνεπάγεται αδικαιολόγητη βεβαιότητα ακρίβειας⁶⁵⁰. Η απόλυτη χρονολόγηση παρέχει την αριθμητική ηλικία ή εύρος σε αντίθεση με τη σχετική χρονολόγηση, η οποία τοποθετεί τα γεγονότα σε τάξη, χωρίς να μετρά χρονικές διαφορές μεταξύ των γεγονότων. Στην αρχαιολογία, η απόλυτη χρονολόγηση βασίζεται συνήθως στις φυσικές, χημικές και βιολογικές ιδιότητες των των τεχνέργων, των κτηρίων ή άλλων κατασκευών που έχουν τροποποιηθεί από ανθρώπους και στις ιστορικές πηγές που προσδίδουν γνωστές ημερομηνίες (νομίσματα και γραπτή ιστορία). Οι τεχνικές περιλαμβάνουν τη δενδροχρονολόγηση, τη χρονολόγηση ραδιενεργού άνθρακα σε ξύλο ή οστά και μεθόδους χρονολόγησης παγιδευμένης ενέργειας, όπως η χρονολόγηση θερμοφωταύγειας των υαλωμάτων των κεραμεικών κ.ά.⁶⁵¹.

Οι ιστορικές και ημερολογιακές μέθοδοι απόλυτης χρονολόγησης βασίζονται σε χρονολογίες και ημερολόγια που καταγράφηκαν από αρχαίους λαούς. Για παράδειγμα, οι αρχαίοι Έλληνες κατέγραψαν την ιστορία τους εκκινώντας από τη χρονολογία των πρώτων Ολυμπιακών Αγώνων το 776 ΠΚΕ. Άλλοι αρχαίοι λαοί, όπως οι Μεσοποτάμιοι, οι Αιγύπτιοι, οι Ρωμαίοι και οι Μάγια διατηρούσαν ημερολόγια, στα οποία κατέγραφαν διάφορα γεγονότα με βάση τις εκάστοτε δυναστείες ή τις περιόδους εξουσίας των ηγεμόνων τους⁶⁵². Ένα από τα πιο ακριβέστερα ημερολόγια είναι αυτό των αρχαίων Αιγυπτίων, που περιλαμβάνει 31 δυναστείες, καλύπτοντας την περίοδο 3000-332 ΠΚΕ⁶⁵³.

Χρησιμοποιώντας τις ιστορικές και ημερολογιακές χρονολογίες -και με την προϋπόθεση ότι τα ημερολόγια μπορούν να συνδεθούν με την ιστορική χρονολογία - οι αρχαιολόγοι μπορούν να προσδιορίσουν με ακρίβεια την ηλικία αντικειμένων, όπως επιγραφών ή νομισμάτων, που αποκαλύπτονται σε «κλειστά» στρώματα. Επιπλέον, οι ιστορικές και ημερολογιακές χρονολογίες είναι χρήσιμες στη διασταυρωμένη χρονολόγηση. Έτσι, εάν, για παράδειγμα, η τυπολογική ακολουθία μιας αρχαιολογικής θέσης ή περιοχής μπορεί να συνδεθεί με τη στρωματογραφική ακολουθία μιας άλλης, η οποία μπορεί να χρονολογηθεί με ακρίβεια με βάση έστω και μια ιστορική ή ημερολογιακή χρονολογία, τότε η

⁶⁵⁰ Toby and Webster 2001, 203 και Henke 2007, 312.

⁶⁵¹ Kelly and Hurst 2012, 87.

⁶⁵² Renfrew and Bahn 2001, 130-131.

⁶⁵³ Renfrew and Bahn 2001, 13.

τυπολογική ακολουθία της πρώτης θέσης ή περιοχής μπορεί και αυτή να αποκτήσει απόλυτες χρονολογίες εκφρασμένες σε έτη.

Οι φυσικές επιστήμες παρέχουν στην αρχαιολογία μια σειρά από μεθόδους απόλυτης χρονολόγησης:

α) Η χρονολόγηση με ραδιάνθρακα (^{14}C) ανακαλύφθηκε το 1949 από τον W. Libby, επιτρέποντας τη χρονολόγηση αρχαιολογικών δεδομένων που έως τότε δεν υπήρχε τρόπος να χρονολογηθούν. Στη φύση, ο άνθρακας υπάρχει ως δύο σταθερά, μη ραδιενεργά ισότοπα, δηλαδή άνθρακας-12 (^{12}C) και άνθρακας-13 (^{13}C). Το ραδιενεργό ισότοπο είναι ο άνθρακας-14 (^{14}C). Ο χρόνος ημιζωής του ^{14}C είναι 5.730 ± 40 χρόνια, οπότε η συγκέντρωσή του στην ατμόσφαιρα θα αναμενόταν να μειωθεί σε μερικές χιλιάδες χρόνια. Ωστόσο, ^{14}C παράγεται συνεχώς στη χαμηλότερη στρατόσφαιρα και την άνω τροπόσφαιρα, κυρίως από τις γαλαξιακές κοσμικές ακτίνες και σε μικρότερο βαθμό από τις ηλιακές κοσμικές ακτίνες⁶⁵⁴. Έτσι, παράγονται νετρόνια, που με τη σειρά τους δημιουργούν ^{14}C όταν συγκρούονται με ισότοπα αζώτου-14 (^{14}N). Μέσω πυρηνικής αντίδρασης δημιουργείται το ^{14}C ⁶⁵⁵. Μόλις παραχθεί, το ^{14}C συνδυάζεται γρήγορα με το οξυγόνο στην ατμόσφαιρα, για να σχηματίσει αρχικά μονοξείδιο (CO)⁶⁵⁶ και τελικά διοξείδιο του άνθρακα (CO_2)⁶⁵⁷. Το διοξείδιο του άνθρακα που παράγεται με αυτόν τον τρόπο διαχέεται στην ατμόσφαιρα, διαλύεται στον ωκεανό και απορροφάται από φυτά μέσω φωτοσύνθεσης. Τα ζώα τρέφονται με φυτά και τελικά ο ραδιάνθρακας κατανέμεται σε όλη τη βιόσφαιρα. Η αναλογία $^{14}\text{C} / ^{12}\text{C}$ είναι περίπου 1,25 μέρη ^{14}C με 1012 μέρη ^{12}C ⁶⁵⁸. Επιπλέον, περίπου το 1% των ατόμων άνθρακα ανήκει στο σταθερό ισότοπο ^{13}C ⁶⁵⁹. Εκπέμποντας ένα βήτα σωματίδιο και ένα αντινετρίνο ηλεκτρονίου, ένα από τα νετρόνια στον ^{14}C πυρήνα αλλάζει σε πρωτόνιο και ο πυρήνας ^{14}C επανέρχεται στο σταθερό (μη ραδιενεργό) ισότοπο ^{14}N ⁶⁶⁰.

Κατά τη διάρκεια της ζωής του, ένα φυτό ή ζώο βρίσκεται σε ισορροπία με το περιβάλλον του, ανταλλάσσοντας άνθρακα είτε με την ατμόσφαιρα είτε με τη

⁶⁵⁴ Bowman 1995, 9-15, επίσης Russell 2011, 16.

⁶⁵⁵ Bianchi and Canuel 2011, 35.

⁶⁵⁶ Lal and Jull 2001, 731-742.

⁶⁵⁷ Queiroz-Alves et al 2018, 278-305.

⁶⁵⁸ Tsipenyuk 1997, 343.

⁶⁵⁹ Bowman 1995, 9-15.

⁶⁶⁰ Taylor and Bar-Yosef 2014, 33.

διατροφή του. Συνεπώς, θα έχει την ίδια αναλογία ^{14}C με την ατμόσφαιρα, ή στην περίπτωση των θαλάσσιων ζώων ή φυτών, με τον ωκεανό. Μόλις πεθάνει, παύει να αποκτά ^{14}C , αλλά το ^{14}C στο βιολογικό του υλικό κατά τη στιγμή του θανάτου θα συνεχίσει να αποσυντίθεται και έτσι ο λόγος των $^{14}\text{C}/^{12}\text{C}$ στα κατάλοιπά του μειώνεται σταδιακά. Επειδή το ^{14}C διασπάται με γνωστό ρυθμό, η αναλογία ραδιοάνθρακα μπορεί να χρησιμοποιηθεί για να καθοριστεί ο χρόνος διακοπής ανταλλαγής σε ένα δεδομένο δείγμα. Όσο μεγαλύτερο είναι το δείγμα ηλικιακά, τόσο λιγότερα ισότοπα ^{14}C θα μείνουν⁶⁶¹.

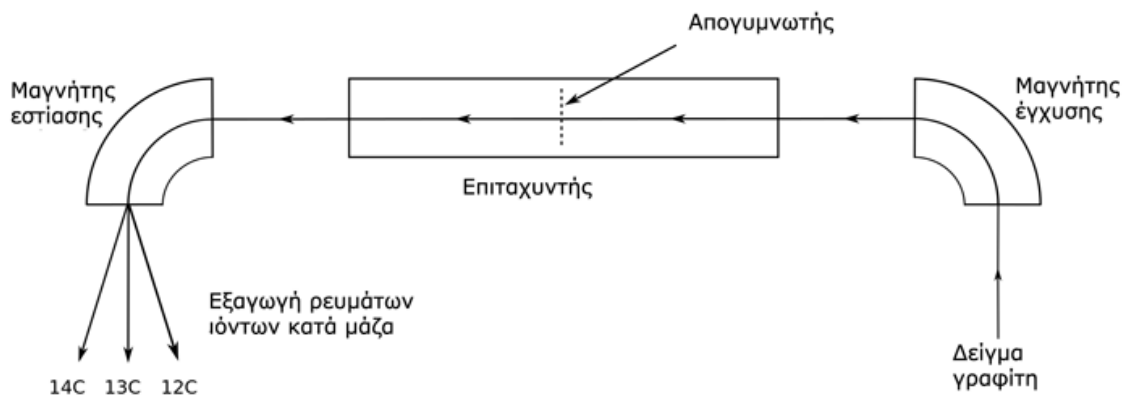
Οι ημερομηνίες που προκύπτουν από ραδιοχρονολόγηση παρουσιάζουν γενικά ένα εύρος τυπικής απόκλισης, αλλά μπορούν, να διορθωθούν με βάση τις χρονολογίες που παρέχει η δένδροχρονολόγηση⁶⁶². Η αξιοπιστία των αποτελεσμάτων της ραδιοχρονολόγησης βελτιώνεται με την επιμήκυνση του χρόνου δοκιμής. Για παράδειγμα, εάν η καταμέτρηση της διάσπασης επί 250 λεπτά είναι αρκετή για να δώσει σφάλμα ± 80 χρόνια, με διάστημα εμπιστοσύνης 68%, τότε ο διπλασιασμός του χρόνου μέτρησης στα 500 λεπτά θα επιτρέψει να μετρηθεί δείγμα με μισό ^{14}C με το ίδιο σφάλμα 80 ετών⁶⁶³. Η χρονολόγηση ραδιοάνθρακα γενικά περιορίζεται σε δείγματα που χρονολογούνται όχι περισσότερο από 50.000 έτη. Παλαιότερα δείγματα δεν έχουν επαρκές ραδιοϊσότοπο ^{14}C για να είναι μετρήσιμα. Παλαιότερες ημερομηνίες έχουν ληφθεί με τη χρήση ειδικών τεχνικών προετοιμασίας δειγμάτων, μεγάλα δείγματα και πολύ μεγάλους χρόνους μέτρησης. Οι συγκεκριμένες τεχνικές μπορούν να επιτρέψουν τη μέτρηση ημερομηνιών μέχρι 60.000 π.π. και σε μερικές περιπτώσεις έως 75.000 χρόνια πριν από το παρόν⁶⁶⁴.

⁶⁶¹ Tsipenyuk 1997, 344.

⁶⁶² Renfrew and Bahn 2001, 143.

⁶⁶³ Bowman 1995, 38–39.

⁶⁶⁴ Walker 2005, 23.



Εικ. 4.1. Απλοποιημένο διάγραμμα λειτουργίας επιταχυντή φασματομετρίας μάζας⁶⁶⁵.

Η χρονική κλίμακα είναι δυνατόν να επεκταθεί στα 80-100.000 χρόνια πριν, χάρη στη φασματομετρία μάζας με επιταχυντή (accelerator mass spectrometry-AMS), για να μετρά απευθείας, τα άτομα ραδιοάνθρακα, αντί να υπολογίζει τις διασπάσεις του⁶⁶⁶. Η AMS μετρά τα ισότοπα ^{14}C και ^{12}C σε ένα δεδομένο δείγμα, προσδιορίζοντας άμεσα την αναλογία $^{14}\text{C}/^{12}\text{C}$. Το δείγμα, συχνά με τη μορφή γραφίτη, φτιάχνεται για να εκπέμπει C-ιόντα (άτομα άνθρακα με ένα μόνο αρνητικό φορτίο), τα οποία εγχύονται σε ένα επιταχυντή. Τα ιόντα επιταχύνονται και διέρχονται μέσω λεπτού φύλλου άνθρακα, το οποίο λειτουργεί ως απογυμνωτής ηλεκτρονίων, έτσι ώστε τα ιόντα να εμφανίζονται με θετικό φορτίο. Τα ιόντα, τα οποία μπορεί να έχουν από 1 έως 4 θετικά φορτία (C+ έως C4+), ανάλογα με το σχέδιο του επιταχυντή, στη συνέχεια περνούν μέσω ενός μαγνήτη που καμπυλώνει την πορεία τους. Τα βαρύτερα ιόντα καμπυλώνουν λιγότερο από τα ελαφρύτερα, έτσι ώστε τα διαφορετικά ισότοπα να εμφανίζονται ως ξεχωριστά ρεύματα ιόντων. Ο ανιχνευτής σωματιδίων καταγράφει τότε τον αριθμό των ιόντων που ανιχνεύονται στο ρεύμα ^{14}C , αλλά επειδή ο όγκος των ^{12}C (και ^{13}C , που απαιτούνται για τη βαθμονόμηση) είναι υπερβολικά μεγάλος για την ανίχνευση μεμονωμένων ιόντων, οι μετρήσεις προσδιορίζονται μετρώντας το ηλεκτρικό ρεύμα που δημιουργείται σε ένα κλωβό Φάραντ⁶⁶⁷. Το μεγάλο θετικό φορτίο που προκαλείται από τα μόρια των δυνάμεων του απογυμνωτή όπως το ^{13}CH , το οποίο έχει βάρος αρκετά κοντά στο

⁶⁶⁵ Aitken 1990, 83.

⁶⁶⁶ Harris 1987, 23-43.

⁶⁶⁷ Aitken 1990, 82-85.

^{14}C για να παρεμβαίνει στις μετρήσεις, αποσυνδέεται, επομένως δεν ανιχνεύεται⁶⁶⁸. Οι περισσότεροι επιταχυντές AMS μετρούν επίσης το $\delta^{13}\text{C}$ του δείγματος, που χρησιμοποιείται για τον υπολογισμό της ηλικίας των ραδιοανθράκων του δείγματος⁶⁶⁹.

Οι ερευνητές έχουν μελετήσει άλλα ραδιενεργά ισότοπα που δημιουργούνται από τις κοσμικές ακτίνες, για να διαπιστωθεί εάν θα μπορούσαν να χρησιμοποιηθούν στη χρονολόγηση αντικειμένων αρχαιολογικού ενδιαφέροντος. Σε αυτά τα ισότοπα περιλαμβάνονται τα ^3He , ^{10}Be , ^{21}Ne , ^{26}Al και ^{36}Cl . Με την ανάπτυξη της AMS στη δεκαετία του '80 έγινε δυνατή η ακριβής μέτρηση των ισοτόπων, ώστε να αποτελέσουν τη βάση χρήσιμων τεχνικών, οι οποίες εφαρμόστηκαν κυρίως για τη χρονολόγηση πετρωμάτων⁶⁷⁰. Τα φυσικώς απαντώμενα ραδιενεργά ισότοπα μπορούν, επίσης, να αποτελέσουν τη βάση άλλων μεθόδων, όπως η χρονολόγηση καλίου-αργού, η χρονολόγηση αργού-αργού και η χρονολόγηση ουρανίου⁶⁷¹. Άλλες τεχνικές χρονολόγησης που ενδιαφέρουν τους αρχαιολόγους περιλαμβάνουν τη θερμοφωταύγεια, οπτικά διεγερόμενη φωταύγεια (optically stimulated luminescence), τον συντονισμό στροφορμής ηλεκτρονίου (electron spin resonance dating-ESR) και χρονολόγηση με τροχιές σχάσης (fission track dating), καθώς και τεχνικές που εξαρτώνται από ετήσιες ζώνες ή στρώματα, όπως η δένδροχρονολόγηση, η τεφροχρονολόγηση και η χρονολόγηση με ιζηματογενή στρώματα ταινιωτών αργίλων (varves)⁶⁷².

Η δένδροχρονολόγηση χρησιμοποιεί τους δακτυλίους ετήσιας ανάπτυξης των δένδρων και ειδικότερα το εύρος των δένδροδακτυλίων, το οποίο μεταβάλλεται ανάλογα με τις ετήσιες κλιματολογικές διακυμάνσεις. Στην αρχαιολογία η δένδροχρονολόγηση χρησιμοποιείται για τη χρονολόγηση θέσεων, κατασκευών, τεχνέργων, πολιτισμικών φάσεων και περιόδων⁶⁷³. Η ανάπτυξη στα δέντρα εμφανίζεται σε ένα στρώμα κυττάρων κοντά στον φλοιό. Ο ρυθμός ανάπτυξης ενός δέντρου αλλάζει σε ένα προβλέψιμο πρότυπο καθ' όλη τη διάρκεια του έτους ανταποκρνόμενος στις εποχιακές κλιματικές αλλαγές, με

⁶⁶⁸ Wiebert 1995, 16.

⁶⁶⁹ Tuniz et al 2004, 395.

⁶⁷⁰ Walker 2005, 77-79.

⁶⁷¹ Walker 2005, 57-77.

⁶⁷² Walker 2005, 93-162.

⁶⁷³ Towner 2002, 68-84.

αποτέλεσμα την εμφάνιση ορατών δακτυλίων ανάπτυξης. Η διατομή των δασικών δένδρων υποδεικνύει μια εναλλαγή ανοικτόχρωμων και σκουρόχρωμων ζωνών, κάθε μία από τις οποίες είναι συνήθως συνεχής γύρω από την περιφέρεια του δέντρου. Αυτές οι ζώνες αντιπροσωπεύουν αυξήσεις εποχικής ανάπτυξης που παράγονται από μεριστρωματικούς ιστούς στο κάμβιο του δέντρου⁶⁷⁴. Η μεγαλύτερη χρονοσειρά βαθμονόμησης με δακτυλίους είναι η χρονοσειρά της γερμανικής βελανιδιάς και πεύκου ~12.400 χρόνια, αποτελούμενη από >5.000 διαφορετικές αλληλοεπικαλυπτόμενες χρονοσειρές από ζωντανά δέντρα, μεσαιωνική ξυλεία στέγης και ξύλο από ποτάμια περιβάλλοντα⁶⁷⁵. Από το 2013, οι παλαιότερες μετρήσεις δακτυλίων δέντρων στο βόρειο ημισφαίριο διαμόρφωσαν μια αλληλοεπικαλυπτόμενη αλληλουχία που εκτείνεται στο παρελθόν από 12.580 έως 13.900 περίπου χρόνια⁶⁷⁶.

Δένδρα του ίδιου είδους που αναπτύσσονται στην ίδια γεωγραφική περιοχή ή στο ίδιο φυσικό περιβάλλον παρουσιάζουν γενικά τις ίδιες διακυμάνσεις στη σειρά του εύρους των δενδροδακτυλίων τους, που μπορούν να μετρηθούν και να αποτυπωθούν σε ένα διάγραμμα («καμπύλη αναφοράς») ώστε να αποτελέσουν μια χρονολογημένη δακτυλιοσειρά ή μια χρονοσειρά δακτυλίων⁶⁷⁷. Συνήθως, τα τμήματα αρχαίου ξύλου περιέχουν μόνον ένα τμήμα δακτυλίων και σπάνια τον εξωτερικό δακτύλιο ή φλοιό απ' όπου αρχίζει η μέτρηση για τον υπολογισμό της ηλικίας του δένδρου, γι' αυτό τα τμήματα αυτά δεν είναι δυνατόν να χρονολογηθούν αφεαυτού. Οι δακτυλιοσειρές τους συγκρίνονται με μια ήδη γνωστή χρονοσειρά δακτυλίων δένδρων του ίδιου είδους, από την ίδια γεωγραφική περιοχή. Συνταιριάζοντας διαδοχικά δακτυλιοσειρές από ζώντα δένδρα και από παλαιότερα δένδρα ή τμήματα ξύλου του ίδιου είδους, οι αρχαιολόγοι μπορούν να κατασκευάσουν μεγάλες και συνεχείς χρονοσειρές δενδροδακτυλίων για μια ολόκληρη γεωγραφική περιοχή, οι οποίες να εκτείνονται στο χρόνο μέχρι 10.000 χρόνια π.π.⁶⁷⁸.

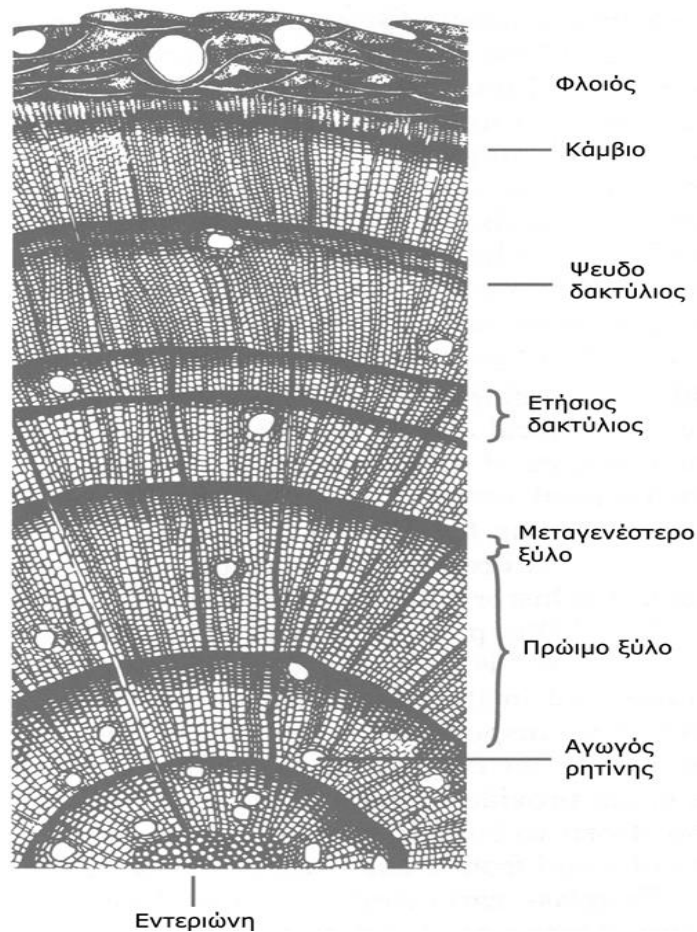
⁶⁷⁴ Vaganov et al. 2006, 368.

⁶⁷⁵ Becker 1993, επίσης Friedrich et al, 2004

⁶⁷⁶ Reimer et al 2013, 1869-1887.

⁶⁷⁷ Fletcher et al 1974, 31-40.

⁶⁷⁸ Renfrew and Bahn 2001, 136



Εικ. 4.2. Σχεδιάσμα της κυτταρικής δομής νεαρού στελέχους κωνοφόρου. Διαφοροποιήσεις στο πάχος των τραχειδών είναι δυνατόν να παράγουν ψευδοδακτυλίους, είτε στο πρώιμο είτε στο μεταγενέστερο ξύλο⁶⁷⁹.

Η δενδροχρονολόγηση παρέχει ακριβείς χρονολογίες και είναι χρήσιμη στην επαλήθευση και διόρθωση των χρονολογιών που προκύπτουν από τη χρονολόγηση με C-14. Μειονέκτημα της μεθόδου είναι ότι απαιτεί δείγματα ξύλου με τουλάχιστον 100 δακτυλίους για ορισμένα είδη⁶⁸⁰ και ότι περιορίζεται σε σχετικά πρόσφατες αρχαιολογικές θέσεις (όχι παλαιότερες των 10.000 ετών) και σε γεωγραφικές περιοχές με αισθητές καιρικές μεταβολές⁶⁸¹.

Η μέθοδος Καλίου/Αργού (Κ-Αγ) εφαρμόζεται στα ηφαιστειογενή πετρώματα και βασίζεται στο γεγονός ότι το κάλιο, που περιέχεται σε πετρώματα τέτοιου είδους με τη μορφή ραδιενεργών ατόμων (Κ-40), διασπάται σε ένα αέριο, το αργό (Ar-40), και φθείρεται με το πέρασμα του χρόνου. Η

⁶⁷⁹ Προσαρμογή από Fritts 1976, 59, εικ. 2.3.

⁶⁸⁰ Baillie 1982, 33, επίσης Towner 2002, 68–84.

⁶⁸¹ Κουκουζέλη κ.ά. 2003, 173.

συγκεκριμένη μέθοδος χρησιμοποιήθηκε για τη χρονολόγηση ρωών λάβας και ηφαιστειακής τέφρας, η οποία σε ορισμένες περιοχές του κόσμου μπορεί να συγκριθεί με τις παγετώδεις αποθέσεις ή να συσχετισθεί με τα πρώιμα απολιθώματα ανθρωποειδών⁶⁸². Το αργόν είναι αέριο που μπορεί να εξαχθεί από ένα δείγμα με θέρμανση. Έτσι, η μέθοδος χρησιμοποιείται για τη χρονολόγηση ηφαιστειακών πετρωμάτων που δεν περιέχουν αργό μετά την ψύξη της λάβας, ρυθμίζοντας έτσι το ισοτοπικό «ρολόι» στο μηδέν. Με την πάροδο του χρόνου, παράγεται και διατηρείται το ⁴⁰Ar μέσα στους μεταλλικούς κρυστάλλους, μέχρι να εξαχθεί με θέρμανση στο εργαστήριο κατά τη διαδικασία της χρονολόγησης⁶⁸³. Η μέθοδος αυτή έχει μεγάλο χρονικό εύρος εφαρμογής από 100.000 μέχρι 4.5 δισεκατομμύρια χρόνια και είναι χρήσιμη για τη χρονολόγηση πρώιμων αρχαιολογικών θέσεων.

Οι θεμελιώδεις παραδοχές στη χρονολόγηση καλίου/αργού είναι ότι (α) δεν έχει μείνει αργό στο ηφαιστειακό υλικό μετά τον σχηματισμό του και (β) το σύστημα παρέμεινε κλειστό από τότε άρχισε η παραγωγή του αργού. Η πρώτη παραδοχή μπορεί να είναι άκυρη στην περίπτωση ορισμένων ωκεάνειων βασαλτικών υλικών που διατηρούν το προηγούμενως σχηματισμένο αργό υπό υψηλή υδροστατική πίεση. Ομοίως, ορισμένα πετρώματα είναι πιθανό να έχουν ενσωματώσει παλαιότερο, πλούσιο σε αργό, υλικό κατά τη διάρκεια του σχηματισμού τους. Τέτοιοι παράγοντες οδηγούν στην υπερεκτίμηση της ηλικίας του δείγματος⁶⁸⁴.

Παρόμοια σφάλματα προκύπτουν από το γεγονός ότι το σύγχρονο αργόν απορροφάται στην επιφάνεια και στο εσωτερικό του δείγματος, με αποτέλεσμα να ακυρώνεται η δεύτερη υπόθεση. Ωστόσο, η ατμοσφαιρική ρύπανση με αργόν μπορεί να μετρηθεί με τη διερεύνηση των διαφόρων ισοτόπων του παρόντος αργού. Το ατμοσφαιρικό αργόν εμφανίζεται ως τρία ισότοπα, ³⁶Ar, ³⁸Ar και ⁴⁰Ar. Δεδομένου ότι ο λόγος ⁴⁰Ar/³⁶Ar στην ατμόσφαιρα είναι γνωστός, οι συγκεκριμένες συγκεντρώσεις των ³⁶Ar και ⁴⁰Ar σε ένα δείγμα μπορούν να

⁶⁸² Loffler 1976, 55-62, επίσης Porter 1979, 161-187 και McDougall and Brown 2006, 205-220.

⁶⁸³ Dalrymple and Lanphere, 1969, 68.

⁶⁸⁴ Fitch 1972, 77-91.

χρησιμοποιηθούν ως μέτρο του βαθμού ατμοσφαιρικής μόλυνσης και να προσαρμοσθεί κατάλληλα η φαινόμενη ηλικία του δείγματος⁶⁸⁵.

Άλλη μέθοδος αρχαιολογικού ενδιαφέροντος είναι η θερμοφωταύγεια. Η φωταύγεια είναι το φως που εκπέμπεται από ένα ορυκτό κρύσταλλο (κυρίως χαλαζία και άστριο) όταν υποβάλλεται σε θέρμανση ή όταν εκτίθεται στο φως. Η μέθοδος της θερμοφωταύγειας εφαρμόζεται σε ανόργανα υλικά κατάλοιπα που είναι προϊόντα θέρμανσης, όπως κεραμικά αντικείμενα, καμένα υλικά από κλιβάνους ή εστίες, μεταλλουργικές σκωρίες, καμένα πυριτολιθικά εργαλεία, λάβα κ.λπ. Βασίζεται στην αρχή ότι όλ' αυτά τα κρυσταλλικά υλικά έχουν την ικανότητα να αποθηκεύουν την ενέργεια της φυσικής ακτινοβολίας που ενυπάρχει σε αυτά (από ραδιενεργές προσμείξεις και από το περιβάλλον). Η θερμοφωταύγεια ενός δείγματος συνάρτηση της ηλικίας του. Όσο παλαιότερο είναι το δείγμα, τόσο μεγαλύτερη θα είναι η ένταση της θερμοφωταύγειας⁶⁸⁶.

Θερμαινόμενα τα δείγματα σε υψηλή θερμοκρασία, εκπέμπουν την αποθηκευμένη ενέργεια με τη μορφή ακτινοβολίας. Έτσι, είναι δυνατόν να χρονολογηθούν τα διάφορα κρυσταλλικά υλικά, προσδιορίζοντας τον χρόνο που μεσολάβησε αφότου αυτά θερμάνθηκαν για τελευταία φορά σε υψηλή θερμοκρασία. Έτσι μετράται η θερμοφωταύγειά τους, δηλαδή η αποθηκευμένη ενέργειά τους, καθώς και η επιδεκτικότητά τους να παράγουν θερμοφωταύγεια. Επειδή η θερμοφωταύγεια αυξάνεται συνεχώς με την πάροδο του χρόνου (εκτός εάν τα υλικά αναθερμανθούν σε περιπτώσεις επανάχρησης), η μέθοδος παρέχει δυνατότητες απεριόριστης χρονολόγησης όσον αφορά στα γεωλογικά υλικά, όπως η λάβα. Όσον αφορά στα κεραμικά και τα πυριτολιθικά εργαλεία, έως τώρα υφίσταται το όριο των 10.000 και 100.000 χρόνων π.π. αντίστοιχα⁶⁸⁷.

Όσον αφορά στη σειρά χρονολογήσεων ουρανίου, σε ένα σύστημα που περιέχει ουράνιο και παραμένει αδιατάρακτο για μεγάλο χρονικό διάστημα (106 έτη), επικρατεί μια δυναμική ισορροπία, στην οποία κάθε θυγατρικό προϊόν θα είναι παρόν σε μια τέτοια ποσότητα, ώστε να φθείρεται με τον ίδιο ρυθμό που σχηματίζεται από το μητρικό ισότοπό του⁶⁸⁸. Η αναλογία ενός ισότοπου προς κάποιο άλλο ίναι ουσιαστικά σταθερή. Ωστόσο, αν το σύστημα διαταραχθεί, δε

⁶⁸⁵ Miller 1972,

⁶⁸⁶ Bradley 2015, 1-2.

⁶⁸⁷ Renfrew and Bahn 2001, 149.

⁶⁸⁸ Broecker and Bender 1972

θα επικρατεί πλέον η ισορροπία παραγωγής και απώλειας και οι σχετικές αναλογίες διαφορετικών ισοτόπων θα βρίσκονται σε ανισορροπία. Μέσω της μέτρησης του βαθμού με τον οποίο επιστρέφει ένα διαταραγμένο σύστημα προϊόντων φθοράς σε μια νέα ισορροπία, μετράται και το χρονικό διάστημα της διαταραχής⁶⁸⁹.

Πολύ περισσότερη εμπιστοσύνη μπορεί να δοθεί σε ημερομηνίες σειρών ουρανίου που λαμβάνονται σε δείγματα ανθρακικών αλάτων από σπηλαιοθέματα (σταλακτίτες και σταλαγμίτες). Τέτοιες αποθέσεις είναι πυκνές και δεν υπόκεινται σε μετααποθετική έκπλυση. Επειδή το ^{230}Th είναι τόσο αδιάλυτο, το νερό από το οποίο κατακρημνίζεται το ανθρακικό σπηλαιοθέμα μπορεί να θεωρηθεί ουσιαστικά απαλλαγμένο από θόριο. Ως εκ τούτου, με την προϋπόθεση ότι η αρχική συγκέντρωση ουρανίου είναι επαρκής, η μέτρηση της αναλογίας $^{230}\text{Th}/^{234}\text{U}$ θα δείξει τη συσσώρευση ^{230}Th με το πέρασμα του χρόνου⁶⁹⁰.

Η ραδιομετρική μέθοδος ουρανίου/θορίου, επίσης αρχαιολογικού ενδιαφέροντος, βασίζεται στη ραδιενεργό διάσπαση των ισοτόπων του ουρανίου σε θυγατρικά προϊόντα, τα οποία αυξάνονται με την πάροδο του χρόνου, καθώς το ουράνιο διασπάται και εξασθενεί και επομένως η συσσώρευσή τους μπορεί να μετρηθεί. Η μέθοδος αυτή είναι ιδιαίτερα χρήσιμη για τη χρονολόγηση υλικών πλουσίων σε ανθρακικό ασβέστιο (CaCO_3), όπως τα σταλαγμιτικά υλικά σπηλαίων⁶⁹¹ και τα δόντια (ζώων και ανθρώπων). Το χρονικό εύρος εφαρμογής της είναι μέχρι στιγμής 50.000-500.000 χρόνια π.π.

⁶⁸⁹ Ivanovich και Harmon 1982

⁶⁹⁰ Harmon et al 1975, 21-33.

⁶⁹¹ Renfrew and Bahn 2001, 154-155.

Κεφάλαιο 5: Το Λογισμικό Digit

Πιθανώς θα αναρωτηθεί κανείς για ποιο λόγο γίνεται εκτεταμένη αναφορά στην αρχαιολογική θεωρία και μέθοδο. Οι παράμετροι κατασκευής ενός συστήματος διαχείρισης ανασκαφικών δεδομένων είναι απόλυτα συνυφασμένοι με την αναλυτική εφαρμογή της αρχαιολογικής θεωρίας και κάτι τέτοιο πρακτικά σημαίνει γνώση της θεωρίας και της μεθόδου σε βάθος, όπως επίσης και των εξελίξεων στην αρχαιολογική θεωρία. Σημαίνει, επίσης, κατανόηση του τρόπου με τον οποίο η ψηφιακή αρχαιολογία επηρεάζει τη θεωρία και το αντίστροφο. Σε διαφορετική περίπτωση το θεωρητικό ή μεθοδολογικό εργαλείο δεν είναι δυνατόν να μετατραπεί σε ψηφιακό εργαλείο, τουλάχιστον με συνέπεια και ακρίβεια.

Η ψηφιακή αρχαιολογία είναι ουσιαστικά εφαρμογή της τεχνολογίας πληροφοριών και ψηφιακών μέσω αναγνώσιμων από τη μηχανή⁶⁹². Περιλαμβάνει τη χρήση ψηφιακής φωτογραφίας, ανίχνευση με laser, αυξημένη πραγματικότητα, εικονική πραγματικότητα και τρισδιάστατη εκτύπωση, ανάμεσα σε άλλες τεχνικές καταγραφής, όπως επίσης, την εισαγωγή ψηφιακών οντοτήτων σε κατάλληλες βάσεις δεδομένων. Η υπολογιστική αρχαιολογία, που καλύπτει μεθόδους ανάλυσης με υπολογιστές, θεωρείται συνήθως υποπεδίο της ψηφιακής αρχαιολογίας, όπως και η εικονική αρχαιολογία⁶⁹³.

Αρκετοί θεωρούν ότι η ψηφιακή αρχαιολογία δεν είναι *per se* εξειδικευμένος κλάδος της αρχαιολογίας ή θεωρητική σχολή, αλλά μέθοδος χρήσης της τεχνολογίας πληροφοριών προς διευκόλυνση των αρχαιολόγων⁶⁹⁴. Θεωρείται, επίσης, στρατηγική διατήρησης ψηφιακών υλικών⁶⁹⁵. Ωστόσο, μια πληρέστερη ματιά στην ιστορία και το παρόν της ψηφιακής πρακτικής στην αρχαιολογία υποδηλώνει έναν πλούτο θεωρητικού έργου που συνδέεται άμεσα με την «αναλογική» αρχαιολογική θεωρία και μέθοδο. Η ψηφιακή αρχαιολογία με τη σειρά της φέρνει μεθοδολογικές αλλαγές στην αρχαιολογία εδώ και μισό αιώνα περίπου. Ωστόσο, δεν αναφέρεται ως ουσιαστικός παράγοντας στην

⁶⁹² Colleen and Stuart 2012, 521-537.

⁶⁹³ Zubrow 2006, 8-28

⁶⁹⁴ Daly et al. 2006, 3.

⁶⁹⁵ Harvey 2005, 110.

ανάπτυξη της φιλοσοφίας της αρχαιολογίας, ούτε αναπτύσσεται στα κείμενα της βασικής αρχαιολογικής θεωρίας, κάτι που εν μέρει οφείλεται στην έλλειψη πλαισίου για την καθοδήγηση και ανάπτυξη της ψηφιακής πρακτικής⁶⁹⁶. Σε κάθε περίπτωση η ψηφιακή αρχαιολογία ακολουθεί πιστά την αρχαιολογική θεωρία και μέθοδο.

Η ιδέα για την εφαρμογή Digit υλοποιήθηκε αρχικά στην ανασκαφή της αρχαίας Φαλάσαρνας το 2003 με περιορισμένες δυνατότητες, που σχετίζονταν κυρίως με συγκεκριμένες ανάγκες της ανασκαφής. Χρησιμοποιήθηκε και για επιφανειακή έρευνα που παρουσιάστηκε αργότερα στο Κρητολογικό συνέδριο. Από το 2006 και μετά ξεκίνησε η ανάπτυξη, η δοκιμή και η χρήση του λογισμικού, έτσι ώστε να καλύπτει όχι μόνον τις ιδιαίτερες ανάγκες της ανασκαφής της αρχαίας Κυμισάλας, αλλά γενικά τις ανάγκες ανασκαφών σε διαφορετικά πεδία.

5.1. Ο τόπος



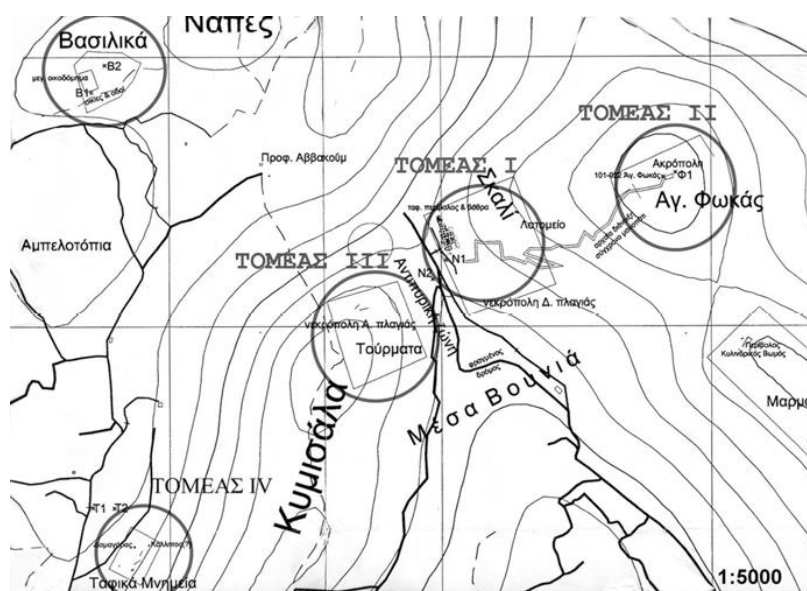
Εικ. 5.1. Γενικός χάρτης της περιοχής της αρχαίας Κυμισάλας

Η αρχαιολογική θέση της Κυμισάλας κείται νοτιοδυτικά της πόλης της Ρόδου, σε ορεινό έδαφος με ανάγλυφο που περιλαμβάνει εμφανή οροπέδια, κοιλάδες και εκτεταμένες δασικές εκτάσεις. Στη γεωγραφική αρχαιολογική θέση

⁶⁹⁶ Dallas 2015, 177.

που ορίζεται από το όρος Ατάβυρος και το όρος Ακραμύτης, την ακτογραμμή (βόρεια) και την επαρχιακή οδό προς τις κοινότητες των Σιαννών και της Μονολίθου, καταγράφονται πολυάριθμες αρχαιολογικές θέσεις με επίκεντρο τον λόφο του Αγίου Φωκά⁶⁹⁷.

Η λεκάνη της Κυμισάλας εκτείνεται ανάμεσα στους λόφους αγ. Φωκάς, Μαρμαρούνια και Κυμισάλα, σε υψόμετρο 289 μ. Μία δεύτερη λεκάνη των Βασιλικών, εκτείνεται βόρεια, από τον λόφο των Βασιλικών και δυτικά, από τον λόφο Ατουμά. Ανατολικά των λόφων αγ. Φωκάς και Χάρακας, στην παραλία της Γλυφάδας, βρίσκεται το αρχαίο λιμάνι της περιοχής, που χρησιμοποιείται έως και σήμερα από αλιευτικά σκάφη. Σημείο αναφοράς για τη γεωγραφική περιοχή είναι και η πηγή Στελιές, κοντά στα Μαρμαρούνια, που πιθανότατα ανέβλυζε από την αρχαιότητα στην ίδια θέση⁶⁹⁸.



Εικ. 5.2. Οι πέντε τομείς έρευνας στην αρχαιολογική γεωγραφική θέση της Κυμισάλας⁶⁹⁹.

Οι τέσσερις τομείς αρχαιολογικής έρευνας που σχετίζονται με το Πανεπιστήμιο Αιγαίου, Τμήμα Μεσογειακών Σπουδών και την ΕΚΠΑ Δωδεκανήσου και ο πέμπτος τομέας τοπογραφικής έρευνας των Βασιλικών και της περιοχής των Μαρμαρουνίων, που σχετίζεται με τη Σχολή Αγρονόμων

⁶⁹⁷ Στεφανάκης και Πατσιαδά 2009-2011, 63-64.

⁶⁹⁸ Stefanakis et al 2015, 263.

⁶⁹⁹ Στεφανάκης και Πατσιαδά 2009-2011, εικόνα 8.

Τοπογράφων Μηχανικών του Εθνικού Μετσόβιου Πολυτεχνείου, διαμορφώνουν μια σύνθετη αρχαιολογική γεωγραφική θέση σε έναν ξεχωριστό βιότοπο⁷⁰⁰. Το οικιστικό πρότυπο διαμορφώνεται από την περιμετρική ανάπτυξη οικισμών, με επίκεντρο την ακρόπολη στον λόφο του αγ. Φωκά σε ένα σύνθετο μορφολογικό τοπίο. Ουσιαστικά, είναι ιεραρχικό σχήμα κατανομής προσαρμοσμένο, όμως, στο περιβάλλον και στην αρχή του μικρότερου κόστους⁷⁰¹.

Η έρευνα υποδεικνύει ότι για την ευρύτερη περιοχή υπάρχουν ενδείξεις κατοίκησης ήδη από τη μυκηναϊκή περίοδο. Αγγεία και χαλκά τέχνηρα της περιόδου βρίσκονται στο Μουσείο της Κοπεγχάγης και στο Αρχαιολογικό Μουσείο της Ρόδου, προϊόντα των ανασκαφών που διεξήγαγε ο Α. Biliotti στα Σιάννα και στον Κάστελο⁷⁰². Υφίστανται, επίσης, προφορικές και γραπτές μαρτυρίες για την ύπαρξη μυκηναϊκών τάφων στην περιοχή της Γλυφάδας⁷⁰³. Σύμφωνα με τον Μ. Ανδρόνικο, που παραθέτει περιγραφές του Maiuri, οι τάφοι που ανασκάφηκαν από τον Biliotti ανήκουν στην υστερομυκηναϊκή περίοδο έως τους γεωμετρικούς χρόνους⁷⁰⁴.

Βέβαια καταγραφές των αρχαιοτήτων της περιοχής παρατηρούνται από τον 15ο αιώνα σε γενικές αναφορές για τα νησιά του Αιγαίου, ενώ από τον 19ο αιώνα περιηγητές, αρχαιοδίφες και αρχαιολόγοι επισκέφθηκαν την αρχαιολογική γεωγραφική θέση της Κυμισάλας και παρέθεσαν περιγραφές των αρχαιοτήτων της⁷⁰⁵. Η οθωμανική διοίκηση του νησιού ελλείπει νομοθετικού πλαισίου, επέτρεψε σε αρχαιολογικές αποστολές μουσείων του εξωτερικού να εμπλουτίσουν τις συλλογές τους με αρχαιότητες της Ρόδου γενικότερα και ειδικότερα της περιοχής Σιαννών και Κυμισάλας. Όταν διαμορφώθηκε νομοθετικό πλαίσιο για τις αρχαιότητες από την τουρκική κυβέρνηση στα τέλη του 19ου αιώνα και οι αρχαιολογικές αποστολές έπαψαν να ενδιαφέρονται και η νεκρόπολη της Κυμισάλας έγινε πεδίο δράσης γηγενών τυμβωρύχων. Αναλυτική περιγραφή για την αρχαιολογική έρευνα της περιόδου στην Κυμισάλα γίνεται σε

⁷⁰⁰ Στεφανάκης Μ.Ι. 2010, 691.

⁷⁰¹ Κουκουζέλη κ.ά 2003, 196.

⁷⁰² Maiuri 1926, 252-253, στο Στεφανάκης και Πατσιαδά 2011, 69.

⁷⁰³ Inglieri 1936, 155 στο ίδιο.

⁷⁰⁴ Ανδρόνικος 1961-1962, 193.

⁷⁰⁵ Στεφανάκης και Πατσιαδά 2009-2011, 67.

σχετική δημοσίευση του επιστημονικού περιοδικού *Ευλιμένη* και σε αρχεία της Εφορείας Προϊστορικών και Κλασικών Αρχαιοτήτων Δωδεκανήσου⁷⁰⁶.

Κατά τον 20ό αιώνα διενεργήθηκε αρχικά έρευνα το 1905 από τον Δανό αρχαιολόγο K.F. Kinch, που κατέγραψε και σχεδίασε τις αρχαιότητες της Κυμισάλας, των Βασιλικών και της ακρόπολης του αγ. Φωκά⁷⁰⁷. Ακολούθησε συστηματική αρχαιολογική έρευνα της Ιταλικής Αρχαιολογικής Σχολής των Αθηνών, το 1913, από τους αρχαιολόγους L. Pernier και G.G. Porro, που διενήργησαν επιφανειακή έρευνα στην περιοχή και στα δυτικά παράλια της νήσου⁷⁰⁸.

Περί το 1915 διενήργησε επιφανειακή έρευνα και ανασκαφή στον ναό της ακρόπολης του Αγίου Φωκά και στην κεντρική νεκρόπολη της Κυμισάλας ο Α. Μαιυρί, Έφορος Αρχαιοτήτων Δωδεκανήσου από το 1916 μέχρι το 1924⁷⁰⁹. Στη συνέχεια επιφανειακές έρευνες στην ύπαιθρο της Ρόδου διενήργησαν το 1970 οι R. Hope Simpson και J.F. Lazenby το 1970, οι οποίοι επισκέφθηκαν τα Βασιλικά, τη νεκρόπολη της Κυμισάλας, τον λόφο του αγ. Φωκά και την περιοχή της Γλυφάδας, διερευνώντας προϊστορικές αρχαιολογικές θέσεις στην περιοχή 48⁷¹⁰. Επιφανειακή έρευνα για προϊστορικές αρχαιότητες σε θέσεις της περιοχής, διενεργήθηκε, επίσης, από τον C. Mee το 1982⁷¹¹.

Στην περιοχή έγιναν, επίσης, ανασκαφές, επιφανειακές έρευνες και καθαρισμοί από την Εφορεία Προϊστορικών και Κλασικών Αρχαιοτήτων Δωδεκανήσου. Το 1968 έγιναν σωστικές ανασκαφές από την Ηώ Ζερβουδάκη στις θέσεις «Σκαλί» και «Τούρματα», στις δυτικές υπώρειες του λόφου του αγ. Φωκά και στις ανατολικές υπώρειες του λόφου της Κυμισάλας⁷¹². Το 1975 υπό την επίβλεψη του αρχαιολόγου Κ. Κατραλή πραγματοποιήθηκαν καθαρισμοί στην ακρόπολη του αγ. Φωκά, σε μνημειακό τάφο στη θέση Αλώνια και στον λόφο των Βασιλικών. Ερευνητική ανασκαφή διενήργησε το 1982 ο Ε. Κακαβογιάννης σε κτήριο των Βασιλικών⁷¹³.

⁷⁰⁶ Στεφανάκης & Πατσιαδά 2009-2011, 63-134.

⁷⁰⁷ Στο ίδιο, 69.

⁷⁰⁸ Pernier 1914, 236-242.

⁷⁰⁹ Βλ. Μαιυρί 1916, 285, στο Στεφανάκης και Πατσιαδά 2009-2011, 69.

⁷¹⁰ Hope Simpson and Lazenby 1973, 146-147.

⁷¹¹ Mee 1982, 105-107.

⁷¹² Στεφανάκης και Πατσιαδά 2009-2011, 69.

⁷¹³ Και για τις δύο ανασκαφές βλ. αντίστοιχα ανασκαφικά ημερολόγια της Εφορείας Δωδεκανήσων.

Η έρευνα της γεωγραφικής αρχαιολογικής θέσης της Κυμισάλας στον 21ο αιώνα, (2006-σήμερα) διεξάγεται από την Εφορεία Προϊστορικών και Κλασικών Αρχαιοτήτων Δωδεκανήσου, με πρόταση που κατέθεσε το Τμήμα Μεσογειακών Σπουδών του Πανεπιστημίου Αιγαίου για κοινή αρχαιολογική έρευνα στην περιοχή. Η αρχαιολογική έρευνα καταγράφεται ως δεύτερη συστηματική περιφερειακή ανασκαφή, μετά την ανασκαφή στο ιερό του Ερεθιμίου Απόλλωνος, στον Θεολόγο⁷¹⁴. Επί του παρόντος η έρευνα είναι επικεντρωμένη στον λόφο του Αγίου Φωκά και στις ανατολικές υπώρειες του λόφου της Κυμισάλας, προκειμένου να εντοπιστεί το χρονολογικό πλαίσιο και η συνολική έκταση της νεκροπόλεως.

Κατά την περίοδο 2006-2010, η ανασκαφική έρευνα διενεργήθηκε στον λόφο του αγ. Φωκά και στον λόφο της Κυμισάλας. Για λόγους χωροθέτησης και διευκόλυνσης της έρευνας η συνολική περιοχή διαιρέθηκε σε πέντε τομείς και ορίστηκε κάναβος με τη χρήση γεωγραφικών συντεταγμένων. Πιο συγκεκριμένα:

Στον τομέα I, ο οποίος διατρέχει τις δυτικές υπώρειες του λόφου του Αγίου Φωκά, περιλαμβάνονται ταφικά μνημεία της Κυμισάλας και διακριτό αρχαίο λατομείο. Στον τομέα αυτό βρίσκεται και η θέση «Σκαλί», στην οποία διενεργήθηκε το 1968 ανασκαφή υπό τη διεύθυνση της Ηούς Ζερβουδάκη.

Στον τομέα II, στην κορυφή της ακροπόλεως του Αγίου Φωκά, βρίσκεται ο ναός, και λίγο χαμηλότερα υπάρχουν αναλημματικοί τοίχοι οχυρωματικού χαρακτήρα. Στον συγκεκριμένο τομέα διενεργήθηκαν επιμέρους καθαρισμοί και επιφανειακές έρευνες για τον προσδιορισμό των ορίων του περιβόλου και για την ανεύρεση της κύριας πύλης του οχυρωματικού περιβόλου και της αρχαίας οδού που οδηγούσε στην ακρόπολη.

Στον τομέα III, στις ανατολικές και στις βορειοανατολικές υπώρειες του λόφου της Κυμισάλας, απαντώνται τάφοι της αρχαϊκής και της κλασικής περιόδου. Στον συγκεκριμένο τομέα εντάσσεται και η θέση «Τούρματα», στην οποία το 1968 διενεργήθηκε η έρευνα της Ηούς Ζερβουδάκη.

Στον τομέα IV, στον οποίο ανήκει συστάδα ελληνιστικών τάφων και ταφικών μνημείων, που εκτείνονται στις βορειοδυτικές υπώρειες του λόφου της Κυμισάλας με σωζόμενα ενεπίγραφα βάθρα.

⁷¹⁴ Παπαχριστοδούλου 1989, 107 κ.ε., 162 κε., 204 κ.ε., στο Στεφανάκης και Πατσιαδά 2009-2011, 70.

Στον τομέα V, του οικισμού των Βασιλικών, ενδιαφέροντος της Σχολής Τοπογράφων του Ε.Μ.Π.

Συνεργάτες στη σύγχρονη αρχαιολογική έρευνα με ερευνητικό και εκπαιδευτικό πρωτοκόλλο συνεργασίας με το Τμήμα Μεσογειακών Σπουδών του Πανεπιστημίου Αιγαίου, είναι η Σχολή Αγρονόμων Τοπογράφων Μηχανικών του Εθνικού Μετσόβιου Πολυτεχνείου. Επίσης, στο πλαίσιο του Κοινοτικού Προγράμματος *Erasmus* και *Erasmus Placement*, συμμετέχουν στην έρευνα μεταπτυχιακοί φοιτητές του Αρχαιολογικού Ινστιτούτου του Πανεπιστημίου Nicolaus Copernicus του Τοριού της Πολωνίας, υπό την εποπτεία του καθηγητή Mariusz Mielczarek. Τα πρώτα συμπεράσματα όσο και η θεωρία πίσω από τη σύγχρονη αρχαιολογική έρευνα έχουν ήδη εκδοθεί σε επιστημονικές περιοδικές εκδόσεις, εν αναμονή των νεότερων στοιχείων που προκύπτουν από τη συνεχιζόμενη ανασκαφική έρευνα⁷¹⁵.

5.2. Η αρχαιολογική θεωρία στην ανασκαφή της Κυμισάλας⁷¹⁶

Το πλαίσιο για την υλοποίηση μιας βάσης δεδομένων και του ανάλογου λογισμικού στην Κυμισάλα περιλαμβάνει τις ουσιαστικές αλληλεξαρτήσεις μεταξύ δραστηριοτήτων, εννοιών και αποτελεσμάτων της εργασίας στο πεδίο και της έρευνας, με έμφαση στην επανάχρηση και ανάλυση τυχόν παρατηρήσεων που παράγονται για κάθε εύρημα ή ομάδα ευρημάτων.

Είναι προφανές ότι η εργασία πεδίου και η ανάλυση συνδέονται, αλλά η πρώτη προσελκύει συνήθως μεγαλύτερη προσοχή από τη δεύτερη. Η πρακτική της ψηφιοποίησης του αρχαιολογικού αρχείου, εφόσον λαμβάνονται υπόψη ζητήματα ορθής πρακτικής και ταξινόμησης, αποκαθιστά σε μεγάλο βαθμό την παρατηρούμενη ανισορροπία και συμβάλλει επίσης στην ολοκληρωμένη παρουσίαση της αρχαιολογικής έρευνας. Όσον αφορά στην ορθή πρακτική εννοούνται οι κανόνες ψηφιοποίησης, έτσι όπως έχουν καθοριστεί από μεγάλους οργανισμούς, ήτοι ICOMOS⁷¹⁷, UNESCO⁷¹⁸, ADS κ.ά. Η καταγραφή και η ψηφιοποίηση των φυσικών χαρακτηριστικών των ιστορικών δομών και τοπίων

⁷¹⁵ Στεφανάκης και Πατσιαδά 2009-2011, 63-134. Επίσης, Stefanakis et al 2015, 259-314.

⁷¹⁶ Καλογερόπουλος 2017, 105-111.

⁷¹⁷ ICOMOS 2019, διαθέσιμο online.

⁷¹⁸ UNESDOC 2003, διαθέσιμο online.

αποτελεί τον ακρογωνιαίο λίθο μιας αειφορικής πολιτισμικής διατήρησης. Οι πληροφορίες που παράγονται καθοδηγούν ουσιαστικά τη λήψη αποφάσεων από μέρους των αρχαιολόγων και συντηρητών. Η αυστηρή τεκμηρίωση μπορεί, επίσης, να εξυπηρετήσει έναν ευρύτερο σκοπό: με την πάροδο του χρόνου, γίνεται ο κύριος τρόπος με τον οποίο οι μελετητές και το κοινό γνωρίζουν και κατανοούν μια αρχαιολογική θέση η οποία αλλάζει ριζικά με την πάροδο του χρόνου και σταδιακά σταδιακά εξαφανίζεται⁷¹⁹.

Σύμφωνα με την Υπηρεσία Αρχαιολογικών Δεδομένων (ADS) και την Αγγλική Κληρονομιά απαιτείται διαρκής διερεύνηση των στρατηγικών διατήρησης και επαναχρησιμοποίησης μεγάλων αρχείων δεδομένων που δημιουργούνται από αρχαιολόγους, ερευνητές και διαχειριστές πολιτιστικών πόρων, οι οποίοι διεξάγουν επιτόπιες επιθεωρήσεις ή ανασκαφικές έρευνες⁷²⁰.

Λαμβάνοντας τα παραπάνω υπόψιν θα πρέπει να σκεφθούμε ότι η αρχαιολογία ως επιστήμη είναι βαθιά συνδεδεμένη με την ιδέα του πειράματος⁷²¹. Το πείραμα (ανασκαφή) δεν είναι δυνατόν να επαναληφθεί από άλλους, τουλάχιστον έτσι όπως επιτάσσει η επιστημονική μέθοδος. Αυτό σημαίνει ότι κατά την προανασκαφική φάση εγείρονται διάφορα ζητήματα, αρχαιολογικής θεωρίας και μεθοδολογίας, τα οποία είναι ζωτικής σημασίας για την παραμετροποίηση οποιασδήποτε σχεσιακής βάσης δεδομένων και του συνδεδεμένου Συστήματος Διαχείρισης Περιεχομένου (CMS).

Εξαρχής, η προσέγγιση στην περίπτωση της ανασκαφής στην Κυμισάλα ήταν ότι τα θεωρητικά ζητήματα συζητούνται εκ των προτέρων, ιδιαίτερα όταν η έρευνα δε συνδέεται με κάποια περιορισμένη αρχαιολογική θέση, αλλά με τη γεωγραφική αρχαιολογική θέση της Κυμισάλας, που περιλαμβάνει πολλαπλά πεδία με αστικό σχεδιασμό, οχυρώσεις και ακρόπολη, ταφικά μνημεία, νεκροταφεία που αντανakλούν κοινωνικές διαστρωματώσεις και εγκαταστάσεις σε μία μεγάλη χρονολογική κλίμακα που αρχίζει από την ύστερη Μινωική, διατρέχει την Κλασική περίοδο και φθάνει ως την Ύστερη Ελληνιστική⁷²².

Η ερμηνεία του «κλασικού» σε μια μεγάλη έκταση χρόνου είναι περίπλοκο ζήτημα, που εξαρτάται σε μεγάλο βαθμό από την κατανόηση του συνολικού

⁷¹⁹ Santana Quintero et al. 1063-1070.

⁷²⁰ ADS 2007, διαθέσιμο online

⁷²¹ Forrest 2008, 61-69.

⁷²² Στεφανάκης και Πατσιαδά, 2009-2011, Stefanakis et al 2015, Stefanakis.

έργου, που πρέπει να παραχθεί. Για αυτό και απαιτείται θεωρητική προσέγγιση για τα διαφορετικά συμφραζόμενα, πριν καν ξεκινήσει κανείς την ανασκαφή, έστω τη δειγματοληψία. Για παράδειγμα, παρατηρείται μια έμφαση στη μνημειακότητα, τάση που έγινε εμφανής στην κλασική αρχαιολογία, η οποία ιστορικά εστίαζε και εν πολλοίς εστιάζει το ανασκαφικό της έργο αποκλειστικά στη μνημειακή αρχιτεκτονική, παραβλέποντας την έρευνα λιγότερο εντυπωσιακών στοιχείων της ανθρώπινης δραστηριότητας, όσο αποκαλυπτικά και αν είναι αυτά τα στοιχεία για την καθημερινή ζωή των ανθρώπων στο παρελθόν⁷²³. Επιπλέον, ο συνήθης ορισμός της κλασικής αρχαιολογίας ή της έννοιας κλασική εποχή δεν μπορεί να θεμελιωθεί σε διαφορετικό πολιτισμικό πλαίσιο. Τι σημαίνει για παράδειγμα κλασική εποχή για τους Κέλτες ή τους Μάγια; Πρακτικό ερώτημα που απαντάται μόνο εν μέρει, αν δοθεί έμφαση στην αστικοποίηση και τη μνημειακότητα που προαναφέρθηκε, αλλά σε ένα διαφορετικό γεωγραφικό και χρονικό πλαίσιο⁷²⁴.

Οι προκαταρκτικές μελέτες στο πλαίσιο ενός αρχαιολογικού τόπου αλλάζουν συχνά τα διαφορετικά επίπεδα των θεωρητικών προσεγγίσεων, όχι αναφορικά με την ανασκαφική μέθοδο, αλλά με τις ερμηνευτικές διαδικασίες. Δεδομένου ότι ο τελικός στόχος είναι η «ακριβής» κατανόηση των διαφορετικών πολιτιστικών πλαισίων με χρονολογική σειρά, διαφορετικές προσεγγίσεις για την ερμηνεία των δεδομένων χρειάζεται να εξεταστούν προσεκτικά και να προαποφασιστούν πριν από κάθε προσπάθεια δοκιμασίας των υποθέσεων που παράγονται στο πεδίο.

Οι στόχοι της ερμηνευτικής φιλοσοφίας διασταυρώνονται με ένα βασικό ερώτημα στην επιστημολογία της αρχαιολογίας: Ποιές είναι οι προϋποθέσεις για την κατανόηση του παρελθόντος των ανθρώπινων κοινωνιών; Το ερμηνευτικό ερώτημα στην προκειμένη περίπτωση δεν είναι τόσο αυτό που καταλαβαίνουμε, αλλά το πώς καταλαβαίνουμε. Υπό ποιές συνθήκες αντιλαμβανόμαστε την

⁷²³ Joyce 2006, 15.

⁷²⁴ Μνημειακές αρχιτεκτονικές κατασκευές και αστικοποίηση για τους Μάγια. Βλ. Coe 1999, 81. Κλασική εποχή για τους Κέλτες δεν αναφέρεται, καθώς παρουσιάζονται ως ορειχάλκινος πολιτισμός που τρέχει παράλληλα με την κλασική εποχή του ελληνορωμαϊκού κόσμου, βλ. Fagan 1996, 219. Κάτι τέτοιο βέβαια υποδεικνύει σε τι βάθος έχουν καθορίσει την άποψή μας για το κλασικό το ιδεώδες και οι ιδέες του Βίνκελμαν και άλλων κλασικιστών, δεδομένου ότι το κλασικό θα έπρεπε να συνδέεται κυρίως με την ωριμότητα ενός πολιτισμικού συστήματος, που δεν καθορίζεται έτσι απλά από τις μνημειακές κατασκευές στην αρχιτεκτονική και την τέχνη.

ετερότητα στο παρόν ή το παρελθόν για παράδειγμα; Δυστυχώς, στην αρχαιολογία, όπως και σε πολλές άλλες κοινωνικές και ανθρωπιστικές επιστήμες, οι όροι αυτοί συχνά απορρίπτονται ως υποκειμενικές και αντιεπιστημονικές προκαταλήψεις, που πρέπει να εξαλειφθούν (ή να αποκρυφθούν) μέσω της εφαρμογής αυστηρών επιστημονικών μεθόδων για την επίτευξη της αντικειμενικής γνώσης. Ωστόσο, είναι σημαντικό σε κάθε περίπτωση να εξετάζονται τα επιστημολογικά «προβλήματα», εγγενή στις προσπάθειες ερμηνείας και κατανόησης του παρελθόντος⁷²⁵.

Οι ανασκαφείς της Κυμισάλας ασχολούνται με μετρήσιμα αντικείμενα και οργανώνουν την κατανόησή τους για τον εμπειρικό κόσμο με τρόπο συστηματικό, κάτι που αντανακλάται στη φιλοσοφία της κατασκευής της σχεσιακής βάσης δεδομένων που χρησιμοποιεί το λογισμικό Digit. Παρόλα αυτά, κάθε είδους ερμηνεία περιλαμβάνει προσωπικές εκτιμήσεις, που εξαρτώνται από το πολιτιστικό, κοινωνικό, πολιτικό και ψυχολογικό υπόβαθρο του αρχαιολόγου⁷²⁶. Σε μια αρκετά απλοποιημένη εκδοχή, η θεωρία είναι μια δεδομένη ακολουθία, μια τάξη στην οποία ταξινομούνται εν γένει τα δεδομένα, διαμορφώνοντας ένα σώμα γνώσης, που μπορεί να συνδέεται ή μη με συγκεκριμένα ερμηνευτικά μοντέλα⁷²⁷. Επιπλέον, η μορφή ταξινόμησης των δεδομένων απαιτεί συγκεκριμένη στρατηγική για τη διαμόρφωση της σχεσιακής βάσης δεδομένων.

Οι τρεις βασικές διεργασίες στην αλυσίδα του αρχαιολογικού έργου στο ανασκαφικό εγχείρημα στην Κυμισάλα είναι η απόκτηση των δεδομένων⁷²⁸, η καταχώρισή τους σε φυσικό και ψηφιακό ημερολόγιο, η ανάλυση (από τα αρχαιολογικά ευρήματα, μέχρι τη δημοσίευση) και οι στρατηγικές της παρατήρησης και της αιτιολόγησης. Οι στρατηγικές της παρατήρησης σχετίζονται με το πώς αντιλαμβάνεται ο αρχαιολόγος διαφορετικά φαινόμενα και πώς επιλέγει τα φαινόμενα, ή «δεδομένα», προκειμένου να τα παρατηρήσει, αποφεύγοντας κατά το δυνατόν τα λάθη της προσωπικής παρατήρησης, ιδιαίτερα στη χρήση αναλογιών⁷²⁹.

⁷²⁵ Johnsen and Olsen 1992, 419-436.

⁷²⁶ Hodder 1987, 7.

⁷²⁷ Thomas 2007, 46.

⁷²⁸ Rahtz and Reilly 2003, 22.

⁷²⁹ Gibbon 2013, 153.

Είναι μάλλον προφανές ότι μεγάλο μέρος της φροντίδας πριν εμπλακούν οι αρχαιολόγοι στις διαδικασίες της φυσικής ανασκαφής, κατευθύνεται προς την επίτευξη της μέγιστης δυνατής αντικειμενικότητας της παρατήρησης και καταγραφής⁷³⁰. Η ακρίβεια στην καταχώριση των πληροφοριών είναι το στέρεο έδαφος, στο οποίο ο αρχαιολόγος μπορεί να ασκήσει ερμηνευτική προσέγγιση, με βάση τις μετρήσεις του πραγματικού κόσμου. Πολλαπλές παρατηρήσεις των ίδιων διαδικασιών έχουν προταθεί ως παράγοντας της αντικειμενικότητας, με τη συμμετοχή διαφορετικών παρατηρητών στη διαδικασία της ηχογράφησης. Κάθε επόπτης που εργάζεται σε ανασκαφικό τετράγωνο στην ανασκαφή στην Κυμισάλας, οφείλει να παρατηρεί το ανασκαφικό έργο που διεξάγεται σε άλλα τετράγωνα και να προσφέρει εκτιμήσεις, εφόσον του ζητηθεί, για τη διαδικασία ή το αρχαιολογικό υλικό⁷³¹. Η γενική μεθοδολογία στην ανασκαφή της Κυμισάλας τείνει προς τον θετικισμό, αν και δεν τίθεται θέμα εμπλοκής στην ατέρμονη συζήτηση για τον ρόλο της αρχαιολογικής πρακτικής ως ανθρωπιστικής ή επιστημονικής επιδίωξης⁷³².

Η αρχαιολογική ερμηνεία στην ανασκαφή της Κυμισάλας

Η αρχαιολογική ερμηνεία στοχεύει στην ανασυγκρότηση του παρελθόντος, συμπεριλαμβανομένης της αξιολόγησης της αλλαγής και της σταθερότητας, της εξέλιξης και της προσαρμογής, των ομοιοτήτων και διαφορών μεταξύ των πληθυσμιακών ομάδων και γενικά όλων των στοιχείων που διαμορφώνουν τον αρχαιολογικό πολιτισμό. Το σημαντικότερο είναι ότι χρησιμοποιούνται σημαντικές κατασκευές στη διαδικασία ερμηνείας - πέρα από τις απλές συλλογές πληροφοριών - που ξεπερνούν τα όρια των περιγραφικών χαρακτηριστικών, εγγενών στα αντικείμενα, τα κτίρια, τις κατασκευές και τα οικοδομημένα. Για παράδειγμα, ενώ ένας αρχαιολόγος στο εργαστήριο μπορεί να καθορίσει τη συμπερίληψη ή τη χημική σύνθεση του πηλού που χρησιμοποιήθηκε για την κατασκευή ενός τεχνέργου, η διαδικασία αυτή καθαυτή στην πραγματικότητα δεν προσφέρει ερμηνεία ή πληροφορίες για την ανακατασκευή του συστημικού πλαισίου την παραγωγή, τη διανομή ή την κατανάλωση αυτού του τεχνέργου.

⁷³⁰ Binford 2009, 56.

⁷³¹ Stefanakis et al 2015, 286.

⁷³² Cohen and Maldonado 2007, 9.

Εάν αυτή η μέθοδος αξιολόγησης ή μέτρησης πρόκειται να χρησιμοποιηθεί σε συνδυασμό με άλλες πληροφορίες σχετικά με την πηγή της πρώτης ύλης, μπορεί να παρέχει ένα μέσο χαρτογράφησης των μορφών επικοινωνίας και αλληλεπίδρασης -ακόμη και ενός συγκεκριμένου εμπορικού δρόμου με μια ακολουθία θέσεων και περιβαλλόντων, των οποίων η γεωμορφολογία παρέχει την ίδια χημική σύνθεση του πηλού.

Συνεπώς, είναι δυνατόν να παραχθούν επεξηγηματικές συνθέσεις νοηματικών δομών, στις οποίες η τελική πρόταση περιέχει πληροφορίες που δεν είναι εγγενείς στην αρχική πρόταση⁷³³. Η ερμηνεία και η κατανόηση είναι ένας στόχος της ανάλυσης και η προσέγγιση απαιτεί στατιστικές μεθόδους που παρέχουν μια εικόνα των σχέσεων μεταξύ των εγγενών πληροφοριών που παρέχονται από το τέχνηργο και όλων των άλλων διαθέσιμων πληροφοριών που σχετίζονται με το πλαίσιο σε πολλαπλές υλικές και χωρικές κλίμακες.

Η διαδικασία δόμησης και συλλογής δεδομένων μέσα στα αναλυτικά και ερμηνευτικά πλαίσια αποτελεί μέρος της προσπάθειας του αρχαιολόγου προσδιορίσει και να ερμηνεύσει τα πολυεπίπεδα πρότυπα που εμπλέκονται σε ένα μόνο αντικείμενο⁷³⁴, έναν σκελετό μέσα σε έναν τάφο, έναν τάφο μέσα σε νεκρόπολη, μέσα σε έναν οικισμό και οι οικισμοί μέσα στο τοπίο των αρχαίων δήμων αντιπροσωπεύουν μια αλυσίδα σχέσεων μεταξύ ατόμων και περιφερειακών θρησκευτικών και τελετουργικών πρακτικών από την αρχαϊκή περίοδο έως την ύστερη αρχαιότητα. Οι απαιτήσεις της ερμηνείας, ωστόσο, δεν περιορίζονται στο πλαίσιο, αλλά επεκτείνονται σε μια κοινωνική και πολιτική μονάδα που σχετίζεται με ιδεολογίες και άλλες κοινωνικές διαμορφώσεις και η αυστηρή μαθηματική προσέγγιση ενός λογαρίθμου δεν είναι ικανή να δώσει απαντήσεις σε προβλήματα κοινωνικής ερμηνείας. Μπορεί μόνο να διατηρήσει δεδομένες κατηγοριοποιήσεις που ορίζονται από την αρχαιολογική μέθοδο και τη θεωρία⁷³⁵.

Στις περισσότερες επιστήμες, οι ερευνητικές στρατηγικές περιλαμβάνουν ή στηρίζονται στη διαδικασία πειραματισμού. Αξιολογώντας τη διάρκεια ενός πειράματος, ορισμένοι παράγοντες μπορεί να θεωρηθούν ότι διατηρούνται ή είναι σταθεροί, ενώ άλλοι μεταβάλλονται, έτσι ώστε να μπορούν να μελετηθούν τα

⁷³³ Salmon 1982, 33

⁷³⁴ Lock and Molynaux 2006, 5.

⁷³⁵ Stefanakis et al, 2015, 289.

αποτελέσματα των αλλαγών. Στην αρχαιολογική ανάλυση, ωστόσο, δεν είναι εξ ολοκλήρου εφικτή η χρήση πειραματισμών με την αληθινή επιστημονική έννοια της λέξης, διότι ενώ οι μέθοδοι μπορούν να επαναληφθούν, το ίδιο το πλαίσιο είναι μοναδικό, δεν μπορεί να αναπαραχθεί και κανονικά καταστρέφεται κατά τη διάρκεια της ανάκτησης. Η ανασκαφή ως μέθοδος δεν επιτρέπει την αναπαραγωγή του πειράματος με τις ίδιες ταξινομικές ή συστημικές μεταβλητές και παραμέτρους⁷³⁶. Επιπλέον, είναι δύσκολο για τους αρχαιολόγους να ανιχνεύουν χρονικά φαινόμενα, χωρίς να ενσωματώνουν πρότυπα -σχέσεις μεταξύ πολλαπλών μεταβλητών και κλίμακες του πλαισίου- που τελικά προέρχονται από καλά χρονολογημένες ομάδες ευρημάτων. Συνεπώς, η αρχαιολογική ανάλυση εξαρτάται εξ ολοκλήρου από ακριβή χρονολόγηση και καταγραφή. Η κατασκευή μιας εφαρμογής όπως η Digit ικανή να καταγράψει με ακρίβεια την πολλαπλότητα των μεταβλητών της ανασκαφικής διαδικασίας και των προϊόντων της κρίνεται ως απαραίτητη.

Οριζόντια και κάθετη δειγματοληψία

Βασικός σκοπός της επαγωγικής στατιστικής είναι η απόκτηση αξιόπιστων συμπερασμάτων σχετικά με διάφορα χαρακτηριστικά του στατιστικού πληθυσμού, με βάση τις πληροφορίες που λαμβάνονται από ένα δείγμα. Η διαδικασία της δειγματοληψίας πρέπει να είναι τυχαία, προκειμένου να αποκτηθούν αμερόληπτα και αντιπροσωπευτικά δείγματα⁷³⁷.

Με όρους γεωγραφικού προσδιορισμού τα παραπάνω σε πρακτικό επίπεδο διαμορφώνουν την ανάγκη καταγραφής πλήρους τοπογραφικού σχεδίου και προσδιορισμό των ανασκαφικών τετραγώνων όχι κατά τη διάρκεια της ανασκαφής, αλλά πριν από οποιαδήποτε ανασκαφική διαδικασία. Σε αυτό το πλέγμα μεταβλητών⁷³⁸ καταχωρούνται τα χωρικά δεδομένα, όπως και τα άυλα μη χωρικά χαρακτηριστικά τους. Είναι, στην πραγματικότητα, μια περιοχή δειγματοληψίας, μια διαίρεση του πληθυσμού σε ίσες περιοχές του δικτύου με τυχαία επιλογή μεταξύ των περιοχών⁷³⁹.

⁷³⁶ Nami 2010, 91-70.

⁷³⁷ Boxill et al 1997, Ch. 4, 36.

⁷³⁸ Ταφικά συμπλέγματα ή άλλες κατασκευές, τέχνηρα, κτήρια ή άλλα οικοδομημένα.

⁷³⁹ Stefanakis et al 2015, 287.

Όσον αφορά στην αρχαιολογία, όμως, η έρευνα δεν είναι από τη φύση της οριζόντια. Στην πραγματικότητα το αρχαιολογικό πείραμα είναι λιγότερο οριζόντιο και περισσότερο κάθετο στον τομέα της ανασκαφικής έρευνας. Κατ' αυτόν τον τρόπο η δειγματοληψία βάσει πλέγματος υπολογίζεται ως το «πρώτο στάδιο». Δεύτερο στάδιο είναι η «τυχαία στρωματοποιημένη δειγματοληψία», που βασίζεται σε αρχαιολογικά στρώματα. Σε κάθε ανασκαμμένο στρώμα πραγματοποιείται με απλή τυχαία επιλογή δειγματοληψία, με ικανό ποσοστό δειγματοληψίας, παρέχοντας μια προανασκαφική γεωλογική εκτίμηση του υπεδάφους και των πιθανών άυλων χαρακτηριστικών των δειγμάτων που έρχονται στην επιφάνεια⁷⁴⁰.

Τεχνικά ζητήματα

Τεχνικά ζητήματα, που σχετίζονται κυρίως με την κάθετη φωτογράφιση του χώρου, οδήγησαν στην κατασκευή σταθερών τετραγώνων στο πλαίσιο των τετραγώνων του Mortimer Wheeler αλληλένδετων με τον τοπογραφικό χάρτη. Εμπνευσμένα από την υποβρύχια αρχαιολογία, όπου τα προβλήματα ορατότητας έκαναν τα σταθερά δίκτυα απαραίτητη επιλογή⁷⁴¹, αυτά τα τετράγωνα κατασκευάζονται είτε από πλαστικούς ή αλουμινένιους και σιδηρούς σωλήνες με διάσταση 4mX4m –υποστηριζόμενα από κάθετους μεταβλητούς άξονες, έτσι ώστε να εξασφαλιστεί η σχετική ακρίβεια⁷⁴² και η αναγκαία για μετρήσεις οριζοντιοποίηση. Η όλη κατασκευή διευκολύνει την κάθετη μέτρηση και τη μέτρηση σε επικλινές έδαφος, καθώς η ανασκαφή στον χώρο της νεκρόπολης διεξάγεται σε επικλινή εδάφη 40° –ενίοτε 60°.

Η κάθετη φωτογράφιση εμπνευσμένη από το άρθρο των Andrew Graham Roulter και Ivor Kerslake για την κάθετη φωτογραφική καταγραφή ενός αρχαιολογικού τόπου⁷⁴³ είναι τεχνικό ζήτημα ιδιαίτερης σημασίας για το εγχείρημα της Κυμισάλας και διεξάγεται είτε με σταθερούς τρίποδες που εφαρμόζουν στο σταθερό τετράγωνο ή γιγάντιους τρίποδες στην περίπτωση ανάγκης κάλυψης μνημείων μεγάλης κλίμακας. Η κάθετη φωτογράφιση βοηθά

⁷⁴⁰ Kipfer 2000, 538.

⁷⁴¹ Catsambis et al 2011, 128.

⁷⁴² Με αυτόν τον όρο εννοείται το εύρος των αποτελεσμάτων που λαμβάνει ο αρχαιολόγος, αν επαναληφθεί η μέτρηση αρκετές φορές. Περί αξιοπιστίας και ακρίβειας βλ. Banning 2006, 11.

⁷⁴³ Roulter and Kerslake 1997, 221-232.

στην ταχύτητα της καταγραφής και την ακρίβεια κλίμακας μέσω προηγμένων εφαρμογών πληροφορικής. Οι φωτογραφίες διορθωμένες στην κατάλληλη κλίμακα διαμορφώνουν φωτομωσαϊκά, που βρίσκουν τη θέση τους στο αρχικό τοπογραφικό σχεδιάγραμμα.



Εικ. 5.3. Κάθετη φωτογράφιση με τη χρήση τρίποδα στον τομέα Ι (φωτ. KARP).

Όταν κατασκευάζονται φωτομωσαϊκά από ένα σύνολο κάθετων εικόνων, προκύπτουν ορισμένα προβλήματα. Μόνο η περιοχή γύρω από την κεντρική προβολή έχει αξιόπιστες χωρικές σχέσεις. Οι περιοχές στα άκρα της εικόνας είναι οι πλέον παραμορφωμένες. Αυτή η στρέβλωση μπορεί να απομακρυνθεί με τη διαδικασία της ορθοδιόρθωσης, κατά την οποία όλα τα σημεία στο επίπεδο της εικόνας διορθώνονται ως προς την κατακόρυφο. Η κάθετη υπερβολή είναι δύσκολο να αφαιρεθεί, καθώς εξαρτάται σε μεγάλο βαθμό από την εστιακή απόσταση του φακού που χρησιμοποιείται. Οι φακοί μεγαλύτερης εστιακής απόστασης παράγουν μικρότερη κάθετη υπερβολή κατά κανόνα. Όταν η παραμόρφωση της εικόνας -τόσο χωρική όσο και κάθετη- αφαιρεθεί, η εικόνα θεωρείται ορθοφωτογραφία, δηλαδή φωτογραφία που διαθέτει μετρητική πληροφορία χάρτη. Τα φωτομωσαϊκά που κατασκευάζονται από ορθοφωτογραφίες είναι πραγματικοί χάρτες, μπορούν να χρησιμοποιηθούν για τον

σχεδιασμό ακριβών περιγραμμάτων⁷⁴⁴ και καταχωρίζονται στη βάση δεδομένων Digit.

Κατά την ανασκαφή ενός συγκεκριμένου τετραγώνου κάθε στρώμα καταγράφεται με κάθετες φωτογραφίες. Χρησιμοποιείται ένας αριθμός των σημείων ελέγχου βαμμένα κόκκινα ή κίτρινα και η φωτογραφία γίνεται αντικείμενο ψηφιακής επεξεργασίας, προκειμένου να διορθωθεί κάθε είδους παραμόρφωση λόγω ύψους. Αυτή η διαδικασία της διόρθωσης καταλήγει σε διορθωμένο σχέδιο μέσω του λογισμικού ηλεκτρονικών υπολογιστών, ακολουθώντας βέβαια τα πρότυπα καλής πρακτικής σε σχέση με τη φωτογράφιση στο πλαίσιο της πολιτιστικής κληρονομιάς⁷⁴⁵.

Η ενασχόληση με τάφους που έχουν ακανόνιστη διαμόρφωση ταφικού θαλάμου είναι χρονοβόρα όσον αφορά στις σχηματικές παραστάσεις, και ένα μέρος της συγκεκριμένης εργασίας πραγματοποιείται με τη χρήση φωτογραμμετρίας. Σημαντικό πλεονέκτημα της χρήσης της φωτογραμμετρίας στην καταγραφή αρχαιολογικών θέσεων είναι η εξοικονόμηση χρόνου στο πεδίο⁷⁴⁶.



Εικ. 5.4. Το ορθοφωτομωσαϊκό της περιοχής της Κυμισάλας (φωτ. KARP-NTUA).

⁷⁴⁴ Garrison 2016, 27.

⁷⁴⁵ Guides to Good Practice, στο *Archaeology Data Service/Digital Antiquity*, online version.

⁷⁴⁶ Fussell 1982, 157-172.

Η δομή των δεδομένων

Η αρχαιολογική έρευνα απαιτεί δραστηριότητες στις οποίες η επεξεργασία των αρχαιολογικών στοιχείων είναι δομημένη σε συνεκτικές ομάδες, ταξινομήσεις, κατηγορίες ή τυπολογίες. Οι τύποι ή οι κατηγορίες εφαρμόζονται συνήθως σε τέχνηρα, αλλά μπορούν να χρησιμοποιηθούν στην ανάλυση του συνολικού πλαισίου ή οποιουδήποτε μέρους του. Ταξινομικές μονάδες όπως οι κατηγορίες και οι τύποι διαμορφώνουν ουσιαστικά το σώμα του υλικού πολιτισμού. Συνεπώς, μέσω της ανάλυσης, της σύγκρισης και συσχετίσεων τύπων ή άλλων μεταβλητών διαμορφώνεται ένα γνωστικό μοντέλο, στο οποίο μπορεί να θεμελιωθεί η παρελθούσα ανθρώπινη συμπεριφορά, καθώς και οι διαφορετικοί παράγοντες που την επηρεάζουν⁷⁴⁷.

Ο Coombs περιγράφει την ανάλυση ως ανίχνευση σχέσεων μοτίβων σε οποιοδήποτε σύνολο δεδομένων⁷⁴⁸. Για παράδειγμα, θα μπορούσε κανείς να ανιχνεύσει μια σχέση μεταξύ των ταφικών πρακτικών, της επεξεργασίας λίθινων τεχνέργων, της διαθεσιμότητας πόρων και των ποσοστών παραγωγής. Η ανάκτηση των στοιχείων για τις ταφικές πρακτικές παράγει έτσι προϊόντα (πληροφορίες, αποτελέσματα, δεδομένα) και οι αρχαιολόγοι ταξινομούν τα προϊόντα του έργου τους στο πεδίο είτε ως «σύνθεση» είτε ως «ερμηνεία». Η *σύνθεση* είναι απλή συλλογή και παρουσίαση δεδομένων. Η *ερμηνεία* όμως απαιτεί επεξεργασία των δεδομένων, προκειμένου να καταλήξει κανείς σε συμπεράσματα σχετικά με την τεχνολογία, τη συμβολική δράση ή την κοινωνική οργάνωση και άλλες πτυχές της πολιτιστικής παραγωγής⁷⁴⁹.

Η σύνθεση ως συμβολική αναπαράσταση

Οι συνθέσεις είναι αναμφισβήτητα το πλέον ορατό αποτέλεσμα της αρχαιολογικής έρευνας. Ο Gardin καθορίζει τη σύνθεση ως σειρά αλληλένδετων προτάσεων που περιγράφουν τα αρχαιολογικά κατάλοιπα, με την οποία διευκολύνεται οποιαδήποτε μελέτη των αρχαίων πληθυσμών⁷⁵⁰. Αυτοί οι τύποι συλλογών είναι συχνά συμβολικές κατασκευές που χρησιμοποιούν μια συγκεκριμένη μεταγλώσσα για να απεικονίσουν το περιεχόμενό τους. Όλα τα

⁷⁴⁷ Stefanakis et al 2015, 287.

⁷⁴⁸ Coombs 1964, 5.

⁷⁴⁹ Stefanakis et al 2015, 288.

⁷⁵⁰ Gardin 1980, 28.

σχέδια, οι χάρτες, η ψηφιοποιημένες απεικονίσεις, ακόμη και ο κώδικας μιας διαδικτυακής εφαρμογής, όπως η Digit, αντιπροσωπεύουν απλοποιήσεις της πραγματικότητας και κατά μία έννοια είναι συμβολικές απεικονίσεις ή αναπαραστάσεις. Προκειμένου να αντλήσει νόημα από τις συνθέσεις πληροφοριών, η αρχαιολογική επιστήμη ακολουθεί τη δική της περιγραφική γλώσσα για την αρχαιολογική μαρτυρία, μια γλώσσα με κανόνες συστηματικής περιγραφής και μετρήσεων. Ένα από τα πλεονεκτήματα αυτής της γλώσσας είναι ότι επιτρέπει τη χρήση υπολογιστών και σχεσιακών βάσεων δεδομένων για τη διαχείριση μεγάλων ποσοτήτων πληροφοριών.

Ωστόσο, τα σχέδια και οι χάρτες που συνοδεύουν μια αρχαιολογική έκδοση είναι απλώς απεικονίσεις ή αντανakλάσεις των τεχνέργων ή αρχαιολογικών κατασκευών. Πρόκειται για κωδικοποιημένες απεικονίσεις που μεταδίδουν συγκεκριμένο τύπο πληροφοριών και παραλείπουν στις περισσότερες περιπτώσεις τον «θόρυβο» ή οποιαδήποτε άλλη πληροφορία που εμποδίζει τη μετάδοση της έννοιας που επιβάλλει ο ανασκαφέας στο σύνολο δεδομένων.

Είναι γεγονός ότι η αρχαιολογική μαρτυρία είναι ουσιαστικά ατελής ή αποσπασματική, δεν ενσωματώνει ή αποδίδει αφεαυτού προφανές νόημα. Προκειμένου να παραχθεί νόημα από το χάος των δεδομένων και να εξαχθούν εύλογα κατανοητές πληροφορίες, είναι ενίοτε απαραίτητο να αποφασιστεί ποιοι τύποι δεδομένων, μετρήσεων και αναλύσεων πρέπει να χρησιμοποιηθούν. Στο KARP επιχειρείται ο συνδυασμός απόλυτων μετρήσεων και κριτηρίων συμπερίληψης δεδομένων, για τη δημιουργία διαφόρων κλιμάκων ανάλυσης. Δεδομένου ότι ο στόχος της αρχαιολογίας σήμερα είναι η κατανόηση των γενικών προτύπων και η δημιουργία μακροσκοπικών αφηγήσεων, μέσω της επεξεργασίας των δεδομένων είναι δυνατόν να συσχετιστούν επαναλαμβανόμενα μοντέλα σε μεγάλες ποσότητες πληροφοριών, προκειμένου να χρησιμοποιηθούν ως αποτελεσματικά, οικονομικά και διαχειρίσιμα εργαλεία ανάλυσης⁷⁵¹.

Οι κατάλογοι δεν είναι αρκετοί

Η έρευνα δεν ολοκληρώνεται με απλούς καταλόγους. Εγγενώς απαιτεί δραστηριότητες, στις οποίες η επεξεργασία της αρχαιολογικής μαρτυρίας συμμετέχει σε κάποιο είδος δομής: ομάδες, κατηγορίες ή τυπολογίες. Οι

⁷⁵¹ Stefanakis et al 2015, 288.

συγκεκριμένες δομές θεωρούνται συνήθως συνδεδεμένες στενά με τέχνηρα. Στην πραγματικότητα εφαρμόζονται στο ευρύτερο πλαίσιο του αρχαιολογικού τύπου, σε χρονολογίες, σε οικοδομημένα, σε οικονομικά συστήματα ή τους ίδιους τους αρχαιολογικούς πολιτισμούς, σε ό, τι γενικά μπορεί να καθοριστεί ως «αντικείμενο». Οι μέθοδος που χρησιμοποιείται προκειμένου να δημιουργηθούν αυτές τις δομές, εν προκειμένω η συστηματική έρευνα, αποτελεί σημαντικό μέρος του αρχαιολογικού έργου εντός και εκτός πεδίου στο εγχείρημα της Κυμισάλας και διαμορφώνει τα θεμέλια για ενδιαφέροντα είδη ανάλυσης⁷⁵².

Ομάδες, κατηγορίες ή τύποι φαινομένων διαμορφώνουν το δομικό υλικό των αρχαιολογικών αναλύσεων και σε μεγάλο βαθμό λειτουργούν ως ταξινομητική αρχή για το λογισμικό Digit. Αυτό που ακολουθεί είναι η σύγκριση, η αναζήτηση συγκεκριμένων συσχετίσεων ή αντιφάσεων μεταξύ των διαφόρων τύπων ή μεταβλητών, ασυνήθιστων ομαδοποιήσεων στον χώρο και τον χρόνο ή άλλα μοντέλα. Τα μοντέλα με τη σειρά τους οδηγούν σε υποθέσεις και εν ευθέτω χρόνω σε συμπεράσματα σχετικά με τις αλλαγές της ανθρώπινης συμπεριφοράς, σε ομοιότητες και διαφορές μεταξύ ομάδων ανθρώπων στο παρελθόν, σε παράγοντες που επηρεάζουν τις ανθρώπινες συμπεριφορές, στον τρόπο με τον οποίο ενδεχομένως όλες αυτές οι υποθέσεις ή τα συμπεράσματα συμβάλλουν στη σύγχρονη ζωή⁷⁵³.

5.3. Digit

Η Digit είναι εφαρμογή λογισμικού που χρησιμοποιείται για τη συλλογή και την ενσωμάτωση δεδομένων σε βάση δεδομένων. Λειτουργεί είτε τοπικά (intranet) είτε διαδικτυακή (internet), σε διακομιστή IIS (Microsoft) ή σε διακομιστή Apache (open source) με τη χρήση του αρθρώματος Μονο ανεξάρτητα από την πλατφόρμα λειτουργίας. Είναι λειτουργική είτε σε πανεπιστημιακό διακομιστή, προσβάσιμο από πολλαπλές ανασκαφές, είτε στον υπολογιστή (desktop), ακόμα και laptop ενός ερευνητή. Είναι γραμμένη σε γλώσσα C# και χρησιμοποιεί ως παρασκήνιο (backend) σχεσιακή βάση

⁷⁵² Kottak 2011, 28-29.

⁷⁵³ Banning, 2002, 1-2.

δεδομένων MySQL, η οποία μπορεί να χρησιμοποιηθεί για την ταυτόχρονη εξυπηρέτηση πολλαπλών διαδικασιών.

Η Digit δοκιμάστηκε ως δικτυακή εφαρμογή κατά τις ανασκαφές 2003-2006 στην αρχαία Φαλάσσαρνα, υπό την επίβλεψη του Μανώλη Στεφανάκη και της Ελπίδας Χατζηδάκη. Η αποκτηθείσα εμπειρία και οι σχετικές εξελίξεις του λογισμικού παρουσιάστηκαν αρχικά στο 10ο Διεθνές Κρητολογικό Συνέδριο⁷⁵⁴. Από το 2006, η Digit έγινε το βασικό λογισμικό του KARP (Kymissala Archaeological Project). Η παρούσα φάση ανάπτυξης είναι η σταθερή έκδοση "Digit V 3.0", και υποστηρίζει ως Σύστημα Διαχείρισης Περιεχομένου (CMS) την *Ευλιμένη*, ένα επιστημονικό περιοδικό που δημοσιεύει άρθρα στην κλασική αρχαιολογία, την επιγραφική, τη νομισματική και την παπυρολογία⁷⁵⁵.

Η πειραματική πρακτική της ενσωμάτωσης της διαχείρισης των αρχαιολογικών δεδομένων με τις δυνατότητες ψηφιακής αποθήκευσης είναι, κατά κάποιο τρόπο, η εισβολή της τεχνολογίας των συστημάτων πληροφορικής σε διάφορους πολιτιστικούς τομείς - μια μορφή πολιτιστικής τεχνολογίας- με αυξανόμενους ρυθμούς τις τελευταίες δύο δεκαετίες. Σημαντικό πρόβλημα από την αρχή αυτής της νέας πρακτικής ήταν ο ορισμός της ποιότητας των ψηφιακών πληροφοριών, η ευκολία πρόσβασης και η δυνατότητα αναφοράς διαφόρων ροών δεδομένων. Εξαρχής η εστίαση ήταν η διατήρηση της πληροφορίας και στη συνέχεια ο χειρισμός της με διαφορετικούς τρόπους⁷⁵⁶. Η χρήση ηλεκτρονικών υπολογιστών και συναφών λογισμικών σε αρχαιολογικά πεδία προσφέρει σαφή πλεονεκτήματα, αλλά η «επιτυχής εφαρμογή απαιτεί εξαρχής μεγάλο προγραμματισμό»⁷⁵⁷.

Οι σχεσιακές βάσεις δεδομένων παρέχουν ταχύτερη πρόσβαση και έχουν ενσωματωμένους μηχανισμούς διαχείρισης της ταυτόχρονης πρόσβασης στα δεδομένα. Το λογισμικό Digit είναι ένα εξειδικευμένο Σύστημα Διαχείρισης Περιεχομένου (CMS) γραμμένο σε C# και χρησιμοποιεί στο παρασκήνιο (back-end) MySQL. Η MySQL είναι ανοικτού πηγαίου κώδικα λογισμικό, χαμηλού

⁷⁵⁴ Καλογερόπουλος, 2006.

⁷⁵⁵ *Ευλιμένη*, μελέτες στην κλασική αρχαιολογία, την επιγραφική, τη νομισματική και την παπυρολογία, διαθέσιμη online.

⁷⁵⁶ Martinez 2001, 9-16.

⁷⁵⁷ Dibble and McPherron 1988, 431-440.

κόστους και υψηλής απόδοσης. Είναι φορητή και υποστηρίζεται από λειτουργικά συστήματα UNIX και MS Windows⁷⁵⁸.

Με τη χρήση του όρου σύστημα διαχείρισης περιεχομένου, εννοούνται προσαρμοσμένες εφαρμογές που βασίζονται σε σχήματα για την αποθήκευση περιεχομένου σε σχεσιακές βάσεις δεδομένων⁷⁵⁹. Η σχεσιακή βάση δεδομένων του CMS "Digit" αποτελείται από κύριους πίνακες που περιγράφουν οντότητες σε μια δεδομένη ιεραρχία και παραμετρικούς πίνακες, που περιγράφουν χαρακτηριστικά αυτών των οντοτήτων σε μια ανασκαφική μονάδα (context). Οι πίνακες αποθηκεύουν ευρετηριασμένα τα δεδομένα, γεγονός που καθιστά δυνατή την αναζήτηση.

Ο παρακάτω πίνακας παρουσιάζει μια συγκεκριμένη ταξινόμηση δεδομένων για οντότητες, όπως ανασκαφικές φάσεις, γεωγραφικές αρχαιολογικές θέσεις, αρχαιολογικές θέσεις, ευρήματα και ανεξάρτητες αλλά σχετικές οντότητες, όπως είναι τα στρώματα, η χρονολόγηση οι χρήστες και οι συνεδρίες. Κάθε φάση είναι μια διακριτή χρονική περίοδος κατά τη διάρκεια της οποίας διερευνάται μια αρχαιολογική γεωγραφική θέση και ανασκάπτεται μια συγκεκριμένη αρχαιολογική θέση που ανήκει στην ευρύτερη γεωγραφική περιοχή. Η ανασκαφή είναι μια διαδικασία που λαμβάνει χώρα σε σκάμματα (contexts) για να αποκαλύψει το αρχαιολογικό αρχείο, μέρος του οποίου αποτελείται από ευρήματα των σκαμμάτων. Στην πραγματικότητα υπάρχει μια σχέση «ένα προς πολλά» σε αυτό το είδος ταξινόμησης, το οποίο επιτρέπει σε κάθε χειριστή να εισάγει και να εξάγει δεδομένα, χρησιμοποιώντας τα κατάλληλα ερωτήματα SQL.

Πίνακας	Τύπος	Σχόλια
articles	InnoDB	Δημοσιεύσεις σχετικές με την ανασκαφή και τη λειτουργία του λογισμικού Digit (οντότητα)
contcategories	InnoDB	Κατηγορίες σκαμμάτων (παραμετρικός πίνακας ιδιοτήτων της οντότητας conts)
contconsists	InnoDB	Άμμος, λίθοι, συσσωματώσεις, ρίζες κ.λπ (παραμετρικός πίνακας ιδιοτήτων της οντότητας conts)

⁷⁵⁸ Converse et al 2004, 6.

⁷⁵⁹ Shklar 2003, 219.

contcorces	InnoDB	Χούμος, αποθέσεις, στρώμα οστρέων, στρώμα καταστροφής, τεχνητή απόθεση, πάτωμα κ.λπ (παραμετρικός πίνακας ιδιοτήτων της οντότητας conts)
contedgedefs	InnoDB	Ο ορισμός της κόψης ως κάθετη, κλιμακωτή κεκλιμένη (παραμετρικός πίνακας ιδιοτήτων της οντότητας conts)
contfindcategories	InnoDB	Τέχνηρα, οικοδομημένα, κτήρια, κατασκευές (παραμετρικός πίνακας ιδιοτήτων της οντότητας contfinds)
contfindgroupings	InnoDB	Ομαδοποιήσεις τεχνέργων και συλλογική αρίθμηση (παραμετρικός πίνακας ιδιοτήτων της οντότητας contfinds)
contfindimages	InnoDB	Φωτογραφίες ευρημάτων (παραμετρικός πίνακας ιδιοτήτων της οντότητας contfinds)
contfinds	InnoDB	Τα εύρηματα στο πλαίσιο του σκάμματος (οντότητες)
contfindtypes	InnoDB	Τύποι ευρημάτων, όπως κεραμικά, λίθινα μεταλλικά, υάλινα τέχνηρα, καύσεις οικοδομημένα (παραμετρικός πίνακας ιδιοτήτων της οντότητας contfinds)
contfindsubtypes	InnoDB	Υποκατηγορίες των τύπων όπως όστρακα, πλήρη αγγεία, λίθινα εργαλεία, οστά, κ.λπ (παραμετρικός πίνακας ιδιοτήτων της οντότητας contfinds)
contgrades	InnoDB	Αποτυπώματα (πασσαλότρυπες) (παραμετρικός πίνακας ιδιοτήτων της οντότητας conts)
contgrids	InnoDB	Ενοποιημένα ή διαφορετικά τετράγωνα ανασκαφής (οριζόντια ή κάθετη ανασκαφή) (παραμετρικός πίνακας ιδιοτήτων της οντότητας conts)
contmethods	InnoDB	Μέθοδος ανασκαφής (παραμετρικός πίνακας ιδιοτήτων της οντότητας conts)
contprofiles	InnoDB	Οριζόντιο, επικλινές προφίλ σκάμματος, κ.λπ (context attribute) (παραμετρικός πίνακας ιδιοτήτων της οντότητας conts)

conts	InnoDB	Ανασκαφικά σκάμματα, ανεξαρτήτως σχήματος και έκτασης (οντότητα)
contslopes	InnoDB	Βαθμοί κλίσης εδάφους σε επικλινές προφίλ (παραμετρικός πίνακας ιδιοτήτων της οντότητας conts)
conttextures	InnoDB	Υφή εδάφους σκάμματος: συμπαγής, πυκνή, λεπτή κ.λπ. (παραμετρικός πίνακας ιδιοτήτων της οντότητας conts)
conttypes	InnoDB	Τύπος εδάφους σκάμματος: απόθεση, επιφανειακό στρώμα, κ.λπ. (παραμετρικός πίνακας ιδιοτήτων της οντότητας conts)
countries	InnoDB	Αυτόματος γεωεντοπισμός
datingmethods	InnoDB	Μέθοδος χρονολόγησης (παραμετρικός πίνακας ιδιοτήτων της οντότητας datings)
datings	InnoDB	Χρονολόγηση (οντότητα)
datingtypes	InnoDB	Τύπος χρονολόγησης: Σχετική, απόλυτη (παραμετρικός πίνακας ιδιοτήτων της οντότητας datings)
diary	InnoDB	Ημερολογιακές καταχωρίσεις για συγκεκριμένα σκάμματα (παραμετρικός πίνακας της οντότητας conts)
files	InnoDB	Αρχεία συνδεδεμένα με τις ημερολογιακές καταχωρίσεις (παραμετρικός πίνακας της οντότητας conts)
galleries	InnoDB	Φωτοθήκες (οντότητες)
gallerycategories	InnoDB	Κατηγορίες φωτοθηκών (παραμετρικός πίνακας της οντότητας galleries)
galleryphotos	InnoDB	Φωτογραφίες φωτοθηκών (παραμετρικός πίνακας της οντότητας galleries)
geosites	InnoDB	Γεωγραφικές αρχαιολογικές θέσεις (οντότητες)
phases	InnoDB	Ανασκαφικές φάσεις (οντότητες)
sessions	InnoDB	Αποθήκευση των συνεδριών (sessions) χρηστών
siteclimates	InnoDB	Το κλίμα της ευρύτερης περιοχής (παραμετρικός πίνακας της οντότητας sites)
siteexctechs	InnoDB	Τεχνικές ανασκαφής (παραμετρικός πίνακας της οντότητας sites)

siteexctypes	InnoDB	Τύποι ανασκαφής: κάθετη, οριζόντια κ.ά (παραμετρικός πίνακας της οντότητας sites)
siteforms	InnoDB	Μορφές αρχαιολογικής θέσεις: νεκρόπολη ακρόπολη, σπήλαιο, λιμνήν, οχύρωση κ.λπ (παραμετρικός πίνακας της οντότητας sites)
siteinvmethods	InnoDB	Μεθοδολογία έρευνας: επίγεια, αέρια, κ.λπ (παραμετρικός πίνακας της οντότητας sites)
sites	InnoDB	Αρχαιολογικές θέσεις (οντότητες)
stratumcolors	InnoDB	Χρώμα στρώματος (παραμετρικός πίνακας της οντότητας strata)
stratums	InnoDB	Στρώματα (strata) (οντότητες)
substratums	InnoDB	Στρώσεις (substrata) (παραμετρικός πίνακας της οντότητας strata)
usergrants	InnoDB	Δικαιώματα χρηστών (παραμετρικός πίνακας της οντότητας users)
users	InnoDB	Χρήστες (οντότητες)

Πιν. 1. Η σχεσιακή βάση δεδομένων του λογισμικού Digit

Η διαχείριση και η παραγωγική χρήση του αρχαιολογικού αρχείου είναι ένα περίπλοκο και επίπονο έργο, το οποίο απαιτεί προσεκτικό προγραμματισμό και οργανωτική εμπειρία. Μέχρι πρόσφατα, η αρχαιολογική καταγραφή χρησιμοποιούσε αναλογικά μέσα τεκμηρίωσης, όπως σημειωματάρια, καταλόγους ευρημάτων, δισδιάστατα αναλογικά σχέδια και φωτογραφίες. Αυτά τα μέσα, ωστόσο, παρά τη συμβολή τους στη διαδικασία καταγραφής, καθιστούν δύσκολη την πρόσβαση και τη διασταύρωση των αρχείων δεδομένων, περιορίζοντας έτσι τις ερμηνευτικές ικανότητες των ερευνητών.

Οι ψηφιακές βάσεις δεδομένων, ωστόσο, είναι σε θέση να παρουσιάζουν παραπομπές και κάθε είδους οπτικοακουστικό υλικό που έχει εισαχθεί από τους διαχειριστές της. Αναμφισβήτητα η τεκμηρίωση μιας ανασκαφής είναι ένας ειδικός χώρος της ψηφιακής τεχνολογίας, εξαιτίας των θεωρητικών, μεθοδολογικών και πρακτικών ιδιοτεροτήτων κάθε ανασκαφής. Οι αυξανόμενες τεχνολογικές εξελίξεις και η συνεχιζόμενη συσσώρευση εμπειριών στην ανάπτυξη ψηφιακών πληροφοριακών συστημάτων για αρχαιολογική χρήση

ανοίγουν νέες προοπτικές για την αποτελεσματική και καινοτόμο υιοθέτηση της ψηφιακής τεχνολογίας στην έρευνα⁷⁶⁰.

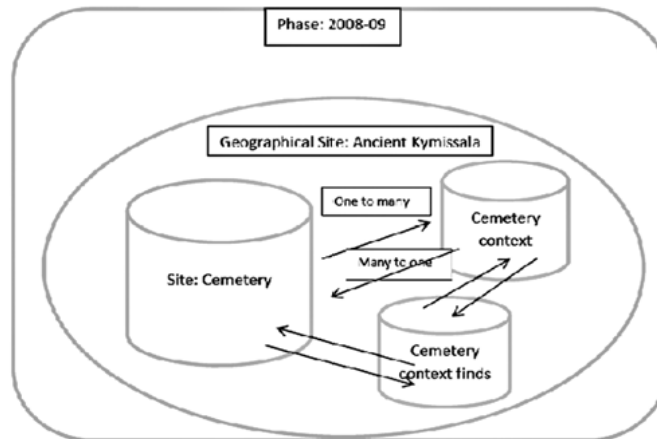
Ιεραρχικά μοντέλα δεδομένων και σχέσεις πινάκων

Η σχέση ένα προς πολλά (1:n), ή το αντίστροφο, επιτρέπει στην εγγραφή ενός πίνακα να συνδέεται σε πολλές εγγραφές ενός άλλου πίνακα χωρίς τους περιορισμούς που επιβάλλονται με την προσφυγή σε περιττά ή περιορισμένο αριθμό πεδίων ενός μόνο πίνακα⁷⁶¹. Κάτι τέτοιο μειώνει το μέγεθος της βάσης δεδομένων και αυξάνει την ευελιξία και την απόδοση των ερωτημάτων που λειτουργούν σε αυτά τα δεδομένα⁷⁶². Έτσι, σύμφωνα με τις απαιτήσεις της ανασκαφής που πραγματοποιείται στην Κυμισάλα, μια συγκεκριμένη ανασκαφική χρονική περίοδος, μια *φάση*, περιλαμβάνει χωρικές οντότητες όπως η *γεωγραφική αρχαιολογική θέση* (geosite), η οποία περιλαμβάνει μια σειρά διακριτών αρχαιολογικών θέσεων (sites). Πρόκειται για μια καθορισμένη γεωγραφική περιοχή με εξίσου καθορισμένα τοπογραφικά χαρακτηριστικά. Αυτή η οριοθετημένη περιοχή είναι και καλά καθορισμένη οικολογική και πολιτιστική περιοχή, μια συγκεκριμένη θέση ή θέσεις που απαρτίζονται από καθορισμένα πλαίσια (τετράγωνα, σκάμματα, contexts). Η συγκεκριμένη χωροχρονική σύνδεση με ένα ανασκαφικό συμβάν, επιτρέπει τη δημιουργία διακριτών αλλά παράλληλων ανασκαφών. Με άλλα λόγια στην ίδια ανασκαφική φάση είναι δυνατόν να συνδεθούν και να γίνουν αντικείμενο επεξεργασίας πολλαπλές ανασκαφές αυτής της ευρύτερης επικράτειας που αποκαλείται γεωγραφική αρχαιολογική θέση. Εν προκειμένω να συνδεθεί με μια συγκεκριμένη ανασκαφική φάση, το ανασκαφικό έργο που διεξάγεται στον Τομέα I στον Τομέα II κ.ο.κ. Τέτοιου είδους σχεδίαση –ουσιαστικά διαμερισματοποίηση- στη βάση δεδομένων εξυπηρετεί στην πολυδιάσπαση του ανασκαφικού έργου, χωρίς την ανάγκη διαμόρφωσης ξεχωριστών βάσεων δεδομένων.

⁷⁶⁰ Stefanakis et al 2015, 310,

⁷⁶¹ Nixon 2012, 216.

⁷⁶² Erickson 2009, 208.



Εικ. 5.5. Οι σχέσεις ένα προς πολλά και το αντίστροφο στη βάση δεδομένων της Digit.

Αρθρωτός σχεδιασμός και επεκτασιμότητα

Το CMS Digit είναι σε θέση να αποθηκεύει, να ανακτά, να επεξεργάζεται, να ενημερώνει και να ελέγχει το περιεχόμενο που είναι αποθηκευμένο σε μια βάση δεδομένων με τυπικά εργαλεία σύνταξης - Γραφική Διεπαφή Χρήστη (GUI). Επιτρέπει στους δημιουργούς περιεχομένου να υποβάλλουν περιεχόμενο χωρίς καμία τεχνική γνώση HTML ή την ανάγκη μεταφόρτωσης αρχείων στο διακομιστή. Η "Digit" έχει επίσης ένα αρθρωτό σχεδιασμό που επιτρέπει την ανάπτυξη πολλαπλών ερωτημάτων και απαντήσεων, σε δομημένη γλώσσα ερωτήσεων (Structured Query Language - SQL), και είναι εφαρμογή από την πλευρά του διακομιστή (side server).

Στην πράξη, αυτό σημαίνει ότι μπορεί να επεκταθεί ανάλογα μέσω νέων μονάδων⁷⁶³, χωρίς να αλλάξει ο πυρήνας της εφαρμογής λογισμικού. Είναι επίσης σε θέση να παρέχει το κατάλληλο περιβάλλον για τη δημοσίευση περιεχομένου. Η λειτουργικότητα δημιουργίας περιεχομένου περιλαμβάνει τον διαχωρισμό της παρουσίασης και του περιεχομένου, τη χρήση εικόνων σε διαφορετικά περιβάλλοντα και την υποστήριξη μεταδεδομένων. Άλλα χαρακτηριστικά επιτρέπουν την ενσωμάτωση πολλαπλών φορμά στην απόδοση κειμένου.

Έχοντας κατά νου ότι ένα σύστημα διαχείρισης περιεχομένου, ακόμη και εκείνο που κατασκευάζεται για συγκεκριμένη χρήση, είναι μια έννοια και όχι

⁷⁶³ Η μονάδα απαντάται και ως άρθρωμα στην ελληνική βιβλιογραφία.

προϊόν και αγκαλιάζει ένα σύνολο διαδικασιών⁷⁶⁴, το λογισμικό της Digit χειρίζεται όλες τις απαραίτητες διαδικασίες για την τεκμηρίωση του αρχαιολογικού μητρώου, από ημερήσια αρχεία καταγραφής, έως την ταξινόμηση και απεικόνιση των ευρημάτων. Επιπλέον, μπορεί να εξάγει πληροφορίες, γενικές ή συγκεκριμένες, σε διάφορες μορφές, από απλό κείμενο έως έγγραφα PDF. Σε κάθε περίπτωση, τόσο οι προγραμματιστές τέτοιων ειδικών CMS όσο και οι αρχαιολόγοι θα πρέπει να λάβουν υπόψη ότι τα δεδομένα δεν είναι πραγματικά αντικείμενα –είναι μετρήσεις μόνον και παρατηρήσεις που κάνουμε σχετικά με αυτά τα αντικείμενα, και λίγο-πολύ κατασκευές ή εντυπώσεις για τον τρόπο που βλέπουμε τα πράγματα⁷⁶⁵.

5.4. Εργαλεία

Παρακάτω περιγράφονται τα εργαλεία που χρησιμοποιήθηκαν για την κατασκευή του λογισμικού και η αρχιτεκτονική του συστήματος

Για τη διαμόρφωση της βάσης δεδομένων χρησιμοποιήθηκε το ανοικτού κώδικα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (database management system - RDBMS) MySQL Workbench και συγκεκριμένα η έκδοση 5.7.22 που κυκλοφόρησε στις 21 Οκτωβρίου, 2015. Το σύστημα διαχείρισης MySQL χρησιμοποιεί ως γλώσσα προγραμματισμού τη C και τη C++. Ο SQL συντακτικός αναλυτής της MySQL (parser) γράφτηκε αρχικά σε γραμματική YACC⁷⁶⁶, αλλά σταδιακά από την έκδοση 5.1 του MySQL Workbench χρησιμοποιείται ένα άλλο εργαλείο αναγνώρισης γλώσσας το ANTLR⁷⁶⁷. Ο συντακτικός αναλυτής στις γλώσσες προγραμματισμού αναλύει τον κώδικα σε επί μέρους τμήματα, τα οποία χρησιμοποιούνται για την κατασκευή μεταγλωττιστών (compilers)⁷⁶⁸ και υπολογίζει τη συντακτική δομή από συμβολοσειρές (strings)⁷⁶⁹.

Ο ένας λόγος προτίμησης της MySQL για την εφαρμογή Digit είναι ότι συνεργάζεται με πολλές ψηφιακές πλατφόρμες (λειτουργικά συστήματα) και ότι

⁷⁶⁴ Browning and Lowndes 2001

⁷⁶⁵ Sullivan 1978, 189.

⁷⁶⁶ Pipes and Kruckenberg 2006, 136.

⁷⁶⁷ Lischke 2015, διαθέσιμο online.

⁷⁶⁸ Puntambekar 2010, 1-4.

⁷⁶⁹ Metsker 2001, 7.

προσφέρεται με άδεια GPL 2, για εγχειρήματα ανοικτού κώδικα⁷⁷⁰. Επιπλέον, η υποστήριξη του MySQL Workbench από το επίσημο εγχειρίδιο για την community version είναι δωρεάν⁷⁷¹.

Ένας δεύτερος λόγος προτίμησης είναι ότι γενικά η MySQL έχει λάβει θετικές κριτικές για την ταχύτητα και τη σταθερότητά της και η διαδικασία δημιουργίας λογαριασμών με ειδικά επίπεδα πρόσβασης στα δεδομένα την καθιστούν πολυχρηστική⁷⁷² και πολυνηματική⁷⁷³. Για τη βελτίωση του πολυχρονισμού (concurrency), πολλαπλά νήματα μπορούν να λειτουργούν με τη μνήμη cache ταυτόχρονα⁷⁷⁴. Για τη δημιουργία της βάσης δεδομένων digit, των πινάκων, των views και των select statements που υπάρχουν στον πηγαίο κώδικα χρησιμοποιήθηκε το MySQL Workbench 6.3.10 για Microsoft Windows⁷⁷⁵.

Γλώσσα προγραμματισμού

Η υψηλή γλώσσα προγραμματισμού που χρησιμοποιήθηκε είναι η C#. Η C# παρέχει πλήρη υποστήριξη για τον αντικειμενοστρεφή προγραμματισμό, συμπεριλαμβανομένης της ενθυλάκωσης⁷⁷⁶, της κληρονομικότητας⁷⁷⁷ και του πολυμορφισμού⁷⁷⁸.

Αναπτύχθηκε από τη Microsoft, έχει συντακτικό που βασίζεται στην C/C++, και χρησιμοποιείται για όλα τα είδη εφαρμογών (desktop, web, mobile). Ανήκει στις μεταγλωττισμένες γλώσσες (compiled languages) ενδιάμεσης μορφής, που χρειάζονται να διερμηνευθούν, προκειμένου να εκτελεστούν⁷⁷⁹. Εκτελείται από εικονική μηχανή (virtual machine) και αποτελεί τμήμα του .NET Framework. Η

⁷⁷⁰ Oracle 2010, διαθέσιμο online.

⁷⁷¹ Oracle 2019, διαθέσιμο online.

⁷⁷² Norman 2003, 135.

⁷⁷³ Reese et al 2002, 7.

⁷⁷⁴ Schwartz et al 2008, 295.

⁷⁷⁵ Oracle 2019b, διαθέσιμο online.

⁷⁷⁶ Απόκρυψη των δεδομένων και της υλοποίησης ενός αντικειμένου, γνωστή και ως *απόκρυψη δεδομένων* (data hiding). Βλ. Hunt 2002, 12.

⁷⁷⁷ Η δυνατότητα δημιουργίας νέων κλάσεων (υποκλάσεων), βασισμένων σε υπάρχουσα (υπερ)κλάση πρόγονο. Βλ. Rama Krishna Rao 2006, 141.

⁷⁷⁸ Η δυνατότητα συνύπαρξης πολλαπλών κλάσεων που μπορούν να χρησιμοποιηθούν εναλλακτικά, παρόλο που κάθε κλάση εφαρμόζει τις ίδιες ιδιότητες ή μεθόδους με διαφορετικούς τρόπους, Βλ. Deitel and Deitel 2005, 390.

⁷⁷⁹ Griffiths et al 2010, 20.

γλώσσα έγινε αποδεκτή ως διεθνές πρότυπο (ECMA-334)⁷⁸⁰ και ISO (ISO/IEC 23270:2006)⁷⁸¹. Η C# ως γλώσσα προγραμματισμού είναι σχεδιασμένη για την Υποδομή Κοινής Γλώσσας (CLI)⁷⁸². Όντας αντικειμενοστρεφής, προωθείται ως φυσική επιλογή για προγραμματισμό στην πλατφόρμα .NET που περιλαμβάνει κοινή μηχανή εκτέλεσης και μια πλούσια βιβλιοθήκη κλάσεων. Προβάλλεται, επίσης, ο ισχυρισμός ότι προωθεί τον αντικειμενοστρεφή προγραμματισμό, καθώς και ότι ενθαρρύνει τις καλές πρακτικές σχεδιασμού⁷⁸³. Η C# μοιάζει με την Java και τη C++ από πολλές απόψεις, αλλά είναι θεμελιωδώς διαφορετική ως προς τη συμπεριφορά της⁷⁸⁴.

Όσον αφορά στην υποδομή κοινής γλώσσας (CLI) για την οποία είναι σχεδιασμένη η C#⁷⁸⁵ είναι μια ανοιχτή προδιαγραφή (τεχνικό πρότυπο) που αναπτύχθηκε από τη Microsoft και τυποποιήθηκε από ISO⁷⁸⁶ και ECMA⁷⁸⁷. Η CLI περιγράφει τον εκτελέσιμο κώδικα και το περιβάλλον εκτέλεσης που επιτρέπει σε πολλαπλές υψηλές γλώσσες να χρησιμοποιούνται σε διαφορετικές πλατφόρμες υπολογιστή, χωρίς να ξαναγράφεται για ειδικές αρχιτεκτονικές. Αυτό σημαίνει ότι είναι αγνωστική πλατφόρμα⁷⁸⁸. Το .NET Framework και το Mono, ειδικού ενδιαφέροντος για την εκτέλεση του λογισμικού Digit σε περιβάλλον Apache (.NET σε περιβάλλον Linux), είναι εφαρμογές του CLI.

Γενικά το .NET Framework είναι πλατφόρμα λογισμικού που αναπτύχθηκε από τη Microsoft και λειτουργεί κυρίως σε λειτουργικό σύστημα Microsoft Windows. Περιλαμβάνει μεγάλη βιβλιοθήκη κλάσεων Framework Class Library (FCL) και παρέχει γλωσσική διαλειτουργικότητα σε διάφορες γλώσσες προγραμματισμού⁷⁸⁹. Αυτό πρακτικά σημαίνει ότι κάθε γλώσσα μπορεί να χρησιμοποιεί κώδικα γραμμένο σε άλλες γλώσσες. Τα προγράμματα που γράφονται για το .NET Framework εκτελούνται σε περιβάλλον λογισμικού.

Το Mono είναι ένα ελεύθερο και ανοικτού κώδικα πρόγραμμα για τη δημιουργία πλατφόρμας λογισμικού συμβατής με το πρότυπο ECMA, το οποίο

⁷⁸⁰ ECMA International 2017, διαθέσιμο online.

⁷⁸¹ ISO/IEC 2018, διαθέσιμο online.

⁷⁸² Watkins et al 2003, 348.

⁷⁸³ Williams 2002, 34-36.

⁷⁸⁴ Grosso et al 2003, 4.

⁷⁸⁵ Wills 2007, 61.

⁷⁸⁶ ISO 2012, διαθέσιμο online.

⁷⁸⁷ ECMA 2013, διαθέσιμο online.

⁷⁸⁸ Stutz et al 2003, 22.

⁷⁸⁹ Kogent Solutions 2008, 2.

περιλαμβάνει μεταγλωττιστή (compiler) C# και εκτέλεση κοινής γλώσσας (Common Language Runtime), δυνατότητες που ενδιαφέρουν για τη λειτουργία της Digit, σε περιβάλλον Linux. Οι βιβλιοθήκες των βασικών κλάσεων είναι συμβατές με την πλατφόρμα .NET Framework με προτυποποίηση ECMA⁷⁹⁰. Ο δηλωμένος στόχος του εγχειρήματος δεν ήταν μόνο η ανάπτυξη της δυνατότητας εκτέλεσης εφαρμογών Microsoft .NET σε πολλαπλές πλατφόρμες, αλλά και η δημιουργία εργαλείων ανάπτυξης για προγραμματιστές του Linux⁷⁹¹.

Πλατφόρμα ανάπτυξης της Digit

Οι web σελίδες δημιουργήθηκαν σε περιβάλλον Asp.NET Webforms. Το ASP.NET Web Forms είναι πλαίσιο εφαρμογών ιστού και ένα από τα πολλά μοντέλα προγραμματισμού που υποστηρίζονται από την τεχνολογία Microsoft ASP.NET. Οι εφαρμογές Web Forms μπορούν να γραφτούν σε οποιαδήποτε γλώσσα προγραμματισμού υποστηρίζει το Common Language Runtime, όπως η C# ή η Visual Basic. Τα βασικά δομικά στοιχεία των σελίδων Web Forms είναι στοιχεία ελέγχου διακομιστή, τα οποία είναι επαναχρησιμοποιήσιμα και είναι υπεύθυνα για την απόδοση σήμανσης HTML και την ανταπόκριση σε γεγονότα. Στην προκειμένη περίπτωση χρησιμοποιεί την τεχνική που ονομάζεται κατάσταση προβολής (view state) για να διατηρεί την κατάσταση των ελέγχων του διακομιστή μεταξύ κυκλικά επαναλαμβανόμενων αιτημάτων HTTP⁷⁹².

Άλλες πλατφόρμες που χρησιμοποιήθηκαν στην Digit για να υποστηρίξουν την εμπειρία του χρήστη από την πλευρά του φυλλομετρητή (client side) και το διαχειριστικό τμήμα της είναι τα HTML5, CSS3, jquery-3.3.1, jquery-ui-1.12.1⁷⁹³, CKEditor⁷⁹⁴, ImageResizer.3.4.3⁷⁹⁵, όλα ανοικτού κώδικα. Για τον γεωεντοπισμό χρησιμοποιήθηκε η υπηρεσία Google Maps, αν και υπάρχει η δυνατότητα χρήσης άλλων εργαλείων, όπως το WorldWind Java⁷⁹⁶, που αποτελεί και μελλοντική προοπτική για την Digit.

⁷⁹⁰ Nayyeri and White 2014, 167.

⁷⁹¹ Caton, M. 2001, διαθέσιμο online.

⁷⁹² Esposito 2009, 42.

⁷⁹³ Meloni and Kyrnin 2018.

⁷⁹⁴ CKEditor Ecosystem 2019, διαθέσιμο online.

⁷⁹⁵ Nuget 2019, διαθέσιμο online.

⁷⁹⁶ Ryan 2017, 62.

Περιβάλλον ανάπτυξης

Ως περιβάλλον ανάπτυξης χρησιμοποιήθηκε το Visual Studio Community 2015 από την Microsoft. Το Visual Studio παρέχεται δωρεάν, αλλά δεν είναι ανοικτού κώδικα. Υποστηρίζει διαφορετικές γλώσσες προγραμματισμού και επιτρέπει στον επεξεργαστή κώδικα και στον εντοπιστή σφαλμάτων να υποστηρίζει (σε διαφορετικούς βαθμούς) σχεδόν οποιαδήποτε γλώσσα προγραμματισμού. Η C#, εν προκειμένω είναι ενσωματωμένη στο Visual Studio⁷⁹⁷.

Η έκδοση του Visual Studio Community 2015 είναι ολοκληρωμένο περιβάλλον ανάπτυξης (integrated development environment, IDE), όπως και η Professional έκδοση, αλλά απευθύνεται σε φοιτητές, μικρές ομάδες προγραμματιστών και εγχειρήματα ανοιχτού πηγαίου κώδικα. Βασισμένη στην Professional, χρησιμοποιεί εργαλεία υψηλής παραγωγικότητας, ενσωματωμένα στη σουίτα λογισμικού. Επίσης, υποστηρίζει πρόσθετα (plug-ins) που στοχεύουν στην αύξηση της παραγωγικότητας και επιτρέπουν στους προγραμματιστές να στοχεύουν σε πολλαπλές πλατφόρμες⁷⁹⁸.

Μεθοδολογία

Για την Digit χρησιμοποιήθηκε η προσέγγιση της αντικειμενοστρεφούς ανάλυσης και σχεδιασμού (Object oriented analysis and design - OOAD). Ως αντικειμενοστρεφής ανάλυση εννοείται η ανάπτυξη ενός αντικειμενοστρεφούς μοντέλου ενός συστήματος λογισμικού⁷⁹⁹. Είναι το μοντέλο των λειτουργικών απαιτήσεων του συστήματος και είναι ανεξάρτητο από τους περιορισμούς υλοποίησης. Ο αντικειμενοστρεφής σχεδιασμός στρέφεται προς την ανάπτυξη ενός αντικειμενοστρεφούς μοντέλου για την υλοποίηση των απαιτήσεων της εφαρμογής. Η αντικειμενοστρεφής ανάλυση και σχεδιασμός χρησιμοποιούν αντικειμενοστρεφή προσέγγιση για την οικοδόμηση εννοιολογικών και λογικών μοντέλων του συστήματος⁸⁰⁰. Τα αντικειμενοστραφή προγράμματα χρησιμοποιούν κλάσεις, αντίθετα με τον δομημένο προγραμματισμό που

⁷⁹⁷ Novak 2011, 20.

⁷⁹⁸ Powers and Snell 2016, 13-15.

⁷⁹⁹ Βεσκούκης, Β. 2015, 468.

⁸⁰⁰ Ashrafi and Ashrafi 2014, 53.

χρησιμοποιεί ρουτίνες, οι οποίες περιέχουν τόσο δεδομένα όσο και ρουτίνες για να χειρίζονται τα δεδομένα⁸⁰¹.

Τα αντικείμενα

Τα αντικείμενα είναι ανεξάρτητες οντότητες, που ενθυλακώνουν πληροφορίες για την κατάσταση και την απεικόνισή τους. Τα αντικείμενα είναι στιγμιότυπα μιας κλάσης⁸⁰². Πραγματικά ή εικονικά, είναι εγκλωβισμένα, δηλαδή τα χαρακτηριστικά και οι λειτουργίες ενός αντικειμένου υπάρχουν μέσα σε ένα θύλακα, μια κλειστή περιοχή ή μαύρο κουτί, που χωρίζει τον ζωτικό τους χώρο σε δύο περιοχές, τον δημόσιο, εκτεθειμένο στον έξω κόσμο, και τον ιδιωτικό, γνωστό μόνο στο αντικείμενο εαυτό. Η απόκρυψη πληροφοριών είναι αποτέλεσμα και πλεονέκτημα της ενθυλάκωσης και προστατεύει τα δεδομένα που κατέχει το αντικείμενο από την κατάχρηση ή την κακή χρήση⁸⁰³.

Όπως συμβαίνει με το DNA και τη γενετική στον πραγματικό κόσμο, η κληρονομικότητα είναι ένας μηχανισμός με τον οποίο τα εικονικά αντικείμενα που ανήκουν σε μια υποκλάση, κληρονομούν τα χαρακτηριστικά της υπερκλάσης. Ο τρόπος με τον οποίο γίνεται αυτό στην πραγματικότητα ανήκει στη σφαίρα συγκεκριμένων τεχνολογιών, αλλά όλοι οι μηχανισμοί εικονικής κληρονομικότητας έχουν ένα κοινό χαρακτηριστικό γνώρισμα, είναι διαρκώς λειτουργικοί. Εάν η υπερκλάση μαθαίνει να κάνει κάτι καλύτερο, το ίδιο συμβαίνει και με όλες τις περιπτώσεις της υποκλάσης. Η υποκλάση κληρονομεί όλες τις ιδιότητες και τις μεθόδους της υπερκλάσης και στη συνέχεια μπορούν να προστεθούν οι δικές της ιδιότητες και μέθοδοι⁸⁰⁴. Είναι ένας μηχανισμός αφαίρεσης, που χρησιμοποιείται για την κατηγοριοποίηση οντοτήτων και διευκολύνει την επαναχρησιμοποίηση τόσο στο επίπεδο του σχεδιασμού όσο και στο επίπεδο του προγραμματισμού, όπου η εικονική κληρονομικότητα χρησιμοποιείται για την εξειδίκευση των στοιχείων στο πλαίσιο⁸⁰⁵. Σε πρακτικό επίπεδο η κληρονομικότητα επιτρέπει στους προγραμματιστές να επαναχρησιμοποιήσουν τους ορισμούς για δομές που διαμορφώθηκαν στο παρελθόν, μειώνοντας έτσι τον φόρτο εργασίας κατά την ανάπτυξη προγραμμάτων.

⁸⁰¹ Attaway 2017, 365-406.

⁸⁰² Craig 2001, 169.

⁸⁰³ Ashrafi and Ashrafi 2014, 58.

⁸⁰⁴ Σταματόπουλος 2015, 8, διαθέσιμο online.

⁸⁰⁵ Ciancarini et al 2013, 199.

Ο πολυμορφισμός είναι αποτέλεσμα της εξειδίκευσης και συνδέεται στενά με την κληρονομικότητα. Όλα τα θηλαστικά κινούνται, αλλά ενώ τα ελάφια κινούνται σε τέσσερα πόδια, οι άνθρωποι περπατούν στα δύο πόδια, οι πίθηκοι στα δύο ή στα τέσσερα πόδια και πηδούν από δέντρο σε δέντρο, ενώ οι νυχτερίδες πετούν⁸⁰⁶.

Όσον αφορά στον αντικειμενοστρεφή σχεδιασμό της Digit, δημιουργήθηκαν κλάσεις που αναπαριστούν τα αντικείμενα του επιπέδου εφαρμογής. Δηλαδή δημιουργήθηκε κλάση για το πλαίσιο, τη φάση, την αρχαιολογική θέση, το εύρημα και άλλα. Όλες οι κλάσεις που δημιουργήθηκαν για τη μοντελοποίηση του προβλήματος ή του χώρου στον οποίο κινείται η Digit, διαμορφώθηκαν στον ονοματοχώρο *Digit.Core.Entities* στο Visual Studio.

Κάθε κλάση έχει ιδιότητες. Για παράδειγμα η *DgCont* (που μοντελοποιεί ή αναπαριστά ένα ανασκαφικό πλαίσιο) έχει ιδιότητες όπως το *ContId*, *Phase*, *Site*, *Name*, *Summary* κ.ά. Επειδή όλες οι κλάσεις στο επίπεδο της εφαρμογής *business classes* (ή *entities*) μοιράζονται κοινές ιδιότητες (*createDt*, *createBy*, *updateDt*, *updateB*) είναι απόγονοι της κλάσης *DgObject*, στην οποία και υπάρχουν αυτές οι ιδιότητες.

Στη συνέχεια δημιουργήθηκαν οι πίνακες στους οποίους αποθηκεύονται τα δεδομένα της Digit. Κάθε κλάση στο επίπεδο της εφαρμογής (*business domain*) αντιστοιχεί σε έναν πίνακα στη βάση δεδομένων με το ίδιο όνομα, και ως στήλες οι ιδιότητες (*properties*) της κλάσης.

Ουσιαστικά, αποθηκεύεται η κλάση *DgCont* στη μνήμη της εφαρμογής (από στοιχεία που καταχωρεί ο χρήστης σε μία *web* φόρμα) σαν μία σειρά (*row*)⁸⁰⁷ στον πίνακα *conts* της *mysql* ή το αντίστροφο. Μία σειρά του πίνακα *conts* στη *mysql* μετατρέπεται σε *DgCont*, για να αποδοθεί σε *HTML*.

Η αντιστοιχία των κλάσεων και των πινάκων φαίνεται, για παράδειγμα, στην κλάση `public class DgCont` σε αντιστοιχία με τον πίνακα `conts`.

Στην `public sealed class DigitSystem` αναπτύσσονται οι μέθοδοι:

```
public DgCont GetContById(DgAccessToken accessToken, Int32 contId)
```

⁸⁰⁶ Liberty and MacDonald 2006, 194.

⁸⁰⁷ Στο πλαίσιο της σχεσιακής βάσης δεδομένων, η σειρά αντιπροσωπεύει ένα απλό, δομημένο στοιχείο δεδομένων σε έναν πίνακα. Με απλά λόγια, ένας πίνακας βάσης δεδομένων μπορεί να θεωρηθεί ότι αποτελείται από σειρές και στήλες. Βλ. Technopedia 2018, διαθέσιμο online.


```
public IList<DgCont> GetConts(DgAccessToken accessToken, string
whereClause = null, string orderByClause = "order by name")
```

```
public DgCont CreateCont(DgAccessToken accessToken, DgCont cont)
```

```
public DgCont UpdateCont(DgAccessToken accessToken, DgCont cont)
```

Οι παραπάνω μέθοδοι χρησιμοποιούν αντίστοιχες μεθόδους από την `internal sealed class DigitDataLayer` για να συνδεθούν με τη mysql:

```
public DgCont GetContById(DgAccessToken accessToken, Int32 contId)
```

```
public IList<DgCont> GetConts(DgAccessToken accessToken, string
whereClause, string orderByClause)
```

```
public DgCont CreateCont(DgAccessToken accessToken, DgCont cont)
```

```
public DgCont UpdateCont(DgAccessToken accessToken, DgCont cont)
```

Η συγκεκριμένη παραδειγματική αντιστοιχία ισχύει για όλους σχεδόν τους πίνακες και τις αντίστοιχες οντότητες στον κώδικα, στο επίπεδο της εφαρμογής.

Αρχιτεκτονική του συστήματος

Η αρχιτεκτονική του συστήματος είναι ένα κλασικό σύστημα τριών επιπέδων (3-Tier). Πρόκειται για αρχιτεκτονική Client-Server η οποία έχει ως χαρακτηριστικό της τη διαμόρφωση των τμημάτων που αποτελούν το σύνολο της εφαρμογής σε τρία επίπεδα (layers ή tiers). Ενώ οι έννοιες layer (επίπεδο) και tier (βαθμίδα) χρησιμοποιούνται συχνά εναλλακτικά, υφίσταται η άποψη πως υπάρχει διαφορά στις δύο έννοιες. Αυτή η άποψη υποστηρίζει ότι το επίπεδο είναι ένας μηχανισμός λογικής δομής για τα στοιχεία που αποτελούν τη λύση (solution) του λογισμικού, ενώ η βαθμίδα αναπαριστά ένα φυσικό δομικό μηχανισμό στην υποδομή του συστήματος⁸⁰⁸. Το αρχιτεκτονικό πρότυπο της διαστρωμάτωσης σε επίπεδα χρησιμοποιείται για εφαρμογές που είναι δυνατόν να αναλυθούν σε ομάδες λειτουργιών, όπου η κάθε ομάδα βρίσκεται σε ένα συγκεκριμένο επίπεδο αφαίρεσης⁸⁰⁹.

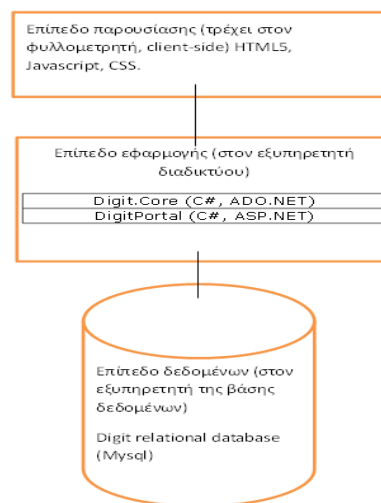
Τα πλεονεκτήματα μιας τέτοιας αρχιτεκτονικής συνοψίζονται ως εξής: Υποστηρίζονται πολλαπλές υπολογιστικές γλώσσες⁸¹⁰. Στο ενδιάμεσο επίπεδο εφαρμογής είναι δυνατόν να μοιραστεί σε διαφορετικούς εξυπηρετητές,

⁸⁰⁸ Fowler et al 2002, 33.

⁸⁰⁹ Buschmann et al., 1996, 31-51.

⁸¹⁰ Information Resources Management Association 2001, 323.

εξασφαλίζοντας ισορροπία φορτίου⁸¹¹. Περαιτέρω, εκμηδενίζονται οι πιθανοί περιορισμοί στη σύνδεση πολλαπλών χρηστών, καθώς η βάση δεδομένων επικοινωνεί μόνο με το επίπεδο εφαρμογής⁸¹². Όσον αφορά σε ζητήματα ασφαλείας, η πρόσβαση σε τμήματα του λογισμικού εξασφαλίζεται μόνο κατόπιν παροχής δικαιωμάτων από τον διαχειριστή του συστήματος⁸¹³. Τέλος, εξειδικευμένοι τεχνικοί στην ανθρώπινη διεπαφή, στις βάσεις δεδομένων, ή στον προγραμματισμό είναι δυνατόν να συμβάλλουν καθοριστικά στην ποιότητα του λογισμικού.



Εικ. 5.6. Γραφική παράσταση των τριών επιπέδων της Digit

Ο διακομιστής της βάσης δεδομένων είναι δυνατόν να βρίσκεται στην ίδια μηχανή με τον διακομιστή ιστού και τον client. Για μεγαλύτερα φορτία τοποθετείται σε μια μηχανή ο database server (mysql) και σε μία άλλη μηχανή ο web server (IIS ή Apache).

⁸¹¹ Venugopal and Patnaik 2011, 472-473.

⁸¹² Lhotka 2006, 14.

⁸¹³ Microsoft Docs, 2019, διαθέσιμο online.

Συμπεράσματα, δυνατότητες, μελλοντικές προοπτικές

Στην παρούσα έρευνα διερευνήθηκαν αναλυτικά ζητήματα αρχαιολογικής θεωρίας και μεθόδου, όπως επίσης και η σχέση αρχαιολογικής θεωρίας και υπολογιστικής αρχαιολογίας με σκοπό την ανάπτυξη εφαρμογής λογισμικού. Διαμορφώθηκε αρχαιολογική οντολογική ιεραρχία για την κατασκευή σχεσιακής βάσης δεδομένων και κλάσεις σχετικές με την αρχαιολογική θεωρία και μέθοδο. Επίσης, καταγράφηκαν τα απαραίτητα εργαλεία για την κατασκευή ενός ημερολογιοκεντρικού λογισμικού.

Δεδομένου του χρόνου ανάπτυξης, αποσφαλμάτωσης, μοντελοποίησης και δοκιμής στο πεδίο, το λογισμικό Digit μπορεί να λειτουργήσει ως εργαλείο ψηφιακής καταχώρισης, επεξεργασίας και ανάκτησης ανασκαφικών δεδομένων και επεξεργασμένων ή ανεπεξέργαστων αρχείων τεκμηρίωσης. Δεδομένου, επίσης, του ημερολογιοκεντρικού χαρακτήρα του είναι λειτουργικό τόσο στο ανασκαφικό πεδίο όσο και στο αρχαιολογικό εργαστήριο.

Είναι γενικό εργαλείο ψηφιοποίησης ανασκαφικών δεδομένων και δεν περιορίζει τις ανάγκες ιδιαίτερων ανασκαφικών πεδίων, που μοιράζονται τον ίδιο κορμό αρχαιολογικής θεωρίας. Το κενό που καλύπτει το λογισμικό είναι η έλλειψη εργαλείων ανοικτού κώδικα στο συγκεκριμένο είδος, μια και οι λιγοστές προηγούμενες προσπάθειες τέτοιου είδους αποσύρθηκαν γρήγορα από το διαδίκτυο ή διαμορφώθηκαν εξ αρχής για κλειστά συστήματα συγκεκριμένων ανασκαφών, με μοναδική πιθανώς εξαίρεση στον κανόνα το λογισμικό Ark (Archaeological Recording Kit) με τελευταία ανανέωση το 2015⁸¹⁴. Το δεύτερο κενό είναι η εξ αρχής προσαρμογή του λογισμικού στην τρέχουσα αρχαιολογική θεωρία με δυνατότητες αναβάθμισης εξαιτίας του αρθρωτού χαρακτήρα της για μεταγενέστερες θεωρητικές προσεγγίσεις.

Η δυσκολία διαμόρφωσης λογισμικών γενικής ψηφιοποίησης ανασκαφικών δεδομένων έγκειται αφενός στον κατακερματισμό της αρχαιολογικής θεωρίας σε πολύ αφηρημένα επίπεδα ανάλυσης, που δεν εξυπηρετούν την ανασκαφική διαδικασία, αφετέρου στα διαφορετικά πεδία εφαρμογής που εντοπίζουν το ενδιαφέρον τους οι αρχαιολόγοι –κυρίως GIS εφαρμογές. Στο συγκεκριμένο πεδίο της αρχαίας Κυμισάλας υπήρξε άπλετη

⁸¹⁴ Ark (Archaeological Recording Kit), διαθέσιμη online.

ελευθερία στον σχεδιασμό και τη μοντελοποίηση, ώστε να διαμορφωθεί λογισμικό γενικής και όχι ειδικής χρήσης.

Η μελλοντική ανάπτυξη της Digit προφανώς είναι έργο ομάδας τεχνικών λογισμικού, από τη διαμόρφωση τεχνικού εγχειριδίου για την εγκατάσταση και χρήση, τον σχεδιασμό της εμφάνισης, της ταχύτητας και της αμεσότητας πρόσβασης της επιθυμητών σελίδων, την επέκταση της βάσης δεδομένων για μελλοντικές απαιτήσεις, τη χρήση ανοικτών βιβλιοθηκών για τρισδιάστατες απεικονίσεις και συναφή συστήματα γεωπληροφορικής και κυρίως όλα εκείνα τα στοιχεία που θα την κάνουν εύκολη στη χρήση, από μη εξοικειωμένους χρήστες σε ζητήματα διαχείρισης περιεχομένου.

Σε κάθε επίπεδο της αρχιτεκτονικής 3 Tier του συστήματος ο ειδικευμένος τεχνικός λογισμικού είναι δυνατόν να βελτιώσει την υφιστάμενη σχεδίαση διασυνδέσεων ανθρώπου-υπολογιστή (GUI Designer), ένας προγραμματιστής C# να υλοποιήσει βελτιώσεις στον κώδικα και ένας ειδικός σχεδιασμού βάσεων δεδομένων να επεκτείνει και να βελτιστοποιήσει την εφαρμογή στο επίπεδο των δεδομένων.

Κάτι τέτοιο βέβαια είναι δυνατόν να γίνει μόνο σε επίπεδο πανεπιστημιακού εγχειρήματος, με τη συμβολή αρχαιολόγων, γεωγράφων, τοπογράφων και επιστημόνων της πολιτισμικής τεχνολογίας, προκειμένου η Digit να διατηρήσει τον ανοικτό χαρακτήρα της και να εξασφαλίσει τη συνέχειά της ως εφαρμογή της πολιτισμικής τεχνολογίας.

Βιβλιογραφία

- Aali, A., Stöllner, T., Abar, A., Rühli, F. 2012, The Salt Men of Iran: The Salt Mine of Douzlākh, Chehrābād. *Archäologisches Korrespondenzblatt. Mainz: Verlag des Römisch-Germanischen Zentralmuseums*. 42(1), 61–81.
- Aitken, M.J. 1990, *Science-based Dating in Archaeology*, London: Longman
- Ammerman, A. 1985, Plow-Zone Experiments in Calabria, Italy, *Journal of Field Archaeology*, 12, 33-40.
- Ανδρόνικος, Μ. 1961-1962, *Ελληνικά επιτάφια μνημεία*, ΑΔ 17, Μελέται, 152-210.
- Andrus, C.F.T. 2011, Shell midden sclerochronology, *Quaternary Science Reviews* 30, 2892-2905.
- Anglim Kreder, J., Nintrup J. 2014, Antiquity Meets the Modern Age: EBAY's Potential Criminal Liability for Counterfeit and Stolen International Antiquity Sales, *Journal of Law, Technology & the Internet*, (5), 143-178.
- AOC 2015, *An Introduction to LiDAR for Archaeology*, AOC Archaeology Group.
- Ascher, R. 1961, Analogy in Archaeological Interpretation, *Southwestern Journal of Anthropology*, (17)4, 317–325.
- Ascher, R. 1968, Time's arrow and the archaeology of a contemporary community, in Chang, K.C. (ed.) *Settlement archaeology*, (43-52) Palo Alto: National Press Books.
- Ashrafi, N., Ashrafi H. 2014, *Object Oriented Systems Analysis and Design*, Harlow: Pearson Education.
- Asouti, E. 2003, Wood Charcoal from Santorini (Thera): New Evidence for Climate, Vegetation and Timber Imports in the Bronze Age Aegean, *Antiquity*.
- Attaway, S. 2017, Introduction to Object-Oriented Programming and Graphics, *Matlab*, 365-406.
- Baillie, M.G.L. 1982, *Tree-ring dating and archaeology*, Chicago: The University of Chicago Press.

- Balme, J., Paterson, A. (eds.) 2006, *Archaeology in Practice: a Student Guide to Archaeological Analyses*. Malden, MA: Blackwell.
- Banning, E.B. 2002, *Archaeological Survey*, NY: Springer Science.
- Banning, E.B. 2006, *The Archaeologist's Laboratory: The Analysis of Archaeological Data*, Berlin: Springer Science.
- Barfield, T. (ed.) 1997, *The Dictionary of Anthropology*, Oxford: Blackwell.
- Barker, P. 1993, *Techniques of Archaeological Excavation*, London and NY: Routledge.
- Barrett, J., 2000, A thesis on agency, in Dobres, M.A. and J. Robb (eds), *Agency in archaeology*, (61–68) London.
- Baskerville, R. 1991. Risk Analysis as a Source of Professional Knowledge. *Computers & Security*, 10(8), 749–764.
- Beck, M.E. 2015, Natural and Cultural Formation Processes, in Carver, M., Gaydarska, B., Montón-Subías, S. (eds.) *Field Archaeology from Around the World*, (11-17) NY: Springer.
- Ben-Dor, E., Portugali, J., Kochavi, M., Shimoni, M., & Lipaz Vinitzky 1999, Airborne Thermal Video Radiometry and Excavation Planning at Tel Leviah, Golan Heights, Israel, *Journal of Field Archaeology*, (26)2, 117-127.
- Βεσκοῦκης, Β. 2015, *Στοιχεία τεχνολογίας λογισμικού*, Αθήνα: Σύνδεσμος Ελληνικών Ακαδημαϊκών Βιβλιοθηκών – ΕΜΠ.
- Bettinger, R.L. 1991. *Hunter-Gatherers: archaeological and evolutionary theory*. New York: Plenum Press.
- Bewley, R.H., Rackowski, W., (eds.), 2001, *Aerial Archaeology, Developing Future Practice*, NATO Science Series 1.337, and Amsterdam: IOS Press.
- Bianchi, T.S., Canuel, E.A. 2011, *Chemical Markers in Aquatic Ecosystems*. Princeton: Princeton University Press.
- Binford, L.R. 1964. A Consideration of Archaeological Research Design, *American Antiquity*, (29)4, 425-441.
- Binford, L.R. 1965, Archaeological Systematics and the Study of Culture Process, *American Antiquity*, 31(2), 203-210.

- Binford, S.R., Binford, L.R. (eds.) 1968, *New perspectives in archaeology*, Chicago: Aldine Publishing Company.
- Binford, L.R. (ed.) 1977. *For Theory Building in Archaeology*, New York: Academic Press.
- Binford, L.R. 1981. Behavioral archaeology and the "Pompeii premise" *Journal of Anthropological Research* 37, 195-208.
- Binford, L.R. 2009, *Debating Archaeology: Updated Edition*, Oxford: Left Coast Press.
- Birx, H.J. (ed.) 2010, *21st Century Anthropology: A Reference Handbook*, Sage Publications.
- Black, G. A., Johnston, R. B. 1962, A Test of Magnetometer as an Aid to Archaeology, *American Antiquity*, (28)2, 199-205.
- Blackman, E. 1971, Opaline Silica Bodies in the Range Grasses of Southern Alberta, *Canadian Journal of Botany* 49(5), 769-781.
- Boas, F. 1920, The Methods of Ethnology, *American Anthropologist* (22)4, 311-321.
- Boehler 2006, Comparison of 3D Laser Scanning and Other 3D Measurement Techniques, in Baltsavias, M., Gruen, A., Van Gool, L., Pateraki, M. (eds.), *Recording, Modeling and Visualization of Cultural Heritage*, (89-99), Leiden: Taylor & Francis.
- Borgman, C., Bates, M., Cloonan, M., Efthimiadis, E., Gilliland-Swetland, A., Kafai, Y., Leazer, G., Maddox, A. 1996, Social aspects of digital libraries. *Final report of the UCLA-NSF social aspects of digital libraries workshop held at UCLA, Los Angeles, USA, February, 15-17.*
- Bortolini, E. 2016, Typology and Classification, in Hunt, A. (ed.) *The Oxford Handbook of Archaeological Ceramic Analysis*, (651-671), Oxford: Oxford University Press.
- Boudon, R. 1991. Review: What Middle-Range Theories are. *Contemporary Sociology*, 20(4), 519-522.
- Bourricaud, F. 1981. *The sociology of Talcott Parsons*. Chicago: University of Chicago Press.
- Bowman, S. 1995, *Radiocarbon Dating*, London: British Museum Press.

- Boxill, I., Chambers, C., Wint, E. 1997, *Introduction to Social Research With Applications to the Caribbean*, Jamaica: University of the West Indies Press.
- Bradley, R. 2000, *An archaeology of natural places*, London: Routledge
- Bradley, R. 2006, Bridging the Two Cultures. Commercial Archaeology and the Study of Prehistoric Britain, *Antiquaries Journal*, 86, 1-13.
- Bradley, R. 2015, *Paleoclimatology: Reconstructing Climates of the Quaternary*. Oxford: Elsevier
- Bronitsky, G. 1986, The use of materials science techniques in the study of pottery construction and use, in Schiffer, M.B. (ed.) *Advances in Archaeological Method and Theory*. Vol. 9, (209-276), Orlando, Florida: Academic Press.
- Brown, A.G., Petit, F., James, L.A. 2006, Archaeology and human artefacts in Piegay, H., Kondolf, M. (eds.) *Tools in Fluvial Geomorphology*, (40-56), N.Y.: Wiley.
- Brück, J. 2005, Experiencing the past: The development of a phenomenological archaeology in British prehistory, *Archaeological Dialogues* (12)1, 45-72.
- Bryant, C.G.A., Jary, D. 1991, Coming to terms with Anthony Giddens. In C.G.A. Bryant & D. Jary (eds.), *Giddens' theory of structuration: A critical appreciation* (1-32). New York, NY: Routledge.
- Burger, O., Todd, L.C. Burnett, P., Stohlgren, T.J., Stephens, D. 2002, Multi-Scale and Nested-Intensity Sampling Techniques for Archaeological Survey, *Journal of Field Archaeology*, (29) 3/4, 409-423.
- Burmeister, St. 2017, Innovation as a Possibility. Technological and Social Determinism in Their Dialectical Resolution, in Burmeister, St. and Bernbeck, R. (eds.), *The Interplay of People and Technologies. Archaeological Case Studies on Innovation*, Berlin: Edition Topoi, 21-42, (23).
- Buschmann, F., Maunier, R., Rohnert, H., Sommerlad, P., Stal M. 1996, *A System of Patterns*, (vol 1), of *Pattern-Oriented Software Architecture*, Chichester: John Wiley.
- Butzer, K. W. 1971, *Environment and archaeology*, Chicago: Aldine.

- Butzer K.W. 1980, Context in archaeology: an alternative perspective. *Journal of Field Archaeology* 7, 417-422.
- Byers, S.N., Akoshima, K., Curran, B. 1989, Determination of adult stature from metatarsal length, *American Journal of Physical Anthropology*, 79, 275–279.
- Cameron, D.W. 1993, Uniformitarianism and Prehistoric Archaeology, *Australian Archaeology*, 36, 42-49.
- Cannon-Brookes, P. 1994, Antiquities in the market-place: Placing a price on documentation, *Antiquity* 68(295), 349-350.
- Carabott, Ph., Hamilakis, Y., Papargyriou, E. 2015, Capturing the eternal light: photography and Greece, photography of Greece, in Carabott, Ph., Hamilakis Y. and E. Papargyriou (eds.) *Camera Graeca. Photographs, narratives, materialities*, (3–21) Farnham: Ashgate.
- Carter, H., Mace A.C. 1923, *The tomb of Tutankh Amen*, Vol. 1. London: Cassell and Co.
- Carver, G. 2012, How to Archaeologize with a Hammer, in Cobb, H., Harris, O.J.T., Jones, C. and P. Richardson, (eds.), *Reconsidering Archaeological Fieldwork: Exploring On-Site Relationships Between Theory and Practice*, (15-29) New York: Springer.
- Carver, M.O.H. 2006, Thinking Allowed, *Rescue News* 100, 6-8.
- Carver, M.O.H. 2015, Field Method in Archaeology: Overview, in Carver, M., Gaydarska, B. Montón-Subías, S. (eds.) *Field Archaeology from Around the World*, (3-9), NY: Springer.
- Catsambis, A., Ford, B., Hamilton, D. (eds.) 2011, *The Oxford Handbook of Maritime Archaeology*. New York and Oxford: Oxford University Press
- Chadwick, A. 2003, Post-processualism, professionalization, and archaeological methodologies: Towards reflective and radical practice, *Archaeological Dialogues*, (10)1, 97-117.
- Chan Y.K., Koo V.C. 2008, An Introduction To Synthetic Aperture Radar (SAR), *Progress In Electromagnetics Research B*, (2), 27–60.
- Chan, Y.K., Lim, S.Y. 2008, Synthetic Aperture Radar (SAR) Signal Generation *Progress In Electromagnetics Research B*, (1), 269–290.

- Chazan, M. 2017, *World Prehistory and Archaeology: Pathways Through Time*, London: Routledge.
- Chenhall, R. 1975, A rationale for archaeological sampling, in Mueller, W. (ed.) *Sampling in Archaeology*, (3-28), Tucson: University of Arizona Press.
- Chiba, M., Terazawa, K. 1998, Estimation of stature from somatometry of skull, *Forensic Science International*, 97, 87-92.
- Childe, V.G. 1956, *Piecing Together the Past: The Interpretation of Archaeological Data*. London: Routledge.
- Ciancarini, P., Fantechi, A. Gorrieri, R. 2013, Formal Methods for Open Object-Based Distributed Systems: IFIP TC6 / WG6.1, in *Third International Conference on Formal Methods for Open Object-Based Distributed Systems (FMOODS), February 15-18, 1999, Florence, Italy*, Berlin: Springer.
- Clark, A. 1996, *Seeing Beneath the Soil*, New York: Routledge.
- Clarke, D.L., 1972, *Analytical archaeology*, London: Methuen and Co., Ltd.
- Clarke, D.L. 1973, Archaeology: The Loss of Innocence. *Antiquity* 47, 6-18.
- Cobb, H., Harris, O.J.T., Jones, C., Richardson, P. 2012, *Reconsidering Archaeological Fieldwork: Exploring On-Site Relationships Between Theory and Practice*, New York: Springer.
- Cochrane, E. E. 2001, Style, Function, and Systematic Empiricism: The Conflation of Process and Pattern, in Hurt, T.D., Rakita, G.F.M. (eds), *Style and Function: Conceptual Issues in Evolutionary Archaeology*, (183-203), Westport, CT: Bergin and Garvey.
- Coe, M.D. 1999, *The May*, New York: Thames and Hudson.
- Cohen, L., Maldonado, A. 2007, Research Methods in Education, *British Journal of Educational Studies* 55, 9.
- Cole, M.A., Linford, N.T., Payne, A.P., Linford, P.K. 1995, Soil magnetic susceptibility measurements and their application to archaeological site investigation, in Beavis, J., Barker, K. (eds), *Science and Site: Evaluation and Conservation*, (Occasional Paper 1, 114-162), Bournemouth University: School of Conservation Sciences.
- Colleen, M., Stuart, E. 2012, DIY and digital archaeology: what are you doing to participate? *World Archaeology*, 44(4), 521-537

- Coles, J. 1972, *Field Archaeology in Britain*, London: Methuen.
- Coles, J. 2014, *Field Archaeology in Britain*, London: Routledge.
- Collins, M.B. 1975, The sources of bias in processual data: an appraisal, in Mueller, J.W. (ed.) *Sampling in archaeology*, (26-32) Tucson: University of Arizona Press.
- Converse, T., Park, J., Morgan, C. 2004, *PHP5 and MySQL Bible*, New Jersey: Wiley.
- Cobb, H., Croucher, K. 2014, Assembling archaeological pedagogy: a theoretical framework for valuing pedagogy in archaeological interpretation and practice, *Archaeological Dialogues* 21(2), 197-216.
- Coombs, C. H. 1964, *A Theory of Data*. New York: John Wiley.
- Coope, G.R., Brophy, J.A., 1972, Late glacial environmental changes indicated by a coleopteran succession from North Wales, *Boreas*, 1, 97-142
- Corcos, N. 2001, Churches as prehistoric ritual monuments: a phenomenological perspective from Somerset, *Assemblage* 6, 22-54.
- Cowgill, G.L. 1967, Computer Applications in Archaeology, *Computers and the Humanities* 2(1), 17-23.
- Craig, I. 2001, *The Interpretation of Object-Oriented Programming Languages*, Berlin: Springer
- Crawford, O.G.S. 1929, *Air-photography for Archaeologists*, Ordnance Survey, London: H.M. Stationery Office.
- Cronyn, J.M. 1990, *The Elements of Archaeological Conservation*, London and New York: Routledge.
- Dallas, C. 2015, Curating archaeological knowledge in the digital continuum: from practice to infrastructure, *Open Archaeology*, 1(1), 176-207.
- Daly, P., Evans, Th.L., Daly, P.T. 2006, *Digital Archaeology: Bridging Method and Theory*, Oxon: Psychology Press.
- Dalrymple, G.B., Lanphere, M.A. 1969, *Potassium-Argon Dating: Principles, Techniques and Applications to Geochronology*, San Francisco: W.H. Freeman.
- Daniel, G. 1975, *A Hundred and Fifty Years of Archaeology*, London: Duckworth.

- Daniels D.J. (ed.) 2004, *Ground Penetrating Radar*, Knoval: Institution of Engineering and Technology.
- Darnay, A., Franklin, W.E. 1972, *Salvage markets for materials in solid wastes*, Washington, D.C: U.S. Environmental Protection Agency.
- Darvill, T. 2002, *Oxford concise dictionary of archaeology*, Oxford: Oxford University Press.
- David, A. 2001, Overview: the role and practice of archaeological prospection, in Brothwell, D.R. and Pollard, A.M. *Handbook of archaeological sciences*, (521-527) Chichester: John Wiley and Sons.
- David, B., Thomas, J. (eds.) 2016, *Handbook of landscape archaeology*, N.Y.: Routledge.
- Davidson, D.A. 1985, Geomorphology and Archaeology, in Rapp, G. Jr., Gifford, J.A. (eds.) *Archaeological Geology*, (25-55), New Haven: Yale Univ. Press.
- Deitel, H., Deitel, P.J. 2005, *C# for Programmers*, New Jersey: Prentice Hall.
- De Laet, S.J. 1957, *Archaeology and Its Problems*, New York, NY: Macmillan.
- Dean, J.S. 1978, Independent Dating in Archaeological Analysis, στο Schiffer M. B. (ed.) *Advances in Archaeological Method and Theory*, Vol. 1, (223-255) London: Academic Press.
- Dibble, H.L., McPherron, S. P. 1988, On the Computerization of Archaeological Projects, *Journal of Field Archaeology*, 15(4), 431-440.
- Dolby, N. 2008, The Use of Wood Charcoal in Landscape Archaeology, in David, B., Thomas, J., (eds.) *Handbook of Landscape Archaeology*, (451-456), Walnut Creek: Left Coast Press.
- Doneus, M., Neubauer, W. 2006, Laser Scanners for 3D Documentation of Stratigraphic Excavations, in Baltsavias, M., Gruen, A., Van Gool, L., Pateraki, M. (eds.), *Recording, Modeling and Visualization of Cultural Heritage*, (193-203) Leiden: Taylor & Francis
- Dowman, E.A. 1970, *Conservation in field archaeology*, London: Methuen.
- Drewett, P.L. 1999, *Field Archaeology: An Introduction*, London: UCL Press.
- Dunaway R.B. 2002, *The Book of Visual Studio .NET: A Guide for Developers*, San Francisco: No Starch Press.

- Dunnell, R.C. 1970, Seriation Method and Its Evaluation, *American Antiquity* 35, 305–319.
- Dunnell, R.C. 1988, Low-density archaeological records from plowed surfaces: Some preliminary considerations, *American Archaeology*, 7, 29-38.
- Dunnell, R.C. Dancey, W.S. 1983, The siteless survey: A regional scale data collection strategy, in Schiffer M.B. (ed.), *Advances in Archaeological Method and Theory*, vol. 6, (267-287), New York: Academic.
- Earle, T.K., Preucel, R.W. 1987, Processual Archaeology and the Radical Critique, *Current Anthropology* 28(4), 501–538.
- Edwards, K.J. 1979, Palynological and Temporal Inference in the Context of Prehistory with Special Reference to the Evidence from Lake and Peat Deposits, *Journal of Archaeological Science*, 6, 255–270.
- Esarey, D., Pauketat, T.R. 1992, *The Lohmann Site: An Early Mississippian Center in the American Bottom (II-S-49)*. Illinois: Department of Transportation and University of Illinois Press, Urbana.
- Esposito, D. 2009, *Microsoft ASP.NET and AJAX: Architecting Web Applications*, London: Microsoft Press by Pearson.
- Evans T. L., Daly, P. 2006, *Digital archaeology: bridging method and theory*, New York: Routledge.
- Fagan, B.M. 1996, *The Oxford Companion to Archaeology*, London and New York: Oxford University Press.
- Fagan, B.M. 1997, *In the Beginning: An Introduction to Archaeology*, New York: Longman.
- Farina, G. 2014, Some reflections on the phenomenological method. *Dialogues in Philosophy, Mental and Neuro Sciences*, 7(2), 50-62.
- Feibleman, J. K. 1967, The Philosophy of Tools, *Social Forces*, 45(3), 329-337.
- Fisher, K.E., Erdelez S., McKechnie, L., American Society for Information Science and Technology 2005. *Theories of Information Behavior*, NJ: Information Today, Inc.
- Fitch, J.R. 1972, Selection of suitable material for dating and the assessment of geological error in potassium-argon age determination, in Bishop, W.W., Miller, J.A. (eds.), *Calibration of Hominoid Evolution*, (77–91), Edinburgh: Scottish Academic Press.

- Fletcher, J.M., Tapper, M.C., Walker, F.S. 1974, Dendrochronology: a reference curve for slow grown oaks, A.D. 1230 to 1546, *Archaeometry*, 16: 31-40.
- Fritts, H.C. 1976, *Tree Rings and Climate*, London: Academic Press.
- Flatman, J. 2015, *Archaeology: A Beginner's Guide*, London: Oneworld Publications.
- Fleming, A. 2005, Megaliths and post-modernism: The case of Wales, *Antiquity*, 79, 921-932.
- Fletcher, J.M., Tapper, M.C., Walker, F.S. 1974, Dendrochronology: a reference curve for slow grown oaks, A.D. 1230 to 1546, *Archaeometry*, 16, 31-40.
- Flint, R.F. 1976, Physical evidence of quaternary climatic change, *Quaternary Research* 6, 519-528.
- Forrest, C. 2008, The Nature of Scientific Experimentation in Archaeology: Experimental Archaeology from the Nineteenth to the mid Twentieth Century, in P. Cunningham, J. Heeb and R. Paardekooper (eds.) *Experiencing Archaeology by Experiment. Proceedings of the Experimental Archaeology Conference, Exeter 2007*, (61-69), Oxford: Oxbow Books.
- Foucault, M. 1973, *The Order of Things*. New York: Vintage Books
- Foucault, M. 2002, *Archaeology of knowledge*, London: Routledge.
- Fowler, M.J.F. 2010, Satellite imagery and archaeology, in Cowley, D.C., Standring, R.A., Abicht, M.J. (eds) *Landscapes through the Lens: Aerial Photographs and Historic Environments*, (99-110), Oxford: Oxbow.
- Fowler, M., Rice, D., Foemmel, M., Hieatt, E., Mee, R., Stafford, R. 2002, *Patterns of Enterprise Application Architecture*, London: Pearson Education,
- Frankel, D. 1993, The Excavator: Creator or Destroyer? *Antiquity* 67, 875-877.
- Fully, G. 1956, New method of determination of the height, *Annales de Medecine Legale, Criminologie, Police Scientifique et Toxicologie*, 36, 266-273.

- Fussell, A. 1982, Terrestrial Photogrammetry in Archaeology, *World Archaeology* 14.2, *Photogrammetry/ Miscellany* (Oct): 157-172.
- Gardin, J.C. 1980, *Archaeological Constructs: An Aspect of Theoretical Archaeology*, Cambridge: Cambridge University Press.
- Garrison, E. 2016, *Techniques in Archaeological Geology*, Berlin: Springer.
- Gell, A. 1998, *Art and agency*, Oxford: Clarendon Press.
- Gelling, M. 2011, Place-Names and Archaeology, in Hinton, D.A., Crawford, S., Hamerow, H. (eds.), *The Oxford Handbook of Anglo-Saxon Archaeology*, (988-1002), Oxford: Oxford University Press.
- Gibbon, G. 2013, *Critically Reading the Theory and Methods of Archaeology: An Introductory Guide*, New York: Rowman and Littlefield.
- Gibbons M.T. 2006. Hermeneutics, Political Inquiry, and Practical Reason: An Evolving Challenge to Political Science, *The American Political Science Review*, 100(4), Thematic Issue on the Evolution of Political Science, in Recognition of the Centennial of the Review, 563-571.
- Giddings, J.L. 1966, Cross-dating the archaeology of northwestern Alaska, *Science*, 153, 127-135.
- Gifford, D.P. 1980, Ethnoarcheological contributions to the taphonomy of human sites, in Behrenmeyer, A.K. and Hill. A.P. (eds.) *Fossils in the making: vertebrate taphonomy and palaeoecology* (94-106). Chicago and London: the University of Chicago Press.
- Goffer, Z. 2007, *Archaeological chemistry*, New Jersey: John Wiley & Sons.
- Gosden, C., 1994, *Social being and time*, Oxford: Blackwell.
- Grant, J., Gorin, S., Fleming, N. 2005, *The Archaeology Coursebook: an introduction to study skills, topics and methods*, London and NY: Routledge.
- Gräslund, B. 1985–1987. Oscar A Montelius, in *Svenskt biografiskt lexicon*, 25, Stockholm: National Archives of Sweden.
- Greathouse, G.A., Flerer, B., Wessel, C.J. 1954, Chemical and physical agents of deterioration, in Greathouse, G.A., Wessel, C.J. (eds.) *Deterioration of materials: causes and preventive techniques*, (71-174) New York: Reinhold.

- Green, H.J.M. 1961, An analysis of archaeological rubbish deposits, *Archaeological News Letter* 7, 51-54.
- Green, D., Cosmas, J., Itagaki, T., Waelkens, M., Degeest, R., Grabczewski, E. 2001, A real time 3D stratigraphic visual simulation system for archaeological analysis and hypothesis testing, *Proceedings of the 2001 Conference on Virtual Reality, Archeology, and Cultural Heritage*, 271-376.
- Griffiths, I., Adams, M. Liberty, J. 2010, *Programming C# 4.0: Building Windows, Web, and RIA Applications for the .NET 4.0 Framework*, Sebastopol, California: O'Reilly.
- Grimshaw, R.W. 1971, *The chemistry and physics of clays and allied ceramic materials*, New York: Wiley.
- Grosso, A. Gozzi, A. Coccoli, M. Boccalatte, A. 2003, An Agent Programming Framework Based on the C# Language and the CLI, in *1st Int. Workshop on C# and .NET Technologies on Algorithms, Computer Graphics, Visualization, Computer Vision and Distributed Computing February 6-8, 2003*, Plzen, Czech Republic: UNION Agency – Science Press.
- Guzinski, R., Spondylis, E., Michalis, M., Tusa, S., Brancato, G., Minno, L., Hansen, L.B. 2016, Exploring the Utility of Bathymetry Maps Derived With Multispectral Satellite Observations in the Field of Underwater Archaeology, *Open Archaeology* 2, 243-263.
- Hamilakis, Y. 1999, La trahison de archéologues: Archaeological practice as intellectual activity in postmodernity, *Journal of Mediterranean Archaeology*, (12)1, 60-79.
- Hamilakis, Y. 2007. *The Nation and its Ruins: Antiquity, Archaeology, and National Imagination in Greece*. Oxford: Oxford University Press.
- Hamilakis, Y., Jones, A.M. 2017. Archaeology and Assemblage, *Cambridge Archaeological Journal* 27(1), 77–84.
- Hamilton, S., Whitehouse, R. 2006, Phenomenology in practice: towards a methodology for a 'subjective' approach, *European Journal of Archaeology* 9, 31-71.
- Hardesty, D. L. 2008, Goals of Archaeology, Overview, in Pearsall D.M. (ed.) *Encyclopedia of Archaeology* (1414–1416), Amsterdam: Elsevier.

- Hardesty, D. L. 2010, *Archaeology*, Vol. 1, Paris: UNESCO EOLSS Publications.
- Harmon, R.S., Thompson, P., Schwarcz, H.P., Ford, D.C., 1975, Uranium-series dating of speleothems, *National Speleological Society Bulletin*, 37, 21-33.
- Harris D.R. 1987, The impact on archaeology of radiocarbon dating by accelerator mass spectrometry, *Phil. Trans. R. Soc. Lond. A* 323, 23-43.
- Harris, E.C. 1979, The laws of archaeological stratigraphy, *World Archaeology*, 11, 111-117.
- Harris, E.C. 1989, *Principles of Archaeological Stratigraphy*, Cambridge: Elsevier.
- Harris, E.C., Brown, M.L., Brown, G.J. (eds.) 1993. *Practices of archaeological stratigraphy*, London: Academic Press
- Harrison, R., 2011, Surface assemblages: Towards an archaeology in and of the present, *Archaeological Dialogues* 18(2), 141-161.
- Harrison, B. 2009, *Seeking a Role: The United Kingdom 1951-1970*, Oxford: Oxford University Press.
- Hartel, J., 2017, Information behaviour, visual research, and the information horizon interview: three ways, *Information Research*, 22(1), CoLIS paper 1635.
- Harvey, D.R. 2005, *Preserving Digital Materials*, Munchen: K.G. Saur Verlag.
- Hasle, G.R., Syvertsen, E.E., Steidinger, K.A., Tangen, K. 1996, Marine Diatoms, in Tomas C.R. (ed.), *Identifying Marine Diatoms and Dinoflagellates*, (5-385), Academic Press.
- Hassan F.A. 1979, Geoarchaeology: The Geologist and Archaeology, *American Antiquity*, (44)2, 267-270.
- Hawthorn, J. 2006, Theories of the Gaze in Waugh, P. (ed.) *Literary Theory and Criticism*, (508-518) Oxford: Oxford University Press.
- Henke, W. 2007, *Handbook of paleoanthropology*, New York: Springer.
- Hester, T.R., Shafer, H.J., Feder, K.L. 2016, *Field methods in archaeology*, London and N.Y: Routledge.
- Higginbotham, E. 1985, Techniques in Historical Archaeology, *Australian Journal of Historical Archaeology*, 3, 8-14.

- Hjort, J., Karjalainen, O., Aalto, J., Westermann, S., Romanovsky, V.E., Nelson, F.E., Etzelmüller, B., Luoto, M. 2018, Degrading permafrost puts Arctic infrastructure at risk by mid-century, *Nature Communications* 9(1), 1-11.
- Hodder, I. 1982, Theoretical archaeology: A reactionary view, in Hodder, I. (ed.), *Symbolic and Structural Archaeology*, Cambridge: Cambridge University Press.
- Hodder, I. 1982b, Sequences of structural change in the Dutch Neolithic, in Hodder, I. (ed.) *Symbolic and Structural archaeology*, (162-177) Cambridge: Cambridge University Press.
- Hodder, I. 1984, Archaeology in 1984, *Antiquity* 58, 25–32.
- Hodder, I. 1989, Writing archaeology: Site reports in context, *Antiquity*, 63, 268-274.
- Hodder, I. 1987, The Contextual Analysis of Symbolic Meanings, in Hodder, I. (ed.) *The Archaeology of Contextual Meanings*, (1-10) Cambridge: Cambridge University Press.
- Hodder, I. 1991, Interpretive Archaeology and Its Role, *American Antiquity*, 56(1), 7–18.
- Hodder, I. 1995, *Theory and Practice in Archaeology*, London: Routledge.
- Hodder, I. 1999, *The Archaeological Process: An introduction*. Oxford: Blackwell.
- Hodder, I., Hutson, S. 2003, *Reading the past: Current approaches to interpretation in archaeology*, Cambridge: Cambridge University Press.
- Holland, T.D. 1995, Estimation of adult stature from the calcaneus and talus, *American Journal of Physical Anthropology*, 96, 315–320.
- Hollesen, J., Matthiesen, H., Elberling, B. 2017, Impact of climate change on Arctic archaeological site, *Archaeometry*, 59(6), 1175–1189.
- Hope Simpson, R., Lazenby, J. F. 1973, Notes from the Dodecanese III, *BSA* 68, 146-147.
- Horne, L. 1983, Recycling an Iranian village: ethnoarchaeology in Baghestan, *Archaeology*, 36(4), 16-21.

- Hortolà, P. 2017. From antiquities to memorabilia: a standardized terminology for ancestral artefacts according to manufacture date, *Studia Antiqua et Archaeologica*, 23 (2), 213–225.
- Humphreys, A.J.B. 1993, The significance of place names in archaeological research, *African Studies*, (52)1, 43-53.
- Hunt, J. 2002, *Guide to C# and Object Orientation*, Berlin: Springer, 12.
- Hunter, L., Leahey E. 2008, Collaborative Research in Sociology: Trends and Contributing Factors, *American Sociologist* 39, 290–306.
- Huvila, I. 2009, Analytical information horizon maps, *Library & Information Science Research*, 31, 18–28.
- Huvila, I. 2012, *Information Services and Digital Literacy: In search of the boundaries of knowing*, Oxford: Chantos Publishing.
- Hyland, D.C., Adovasio, J.M., Illingworth, J.S. 2003, The Perishable Artifacts, in MacNeish, R.S., Libby J.G. (eds.) *Pendejo Cave*, (297-416), Albuquerque: University of New Mexico Press.
- Information Resources Management Association 2001, *Managing Information Technology in a Global Economy*, Los Angeles.
- Introna, F., Di Vella, G., Petrachi, S. 1993, Determination of height in life using multiple regression of skull parameters, *Bollettino della Società Italiana di Biologia Sperimentale*, 69, 153–160.
- Irny, S.I., Rose, A.A. 2005, Designing a Strategic Information Systems Planning Methodology for Malaysian Institutes of Higher Learning, *Issues in Information System*, 6(1), 325-331.
- Jashemski, W. 1979, Pompeii and Mount Vesuvius, 79, in Sheets P.D., Grayson D.K. (eds.) *Volcanic activity and human ecology*, (587-622) New York: Academic Press.
- Jit, I., Singh, S. 1956, Estimation of stature from clavicles, *Indian Journal of Medical Research*, 44, 137–155.
- Johnsen, H., Olsen, B. 1992, Hermeneutics and Archaeology: On the Philosophy of Contextual Archaeology, *American Antiquity*, (57)3, 419-436.

- Johnson, M. 2000, Self-made men and the staging of agency, in Dobres, M.A. and J. Robb (eds.) *Agency in archaeology*, (213–231) London: Routledge
- Johnson, M. 2002, *Behind the castle gate. From medieval to Renaissance*, London: Routledge.
- Johnson, M. 2006, On the nature of theoretical archaeology and archaeological theory, *Archaeological Dialogues*, 13(2), 117–132.
- Johnson, M. 2007, *Ideas of landscape*, Oxford: Blackwell.
- Johnson, M. 2010, *Archaeological Theory: An Introduction* (second edition). Oxford: Blackwell.
- Jones, A. 2003, *Archaeological theory and scientific practice*, Cambridge: Cambridge University Press.
- Jones, L.H.P., Handreck, K.A. 1967, Silica in Soils, Plants, and Animals, in Nolman, A.G. (ed.), *Advances in Agronomy*, Vol. 19 (107-149) New York: Academic Press
- Joyce, R.A. 2002, *The Languages of Archaeology*. Oxford: Blackwell.
- Joyce, R.A. 2006. The monumental and the trace: archaeological conservation and the materiality of the past, in N. Agnew and J. Bridgland (eds.) *Of the Past, for the Future: Integrating Archaeology and Conservation: Proceedings of the Conservation Theme at the 5th World Archaeological Congress: 13- 18*. Los Angeles: Getty Conservation Institute.
- Καλογερόπουλος, 2006, Digit: Ζητήματα ψηφιοποίησης ανασκαφικών δεδομένων, *Ι΄ Διεθνές Κρητολογικό Συνέδριο 1-8 Οκτωβρίου*.
- Καλογερόπουλος, Κ. 2014, *Κυμισάλα, η θεωρία πίσω από το ανασκαφικό έργο», Κυμισάλα: Αρχαιολογία-Εκπαίδευση-Αειφορία, Πρακτικά ημερίδας*, Ρόδος, 4-5 Απριλίου.
- Καλογερόπουλος, Κ. 2017, Η Αρχαιολογική Θεωρία στην Ανασκαφή της Κυμισάλας, in Stefanakis, M. (ed.) *Archaeology - Education - Sustainability*, (105-111), Oxford: Archaeopress.
- Kang, B. W. 2005, An Examination of an Intermediate Sociopolitical Evolutionary Type between Chieftdom and State, *Arctic Anthropology*, 42(2), 22-35.

- Κατσιάνης, Μ. 2009, *Ανασκαφική Μεθοδολογία και Σχεδιασμός Πληροφοριακού Συστήματος για τη Διαχείριση Αρχαιολογικών Τεκμηρίων*, [διδακτορική διατριβή], Θεσσαλονίκη: Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης.
- Kelly, R.L., Thomas, D.H. 2014, *Archaeology: down to earth*, Belmont, CA: Wadsworth, Cengage Learning.
- Kibblewhite, M., Tóth, G., Hermann, T. 2015, Predicting the preservation of cultural artefacts and buried materials in soil, *Science of the Total Environment*, 529, 249-263.
- Kipfer, B.A. 2000, *Encyclopedic Dictionary of Archaeology*, New York: Springer.
- Kirtsideli, I.Y. Vlasov, D.Y., Abakumov, E.V., Barantsevich, E.P., Novozhilov, Y.K., Krylenkov, V.A., Sokolov, V.T. 2017, Airborne fungi in arctic settlement Tiksi (Russian Arctic, coast of the Laptev Sea), *Czech Polar Reports* (7)2, 300-310.
- Knapp, A.B. 1998, Social approaches to the archaeology and anthropology of mining in Knapp, A.B., Pigott, V.C. and E.W. Herbert (eds.) *Social Approaches to an Industrial Past*, London: Routledge.
- Knorzer, Ê. H. 1984, The prospects of the palaeoethnobotanical examination of cesspits, in van Zeist, W., Casparie W.A. (eds.) *Plants and ancient man*, (331-338) Boston: A. A. Balkema.
- Kofler, M. 2006, *The Definitive Guide to MySQL 5*, N.Y.: Apress.
- Kogent Solutions 2008, *C# 2008 Programming: Covers .Net 3.5 Black Book*, New Delhi: Dreamtech Press.
- Kohl, P. 2004, Comments on Adam T. Smith's 'The end of the essential archaeological subject', *Archaeological dialogues* 11, 20-21.
- Koopman, B.O. 1980, *Search and Screening: General Principles with Historical Applications*, New York: Pergamon Press.
- Kopytoff, I. 1986, The cultural biography of things: commoditization as process, in Appadurai, A. (ed.) *The social life of things: Commodities in cultural perspective* (64-91) Cambridge: Cambridge University Press.
- Κουκουζέλη, Α. κ.ά. 2003, *Αρχαιολογία στον Ελληνικό Χώρο*, Τόμ. Α', Πάτρα: ΕΑΠ.

- Kottak, C.P. 2011, *Window on Humanity: A Concise Introduction to Anthropology*, New York: McGraw-Hill.
- Kraker, J.J., Shott, M.J., Welch P.D. 1983, Design and Evaluation of Shovel-test Sampling in Regional Archaeological Survey. *Journal of Field Archaeology* 10(4), 469-480.
- Kreuz, A. 1992, Charcoal from Ten Early Neolithic Settlements in Central Europe and its Interpretation in Terms of Woodland Management and Wildwood Resources, *Bulletin de la Societe Botanique de France* 139, 383-394
- Krieger, W. H. 2006, *Can there be a Philosophy of Archaeology? Processual Archaeology and the Philosophy of Science*, Lanham: Lexington Books.
- Kristiansen, K. 2004, Genes versus agents. A discussion of the widening theoretical gap in archaeology, *Archaeological dialogues* 11, 77-98.
- Kuhn, T. S. 1970, *The structure of scientific revolutions*, Chicago, IL: University of Chicago.
- Κωτσάκης, Κ. 1998, Αντικείμενα και αφηγήσεις. Η ερμηνεία του υλικού πολιτισμού στη σύγχρονη Αρχαιολογία, *Επτάκυκλος*, (2)10, 11-23.
- Lal, D., Jull, A.J.T. 2001, In-situ cosmogenic ¹⁴C: production and examples of its unique applications in studies of terrestrial and extraterrestrial processes, *Radiocarbon*, (43)2, 731-742.
- Lambers, K., Remondino, F. 2008, Optical 3D Measurement Techniques, in Posluschny, A. (ed.) *Archaeology: Recent Developments and Applications, in Layers of Perception: Proceedings of 35th International Conference on Computer Applications and Quantitative Methods in Archaeology, Berlin, Germany April 2-6, 2007*, (27-35), Bonn: Habelt.
- Lampeter Archaeology Workshop, 1997, Relativism, objectivity and the politics of the past, *Archaeological dialogues* 2, 164-198.
- Leakey, R.E., Roger L. 1978, *People of the Lake: mankind and its beginnings*. Doubleday, Garden City: Anchor Press.
- Leakey, M.D. 1979, 3.6 million years old: footprints in the ashes of time. *National Geographic* 155, 446-457.

- Leone, M.P. 1986, Symbolic, Structural and Critical Archaeology, in Meltzer, D.J., Fowler, D.D., Sabloff, J.A. (eds.) *American Archaeology Past and Future*, 415-438. Washington, D.C.: Smithsonian Institution Press.
- Leone, M.P., Potter, P.B. Jr., Shackel, P.A. 1987, Toward a Critical Archaeology *Current Anthropology*, 28(3), 283-302.
- Lewarch, D.E., O'Brien, M.J. 1981, The expanding role of surface assemblages in archaeological research. *Advances in Archaeological Method and Theory*, 4, 297-342.
- Lhotka, R. 2006, *Expert C# 2005 Business Objects*, New York: Apress.
- Libby, A. 2015, *Mastering jQuery*, Birmingham: Pact Publishing.
- Liberty, J., MacDonald, B. 2006, *Learning C# 2005*, Sebastopol, California: O'Reilly.
- Lipo, C.P., Madsen, M.E., Dunnell, R.C., Hunt, T. 1997, Population Structure, Cultural Transmission, and Frequency Seriation, *Journal of Anthropological Archaeology*, 16, 301-334.
- Linford, N. 2005, Archaeological applications of naturally occurring nanomagnets, International Conference on Fine Particle Magnetism (London, 20-22 September 2004) *Journal of Physics: Conference Series*, 17, 127-144.
- Linington, R.E. 1966, Test use of a gravimeter on Etruscan chamber tombs at Cerveteri, *Prospezioni Archaeologiche*, 1, 37-41.
- Lister, F.C., Lister, R.H. 1981, The recycled pots and potsherds of Spain. *Historical Archaeology* 15, 66-78.
- Lock, G. 2003, *Using computers in archaeology: towards virtual pasts*, London and NY: Routledge.
- Lock, G., Molynaux, B.L. (eds.) 2006, *Confronting Scales in Archaeology*, New York: Springer.
- Loffler, E., 1976, Potassium-argon dates and pre-Wurm glaciations of Mount Giluwe volcano, Papua, New Guinea, *Z Gletscherk Glazialgeol*, 12, 55-62.
- Llobera, M. 2011, Archaeological Visualization: Towards an Archaeological Information Science (AISC), *Journal of Archaeological Method and Theory*, 18, 193-223.

- Lucas, G. 2001a, *Critical approaches to fieldwork. Contemporary and historical archaeological practice*, London: Routledge.
- Lucas, G. 2001b, Destruction and the Rhetoric of Excavation, *Norwegian Archaeology Review* 34, 35–46.
- Lucas, G. 2012, *Understanding the Archaeological Record*, Cambridge: Cambridge University Press.
- Lucy, S. 1997, Housewives: warriors and slaves? Sex and gender in Anglo-Saxon burials in Moore, J., Scoot, E., (eds.) *Invisible People and Processes: Writing Gender and Childhood into European Archaeology* (150–168) London and New York: Leicester University Press.
- Lundy, J.K. 1983, Regression equations for estimating living stature from long limb bones in the South African Negro, *South African Journal of Science*, 79, 337–338.
- Lundy, J.K. 1985, The mathematical versus anatomical methods of stature estimate from long bones, *American Journal of Forensic Medicine and Pathology*, 6, 73–76.
- Lundy, J.K. 1998, Forensic Anthropology: What Bones Can Tell Us, *Laboratory Medicine*, (29)7, 423-427.
- Lundy, J.K., Feldesman, M.R. 1987, Revised equations for estimating living stature from long bones of the South African Negro, *South African Journal of Science*, 83, 54–55.
- Lyman, R.L. 1994, *Vertebrate taphonomy*, Cambridge: Cambridge University Press.
- Lyman, R.L. 2012. A Historical Sketch on the Concepts of Archaeological Association, Context, and Provenience, *Journal of Archaeological Method and Theory*, 19(2), 207–240.
- Lyman, R.L., O'Brien, M.J., and Dunnell, R.C. 1997, *The Rise and Fall of Culture History*, New York: Plenum Press.
- Martinez, P. 2001, Digital Realities and Archaeology: A Difficult Relationship or a Fruitful Marriage? in *VAST '01: Proceedings of the 2001 Conference on Virtual Reality, Archeology and Cultural Heritage*, (9–16), New York.

- McDougall, I., Brown, F.H. 2006, Precise $^{40}\text{Ar}/^{39}\text{Ar}$ geochronology for the upper Koobi Fora Formation, Turkana Basin, northern Kenya, *J. Geol. Soc.* 163, 205-220.
- Macklin, M.G. 1995, Archaeology and the river environment in Britain: A prospective review, in, Barham, A.J., Macphail, R.I. (eds.) *Archaeological Sediments and Soils: Analysis, Interpretation and Management, 10th Anniv. Conf. Assn. Environmental Archaeology, 1989*, (205-220), London: Institute of Archaeology, University College.
- Macleod N. 2005, *Principles of Stratigraphy*, London: Natural History Museum.
- Maher, B.A. 1986, Characterization of soils by mineral magnetic measurements, *Physics of the Earth and Planetary Interiors*, 42, 76-92.
- Majewski, T., O'Brien, M.J. 1987, The use and misuse of ceramics in archaeological analysis in Schiffer, M.B. *Advances in Archaeological Theory and Method*, Vol. 11, (97-209), San Diego and London: Academic Press.
- Martin J., 2016, *Visual Studio 2015 Cookbook* Birmingham: Packt Publishing.
- Matthiesen, H., Jensen, J.B., Gregory, D., Hollesen, J., Elberling, B. 2014, Degradation of archaeological wood under freezing and thawing conditions, *Archaeometry*, 56, 479-495.
- Ματζάνας, Χ. 2001, Πειραματική λάξευση πυριτολίθου: μέθοδοι και τεχνικές, *Corpus* 30, 56-69.
- McDaniel A. 2011, *HTML5: Your visual blueprint for designing rich Web pages and applications*, San Francisco: Wiley.
- McGuire, R. H. 2008, *Archaeology as Political Action*, Berkeley: University of California Press.
- McPherron S. J. P. 2005, Artifact orientations and site formation processes from total station proveniences, *Journal of Archaeological Science* 32, 1003-1014.
- McPherron, S. J. P., Dibble, H.L. 2002, Using computers in adverse field conditions: Tales from the Egyptian desert, *The SAA Archaeological Record*, 3, 28-32.
- Meadows, L. Jantz, R.L. 1992, Estimation of stature from metacarpal lengths, *Journal of Forensic Science*, 37, 147-154.

- Mee, C. 1982, *Rhodes in the Bronze Age: An Archaeological Survey*. Warminster: Aris and Phillips.
- Meloni, J.C., Kyrnin, J. 2018, *HTML, CSS, and JavaScript All in One: Covering HTML5, CSS3, and ES6, Sams Teach Yourself*, Carmel, Indiana: Sams Publishing.
- Meskell, L. 2004, *Object worlds in ancient Egypt. Material biographies past and present*, Oxford: Berg.
- Mesoudi, A., Whiten A., Laland, K.N. 2006, Towards a unified science of cultural evolution, *The Behavioral and Brain Sciences*, 29(4), 329-347.
- Metsker, S.J. 2001, *Building parsers with Java*, Boston: Addison-Wesley.
- Miller, D., Tilley C. 1984. Ideology, Power, and Prehistory: An Introduction, in Miller, D. and C. Tilley, (eds.) *Ideology, Power, and Prehistory*, 1-15. Cambridge: Cambridge University Press.
- Miller, D. 1984. Modernism and suburbia as material ideology, in Miller D. and C. Tilley (eds.) *Ideology, Power and Prehistory*, Cambridge: Cambridge University Press.
- Miller, N.F., Smart, T.L. 1984, Intentional Burning of Dung as Fuel: A Mechanism for the Incorporation of Charred Seeds into the Archaeological Record, *Journal of Ethnobiology*, 4, 15-28.
- Monastersky, R.1990. Fingerprints in the sand: Federal agents use dirty evidence against archaeological thieves, *Science News*, 138(25), 392-394.
- Moore, H., 1996. *Space, text and gender: An anthropological study of the Marakwet of Kenya*, N.Y.: The Guilford Press.
- Morgan, A., 1973, Late Pleistocene environmental changes indicated by fossil insect faunas of the English Midland, *Boreas*, 2, 173-212.
- Morgan, C. 2013, Publishing archaeological research, *Mélanges de la Casa de Velázquez*, (43)2, 279-288.
- Moro-Abadám, O., Palacio-Pérez, E. 2015, Rethinking the Structural Analysis of Paleolithic Art: New Perspectives on Leroi-Gourhan's Structuralism *Cambridge Archaeological Journal* 25(3), 1-16.
- Mukherjee, S. 2012, *Applied Mineralogy: Applications in Industry and Environment*, Berlin: Springer Science & Business Media.

- Mueller, J.W. 1974, The Use of Sampling in Archaeological Survey, *Memoirs of the Society for American Archaeology*, 28, 1-91.
- Murray, T. 2011, Archaeologists and Indigenous People: A Maturing Relationship? *Annual Review of Anthropology* 40, 363-378.
- Musgrave, J.H., Harneja N.K. 1978, The estimation of adult stature from metacarpal bone length, *American Journal of Physical Anthropology*, 48, 113-119.
- Nami, H.G. 2010, Theoretical Reflections on Experimental Archaeology and Lithic Technology, in Nami, H.G. (ed.), *Experiments and Interpretation of Traditional Technologies: Essays in Honor of Errett Callahan*, (91-170), Buenos Aires.
- Nayyeri, K., White D. 2014, *Pro ASP.NET SignalR: Real-Time Communication in .NET with SignalR 2.1167*, New York: Apress
- Nixon, R. 2012, *Learning PHP, MySQL, JavaScript, and CSS: A Step-by-Step Guide to Creating Dynamic Websites*, Sebastopol: O'Reilly Media.
- Norman, M. 2003, *Database Design Manual: using MySQL for Windows*, Berlin and Heidelberg, Springer.
- North, N.A., MacLeod, I.D 1987, Corrosion of metals, in Colin Pearson, (ed.) *Conservation of Marine Archaeological Objects*, (68-98), Oxford: Butterworth-Heinemann.
- Novak, I. 2011, *Beginning Microsoft Visual Studio LightSwitch Development*, New York: Wiley.
- O'Brien, M.J., Lyman, R.L. 1999, *Seriation, Stratigraphy, and Index Fossils: The Backbone of Archaeological Dating*. New York: Kluwer.
- O'Connell, J.F. 1995, Ethnoarchaeology Needs a General Theory of Behavior, *Journal of Archaeological Research*, 3(3), 205-255.
- Olsen, B. 2010, *In Defense of Things: Archaeology and the Ontology of Objects*, Plymouth: Altamira Press.
- Olsen, B. 2006, Archaeology, hermeneutics of suspicion and phenomenological trivialization, *Archaeological Dialogues* (13)2, 144-150.
- Ortner, S.B. 1984, Theory in anthropology since the sixties, *Society for Comparative Study of Society and History*, (26)1, 126-166.

- Orton, C. 2002, *Sampling in Archaeology*, Cambridge: Cambridge University Press.
- Παπαχριστοδούλου, Ι.Χ. 1989, *Οι Αρχαίοι Ροδιακοί Δήμοι. Ιστορική Ανασκόπηση. Η Ιαλυσία*. Αθήνα: Βιβλιοθήκη της εν Αθήναις Αρχαιολογικής Εταιρείας, 110.
- Patrik, L. E. 1985, Is There an Archaeological Record? *Advances in Archaeological Method and Theory*, 8, 27-62.
- Pearsall, D.M. 1982, Applications of a New Paleoethnobotanical Technique in Archeology, *American Anthropologist*, (84)4 862-871.
- Pearson, M.P. 1984. Economic and ideological change: Cyclical growth in the pre-state societies of Jutland in Miller, D., Tilley C. (eds.) *Ideology, Power and Prehistory*, 69-92, Cambridge: Cambridge University Press.
- Pelegrin, J. 2015. Lithics and Archaeology, in Wright J. (ed.) *International Encyclopedia of the Social & Behavioral Sciences* (Second Edition), Amsterdam: Elsevier, 260-264.
- Peltz, L. and Myrone, M. 1999: 'Introduction: 'mine are the subjects rejected by the historian': antiquarianism, history and the making of modern culture', in M. Myrone and L. Peltz (eds.), *Producing the Past: aspects of antiquarian culture and practice 1700-1850*. Aldershot & Brookfield: Ashgate, 1-13.
- Pernier, L., 1914. *Ricognizioni archeologiche nelle Sporadi*, *BdA* 8, 236-242.
- Perrin, K. *etal*, 2014. *A Standard and Guide to Best Practice for Archaeological Archiving in Europe: EAC Guidelines 1*, Namur: Europae Archaeologia Consilium (EAC).
- Πετρόχειλος, Ν. 2002-2005. Το κλασικό νεκροταφείο στην Αγία Ελεούσα της Άνδρου, *AAA* 35-38, 107-119.
- Pipes, J., Kruckenberg, M. 2006, *Pro MySQL*, New York: Apress.
- Pitts, M. 2011. *Hengeworld*, N.Y.: Random House.
- Platthy, J. 1993, UNESCO and trafficking in illicit treasures, *Journal of Prehistoric Religion*, 7, 42-48.
- Plog, S., Plog, F., Wait, W. 1978, Decision Making in Modern Surveys, in Schiffer, M.B. (ed.) *Advances in Archaeological Method and Theory*, Vol. 1. (383-421), New York: Academic Press.

- Pluciennik, M. 1999, Archaeological Narratives and Other Ways of Telling, *Current Anthropology*, (40)5, 653-678.
- Poidebard, A. 1934, *La Trace de Rome dans le Desert de Syrie; Le Limes de Trajan de la Conquete Arabe. Recherches Aeriennes (1925-1932)*, Paris: Geuthner.
- Pollard, M.A. 1999, Geoarchaeology: An Introduction, *Geological Society London Special Publications*, 165, 7-14.
- Pollard, J., Reynolds, A. 2002, *Avebury: The biography of a landscape*, Stroud: Tempus.
- Porter, S.C., 1979, Hawaiian glacial ages, *Quat. Res.* 12, 161-187.
- Poulter, G.A., Kerslake, I. 1997, Vertical Photographic Site Recording: The 'Holmes Boom', *Journal of Field Archaeology*, 24.2: 221-232.
- Pourgouris M. 2011. The Making of Modern Greece: Nationalism, Romanticism & the Uses of the Past (1797-1896), *Journal of Modern Greek Studies*, 29(2), 291-295.
- Powers, L. Snell, M. 2016, *Microsoft® Visual Studio® 2015 Unleashed*, London: Pearson Education
- Preucel R.W. 1995, The Post-processual Condition, *Journal of Archaeological Research*, 3(2), 147-175.
- Price T.D., Burton J.H. 2011, *An Introduction to Archaeological Chemistry*, New York: Springer.
- Proudfoot, B. 1976, The analysis and interpretation of soil phosphorus in archaeological contexts, in Davidson, D.A., Shackley, M.L. (eds), *Geoarchaeology*, (93-113) London: Duckworth.
- Puntambekar, A.A. 2010, *Compiler Design*, Pune, India: Technical Publications.
- Punt, W., Reitsma, T., Reuvers, A. 1974, Caprifoliaceae, *Review of Paleobotany and Palynology* 17, 5-29.
- Queiroz-Alves, E., Macario, K., Ascough, P., Bronk, R.C. 2018, The worldwide marine radiocarbon reservoir effect: Definitions, mechanisms and prospects, *Reviews of Geophysics*. (56)1, 278-305

- Raab, L.M., Goodyear, A.C. 1984, Middle-Range Theory in Archaeology: A Critical Review of Origins and Applications, *American Antiquity*, 49(2), 255-268.
- Rahtz, S., Reilly, P. 2003, *Archaeology and the Information Age*, Oxon, UK: Routledge.
- Rama Krishna Rao, B. 2006, *Programming with C#: Concepts and Practice*, New Delhi: PHI Learning PV.
- Ramenofsky, A.F., Steffen, A. (eds.) 1998, *Unit Issues in Archaeology: Measuring Time, Space, and Material*, Salt Lake City: The University of Utah Press.
- Rathje, W.L., Schiffer, M.B. 1982. *Archaeology*, New York: Harcourt Brace Jovanovich.
- Reese, G., Yarger, R., King, T., Williams, H.E. 2002, *Managing & Using MySQL: Open Source SQL Databases for Managing Information*, Sebastopol: O'Reilly.
- Reid, J.J., Schiffer, M.B., Neff J.M. 1975. Archaeological considerations of intra-site sampling, in Mueller, J. (ed.) *Sampling in archaeology*, (209-224), Tucson: University of Arizona Press.
- Reimer, P.J. et al. 2013, IntCal13 and Marine13 radiocarbon age calibration curves 0–50,000 years cal BP, *Radiocarbon*, (55)4, 1869–1887.
- Renfrew, C. 1972. *The Emergence of Civilisation: The Cyclades and the Aegean in the Third Millennium B.C.*, London: Methuen.
- Renfrew, C. 1976, Introduction, in Davidson, D.A., Shackley, M.L. (eds.), *Geoarchaeology: Earth Science and the Past*, (1-12), London: Duckworth.
- Renfrew, C., Bahn, B. 2001. *Αρχαιολογία: Θεωρίες, μεθοδολογία και πρακτικές εφαρμογές*, Αθήνα: Ινστιτούτο του Βιβλίου-Α. Καρδαμίτσα.
- Renfrew, C. Bahn, P. 2016. *Archaeology: Theories, Methods and Practice*. London: Thames and Hudson.
- Renfrew, C., Zubrow, E. B. W., (eds.) 2004. *The Ancient Mind: Elements of Cognitive Archaeology*, New Directions in Archaeology, Cambridge: Cambridge University Press.
- Renfrew C. 2001. *The Emergence of Civilization*, Oxford: Oxbow.

- Rice, Patricia C. 1990. Introductory Archaeology: The Inexpensive Laboratory. *Anthropology & Education Quarterly* 21(2), 167–172.
- Richard, C. 1996. Henges and Water, *Journal of Material Culture*, 1, 313-336.
- Richardson, M., Gajewski, B. 2002, Archaeological Sampling Strategies, *Journal of Statistics Education*, (10)3, 1-9.
- Riede, F., Apel, J., Darmark, K. 2012, Cultural evolution and archaeology: Historical and current trends, in Berge, R., Jasinski, M. E. & Sognnes, K. (eds.) *N-TAG TEN. Proceedings of the 10th Nordic TAG conference at Stiklestad, Norway 2009*, Oxford: Archaeopress.
- Robb, J. 2004, Agency: A personal view, *Archaeological dialogues* 11, 103–6.
- Robin, C. 2006, Gender, farming, and long-term change: Maya historical and archaeological perspectives, *Current anthropology*, 47(3), 409-433.
- Roosevelt, C.H., Cobb, P., Moss, E., Olson, B.R. & S. Ünlüsoy 2015, Excavation is Destruction Digitization: Advances in Archaeological Practice, *Journal of Field Archaeology*, (40)3, 325-346.
- Rowe, J.H. 1970, Stratigraphy and Seriation, in Fagan B.M. (ed.) *Introductory Readings in Archaeology*, (58-69), Boston: Little Brown & Co.
- Russell, N. 2011, *Marine radiocarbon reservoir effects (MRE) in archaeology: temporal and spatial changes through the Holocene within the UK coastal environment (PhD thesis)*, Glasgow, Scotland UK: University of Glasgow.
- Ryan, A.M. 2017, *The Power of Java*, New York, Cavendish Square Publishing
- Rye, O. 1981, *Pottery technology: principles and reconstruction*. Washington, D.C.: Taraxacum.
- Sahney, S., Benton, M.J. 2008, Recovery from the most profound mass extinction of all time, *Proceedings of the Royal Society B: Biological Sciences*, 275, 759–765.
- Salmon, M.H. 1978, What Can Systems Theory Do for Archaeology? *American Antiquity*, 43(2), Contributions to Archaeological Method and Theory, 174-183.
- Salmon, M.H. 1982, *Philosophy and Archaeology*. New York: Academic Press.
- Santana Quintero, M., Fai, S., Smith, L., Duer, A., Barazzetti, L. 2019, Ethical Framework for Heritage Recording Specialists Applying Digital

- Workflows for Conservation, *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XLII-2/W15, 1063-1070.
- Savolainen, R., Kari, J. 2004. Placing the internet in information source horizons. A study of information seeking by internet users in the context of self-development, *Library and Information Science Research*, 26, 415–433.
- Sawyer McFarland, D. 2012. *CSS3: The Missing Manual*, Farnham: O'Reilly.
- Schreibman, S., Siemens, R., Unsworth, J. 2008, *A Companion to Digital Humanities*, New Jersey: Wiley-Blackwell.
- Schiffer, M.B. 1972, Archaeological Context and Systemic Context, *American Antiquity*, 37(2), 156-165.
- Schiffer M.B. 1976, *Behavioral archeology*, New York: Academic Press.
- Schiffer, M.B. 1978, Chipped stone and human behavior at the Joint site, in Grebinger, P.F. (ed.) *Discovering past behavior: experiments in the archaeology of the American Southwest*, (141-163) New York: Gordon and Breach.
- Schiffer, M.B., Sullivan, A.P., Klinger, T.C. 1978, The design of archaeological surveys, *World Archaeology* 10, 1-28.
- Schiffer, M.B. 1980, Review of *For Theory Building in Archaeology: Essays on Faunal Remains, Aquatic Resources, Spatial Analysis, and Systemic Modeling*, edited by Lewis R. Binford, *American Antiquity*, 45, 377.
- Schiffer, M.B. 1996. *Formation Processes of the Archaeological Record*, Salt Lake City: University of Utah Press.
- Schiffer, M.B., Sullivan, A.P. Klinger, T.C. 1978, The Design of Archaeological Surveys, *World Archaeology* 10(1), 1-28.
- Schiffer, M.B., Rathje, W.L. 1973, Efficient exploitation of the archeological record: penetrating problems, in Redman, C.L. *Research and theory in current archeology*, (169-179) New York: Wiley.
- Schwartz, B., Zaitsev, P., Tkachenko, V., Zawodny, J.D., Lentz, A., Balling, D.J. 2008, *High Performance MySQL: Optimization, Backups, Replication, and More*, Sebastopol, California: O'Reilly.

- Scott, H., Stephenson, J. 2016, Testing UAV (drone) aerial photography and photogrammetry for archaeology, *ResearchGate* RG: 2.1.4306.3443, 1-43.
- Seidl, N.P. 2008, Significance of Toponyms, with Emphasis on Field Names, for Studying Cultural Landscape, *Acta geographica Slovenica*, (48)1, 33–56
- Sera-Shriar, E. 2013, What is armchair anthropology? Observational practices in 19th-century British human sciences, *History of the Human Sciences*, 27(2), 26-40.
- Serola, S. 2006. City planners' information seeking behavior: Information channels used and information types needed in varying types of perceived work tasks, in *Proceedings of the 1st International Conference on Information Interaction in Context* (42–45). New York: ACM Press.
- Service, E.R. 1971. *Primitive social organization*. 2nd. ed. New York: Random House.
- Shanks, M., Tilley, C. 1992, *Re-constructing Archaeology: Theory and Practice*, London: Routledge.
- Sharer, R. J., Ashmore, W. 2003, *Archaeology: Discovering Our Past*, Boston: McGraw Hill.
- Shennan, S., 2002, *Genes, memes and human history: Darwinian archaeology and cultural evolution*, London: Thames and Hudson.
- Shklar, L. 2003, *Web Application Architecture: Principles, Protocols and Practices*, New Jersey: Wiley.
- Shoet, I.M., Rosenfeld, Y., Puterman M., Gilboal E. 1999, Deterioration Patterns for Maintenance Management: A Methodological Approach, in Lacasse, M.A., Vanier, D.J. (eds). *Durability of Building Materials and Components*, (1666-1678) Institute for Research in Construction, Ottawa: National Research Council Canada.
- Sillar, W. 2004, Acts of god and active material culture. Agency and commitment in the Andes, in Gardner A. (ed.) *Agency uncovered: Archaeological perspectives on social agency, power and being human*, (153-190) London: CRC Press.
- Skeet, J. 2014, *C# in Depth*, NY: Manning Publications.

- Smith, A.T. 2004, The end of the essential archaeological subject, *Archaeological dialogues* 11, 1-19
- Sonnenwald, D. H. 1999,. Evolving perspectives of human information behavior: Contexts, situations, social networks and information horizons, in T. D. Wilson & D. K. Allen (eds.), *Exploring the Contexts of Information Behavior: Proceedings of the Second International Conference in Information Needs, Seeking and Use in Different Contexts (13-15 August 1998, Sheffield, UK)*, Cambridge: Taylor Graham, 176-190.
- Sonnenwald, D.H., Wildemuth, B.M., Harmon, G.L. 2001, A research method using the concept of information horizons: An example from a study of lower socioeconomic students' information seeking behaviour, *The New Review of Information Behaviour*, 2, 65-86.
- Spaulding, A.C. 1953, Statistical Techniques for the Discovery of Artifact Types, *American Antiquity* (18)4, 305-313.
- Stanley, S.M. 1999, *Earth System History*, New York: W.H. Freeman and Company.
- Steen-McIntyre, V. 1985, Tephrochronology and its application to archaeology, in Rapp, G. Jr., Gifford, J.A. (eds.) *Archaeological geology*, (265-302) New Haven: Yale University Press.
- Stefanakis, M.I., Kalogeropoulos, K., Georgopoulos, A., Bourbou, Chr. 2015, The Kymissala (Rhodes) Archaeological Research Project (KARP): Multi-Disciplinary Experimental Research and Theoretical Issues, in Haggis, C. Antonaccio, C.M. (eds.) *Classical Archaeology in Context. Theory and Practice in Excavation in the Greek World*, (259-314), Mouton: Walter de Gruyter.
- Στεφανάκης, Μ.Ι., Πατσιαδά, Β. 2009-2011, Η αρχαιολογική έρευνα στον δήμο Κυμισαλέων (Ρόδος) κατά τα έτη 2006-2010: Μια πρώτη παρουσίαση, *Ευλιμένη*, 10-12, 63-134.
- Στεφανάκης, Μ.Ι. 2010, Πρόταση ίδρυσης αρχαιολογικού-οικολογικού πάρκου στη Ρόδο (αρχαία Κυμισάλα): Μια πρόκληση για την τοπική ανάπτυξη, στο Μπεριάτος, Η. και Μ. Παπαγεωργίου (επιμέλεια), *Χωροταξία* -

- Πολεοδομία - Περιβάλλον στον 21ο Αιώνα: Ελλάδα - Μεσόγειος*. Βόλος: Πανεπιστημιακές Εκδόσεις Θεσσαλίας.
- Stein, J.K. 1986, Coring Archaeological Sites, *American Antiquity*, (51)3, 505-527.
- Steward, J.H. 1955, *Theory of culture change*, Urbana: University of Illinois Press.
- Storey, A.A., Jones, T.L. 2011, Diffusionism in Archaeological Theory, in Jones, T. L. et al (eds.) *Polynesians in America: Pre-Columbian Contacts with the New World*, New York: Altamira Press.
- Stothers, D.M., Tucker, P.M. 2006, *The Fry Site: Archaeological and Ethnohistorical Perspectives on the Maumee River Ottawa of Northwest Ohio*, Morrisville: Lulu.com.
- Stutz, L.N. 2003, Embodied rituals and ritualized bodies: tracing ritual practices in late Mesolithic burials. *Acta Archaeologica Lundensia*, series in 80(46), Lund: Wallin and Dahlholm Boktryckeri AB.
- Stutz, D., Neward, T., Shilling, G. 2003, *Shared Source CLI Essentials*, Sebastopol, California: O'Reilly
- Sullivan, A.P. (ed.) 1998, *Surface Archaeology*, Albuquerque: University of New Mexico Press.
- Sutton, M.Q., Yohe II, R.M. 2008, *Archaeology: the science of the human past*, Boston: Pearson/Allyn and Bacon Publishers, 248-264.
- Sylenki, A.T. 1984, Understanding the relationship between surface and subsurface remains: An approach to isolating potential sources of variation, in Plog, S., Powell S. (eds.), *Papers on the Archaeology of Black Mesa, Arizona*, Vol 2, (14-38), Carbondale: Center for Archaeological Investigations, Southern Illinois University Press.
- Tabor, R. 2004, *Regional Perspectives in Archaeology: From Strategy to Narrative*, Oxford: British Archaeological Reports International series 1203.
- Tarnocai, C. 2009. Arctic Permafrost Soils, in Margesin, R. (ed.) *Permafrost Soils*, (3-17) Berlin: Springer.
- Tehrani, J. 2010, The past and future of the evolutionary taxonomy of cultures, *Journal of Evolutionary Psychology*, (8)2, 169-182.

- Tehrani, J.J., Collard, M. 2002, Investigating cultural evolution through biological phylogenetic analyses of Turkmen textiles, *Journal of Anthropological Archaeology*, 21, 443-463.
- Thomas, G. 2007, *Education and Theory: Strangers in Paradigms*, Maidenhead: Open University Press.
- Thomas, J. 2000, Reconfiguring the social, reconfiguring the material, in Schiffer, M.B. (ed.), *Social Theory in Archaeology* (143-155), Salt Lake City: University of Utah Press.
- Thomas, J., 2004, *Archaeology and Modernity*, London: Routledge,
- Thomas, K.D., 2015, Molluscs emergent, Part II: themes and trends in the scientific investigation of molluscs and their shells as past human resources, *Journal of Archaeological Science* 56, 159-167.
- Thompson, R.H., Longacre, W.A. 1966, The University of Arizona Archaeological Field School at Grasshopper, East Central Arizona, *The Kiva* 31, 255-275.
- Tian, C., Wang, L., Kaseke, K.F., Bird, B.W. 2018, Stable isotope compositions ($\delta^{2}\text{H}$, $\delta^{18}\text{O}$ and $\delta^{17}\text{O}$) of rainfall and snowfall in the central United States, *Science Reports*, 8(1), 6712
- Tilley, C. 1982, Social formation, social structures, and social change, in Hodder, I. (ed.) *Symbolic and Structural Archaeology*, (26-38) Cambridge: Cambridge University Press.
- Tilley C. 1991, *Material Culture and Text: The Art of Ambiguity*, N.Y.: Routledge
- Tilley, C. 1994, *A Phenomenology of Landscape: places, paths and monuments*. Oxford: Berg.
- Tilley, C. 1996, The Power of Rocks: Landscape and Topography on Bodmin Moor, *World Archaeology* 28, 161-176.
- Towner, R. H. 2002, Archeological dendrochronology in the southwestern United States, *Evolutionary Anthropology: Issues, News, and Reviews*, (11)2, 68-84.
- Trigger, B.G., 1998, Archaeology and Epistemology: Dialoguing across the Darwinian Chasm. *American Journal of Archaeology*, 102, 1-34.

- Trigger, B.G., 1989, *A history of archaeological thought*, Cambridge: Cambridge University Press.
- Trigger, B.G. 2007, *A History of Archaeological Thought*. New York: Cambridge University Press.
- Trigger B.G. 2007a, *Artifacts and Ideas: Essays in Archaeology*, New Jersey: Transaction Publishers.
- Tringham, R. 1991, Households with faces: The challenge of gender in prehistoric architectural remains, in Gero, J., Conkey, M. (eds.), *Engendering archaeology. Women and prehistory*, (93-131) Oxford: Wiley-Blackwell.
- Tsipenyuk, Y.M. 1997, *Nuclear Methods in Science and Technology*. Bristol, UK: Institute of Physics Publishing.
- Toby, E.S., Webster, D.L., (eds.) 2001, *Archaeology of ancient Mexico and Central America: an encyclopedia*, New York: Garland.
- Tuniz, C., Zoppi, U., Barbetti, M. 2004, Radionuclide dating in archaeology by accelerator mass spectrometry, in Martini, M., Milazzo, M. Piacentini, M. (eds.), *Physics Methods in Archaeometry*, (385-405), Amsterdam: IOS Press.
- Urry, J. 2000. *Sociology beyond societies: mobilities for the twenty-first century*. London: Routledge.
- Vaganov, E.A., Hughes, M.K., Shashkin, A.V., 2006, *Growth Dynamics of Tree-rings: An Image of Past and Future Environments*. Berlin: Springer-Verlag.
- Venugopal, K.R., Patnaik, L.M. (eds.) 2011, *Computer Networks and Intelligent Computing: 5th International Conference on Information Processing, ICIP 2011, Bangalore, India, August 5-7, 2011. Proceedings*, Berlin: Springer. 472-473
- Vischer, D. 1989, Impact of 18th and 19th century river training works: Three case studies from Switzerland, in, pp., Petts, G.E., Möller, H., Roux, A.L. (eds.), *Historical Change of Large Alluvial Rivers: Western Europe*, (19-40), N.Y.: J.Wiley & Sons.

- Wacquant, L. 2003, Positivism, in Outhwaite, W. (ed.), *The Blackwell Dictionary of Twentieth-Century Social Thought*, Oxford: Blackwell Publishing
- Wallace, S. 2012, *Contradictions of Archaeological Theory: Engaging Critical Realism and Archaeological Theory*, UK: Routledge.
- Watkins, D., Hammond, M.J., Abrams, B. 2003, *Programming in the .NET Environment*, Boston: Addison-Wesley Professional.
- Watson, P.J. LeBlanc, S.A., Redman, C.L. 1971, *Explanation in Archeology: An Explicitly Scientific Approach*, New York, NY: Columbia University Press.
- Webmoor, T., and Witmore, C.L. 2008, Things are Us! A Commentary in Human/Things Relations under the Banner of a 'Social' Archaeology, *Norwegian Archaeology Review*, (41)1, 53–70.
- Webster, G. S. 2008, Culture history: a culture-historical approach, in Bentley, R.A., Maschner, H.D.G, and C. Chippindale, (eds.) *Handbook of Archaeological Theories*, (11–27) Alta Mira Press.
- Wessel, C.J., Thom, H.C.S. 1954, Climate and deterioration, in Greathouse, G.A., Wessel C.J. (eds.) *Deterioration of materials: causes and preventive techniques*, (3-70) New York: Reinhold.
- Weymouth J.W. 1986, Geophysical Methods of Archaeological Site Surveying, in Schiffer M.B. (ed.) *Advances in Archaeological Method and Theory*, Vol. 9, (311-395), New York: Academic Press.
- White, L.A. 1959, *The Evolution of Culture*, New York: McGraw-Hill.
- Wiebert, A. 1995, *Development of the Lund AMS System and the Evaluation of a New AMS Detection Technique*, Lund: University of Lund.
- Williams, R. 1973, *The city and the country*, N.Y.: Oxford University Press.
- Williams, M. 2002, *Microsoft Visual C# .NET*, Redmond, Washington: Microsoft Press.
- Wills, D.C. 2007, *Pro Visual C++ 2005 for C# Developers: Featuring C++/CLI*, New York: Apress.
- Wylie A. 1985, Putting Shakertown Back Together: Critical Theory in Archaeology. *Journal of Anthropological Archaeology* 4, 133-147.

- Wylie, A. 1992, The interplay of evidential constraints and political interests: Recent archaeological research on gender, *American Antiquity* 57, 15-35.
- Winter, T. N. 1999, Roberto Busa, S.J., and the Invention of the Machine-Generated Concordance, *The Classical Bulletin*, 75(1), 3-20.
- Whyte, N. 2013, An Archaeology of Natural Places: Trees in the Early Modern Landscape, *Huntington Library Quarterly* 76(4) 499-517.
- Wobst, H.M. 1983, We can't see the forest for the trees: sampling and the shapes of archaeological distributions, in Moore, J., Keene, A. (eds.) *Archaeological Hammers and Theories*, (37-85), New York: Academic Press.
- Wood, W.R., Johnson, D.L. 1978, A Survey of Disturbance Processes in Archaeological Site Formation, *Advances in Archaeological Method and Theory*, 1, 315-381.
- Wright, P.J. 2009, Methodological Issues in Paleoethnobotany: A Consideration of Issues, Methods, and Cases, in VanDerwarker A.M., Peres, T.M. (eds.), *Integrating Zooarchaeology and Paleoethnobotany: A Consideration of Issues, Methods, and Cases*, (37-64) N.Y.: Springer Science.
- Yarrow, T. 2003, Artefactual Persons: The Relational Capacities of Persons and Things in the Practice of Excavation, *Norwegian Archaeological Review* 36, 65-73.
- Yoffee N. and A. Sherratt (eds.) 1993, *Archaeological theory: who sets the agenda?* Cambridge: Cambridge University Press.
- Zubrow E. B. W. 2006, Digital archaeology: a historical context in Evans Th.L. and Daly, P. (eds.), *Digital Archaeology: Bridging method and theory*, (8-28), London and New York: Routledge.

Διαδίκτυο

- ADS (Archaeology Data Service), and English Heritage, 2007, Preservation and Management Strategies for Exceptionally Large Data Formats: 'Big Data', διαθέσιμο στο https://archaeologydataservice.ac.uk/resources/attach/bigdata_final_report_1.3.pdf, ανάκτηση 30.12.2018.
- Ark (Archaeological Recording Kit), διαθέσιμο στο <https://sourceforge.net/projects/arkdb/>, ανάκτηση 27.03.2017.
- Asp.NET Webforms, διαθέσιμο στο <https://www.asp.net/web-forms>, ανάκτηση 22.5.2017.
- Caton, M. 2001, Mono Project aims to bring .Net apps to Linux, *ZDNet*, διαθέσιμο στο <https://www.zdnet.com/article/mono-project-aims-to-bring-net-apps-to-linux/>, ανάκτηση 04.06.2019.
- CKEditor Ecosystem 2019, διαθέσιμο στο <https://ckeditor.com>, ανάκτηση 05.05.2019.
- DigAR Lab. (2011). «Welcome to the DigAR Lab!», διαθέσιμο στο: <http://depts.washington.edu/digarlab/>, ανάκτηση 21.03.2017.
- Ευλιμένη* διαθέσιμη online στο <http://www.eulimene.eu>
- Guides to Good Practice, *Archaeology Data Service/Digital Antiquity*, στο <http://guides.archaeologydataservice.ac.uk/g2gp/Main>, ανάκτηση 04.09.2017.
- ECMA International 2013, *Standard ECMA-335: Common Language Infrastructure (CLI)*, Geneva, διαθέσιμο στο <http://www.ecma-international.org/publications/standards/Ecma-335.htm>, ανάκτηση 23/12/2013.
- ECMA International 2017, *Standard ECMA-334 C# Language Specification*, Geneva, διαθέσιμο στο <https://www.ecma-international.org/publications/files/ECMA-ST/ECMA-334.pdf>, ανάκτηση 08.09.2018.
- Hales, D., & Edmonds, B. 2018. Intragenerational Cultural Evolution and Ethnocentrism. *Journal of Conflict Resolution*, 1-27, διαθέσιμο στο <https://journals.sagepub.com/doi/pdf/10.1177/0022002718780481>, ανάκτηση 30.03.2018.

IBM Archives, «The birth of the IBM PC», διαθέσιμο στο: http://www-03.ibm.com/ibm/history/exhibits/pc25/pc25_birth.html ανάκτηση 29.2.2014

ICOMOS (International Council of Monuments and Sites), 1996. Principles for the recording of monuments, groups of buildings and sites, διαθέσιμο στο <https://www.icomos.org/charters/archives-e.pdf> ανάκτηση 18.02.2019.

ImageResizer 3.4.3., διαθέσιμο στο <https://www.nuget.org/packages/ImageResizer/3.4.3> ανάκτηση 10.05.2014.

ISO 2012, ISO/IEC 23271:2012, Geneva, διαθέσιμο στο <https://www.iso.org/standard/58046.html>, ανάκτηση 05.10.2013.

ISO 2018, ISO/IEC 2018, *Information technology — Programming languages — C#*, Geneva, διαθέσιμο στο https://standards.iso.org/ittf/PubliclyAvailableStandards/c075178_ISO_IEC_23270_2018.zip ανάκτηση 12.09.2018.

Jquery-ui-1.12.1., διαθέσιμο στο <https://jqueryui.com/download/all/> ανάκτηση 03.06.2017.

Lischke, M. 2015, Parsing in MySQL Workbench: the ANTLR age, *MySQL Workbench*, διαθέσιμο στο <https://mysqlworkbench.org/2015/08/parsing-in-mysql-workbench/>, ανάκτηση 05.09.2018.

Microsoft Docs, 2019, Active Directory administrative tier model, διαθέσιμο στο <https://docs.microsoft.com/en-us/windows-server/identity/securing-privileged-access/securing-privileged-access-reference-material>, ανάκτηση 09.05.2019.

McGranaghan, M. 2017. Ethnographic Analogy in Archaeology: Methodological Insights from Southern Africa. *Oxford Research Encyclopedia of African History*, διαθέσιμο στο <http://oxfordre.com/africanhistory/view/10.1093/acrefore/9780190277734.001.0001/acrefore-9780190277734-e-213>, ανάκτηση 23.03.2017.

- Morgan, C. 2011, *Health and Safety for Academics*, διαθέσιμο στο <https://colleen-morgan.com/2011/03/10/health-and-safety-for-academics/>, ανάκτηση 05.06.2016.
- Nuget 2019, *ImageResizer 3.4.3*, διαθέσιμο στο <https://www.nuget.org/packages/ImageResizer/3.4.3>, ανάκτηση 03.04.2019
- Oracle 2010, *Commercial License for OEMs, ISVs and VARs*, διαθέσιμο στο <https://www.mysql.com/about/legal/licensing/oem/> ανάκτηση 06.09.2010.
- Oracle 2019, *MySQL Workbench*, διαθέσιμο στο <https://dev.mysql.com/doc/workbench/en/> ανάκτηση 02.08.2019.
- Oracle 2019b, *MySQL Community Downloads: MySQL Workbench 6.3.10*, διαθέσιμο στο https://dev.mysql.com/downloads/workbench/?utm_source=tuicool, ανάκτηση 02.08.2018.
- Sonnenwald, D.H., Wildemuth, B.M. 2001, *Investigating information seeking behavior using the concept of information horizons*, Chapel Hill: University of North Carolina, 1-20 διαθέσιμο στο <https://sils.unc.edu/sites/default/files/general/research/TR-2001-01.pdf>, ανάκτηση 05.03.2017.
- Σταματόπουλος, Π. 2015, *Λογικός και συναρτησιακός προγραμματισμός*. [ηλεκτρ. βιβλ.], Αθήνα:Σύνδεσμος Ελληνικών Ακαδημαϊκών Βιβλιοθηκών, διαθέσιμο στο <http://hdl.handle.net/11419/3587>, ανάκτηση 02.02.2018.
- Staten Island Museum at CSI: Archaeology Study, διαθέσιμο στο https://www.brown.edu/Departments/Joukowsky_Institute/courses/greekpast/4782.html, ανάκτηση 24.07.2017.
- Technopedia 2018, *Database Row*, διαθέσιμο στο <https://www.techopedia.com/definition/4425/database-row>, ανάκτηση 29/06/2018.
- WorldWind Java, διαθέσιμο στο <https://worldwind.arc.nasa.gov/java/>, ανάκτηση 30/08/2018.

Operating System Market Share, στο *NET MARKETSHARE*, διαθέσιμο στο <https://www.netmarketshare.com/operating-system-market-share.aspx>, updated 19.07.2018, ανάκτηση 22.9.2017.

Step trenching, *Oxford Reference*, διαθέσιμο στο <https://www.oxfordreference.com/view/10.1093/oi/authority.20110803100531568>, ανάκτηση 30.03.2018

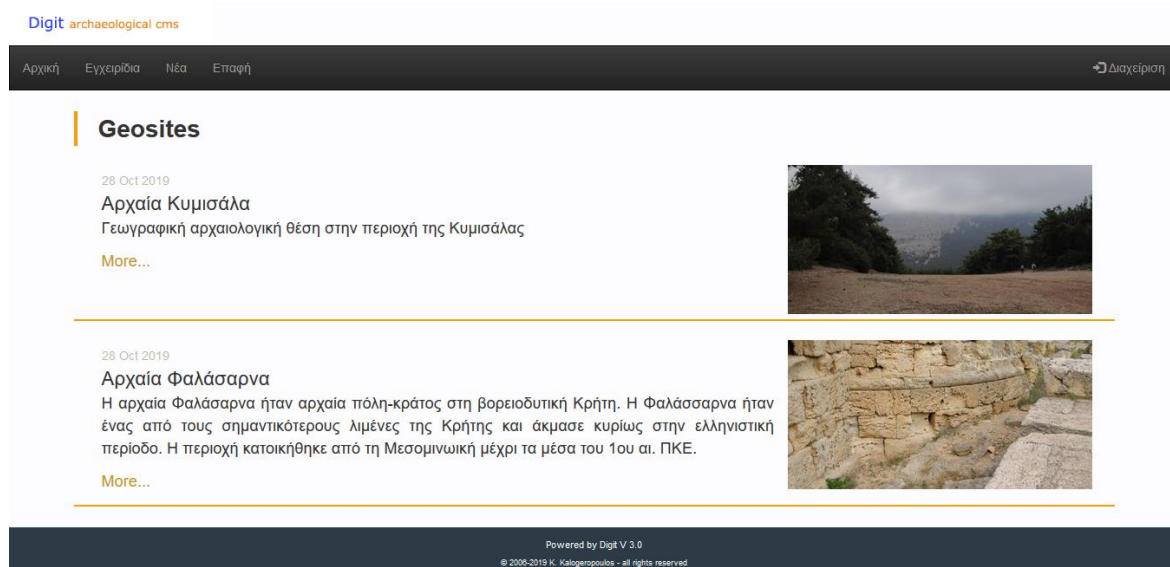
UNESDOC 2003, Guidelines for the preservation of digital heritage, διαθέσιμο στο <https://unesdoc.unesco.org/ark:/48223/pf0000130071>, ανάκτηση 20.06.2019.

Unilinear cultural evolution, *Oxford Reference*, διαθέσιμο στο <http://www.oxfordreference.com/view/10.1093/oi/authority.20110803110706530>, ανάκτηση 22.11.2016

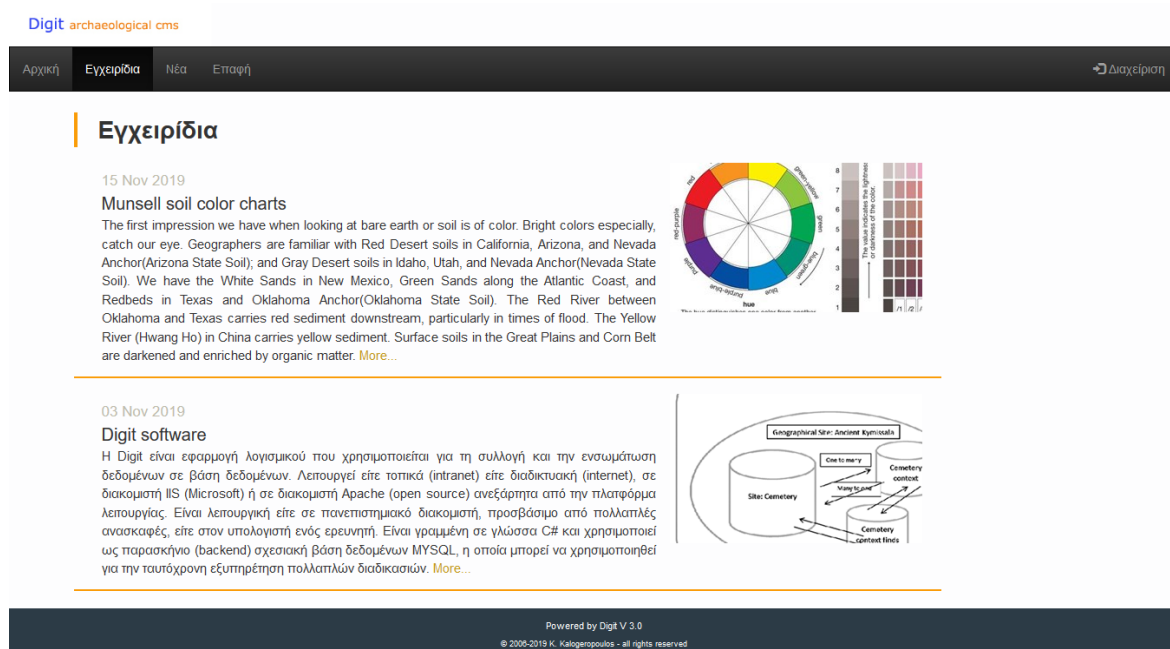
Wikipedia Commons, διαθέσιμο στο https://commons.wikimedia.org/wiki/File:Diagramm_bewuchs.jpg, ανάκτηση 30.03.2017.

Παράρτημα 1: Ο τρόπος λειτουργίας του λογισμικού

Στο παρόν παράρτημα αναπτύσσεται ο τρόπος λειτουργίας του λογισμικού έτσι όπως προκύπτει από τις frontend (crud) σελίδες του, εκκινώντας από την αρχική σελίδα.

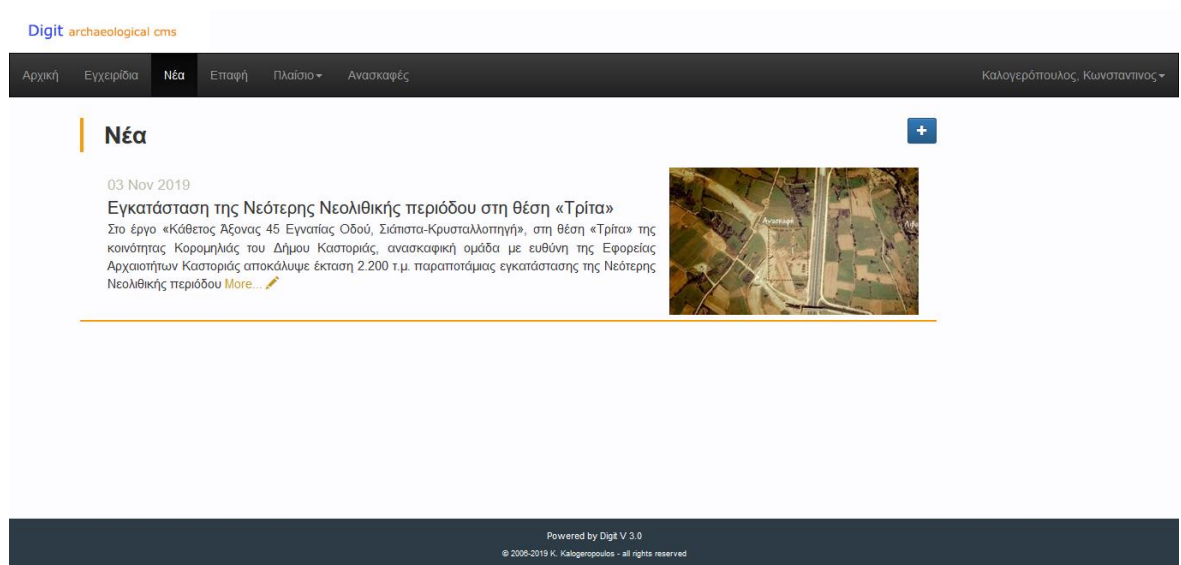


Εικ. 1 Η αρχική σελίδα του λογισμικού Digit



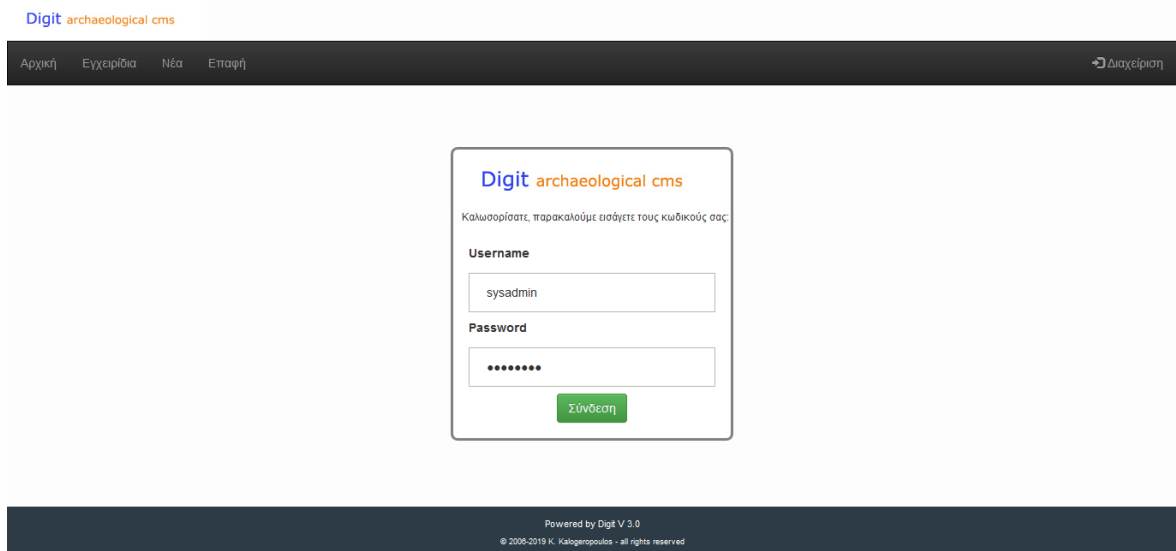
Εικ.2 Η σελίδα των καταχωρισμένων εγχειριδίων και σχετικών άρθρων για τη λειτουργία του λογισμικού Digit

Όπως φαίνεται στις εικ. 1 και 2 ο μη συνδεδεμένος χρήστης μπορεί να δει τις γεωγραφικές αρχαιολογικές θέσεις, εγχειρίδα ή σχετικά άρθρα που αφορούν στη λειτουργία του λογισμικού και αρχαιολογικά νέα ή σχετικά νέα από τον χώρο της πληροφορικής. Δεν μπορεί να προσθέσει να επεξεργαστεί ή να διαγράψει το σχετικό περιεχόμενο των σελίδων. Στην περίπτωση που είναι συνδεδεμένος, μπορεί να επεξεργαστεί σχετικά άρθρα και να προσθέσει περιεχόμενο, όπως φαίνεται στην εικ. 3.

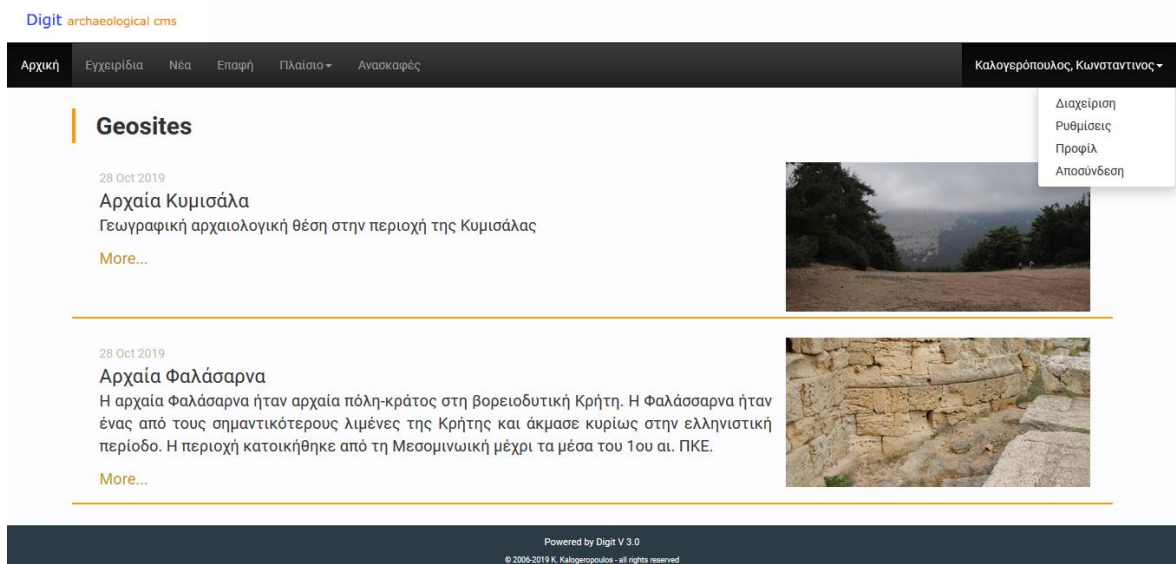


Εικ. 3. Η σελίδα των Νέων με συνδεδεμένο χρήστη

Από τον σύνδεσμο **Διαχείριση** γίνεται η σύνδεση του χρήστη με τη χρήση username και password. Επιλέγοντας *διαχείριση*, ο χρήστης οδηγείται σε σελίδα σύνδεσης, στην οποία υποχρεούται να δώσει τα κατάλληλα διαπιστευτήρια, προκειμένου να περάσει στη σελίδα διαχείρισης, όπως φαίνεται στην εικ. 4. Μέσω του drop down καταλόγου επιλογών, όπως φαίνεται στην εικ. 5, μπορεί να επιλέξει είτε να διαχειριστεί το προσωπικό του προφίλ, να περάσει στη διαχειριστική σελίδα, να επεξεργαστεί ρυθμίσεις του λογισμικού ανάλογα με το επίπεδο χρήσης που μπορεί να διαχειριστεί ή να αποσυνδεθεί από το λογισμικό.



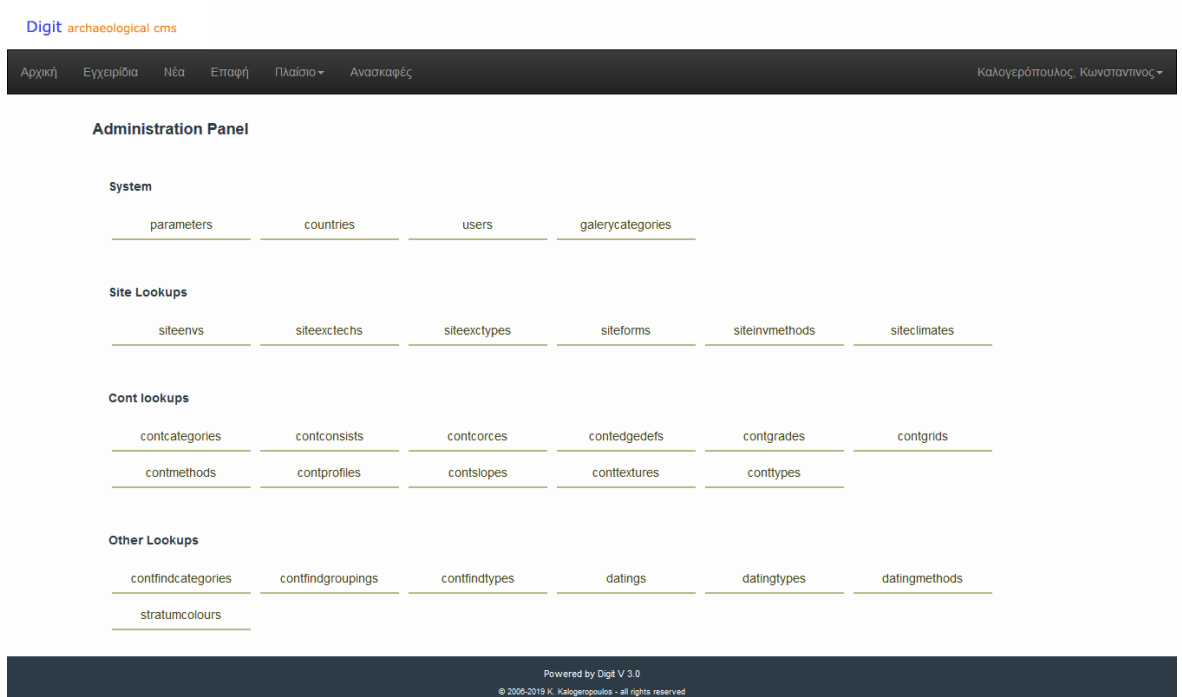
Εικ. 4 Σελίδα σύνδεσης με username και password.



Εικ. 5 Drop down επιλογές του διαχειριστικού συνδέσμου

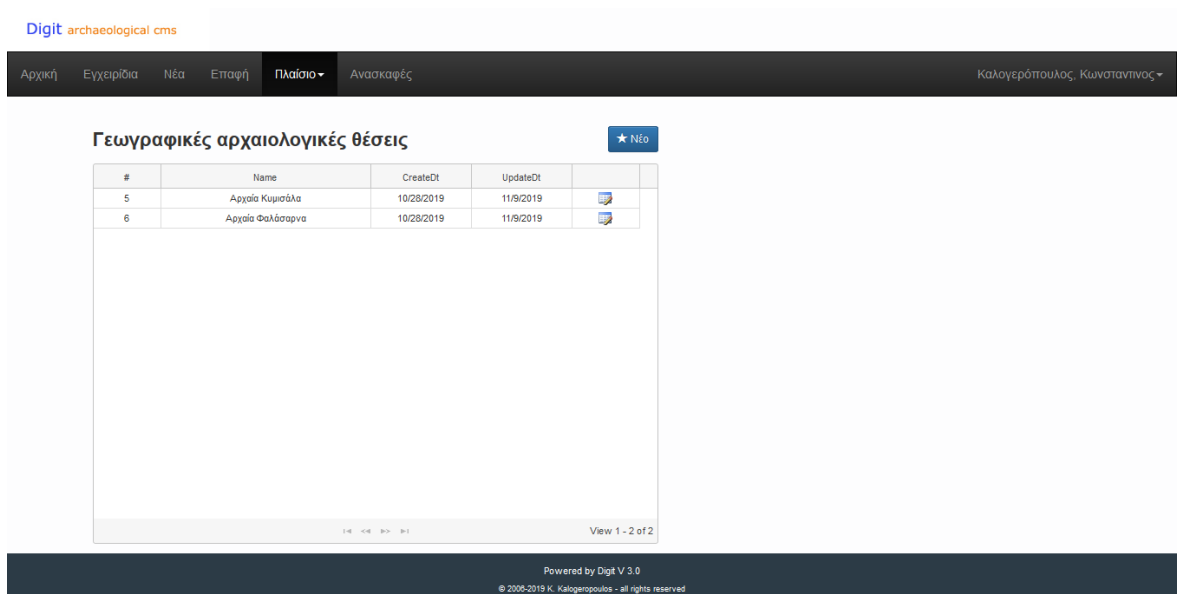
Στη σελίδα διαχείρισης, όπως φαίνεται στην εικ. 6, υπάρχουν όλες οι παράμετροι (lookups) που σχετίζονται με τη διαχείριση γεωγραφικών αρχαιολογικών θέσεων, αρχαιολογικών θέσεων, ανασκαφικών τετραγώνων, ευρημάτων και στρωματογραφικών δεδομένων. Άλλες διαχειριστικές σελίδες σχετίζονται με τους χρήστες του συστήματος και την παροχή δικαιωμάτων, με κατηγορίες φωτοθηκών και γεωγραφικά δεδομένα της χώρας στην οποία γίνεται η αρχαιολογική έρευνα. Επιπλέον παρέχει σύνδεση με τη σελίδα Πλαίσιο, όπου είναι δυνατόν να οριστούν οι αρχαιολογικές γεωγραφικές

θέσεις, οι αρχαιολογικές θέσεις, οι ανασκαφικές φάσεις και οι μεμονωμένες ανασκαφικές ή ερευνητικές μονάδες.



Εικ. 6 Σελίδα διαχείρισης παραμέτρων

Στον σύνδεσμο Πλαίσιο είναι δυνατή η δημιουργία, επεξεργασία και διαγραφή μιας αρχαιολογικής θέσης, εφόσον ο χρήστης έχει τα ανάλογα δικαιώματα, όπως φαίνεται στην εικ. 7



Εικ.7. Δημιουργία επεξεργασία και διαγραφή αρχαιολογικής γεωγραφικής θέσης.

Προκειμένου να είναι δυνατή η διαγραφή μιας γεωγραφικής αρχαιολογικής θέσης, χρειάζεται πρώτα να διαγραφεί όλο το περιεχόμενο που εξαρτάται από τη συγκεκριμένη θέση, δηλαδή οι εξαρτώμενες αρχαιολογικές θέσεις, οι εξαρτώμενες ανασκαφικές θέσεις, τα ανασκαφικά πλαίσια με τα ημερολόγια, τα τέχνηρα και τις εικόνες που περιλαμβάνει. Σε διαφορετική περίπτωση το σύστημα δεν επιτρέπει τη διαγραφή.

Digit archaeological cms

Αρχική Εγχειρίδια Νέα Επαφή **Πλαίσιο** Ανασκαφές Καλογερόπουλος Κωνσταντίνος

Sites

★New Site

#	Geosite	Name	ShowOrder	IsActive	Phases	Conts	Users	CreateDt	
4	Αρχαία Κυμισάλα	Τομέας 1 Νεκρόπολις λόφου αγ. Φωκά	0	✓	2	2	1	10/28/2019	
5	Αρχαία Κυμισάλα	Τομέας 2 Ακρόπολις λόφου αγ. Φωκά	0	✓	1	1	1	11/3/2019	
6	Αρχαία Κυμισάλα	Τομέας 3 Νεκρόπολις λόφου Κυμισάλας	0	✓	1	1	1	11/3/2019	
7	Αρχαία Κυμισάλα	Τομέας 5 Βασιικά	0	✓	2	0	1	11/4/2019	

View 1 - 4 of 4

Powered by Digit V 3.0
© 2008-2019 K. Kalogeropoulos - all rights reserved

Εικ. 8. Δημιουργία, επεξεργασία και διαγραφή αρχαιολογικής θέσης.

Digit archaeological cms

Αρχική Εγχειρίδια Νέα Επαφή **Πλαίσιο** Ανασκαφές Καλογερόπουλος Κωνσταντίνος

Phases

★New Phase

#	Geosite	Site	Name	ShowOrder	IsActive	Users	CreateDt	
4	Αρχαία Κυμισάλα	Τομέας 1 Νεκρόπολις λόφου αγ. Φω	T1 Ανασκαφή 2019	1	✓	1	11/3/2019	
5	Αρχαία Κυμισάλα	Τομέας 5 Βασιικά	Τοπογραφικό έργο 2016	0	✓	1	11/4/2019	
6	Αρχαία Κυμισάλα	Τομέας 5 Βασιικά	Τοπογραφικό έργο 2018	0	✓	1	11/5/2019	
7	Αρχαία Κυμισάλα	Τομέας 2 Ακρόπολις λόφου αγ. Φωκ	Αρχαίος δρόμος (πτεροδεία) 2007	0	✓	1	11/6/2019	
8	Αρχαία Κυμισάλα	Τομέας 3 Νεκρόπολις λόφου Κυμισά	T3 Ανασκαφή 2007	0	✓	0	11/6/2019	
9	Αρχαία Κυμισάλα	Τομέας 1 Νεκρόπολις λόφου αγ. Φω	ΝΔ Πρόποδες αγ. Φωκά (πτεροδεία) 2018	0	✓	0	11/10/2019	

View 1 - 6 of 6

Powered by Digit V 3.0
© 2008-2019 K. Kalogeropoulos - all rights reserved

Εικ. 9. Δημιουργία, επεξεργασία και διαγραφή ανασκαφικής φάσης.

Conts ★ New Cont

#	Site	Phase	Name	Supervisor	Finds	CreateDt	
5	Τομέας 1 Νεκρόπολις Λόφου αγ.	T1 Ανασκαφή 2019	Τομή 44 Τάφος 28 Τετράγωνο 12-4	Καλογερόπουλος,Κωνσταντίνος	11	11/3/2019	
7	Τομέας 3 Νεκρόπολις Λόφου Κι	T3 Ανασκαφή 2007	T. 3 Τάφος 1 Τετράγωνο 1-2	Καλογερόπουλος,Κωνσταντίνος	13	11/6/2019	
8	Τομέας 2 Ακρόπολις Λόφου αγ.	Αρχαίος δρόμος (πτεροδιά) 200	Επιφανειακή επιθεώρηση 2007 αρχαίος δρόμος	Καλογερόπουλος,Κωνσταντίνος	0	11/8/2019	
9	Τομέας 1 Νεκρόπολις Λόφου αγ.	ΝΔ Πρόποδες αγ. Φωκά (πτεροί Επιφανειακή	επιθεώρηση 2018 Λόφος αγ. Φωκά 2)	Καλογερόπουλος,Κωνσταντίνος	0	11/10/2019	

Powered by Digit V 3.0

© 2006-2019 K. Καλογερόπουλος - all rights reserved

Εικ. 10. Δημιουργία, επεξεργασία και διαγραφή ανασκαφικού πλαισίου.

Η δυνατότητα διαγραφής της αρχαιολογικής θέσης, της ανασκαφικής φάσης ή του ανασκαφικού πλαισίου είναι δυνατή, όπως και στην περίπτωση της αρχαιολογικής γεωγραφικής θέσης μόνον όταν έχουν διαγραφεί down top όλες οι εξαρτώμενες ιεραρχικά οντότητες.

 Edit 'Τομή 44 Τάφος 28 Τετράγωνο 12-4' (cancel and return back)

Save Delete

Info Dimensions Location Attrs Users Maps Photos Stratums Information Horizon Graphs

Name

Τομή 44 Τάφος 28 Τετράγωνο 12-4

Summary

Τομή στο ν/δ άκρο του καννάβου

Description

 Source **B I U S** | | | | | Font | Size

Η συγκεκριμένη τομή αποκάλυψε τάφο με κτερίσματα χρονολογημένα στον 3ο αι. ΠΚΕ

Powered by Digit V 3.0

© 2006-2019 K. Καλογερόπουλος - all rights reserved

Εικ. 11. Σελίδα επεξεργασίας ανασκαφικού πλαισίου

Στη σελίδα επεξεργασίας πλαισίου (εικ. 11) παρέχονται σύνδεσμοι σχετικοί με τις διαστάσεις του πλαισίου, τη γεωγραφική του θέση (Geolocation), τις ιδιότητές του και τους συνδεδεμένους χρήστες του, με δυνατότητα προσθαφαίρεσης συνδεδεμένων χρηστών. Παρέχονται, επίσης, σύνδεσμοι για την εξάρτηση χαρτών και εικόνων, όπως και στρωματογραφικών δεδομένων (εικ. 12). Εντέλει υφίστανται σύνδεσμοι για τον ορίζοντα πληροφόρησης (εικ. 12), μέσω του οποίου είναι δυνατή η περαιτέρω μελέτη τεχνέργων ή άλλων πολιτισμικών δεδομένων της περιοχής στο διαδίκτυο ή την τοπική βιβλιοθήκη της ερευνώμενης περιοχής. Με τον σύνδεσμο graphs προβάλλονται στατιστικά στοιχεία που αφορούν στο εν λόγω ανασκαφικό πλαίσιο.

The screenshot shows the 'Stratums' tab in the Digit archaeological cms interface. The table below represents the data shown in the interface:

#	Name	Colour	CreateDt		
+	5	humus	dark brown munsell	11/3/2019	
+	6	Str 1	orange munsell	11/3/2019	
+	7	str 1 θάλαμος φερτές ύλες	light brown munsell	11/3/2019	
+	8	str 2 θάλαμος	light brown munsell	11/3/2019	
+	17	str 3 θάλαμος rock bed	dark gray munsell	11/16/2019	

Εικ. 12. Ο σύνδεσμος στρωματογραφικών δεδομένων ενός ανασκαφικού πλαισίου

Όπως φαίνεται στην εικ 12. τα καταγεγραμμένα στρώματα του ανασκαφικού πλαισίου, βάσει ημερολογίου, καταχωρούνται ιεραρχικά (top down) με τη χρήση της χρωματικής κλίμακας Munshell κατά την ανασκαφική διαδικασία.

Edit 'Τομή 44 Τάφος 28 Τετράγωνο 12-4' (cancel and return back)

Save Delete

Info Dimensions Location Attrs Users Maps Photos Stratums Information Horizon Graphs



Data visualization



Powered by Digit V 3.0

© 2006-2019 Κ. Καλογερόπουλος - all rights reserved

Εικ. 13. Οπτικοποίηση δεδομένων (data visualization) και στατιστικά γραφήματα τεχνέργων ανασκαφικού πλαισίου.

Όντας συνδεδεμένος ο χρήστης βλέπει στον σύνδεσμο **Ανασκαφές** τις ανασκαφικές φάσεις και τις ανασκαφές με τις οποίες είναι συνδεδεμένος και μπορεί να επεξεργαστεί (εικ 14). Στην ίδια σελίδα είναι δυνατή η παρακολούθηση των επεξεργασιών που έγιναν από άλλους χρήστες (flow).

Οι ανασκαφές μου

Τομέας 1 Νεκρόπολις λόφου αγ. Φωκά - Τ1 Ανασκαφή 2019

Τομή 44 Τάφος 28 Τετράγωνο 12-4

Τομέας 1 Νεκρόπολις λόφου αγ. Φωκά - ΝΔ Πρόποδες αγ. Φωκά (περιοδεία) 2018

Επιφανειακή επιθεώρηση 2018 λόφος αγ. Φωκά 2018




Τομέας 2 Ακρόπολις λόφου αγ. Φωκά - Αρχαίος δρόμος (περιοδεία) 2007

Επιφανειακή επιθεώρηση 2007 αρχαίος δρόμος

Τομέας 3 Νεκρόπολις λόφου Κυμισιάς - Τ3 Ανασκαφή 2007

Τ. 3 Τάφος 1 Τετράγωνο 1-2

Latest posts...






- 
Κωνσταντίνος Καλογερόπουλος before 42 days
 147 Σελίδα ημερολογίου
 ...
- 
Κωνσταντίνος Καλογερόπουλος before 42 days
 197 σελίδα ημερολογίου
 Πραγματοποιήθηκε επιφανειακός καθαρισμός και εκ νέου φωτογράφιση τ...
- 
Κωνσταντίνος Καλογερόπουλος before 42 days
 Νοτιοδυτικοί πρόποδες λόφου Αγίου Φωκά, 200 περίπου μέτρα νοτιοανατολικά του χώρου της φετινής α...

Εικ. 14. Η σελίδα Ανασκαφές

Ημερολόγιο Πλαίσιου: Τομή 44 Τάφος 28 Τετράγωνο 12-4

★New Entry

(Τομέας 1 Νεκρόπολις λόφου αγ. Φωκά) - (Τ1 Ανασκαφή 2019)

#	Title	Findings	CreateDt	
16	εγγραφή #26062019	5	11/3/2019	
15	εγγραφή #25062019	2	11/3/2019	
14	εγγραφή #24062019	1	11/3/2019	
13	εγγραφή #21062019	2	11/3/2019	
12	εγγραφή #20062019	1	11/3/2019	

Εικ. 15 Σελίδα πρόσθεσης και επεξεργασίας ημερολογιακών εγγραφών.

Μέσω της σελίδας **Ανασκαφές** και εφόσον ακολουθήσει τους συνδέσμους που παρέχονται ο συνδεδεμένος χρήστης οδηγείται σε σελίδα πρόσθεσης και επεξεργασίας ημερολογιακών εγγραφών για ένα συγκεκριμένο ανασκαφικό πλαίσιο (εικ. 15).

Παράρτημα 2: Οι πίνακες της βάσης δεδομένων Digit και τα αντίστοιχα SQL queries (Mysql Workbench)

Πίνακας	MySQL query
<p>Table: articles</p> <p>Columns: articleId int(11) AI PK site int(11) phase int(11) title varchar(256) summary varchar(2048) body longtext isvisible tinyint(4) author varchar(128) sourceName varchar(128) sourceURL varchar(256) url varchar(256) createDt datetime createBy int(11) updateDt datetime updateBy int(11)</p>	<pre>CREATE TABLE `articles` (`articleId` int(11) NOT NULL AUTO_INCREMENT, `site` int(11) DEFAULT NULL, `phase` int(11) DEFAULT NULL, `title` varchar(256) NOT NULL, `summary` varchar(2048) DEFAULT NULL, `body` longtext, `isvisible` tinyint(4) NOT NULL, `author` varchar(128) DEFAULT NULL, `sourceName` varchar(128) DEFAULT NULL, `sourceURL` varchar(256) DEFAULT NULL, `url` varchar(256) DEFAULT NULL, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`articleId`)) ENGINE=InnoDB DEFAULT CHARSET=utf8;</pre>
<p>Table: contcategories</p> <p>Columns: categoryId smallint(6) AI PK name varchar(128) description longtext createDt datetime createBy int(11) updateDt datetime updateBy int(11)</p>	<pre>CREATE TABLE `contcategories` (`categoryId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`categoryId`)) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=utf8 COMMENT='είδος πλακιδίου';</pre>
<p>Table: contconsists</p> <p>Columns: consistId smallint(6) AI PK name varchar(128)</p>	<pre>CREATE TABLE `contconsists` (`consistId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext,</pre>

<pre> description longtext createDt datetime createBy int(11) updateDt datetime updateBy int(11) </pre>	<pre> `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`consistId`)) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=utf8 COMMENT='συμπερίληψη'; /*!40101 SET character_set_client @saved_cs_client */; </pre>
<p>Table: contcorces</p> <p>Columns:</p> <pre> corceId smallint(6) AI PK name varchar(128) description longtext createDt datetime createBy int(11) updateDt datetime updateBy int(11) </pre>	<pre> CREATE TABLE `contcorces` (`corceId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`corceId`)) ENGINE=InnoDB AUTO_INCREMENT=8 DEFAULT CHARSET=utf8 COMMENT='είδος απόθεσης'; </pre>
<p>Table: contedgedefs</p> <p>Columns:</p> <pre> edgedefId smallint(6) AI PK name varchar(128) description longtext createDt datetime createBy int(11) updateDt datetime updateBy int(11) </pre>	<pre> CREATE TABLE `contedgedefs` (`edgedefId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`edgedefId`)) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8 COMMENT='μορφή πλαισίου'; </pre>
<p>Table: contfindcategories</p> <p>Columns:</p> <pre> categoryId smallint(6) AI PK name varchar(128) description longtext createDt datetime createBy int(11) updateDt datetime updateBy int(11) </pre>	<pre> CREATE TABLE `contfindcategories` (`categoryId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`categoryId`) </pre>

	<pre>) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=utf8 COMMENT='είδη ευρημάτων';</pre>
<p>Table: contfindgroupings</p> <p>Columns: groupingId smallint(6) AI PK name varchar(128) description longtext createDt datetime createBy int(11) updateDt datetime updateBy int(11)</p>	<pre>CREATE TABLE `contfindgroupings` (`groupingId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`groupingId`)) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=utf8 COMMENT='τρόπος αρίθμησης';</pre>
<p>Table: contfinds</p> <p>Columns: contfindid int(11) AI PK cont int(11) phase int(11) name varchar(128) summary varchar(1024) description longtext code varchar(32) catalogue varchar(32) north varchar(32) east varchar(32) type smallint(6) subtype smallint(6) category smallint(6) grouping smallint(6) weight int(11) stratum int(11) substratum int(11) dating smallint(6) datingtype smallint(6) datingmethod smallint(6) createDt datetime createBy int(11) updateDt datetime updateBy int(11)</p>	<pre>CREATE TABLE `contfinds` (`contfindid` int(11) NOT NULL AUTO_INCREMENT, `diary` int(11) NOT NULL DEFAULT '0', `cont` int(11) NOT NULL, `phase` int(11) NOT NULL, `name` varchar(128) NOT NULL, `summary` varchar(1024) DEFAULT NULL, `description` longtext, `code` varchar(32) NOT NULL, `catalogue` varchar(32) NOT NULL, `north` varchar(32) NOT NULL, `east` varchar(32) NOT NULL, `type` smallint(6) NOT NULL, `subtype` smallint(6) DEFAULT NULL, `category` smallint(6) NOT NULL, `grouping` smallint(6) NOT NULL, `weight` int(11) NOT NULL, `stratum` int(11) NOT NULL, `substratum` int(11) DEFAULT NULL, `dating` smallint(6) NOT NULL, `datingtype` smallint(6) NOT NULL, `datingmethod` smallint(6) NOT NULL, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`contfindid`), KEY `fk_contfinds_conts_idx` (`cont`),</pre>

	<pre> KEY `fk_contfinds_contfindcategories_idx` (`category`), KEY `fk_contfinds_contfindgroupings_idx` (`grouping`), KEY `fk_contfinds_contfindtypes_idx` (`type`), KEY `fk_contfinds_datings_idx` (`dating`), KEY `fk_contfinds_datingtypes_idx` (`datingtype`), KEY `fk_contfinds_datingmethods_idx` (`datingmethod`), KEY `fk_contfinds_stratum_idx` (`stratum`), KEY `fk_contfinds_substratum_idx` (`substratum`), KEY `fk_contfinds_contfindsubtypes_idx` (`subtype`), CONSTRAINT `fk_contfinds_contfindcategories` FOREIGN KEY (`category`) REFERENCES `contfindcategories` (`categoryId`), CONSTRAINT `fk_contfinds_contfindgroupings` FOREIGN KEY (`grouping`) REFERENCES `contfindgroupings` (`groupingId`), CONSTRAINT `fk_contfinds_contfindsubtypes` FOREIGN KEY (`subtype`) REFERENCES `contfindsubtypes` (`subtypeid`) ON DELETE NO ACTION ON UPDATE NO ACTION, CONSTRAINT `fk_contfinds_contfindtypes` FOREIGN KEY (`type`) REFERENCES `contfindtypes` (`findtypeid`), CONSTRAINT `fk_contfinds_conts` FOREIGN KEY (`cont`) REFERENCES `conts` (`contid`), CONSTRAINT `fk_contfinds_datingmethods` FOREIGN KEY (`datingmethod`) REFERENCES `datingmethods` (`methodid`), CONSTRAINT `fk_contfinds_datings` FOREIGN KEY (`dating`) REFERENCES `datings` (`datingid`), CONSTRAINT `fk_contfinds_datingtypes` FOREIGN KEY (`datingtype`) REFERENCES `datingtypes` (`typeid`), CONSTRAINT `fk_contfinds_stratum` FOREIGN KEY (`stratum`) REFERENCES `stratums` (`stratumid`), CONSTRAINT `fk_contfinds_substratum` FOREIGN KEY (`substratum`) REFERENCES `substratums` </pre>
--	---

	<pre>(`substratumid`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='ευρήματα';</pre>
<p>Table: <code>contfindtypes</code></p> <p>Columns: findTypeId smallint(6) AI PK code varchar(32) name varchar(128) description longtext createDt datetime createBy int(11) updateDt datetime updateBy int(11)</p>	<pre>CREATE TABLE `contfindtypes` (`findTypeId` smallint(6) NOT NULL AUTO_INCREMENT, `code` varchar(32) NOT NULL, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`findTypeId`)) ENGINE=InnoDB AUTO_INCREMENT=15 DEFAULT CHARSET=utf8 COMMENT='τύπος ευρήματος';</pre>
<p>Table: <code>contfindsubtypes</code></p> <p>Columns: subtypeid smallint(6) AI PK findType smallint(6) code varchar(32) name varchar(128) description longtext createDt datetime createBy int(11) updateDt datetime updateBy int(11)</p>	<pre>CREATE TABLE `contfindsubtypes` (`subtypeid` smallint(6) NOT NULL AUTO_INCREMENT, `findType` smallint(6) NOT NULL, `code` varchar(32) NOT NULL, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`subtypeid`), KEY `FK_contfindsubtypes_contfindtypes_idx` (`findType`), CONSTRAINT `FK_contfindsubtypes_contfindtypes` FOREIGN KEY (`findType`) REFERENCES `contfindtypes` (`findTypeId`) ON DELETE NO ACTION ON UPDATE NO ACTION) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='υπότυπος ευρήματος';</pre>
<p>Table: <code>contgrades</code></p> <p>Columns: gradeId smallint(6) AI PK name varchar(128) description longtext</p>	<pre>CREATE TABLE `contgrades` (`gradeId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL,</pre>

<pre> createDt datetime createBy int(11) updateDt datetime updateBy int(11) </pre>	<pre> `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`gradeId`)) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=utf8 COMMENT='βαθμολόγηση'; </pre>
<p>Table: contgrids</p> <p>Columns:</p> <pre> gridId smallint(6) AI PK name varchar(128) description longtext createDt datetime createBy int(11) updateDt datetime updateBy int(11) </pre>	<pre> CREATE TABLE `contgrids` (`gridId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`gridId`)) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8 COMMENT='ενοποιήσεις'; </pre>
<p>Table: contmethods</p> <p>Columns:</p> <pre> methodId smallint(6) AI PK name varchar(128) description longtext createDt datetime createBy int(11) updateDt datetime updateBy int(11) </pre>	<pre> CREATE TABLE `contmethods` (`methodId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`methodId`)) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=utf8 COMMENT='μέθοδος ανασκαφής πλασιού'; </pre>
<p>Table: contprofiles</p> <p>Columns:</p> <pre> profileId smallint(6) AI PK name varchar(128) description longtext createDt datetime createBy int(11) updateDt datetime updateBy int(11) </pre>	<pre> CREATE TABLE `contprofiles` (`profileId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`profileId`)) ENGINE=InnoDB AUTO_INCREMENT=19 DEFAULT CHARSET=utf8 COMMENT='προφίλ πλασιού'; </pre>
<p>Table: conts</p>	<pre> CREATE TABLE `conts` (`site` int(11) NOT NULL, </pre>

<p>Columns:</p> <p>site int(11) phase int(11) contId int(11) AI PK name varchar(128) summary varchar(1024) description longtext arbitrary tinyint(4) supervisor int(11) height int(11) width int(11) length int(11) mindepth int(11) maxdepth int(11) volume int(11) latitude float longitude float mapZoomLevel smallint(6) showOrder int(11) conttype smallint(6) contmethod smallint(6) contcategory smallint(6) contgrade smallint(6) contslope smallint(6) contprofile smallint(6) contdedgedef smallint(6) contconsist smallint(6) conttexture smallint(6) contcorce smallint(6) contgrid smallint(6) dirName varchar(64) createDt datetime createBy int(11) updateDt datetime updateBy int(11)</p>	<pre> `phase` int(11) NOT NULL, `contId` int(11) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `summary` varchar(1024) DEFAULT NULL, `description` longtext, `arbitrary` tinyint(4) NOT NULL DEFAULT '0', `supervisor` int(11) NOT NULL, `height` int(11) DEFAULT NULL COMMENT 'εκατοστά (cm)', `width` int(11) DEFAULT NULL COMMENT 'εκατοστά (cm)', `length` int(11) DEFAULT NULL, `mindepth` int(11) DEFAULT NULL COMMENT 'εκατοστά (cm)', `maxdepth` int(11) DEFAULT NULL COMMENT 'εκατοστά (cm)', `volume` int(11) DEFAULT NULL COMMENT 'κυβικά εκατοστά (cm)', `latitude` float DEFAULT NULL, `longitude` float DEFAULT NULL, `mapZoomLevel` smallint(6) NOT NULL, `showOrder` int(11) NOT NULL, `conttype` smallint(6) NOT NULL, `contmethod` smallint(6) NOT NULL, `contcategory` smallint(6) NOT NULL, `contgrade` smallint(6) NOT NULL, `contslope` smallint(6) NOT NULL, `contprofile` smallint(6) NOT NULL, `contdedgedef` smallint(6) NOT NULL, `contconsist` smallint(6) NOT NULL, `conttexture` smallint(6) NOT NULL, `contcorce` smallint(6) NOT NULL, `contgrid` smallint(6) NOT NULL, `dirName` varchar(64) NOT NULL, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`contId`), KEY `fk_conts_contcategories_idx` (`contcategory`), KEY `fk_conts_contconsists_idx` </pre>
--	---

	<pre> (`contconsist`), KEY `fk_conts_contcorces_idx` (`contcorce`), KEY `fk_conts_contedgedefs_idx` (`contedgedef`), KEY `fk_conts_contgrades_idx` (`contgrade`), KEY `fk_conts_contmethods_idx` (`contmethod`), KEY `fk_conts_contprofiles_idx` (`contprofile`), KEY `fk_conts_contslopes_idx` (`contslope`), KEY `fk_conts_conttextures_idx` (`conttexture`), KEY `fk_conts_conttypes_idx` (`conttype`), KEY `fk_conts_contgrids_idx` (`contgrid`), KEY `fk_conts_phases_idx` (`phase`), KEY `fk_conts_users_idx` (`supervisor`), KEY `fk_conts_sites_idx` (`site`), CONSTRAINT `fk_conts_contcategories` FOREIGN KEY (`contcategory`) REFERENCES `contcategories` (`categoryId`), CONSTRAINT `fk_conts_contconsists` FOREIGN KEY (`contconsist`) REFERENCES `contconsists` (`consistId`), CONSTRAINT `fk_conts_contcorces` FOREIGN KEY (`contcorce`) REFERENCES `contcorces` (`corceId`), CONSTRAINT `fk_conts_contedgedefs` FOREIGN KEY (`contedgedef`) REFERENCES `contedgedefs` (`edgedefId`), CONSTRAINT `fk_conts_contgrades` FOREIGN KEY (`contgrade`) REFERENCES `contgrades` (`gradeId`), CONSTRAINT `fk_conts_contgrids` FOREIGN KEY (`contgrid`) REFERENCES `contgrids` (`gridId`), CONSTRAINT `fk_conts_contmethods` FOREIGN KEY (`contmethod`) REFERENCES `contmethods` (`methodId`), CONSTRAINT `fk_conts_contprofiles` FOREIGN KEY (`contprofile`) REFERENCES `contprofiles` (`profileId`), CONSTRAINT `fk_conts_contslopes` FOREIGN KEY (`contslope`) REFERENCES `contslopes` (`slopeid`), </pre>
--	---

	<pre> CONSTRAINT `fk_conts_conttextures` FOREIGN KEY (`conttexture`) REFERENCES `conttextures` (`textureid`), CONSTRAINT `fk_conts_conttypes` FOREIGN KEY (`conttype`) REFERENCES `conttypes` (`typeid`), CONSTRAINT `fk_conts_phases` FOREIGN KEY (`phase`) REFERENCES `phases` (`phaseid`), CONSTRAINT `fk_conts_sites` FOREIGN KEY (`site`) REFERENCES `sites` (`siteid`), CONSTRAINT `fk_conts_users` FOREIGN KEY (`supervisor`) REFERENCES `users` (`userid`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Πλαίσια'; </pre>
<p>Table: contslopes</p> <p>Columns: slopeId smallint(6) AI PK name varchar(128) description longtext createDt datetime createBy int(11) updateDt datetime updateBy int(11)</p>	<pre> CREATE TABLE `contslopes` (`slopeId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`slopeId`)) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8 COMMENT='κλίση πεδίου'; </pre>
<p>Table: conttextures</p> <p>Columns: textureId smallint(6) AI PK name varchar(128) description longtext createDt datetime createBy int(11) updateDt datetime updateBy int(11)</p>	<pre> CREATE TABLE `conttextures` (`textureId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`textureId`)) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=utf8 COMMENT='υφή πλαισίου'; </pre>
<p>Table: conttypes</p> <p>Columns: typeId smallint(6) AI PK name varchar(128) description longtext</p>	<pre> CREATE TABLE `conttypes` (`typeid` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, </pre>

<pre> createDt datetime createBy int(11) updateDt datetime updateBy int(11) </pre>	<pre> `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`typeId`)) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=utf8 COMMENT='διαστρωμάτωση'; </pre>
<p>Table: countries</p> <p>Columns:</p> <pre> countryId int(11) AI PK long_name varchar(128) short_name varchar(45) createDt datetime createBy int(11) updateDt datetime updateBy int(11) </pre>	<pre> CREATE TABLE `countries` (`countryId` int(11) NOT NULL AUTO_INCREMENT, `long_name` varchar(128) NOT NULL, `short_name` varchar(45) DEFAULT NULL, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`countryId`)) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8; </pre>
<p>Table: datingmethods</p> <p>Columns:</p> <pre> methodId smallint(6) AI PK name varchar(128) description varchar(1024) createDt datetime createBy int(11) updateDt datetime updateBy int(11) </pre>	<pre> CREATE TABLE `datingmethods` (`methodId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` varchar(1024) DEFAULT NULL, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`methodId`)) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=utf8 COMMENT='μέθοδος χρονολόγησης'; </pre>
<p>Table: datings</p> <p>Columns:</p> <pre> datingId smallint(6) AI PK name varchar(128) description varchar(1024) createDt datetime createBy int(11) updateDt datetime updateBy int(11) </pre>	<p>Χρονολόγηση (οντότητα)</p>
<p>Table: datingtypes</p> <p>Columns:</p> <pre> typeId smallint(6) AI PK name varchar(128) </pre>	<p>Τύπος χρονολόγησης: Σχετική, απόλυτη (παραμετρικός πίνακας ιδιοτήτων της οντότητας datings)</p>

description varchar(1024) createdDt datetime createBy int(11) updatedDt datetime updateBy int(11)	
Table: diary Columns: site int(11) phase int(11) cont int(11) diaryId int(11) AI PK title varchar(256) body longtext dirName varchar(64) createDt datetime createBy int(11) updateDt datetime updateBy int(11)	CREATE TABLE `diary` (`site` int(11) NOT NULL, `phase` int(11) NOT NULL, `cont` int(11) NOT NULL, `diaryId` int(11) NOT NULL AUTO_INCREMENT, `title` varchar(256) DEFAULT NULL, `body` longtext, `dirName` varchar(64) NOT NULL, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`diaryId`), KEY `fk_diary_conts_idx` (`site`,`phase`,`cont`), KEY `fk_diary_phases_idx` (`phase`), KEY `fk_diary_conts_idx1` (`cont`), KEY `fk_diary_users1_idx` (`createBy`), KEY `fk_diary_users2_idx` (`updateBy`), CONSTRAINT `fk_diary_conts` FOREIGN KEY (`cont`) REFERENCES `conts` (`contId`) ON DELETE NO ACTION ON UPDATE NO ACTION, CONSTRAINT `fk_diary_phases` FOREIGN KEY (`phase`) REFERENCES `phases` (`phaseId`) ON DELETE NO ACTION ON UPDATE NO ACTION, CONSTRAINT `fk_diary_sites` FOREIGN KEY (`site`) REFERENCES `sites` (`siteId`) ON DELETE NO ACTION ON UPDATE NO ACTION, CONSTRAINT `fk_diary_users1` FOREIGN KEY (`createBy`) REFERENCES `users` (`userId`) ON DELETE NO ACTION ON UPDATE NO ACTION, CONSTRAINT `fk_diary_users2` FOREIGN KEY (`updateBy`) REFERENCES `users` (`userId`) ON DELETE NO ACTION ON UPDATE NO ACTION) ENGINE=InnoDB DEFAULT CHARSET=utf8;
Table: files	CREATE TABLE `files` (`fileId` int(11) NOT NULL AUTO_INCREMENT,

<p>Columns:</p> <p>fileId int(11) AI PK filetype tinyint(4) ownerName varchar(45) ownerId int(11) name varchar(128) description varchar(1024) imagecategory tinyint(4) orgfilename varchar(128) mngfilename varchar(128) filepath varchar(256) createdt datetime createBy int(11) updatedt datetime updateBy int(11)</p>	<pre> `filetype` tinyint(4) NOT NULL COMMENT 'Είδος αρχείου (0-> εικονα, 1->pdf, 2->word, ...)', `ownerName` varchar(45) NOT NULL COMMENT 'Το ονομα του πίνακα στον οποίο ανήκει η συγκεκριμένη εγγραφή', `ownerId` int(11) NOT NULL COMMENT 'Το id της εγγραφής απο τον owner στον οποίο ανήκει αυτή η εγγραφή', `name` varchar(128) DEFAULT NULL, `description` varchar(1024) DEFAULT NULL, `imagecategory` tinyint(4) DEFAULT NULL, `orgfilename` varchar(128) NOT NULL, `mngfilename` varchar(128) NOT NULL, `filepath` varchar(256) NOT NULL, `createdt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updatedt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`fileId`)) ENGINE=InnoDB DEFAULT CHARSET=utf8; </pre>
<p>Table: galleries</p> <p>Columns:</p> <p>galeryId int(11) AI PK category smallint(6) name varchar(128) description longtext isvisible tinyint(4) createdt datetime createBy int(11) updatedt datetime updateBy int(11)</p>	<pre> CREATE TABLE `galleries` (`galeryId` int(11) NOT NULL AUTO_INCREMENT, `category` smallint(6) NOT NULL, `name` varchar(128) NOT NULL, `description` longtext, `isvisible` tinyint(4) NOT NULL, `createdt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updatedt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`galeryId`)) ENGINE=InnoDB DEFAULT CHARSET=utf8; </pre>
<p>Table: gallerycategories</p> <p>Columns:</p> <p>categoryId smallint(6) AI PK name varchar(128) description longtext createdt datetime createBy int(11) updatedt datetime updateBy int(11)</p>	<pre> CREATE TABLE `gallerycategories` (`categoryId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createdt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updatedt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`categoryId`) </pre>

) ENGINE=InnoDB AUTO_INCREMENT=8 DEFAULT CHARSET=utf8;
Table: galeryphotos Columns: photoId int(11) AI PK galery int(11) fileid int(11) showorder int(11) createDt datetime createBy int(11) updateDt datetime updateBy int(11)	CREATE TABLE `galeryphotos` (`photoId` int(11) NOT NULL AUTO_INCREMENT, `galery` int(11) NOT NULL, `fileid` int(11) NOT NULL, `showorder` int(11) NOT NULL, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`photoId`)) ENGINE=InnoDB DEFAULT CHARSET=utf8;
Table: geosites Columns: geositeId int(11) AI PK name varchar(128) summary varchar(1024) description longtext country int(11) latitude decimal(18,15) longitude decimal(18,15) mapZoomLevel smallint(6) dirName varchar(64) createDt datetime createBy int(11) updateDt datetime updateBy int(11)	CREATE TABLE `geosites` (`geositeId` int(11) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `summary` varchar(1024) DEFAULT NULL, `description` longtext, `country` int(11) NOT NULL, `latitude` decimal(18,15) DEFAULT NULL, `longitude` decimal(18,15) DEFAULT NULL, `mapZoomLevel` smallint(6) NOT NULL, `dirName` varchar(64) NOT NULL, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`geositeId`), KEY `fk_geosites_countries_idx` (`country`), CONSTRAINT `fk_geosites_countries` FOREIGN KEY (`country`) REFERENCES `countries` (`countryId`)) ENGINE=InnoDB DEFAULT CHARSET=utf8;
Table: phases Columns: phaseId int(11) AI PK site int(11) name varchar(128) summary varchar(1024) description longtext start datetime end datetime	CREATE TABLE `phases` (`phaseId` int(11) NOT NULL AUTO_INCREMENT, `site` int(11) NOT NULL, `name` varchar(128) NOT NULL, `summary` varchar(1024) DEFAULT NULL, `description` longtext, `start` datetime NOT NULL, `end` datetime DEFAULT NULL, `showOrder` int(11) NOT NULL, `isactive` tinyint(4) NOT NULL DEFAULT '0',

<pre>showOrder int(11) inactive tinyint(4) dirName varchar(64) createDt datetime createBy int(11) updateDt datetime updateBy int(11)</pre>	<pre>`dirName` varchar(64) NOT NULL, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`phaseId`), KEY `fk_phases_sites_idx` (`site`), CONSTRAINT `fk_phases_sites` FOREIGN KEY (`site`) REFERENCES `sites` (`siteid`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Ανασκαφικές περιόδου';</pre>
<p>Table: sessions</p> <p>Columns:</p> <pre>accessTokenId binary(16) PK user int(11) startdt datetime enddt datetime lastactivity datetime</pre>	<pre>CREATE TABLE `sessions` (`accessTokenId` binary(16) NOT NULL, `user` int(11) NOT NULL, `startdt` datetime NOT NULL, `enddt` datetime DEFAULT NULL, `lastactivity` datetime NOT NULL, PRIMARY KEY (`accessTokenId`), KEY `fk_sessions_users_idx` (`user`), CONSTRAINT `fk_sessions_users` FOREIGN KEY (`user`) REFERENCES `users` (`userid`)) ENGINE=InnoDB DEFAULT CHARSET=utf8;</pre>
<p>Table: siteclimates</p> <p>Columns:</p> <pre>climateId smallint(6) AI PK name varchar(128) description longtext createDt datetime createBy int(11) updateDt datetime updateBy int(11)</pre>	<pre>CREATE TABLE `siteclimates` (`climateId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`climateId`)) ENGINE=InnoDB DEFAULT CHARSET=utf8;</pre>
<p>Table: siteenvs</p> <p>Columns:</p> <pre>environmentId smallint(6) AI PK name varchar(128) description longtext createDt datetime createBy int(11) updateDt datetime</pre>	<pre>CREATE TABLE `siteenvs` (`environmentId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL,</pre>

updateBy int(11)	PRIMARY KEY (`environmentId`)) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=utf8 COMMENT='Περιβάλλοντα αρχαιολογική θέσης';
Table: siteexctechs Columns: exctechId smallint(6) AI PK name varchar(128) description longtext createDt datetime createBy int(11) updateDt datetime updateBy int(11)	CREATE TABLE `siteexctechs` (`exctechId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`exctechId`)) ENGINE=InnoDB AUTO_INCREMENT=8 DEFAULT CHARSET=utf8 COMMENT='Τεχνικές ανασκαφής';
Table: siteexctypes Columns: exctypeId smallint(6) AI PK name varchar(128) description longtext createDt datetime createBy int(11) updateDt datetime updateBy int(11)	CREATE TABLE `siteexctypes` (`exctypeId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`exctypeId`)) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8 COMMENT='Τύπος ανασκαφής';
Table: siteforms Columns: formId smallint(6) AI PK name varchar(128) description longtext createDt datetime createBy int(11) updateDt datetime updateBy int(11)	CREATE TABLE `siteforms` (`formId` smallint(6) NOT NULL AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`formId`)) ENGINE=InnoDB AUTO_INCREMENT=10 DEFAULT CHARSET=utf8 COMMENT='Μορφές αρχαιολογική θέσης(site)';
Table: siteinvmethods	CREATE TABLE `siteinvmethods` (`invmethodId` smallint(6) NOT NULL

<p>Columns:</p> <p>invmethodId smallint(6) AI PK name varchar(128) description longtext createDt datetime createBy int(11) updateDt datetime updateBy int(11)</p>	<pre> AUTO_INCREMENT, `name` varchar(128) NOT NULL, `description` longtext, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`invmethodId`)) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8 COMMENT='Μέθοδοι έρευνας αρχαιολογικής θέσης'; </pre>
<p>Table: sites</p> <p>Columns:</p> <p>siteId int(11) AI PK geosite int(11) name varchar(128) summary varchar(1024) description longtext latitude float longitude float mapZoomLevel smallint(6) showOrder int(11) env smallint(6) form smallint(6) exctype smallint(6) exctech smallint(6) invmethod smallint(6) climate smallint(6) isactive tinyint(1) dirName varchar(64) createDt datetime createBy int(11) updateDt datetime updateBy int(11)</p>	<pre> CREATE TABLE `sites` (`siteId` int(11) NOT NULL AUTO_INCREMENT, `geosite` int(11) NOT NULL, `name` varchar(128) NOT NULL, `summary` varchar(1024) DEFAULT NULL, `description` longtext, `latitude` float DEFAULT NULL, `longitude` float DEFAULT NULL, `mapZoomLevel` smallint(6) NOT NULL, `showOrder` int(11) NOT NULL, `env` smallint(6) DEFAULT NULL, `form` smallint(6) DEFAULT NULL, `exctype` smallint(6) DEFAULT NULL, `exctech` smallint(6) DEFAULT NULL, `invmethod` smallint(6) DEFAULT NULL, `climate` smallint(6) DEFAULT NULL, `isactive` tinyint(1) DEFAULT '0', `dirName` varchar(64) NOT NULL, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`siteId`), KEY `fk_site_siteenvs_idx` (`env`), KEY `fk_site_siteforms_idx` (`form`), KEY `fk_site_siteinvmethods_idx` (`invmethod`), KEY `fk_sites_siteexctechs_idx` (`exctech`), KEY `fk_sites_siteexctypes_idx` (`exctype`), KEY `fk_sites_geosites_idx` (`geosite`), KEY `fk_sites_siteclimates_idx` (`climate`), CONSTRAINT `fk_site_siteenvs` FOREIGN KEY </pre>

	<pre>(`env`) REFERENCES `siteenvs` (`environmentId`), CONSTRAINT `fk_site_siteforms` FOREIGN KEY (`form`) REFERENCES `siteforms` (`formId`), CONSTRAINT `fk_site_siteinvmethods` FOREIGN KEY (`invmethod`) REFERENCES `siteinvmethods` (`invmethodId`), CONSTRAINT `fk_sites_geosites` FOREIGN KEY (`geosite`) REFERENCES `geosites` (`geositeId`), CONSTRAINT `fk_sites_siteclimates` FOREIGN KEY (`climate`) REFERENCES `siteclimates` (`climateId`) ON DELETE NO ACTION ON UPDATE NO ACTION, CONSTRAINT `fk_sites_siteexctechs` FOREIGN KEY (`exctech`) REFERENCES `siteexctechs` (`exctechId`), CONSTRAINT `fk_sites_siteexctypes` FOREIGN KEY (`exctype`) REFERENCES `siteexctypes` (`exctypeId`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='Archaeological site';</pre>
<p>Table: stratums</p> <p>Columns:</p> <pre>stratumId int(11) AI PK cont int(11) name varchar(128) description varchar(1024) colour smallint(6) createDt datetime createBy int(11) updateDt datetime updateBy int(11)</pre>	<pre>CREATE TABLE `stratums` (`stratumId` int(11) NOT NULL AUTO_INCREMENT, `cont` int(11) NOT NULL, `name` varchar(128) NOT NULL, `description` varchar(1024) DEFAULT NULL, `colour` smallint(6) NOT NULL, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`stratumId`), KEY `fk_stratums_conts_idx` (`cont`), KEY `fk_stratums_colors_idx` (`colour`), CONSTRAINT `fk_stratums_colors` FOREIGN KEY (`colour`) REFERENCES `stratumcolours` (`colorId`), CONSTRAINT `fk_stratums_conts` FOREIGN KEY (`cont`) REFERENCES `conts` (`contId`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='στρώμα';</pre>
<pre>updateBy int(11)</pre>	<pre>CREATE TABLE `substratums` (`substratumId` int(11) NOT NULL AUTO_INCREMENT,</pre>

	<pre> `stratum` int(11) NOT NULL, `name` varchar(64) NOT NULL, `description` varchar(1024) DEFAULT NULL, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`substratumId`), KEY `fk_substratums_stratums_idx` (`stratum`), CONSTRAINT `fk_substratums_stratums` FOREIGN KEY (`stratum`) REFERENCES `stratums` (`stratumId`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='στρούσεις'; </pre>
<p>Table: tutorials</p> <p>Columns:</p> <pre> tutorialid int(11) AI PK title varchar(256) summary varchar(2048) body longtext isvisible tinyint(4) author varchar(128) sourceName varchar(128) sourceURL varchar(256) url varchar(256) createDt datetime createBy int(11) updateDt datetime updateBy int(11) </pre>	
<p>Table: usergrants</p> <p>Columns:</p> <pre> usergrantid int(11) AI PK user int(11) site int(11) siteRole tinyint(4) phase int(11) </pre>	<pre> CREATE TABLE `usergrants` (`usergrantid` int(11) NOT NULL AUTO_INCREMENT, `user` int(11) NOT NULL, `site` int(11) DEFAULT NULL, `siteRole` tinyint(4) NOT NULL DEFAULT '0', `phase` int(11) DEFAULT NULL, `phaseRole` tinyint(4) NOT NULL DEFAULT '0', `cont` int(11) DEFAULT NULL, </pre>

<pre> phaseRole tinyint(4) cont int(11) contRole tinyint(4) flags int(11) createDt datetime createBy int(11) updateDt datetime updateBy int(11) </pre>	<pre> `contRole` tinyint(4) NOT NULL DEFAULT '0', `flags` int(11) NOT NULL DEFAULT '0', `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`usergrantid`), KEY `fk_usergrants_users_idx` (`user`), KEY `fk_usergrants_sites_idx` (`site`), KEY `fk_usergrants_phases_idx` (`phase`), KEY `fk_usergrants_conts_idx` (`cont`), CONSTRAINT `fk_usergrants_conts` FOREIGN KEY (`cont`) REFERENCES `conts` (`contId`), CONSTRAINT `fk_usergrants_phases` FOREIGN KEY (`phase`) REFERENCES `phases` (`phaseId`), CONSTRAINT `fk_usergrants_sites` FOREIGN KEY (`site`) REFERENCES `sites` (`siteId`), CONSTRAINT `fk_usergrants_users` FOREIGN KEY (`user`) REFERENCES `users` (`userid`)) ENGINE=InnoDB DEFAULT CHARSET=utf8; </pre>
<p>Table: users</p> <p>Columns:</p> <pre> userId int(11) AI PK login varchar(128) pass varchar(128) salt varchar(128) firstname varchar(128) lastname varchar(128) email varchar(128) telephone varchar(128) isactive tinyint(4) islockedout tinyint(4) issysadmin tinyint(4) mngarticles tinyint(4) mngpublications tinyint(4) mngtutorials tinyint(4) mnggalleries tinyint(4) totallogins int(11) lastlogin datetime failedAtmpCount int(11) failedAtmpWindowStart datetime </pre>	<pre> CREATE TABLE `users` (`userId` int(11) NOT NULL AUTO_INCREMENT, `login` varchar(128) NOT NULL, `pass` varchar(128) NOT NULL, `salt` varchar(128) NOT NULL, `firstname` varchar(128) NOT NULL, `lastname` varchar(128) NOT NULL, `email` varchar(128) NOT NULL, `telephone` varchar(128) DEFAULT NULL, `isactive` tinyint(4) NOT NULL DEFAULT '1', `islockedout` tinyint(4) NOT NULL DEFAULT '0', `issysadmin` tinyint(4) NOT NULL DEFAULT '0', `mngarticles` tinyint(4) NOT NULL DEFAULT '0', `mngpublications` tinyint(4) NOT NULL DEFAULT '0', `mngtutorials` tinyint(4) NOT NULL DEFAULT '0', `mnggalleries` tinyint(4) NOT NULL DEFAULT '0', `totallogins` int(11) NOT NULL DEFAULT '0', `lastlogin` datetime DEFAULT NULL, `failedAtmpCount` int(11) NOT NULL DEFAULT '0', `failedAtmpWindowStart` datetime DEFAULT NULL, `lastLockoutDate` datetime DEFAULT NULL, </pre>

lastLockoutDate datetime avatar varchar(64) dirName varchar(64) createDt datetime createBy int(11) updateDt datetime updateBy int(11)	`avatar` varchar(64) DEFAULT NULL, `dirName` varchar(64) NOT NULL, `createDt` datetime NOT NULL, `createBy` int(11) NOT NULL, `updateDt` datetime NOT NULL, `updateBy` int(11) NOT NULL, PRIMARY KEY (`userId`)) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8 COMMENT='Χρήστες';
---	--

Παράρτημα 3: Υλοποίηση της κλάσης DigitSystem

Στο παρόν παράρτημα επισυνάπτεται η υλοποίηση της κλάσης *DigitSystem*. Η *DigitSystem* περιέχει το public API του Digit CMS. Οποιος θέλει να χρησιμοποιήσει το Digit CMS, για τη δημιουργία κάποιου τρίτου εργαλείου, το μόνο που χρειάζεται είναι να κάνει reference το Digit.Core.dll και να δημιουργήσει ένα στιγμιότυπο (instance) της κλάσης *DigitSystem*. Η *DigitSystem* βρίσκεται στο επίπεδο εφαρμογής (Business Layer) του συστήματός και «βλέπει» την mysql μέσω της κλάσεως *DigitDataLayer*, που είναι wrapper όλων των queries προς τη mysql.

Οι μέθοδοι που ασχολούνται με κάθε οντότητα του συστήματος (*DgUser*, *DgCountry*, *DgSiteClimate*, *DgSiteEnv* κ.ά.) βρίσκονται ομαδοποιημένες και συνήθως υπάρχουν για κάθε τέτοια οντότητα τουλάχιστον οι βασικές CRUD διεργασίες.

Για παράδειγμα, για την *DgSiteClimate* θα βρούμε για διάβασμα τις εξής μεθόδους:

```
public DgSiteClimate GetSiteClimateById(DgAccessToken  
accessToken, ....)
```

```
public IList<DgSiteClimate> GetSiteClimates(DgAccessToken  
accessToken, .....)
```

```
public IList<DgSiteClimate> GetSiteClimates(DgAccessToken  
accessToken, int pageIndex, ....)
```

Για δημιουργία:

```
public DgSiteClimate CreateSiteClimate(DgAccessToken accessToken,  
...)
```

Για μεταβολή:

```
public DgSiteClimate UpdateSiteClimate(DgAccessToken accessToken,  
...)
```

Για διαγραφή:

```
public void DeleteSiteClimate(DgAccessToken accessToken, ...)
```

```

1. using Digit.Core;
2. using Digit.Core.Builtins;
3. using Digit.Core.DLL;
4. using Digit.Core.Entities;
5. using System;
6. using System.Collections.Generic;
7. using System.Configuration;
8. using System.Globalization;
9. using System.IO;
10. using System.Net;
11. using System.Xml.Linq;
12.
13. namespace Digit
14. {
15.     /// <summary>
16.     /// Here we will find the API for our cms system
17.     /// </summary>
18.     public sealed class DigitSystem
19.     {
20.         #region support stuff
21.         DigitDataLayer m_digitDLL;
22.         internal DigitDataLayer DigitDLL
23.         {
24.             get
25.             {
26.                 if (m_digitDLL == null)
27.                 {
28.                     m_digitDLL = new DigitDataLayer();
29.                 }
30.                 return m_digitDLL;
31.             }
32.         }
33.
34.         #endregion
35.
36.         static public Decimal Default_Latitude
37.         {
38.             get
39.             {
40.                 return 39.24530374534449m;
41.             }
42.         }
43.         static public Decimal Default_Longitude
44.         {
45.             get
46.             {
47.                 return 21.918928467075034m;
48.             }
49.         }
50.         static public Decimal Default_MapZoomLevel
51.         {
52.             get
53.             {
54.                 return 7;
55.             }
56.         }
57.
58.         #region Login stuff
59.         /// <summary>

```

```

60.         /// Κάνει Login τον χρήστη στο σύστημα μας. Εάν το login είναι επιτυχημένο επιστρέφει πίσω
        ένα έγκυρο DgAccessToken
61.         /// </summary>
62.         /// <param name="logOnToken"></param>
63.         /// <param name="pswdToken"></param>
64.         /// <param name="ipaddress"></param>
65.         /// <returns></returns>
66.         public DgAccessToken LogOnUser(string logOnToken, string pswdToken, string ipaddress = nul
1)
67.         {
68.             if (Utility.ValidateParameter(ref logOnToken, true, true, true, 72) == false)
69.                 return null;
70.             if (Utility.ValidateParameter(ref pswdToken, true, true, false, 128) == false)
71.                 return null;
72.             if (Utility.ValidateParameter(ref ipaddress, false, false, true, 50) == false)
73.                 return null;
74.
75.             var userPassword = GetPasswordWithFormat(logOnToken, pswdToken);
76.             if (userPassword == null)
77.                 return null;
78.
79.             DateTime dtNow = Utility.Now();
80.             var accessToken = DigitDLL.OpenAccessToken(userPassword.UserId, ipaddress, dtNow);
81.             if (accessToken == null)
82.                 return null;
83.
84.             return accessToken;
85.         }
86.         /// <summary>
87.         /// Επιβεβαιώνει την εγκυρότητα ενός accessTokenId
88.         /// </summary>
89.         /// <param name="accessTokenId"></param>
90.         /// <param name="ipaddress"></param>
91.         /// <returns></returns>
92.         public DgAccessToken ValidateAccessToken(Guid accessTokenId, string ipaddress = null)
93.         {
94.             if (Utility.ValidateParameter(ref ipaddress, false, false, true, 50) == false)
95.                 return null;
96.
97.             DateTime dtNow = Utility.RoundToSeconds(DateTime.UtcNow);
98.             var accessToken = DigitDLL.ValidateAccessToken(accessTokenId, dtNow);
99.
100.            return accessToken;
101.        }
102.        /// <summary>
103.        /// Τερματίζει τη συνεδρία ενός χρήστη απο το σύστημά μας
104.        /// </summary>
105.        /// <param name="accessToken"></param>
106.        public void LogOffUser(DgAccessToken accessToken)
107.        {
108.            if (accessToken == null) throw new ArgumentNullException("accessToken");
109.
110.            LogOffUser(accessToken.AccessTokenId);
111.        }
112.        /// <summary>
113.        /// Τερματίζει τη συνεδρία ενός χρήστη απο το σύστημά μας
114.        /// </summary>
115.        /// <param name="accessTokenId"></param>
116.        public void LogOffUser(Guid accessTokenId)
117.        {
118.            DateTime dtNow = Utility.Now();
119.
120.            DigitDLL.CloseAccessToken(accessTokenId, dtNow);

```

```

121.     }
122.
123.     /// <summary>
124.     ///
125.     /// </summary>
126.     /// <param name="logOnToken"></param>
127.     /// <param name="password"></param>
128.     /// <returns></returns>
129.     DgUserPassword GetPasswordWithFormat(string logOnToken, string password)
130.     {
131.         DateTime dtNow = Utility.Now();
132.         var userPassword = DigitDLL.GetPasswordWithFormat(logOnToken, dtNow);
133.         if (userPassword == null)
134.             return null;
135.
136.
137.         if (userPassword.IsLockedOut)
138.             return null;
139.         if (!userPassword.IsActive)
140.             return null;
141.
142.         string encodedPasswd = Utility.EncodePassword(password, userPassword.PswdSalt);
143.
144.         bool isPasswordCorrect = userPassword.PswdFromDB.Equals(encodedPasswd);
145.
146.
147.         if (isPasswordCorrect && userPassword.FailedPasswordAttemptCount == 0)
148.             return userPassword;
149.
150.         DigitDLL.UpdateUserInfo(
151.             logOnToken,
152.             isPasswordCorrect,
153.             5/*μπορεί να βάλει λάθος κωδικό στη σειρά πέντε φορές*/,
154.             120/*120 δευτερόλεπτα λάθος προσπαθειών και κλειδώνουμε*/,
155.             dtNow,
156.             userPassword.FailedPasswordAttemptCount,
157.             userPassword.FailedPasswordAttemptWindowStart
158.         );
159.
160.
161.         if (isPasswordCorrect)
162.             return userPassword;
163.
164.         return null;
165.     }
166.     /// <summary>
167.     ///
168.     /// </summary>
169.     /// <param name="pswdToken"></param>
170.     void CheckPassword(ref string pswdToken)
171.     {
172.         /*Retrieve required SecurityPolicy values:*/
173.         int MinRequiredPasswordLength = 8;
174.         int MinRequiredNonAlphanumericCharacters = 2;
175.
176.
177.         if (!Utility.ValidateParameter(ref pswdToken, true, true, false, 128))
178.         {
179.             throw new DgException(SR.GetString(SR.Password_is_invalid));
180.         }
181.         if (pswdToken.Length < MinRequiredPasswordLength)
182.         {

```

```

183.         throw new DgException(SR.GetString(SR.Password_is_too_small, MinRequiredPasswordLe
ngth));
184.     }
185.     int count = 0;
186.     for (int i = 0; i < pswdToken.Length; i++)
187.     {
188.         if (!char.IsLetterOrDigit(pswdToken, i))
189.         {
190.             count++;
191.         }
192.     }
193.     if (count < MinRequiredNonAlphanumericCharacters)
194.     {
195.         throw new DgException(SR.GetString(SR.Password_MinRequiredNonAlphanumericCharacter
s, MinRequiredNonAlphanumericCharacters.ToString(CultureInfo.InvariantCulture)));
196.     }
197. }
198. #endregion
199.
200.
201.     /// <summary>
202.     ///
203.     /// </summary>
204.     /// <param name="accessToken"></param>
205.     /// <param name="table_name"></param>
206.     /// <returns></returns>
207.     public bool ExistsEntityName(DgAccessToken accessToken, string table_name)
208.     {
209.         if (string.IsNullOrEmpty(table_name)) throw new ArgumentNullException("table_name
");
210.         table_name = table_name.ToLowerInvariant();
211.
212.         return DigitDLL.ExistsEntityName(accessToken, table_name);
213.     }
214.
215.     #region Users
216.     /// <summary>
217.     ///
218.     /// </summary>
219.     /// <param name="accessToken">Αντιπροσωπεύει τον χρήστη (authorization context) που καλεί
αυτή τη μέθοδο.</param>
220.     /// <param name="whereClause"></param>
221.     /// <param name="orderByClause"></param>
222.     /// <returns></returns>
223.     public IList<DgUser> GetUsers(DgAccessToken accessToken, string whereClause = null, string
orderByClause = "order by lastname,firstname")
224.     {
225.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
226.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
227.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClause");
228.
229.         #region SecurityLayer
230.
231.         #endregion
232.
233.         return DigitDLL.GetUsers(accessToken, whereClause, orderByClause);
234.     }
235.     /// <summary>
236.     ///
237.     /// </summary>
238.     /// <param name="accessToken">Αντιπροσωπεύει τον χρήστη (authorization context) που καλεί
αυτή τη μέθοδο.</param>

```

```

239.     /// <param name="pageIndex">0 αριθμός της σελίδας την οποία ζητάμε απο το σύστημα. Η αριθμ
ηση αρχίζει απο το ένα (1).</param>
240.     /// <param name="pageSize">Το πλήθος των εγγραφών που θα επιστρέφει σε κάθε σελίδα το σύστ
ημα μας</param>
241.     /// <param name="totalRows">Εδώ το σύστημα θα βάλει το συνολικό πλήθος των εγγραφών που ικ
ανοποιούν την κλήση μας</param>
242.     /// <param name="whereClause">Partial sql with the custom where clause. It must begin with
'where'</param>
243.     /// <param name="orderByClause">Partial sql with the custom ordr by clause. It must begin
with 'order by'</param>
244.     /// <returns></returns>
245.     public IList<DgUser> GetUsers(DgAccessToken accessToken, int pageIndex, int pageSize, ref
int totalRows, string whereClause = null, string orderByClause = "order by lastname,firstname")
246.     {
247.         if (accessToken == null) throw new ArgumentNullException("accessToken");
248.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
249.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClause");

250.
251.         #region SecurityLayer
252.
253.         #endregion
254.
255.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
256.         int startRowIndex = ((pageIndex - 1) * pageSize);
257.
258.         return DigitDLL.GetUsers(accessToken, startRowIndex, pageSize, ref totalRows, whereCla
use, orderByClause);
259.     }
260.     /// <summary>
261.     ///
262.     /// </summary>
263.     /// <param name="accessToken">Αντιπροσωπεύει τον χρήστη (authorization context) που καλεί
αυτή τη μέθοδο.</param>
264.     /// <param name="userId"></param>
265.     /// <returns></returns>
266.     public DgUser GetUserById(DgAccessToken accessToken, Int32 userId)
267.     {
268.         #region input parameters validation
269.         if (accessToken == null) throw new ArgumentNullException("accessToken");
270.         #endregion
271.
272.         #region SecurityLayer
273.         #endregion
274.
275.         return DigitDLL.GetUserById(accessToken, userId);
276.     }
277.     /// <summary>
278.     ///
279.     /// </summary>
280.     /// <param name="accessToken">Αντιπροσωπεύει τον χρήστη (authorization context) που καλεί
αυτή τη μέθοδο.</param>
281.     /// <param name="email"></param>
282.     /// <returns></returns>
283.     public DgUser GetUserByEmail(DgAccessToken accessToken, string email)
284.     {
285.         #region input parameters validation
286.         if (accessToken == null) throw new ArgumentNullException("accessToken");
287.         #endregion
288.         Utility.CheckParameter(ref email, true, true, true, 256, "email");
289.
290.         #region SecurityLayer
291.         #endregion

```

```

292.
293.         email = email.Replace("'", "").ToLowerInvariant();
294.
295.         return DigitDLL.GetUserByEmail(accessToken, email);
296.     }
297.     /// <summary>
298.     ///
299.     /// </summary>
300.     /// <param name="accessToken">Αντιπροσωπεύει τον χρήστη (authorization context) που καλεί
    αυτή τη μέθοδο.</param>
301.     /// <param name="accessTokenId"></param>
302.     /// <returns></returns>
303.     public DgUser GetUserFromAccessToken(DgAccessToken accessToken, Guid accessTokenId)
304.     {
305.         #region input parameters validation
306.         if (accessToken == null) throw new ArgumentNullException("accessToken");
307.         #endregion
308.
309.         #region SecurityLayer
310.         #endregion
311.
312.         return DigitDLL.GetUserByFromAccessToken(accessToken, accessTokenId);
313.     }
314.     /// <summary>
315.     /// Ξεκλειδώνει έναν χρήστη
316.     /// </summary>
317.     /// <param name="accessToken">Αντιπροσωπεύει τον χρήστη (authorization context) που καλεί
    αυτή τη μέθοδο.</param>
318.     /// <param name="userId"></param>
319.     public void UnlockUser(DgAccessToken accessToken, Int32 userId)
320.     {
321.         if (accessToken == null) throw new ArgumentNullException("accessstoken");
322.
323.         #region SecurityLayer
324.         if (accessToken.IsSysAdmin == false)
325.         {
326.             throw new DgException("Only a SysAdmin can do that!");
327.         }
328.         #endregion
329.
330.         DigitDLL.UnlockUser(accessToken, userId);
331.     }
332.     /// <summary>
333.     ///
334.     /// </summary>
335.     /// <param name="accessToken">Αντιπροσωπεύει τον χρήστη (authorization context) που καλεί
    αυτή τη μέθοδο.</param>
336.     /// <param name="userId"></param>
337.     /// <param name="pswdToken"></param>
338.     /// <returns></returns>
339.     public DgUser ChangePassword(DgAccessToken accessToken, Int32 userId, string pswdToken)
340.     {
341.         #region input parameters validation
342.         if (accessToken == null) throw new ArgumentNullException("accessToken");
343.         Utility.CheckParameter(ref pswdToken, true, true, false, 64, "pswdToken");
344.         #endregion
345.
346.         #region SecurityLayer
347.         if (accessToken.IsSysAdmin == false)
348.         {
349.             throw new DgException("Only a SysAdmin can do that!");
350.         }
351.         #endregion

```

```

352.
353.         var existingUser = DigitDLL.GetUserById(accessToken, userId);
354.         if (existingUser == null) throw new DgException(SR.GetString(SR.There_is_no_item_with_
id, "User", userId));
355.
356.
357.         /*Ελέγχουμε το password με βάση system-wide τιμές του Security-Policy:*/
358.         CheckPassword(ref pswdToken);
359.
360.         string salt = Utility.GenerateSalt();
361.         string pass = Utility.EncodePassword(pswdToken, salt);
362.         if (pass.Length > 256)
363.         {
364.             throw new DgException(SR.GetString(SR.Password_is_invalid));
365.         }
366.
367.
368.         existingUser.Pass = pass;
369.         existingUser.Salt = salt;
370.         return DigitDLL.UpdateUser(accessToken, existingUser);
371.     }
372.
373.     /// <summary>
374.     ///
375.     /// </summary>
376.     /// <param name="accessToken">Αντιπροσωπεύει τον χρήστη (authorization context) που καλεί
αυτή τη μέθοδο.</param>
377.     /// <param name="firstName"></param>
378.     /// <param name="lastName"></param>
379.     /// <param name="email"></param>
380.     /// <param name="logOnToken"></param>
381.     /// <param name="pswdToken"></param>
382.     /// <param name="isSysadmin"></param>
383.     /// <param name="mngarticles"></param>
384.     /// <param name="mnggalleries"></param>
385.     /// <param name="mngpublications"></param>
386.     /// <param name="mngtutorials"></param>
387.     /// <returns></returns>
388.     public DgUser CreateUser(DgAccessToken accessToken, string firstName, string lastName, str
ing email, string logOnToken, string pswdToken, bool isSysadmin = false, bool mngarticles = false,
bool mnggalleries = false, bool mngpublications = false, bool mngtutorials = false)
389.     {
390.         #region input parameters validation
391.         if (accessToken == null) throw new ArgumentNullException("accessToken");
392.         #endregion
393.
394.         var user = new DgUser();
395.         user.Firstname = firstName;
396.         user.Lastname = lastName;
397.         user.Email = email;
398.         user.Issysadmin = isSysadmin;
399.         user.Mngarticles = mngarticles;
400.         user.Mnggalleries = mnggalleries;
401.         user.Mngpublications = mngpublications;
402.         user.Mngtutorials = mngtutorials;
403.
404.         return _CreateUser(accessToken, user, logOnToken, pswdToken);
405.     }
406.     /// <summary>
407.     ///
408.     /// </summary>
409.     /// <param name="accessToken"></param>
410.     /// <param name="user"></param>

```



```

411.     /// <param name="logOnToken"></param>
412.     /// <param name="pswdToken"></param>
413.     /// <returns></returns>
414.     DgUser _CreateUser(DgAccessToken accessToken, DgUser user, string logOnToken, string pswdToken)
415.     {
416.         #region input parameters validation
417.         if (accessToken == null) throw new ArgumentNullException("accessToken");
418.         if (user == null) throw new ArgumentNullException("user");
419.
420.         user.ValidateInstance();
421.
422.         Utility.CheckParameter(ref logOnToken, true, true, false, 64, "logOnToken");
423.         Utility.CheckParameter(ref pswdToken, true, true, false, 64, "pswdToken");
424.         #endregion
425.
426.         #region SecurityLayer
427.         if (accessToken.IsSysAdmin == false)
428.         {
429.             throw new DgException("Only a SysAdmin can do that!");
430.         }
431.         #endregion
432.
433.
434.         #region Ελεγχος του email του χρήστη μας:
435.         if (string.IsNullOrEmpty(user.Email))
436.         {
437.             throw new DgException("Users are required to have an email address!");
438.         }
439.         user.Email = user.Email.ToLowerInvariant();
440.         if (!Utility.EmailIsValid(user.Email))
441.         {
442.             throw new DgException(string.Format("Email address '{0}' is in use!", user.Email));
443.         }
444.
445.         if (DigitDLL.GetUserByEmail(accessToken, user.Email) != null)
446.         {
447.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Email", user.Email));
448.         }
449.         #endregion
450.
451.         #region credentials
452.         /*Ελεγχος του logOnToken του χρήστη μας:*/
453.         if (string.IsNullOrEmpty(logOnToken))
454.         {
455.             throw new DgException("Users are required to have a username!");
456.         }
457.         logOnToken = logOnToken.Trim().ToLowerInvariant();
458.         if (logOnToken.Length < 6)
459.         {
460.             throw new DgException("Username must be greater than 6 characters");
461.         }
462.         if (DigitDLL.GetUserByLogin(accessToken, logOnToken) != null)
463.         {
464.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Username", logOnToken));
465.         }
466.         /*Ελέγχουμε το password με βάση system-wide τιμές του Security-Policy:*/
467.         CheckPassword(ref pswdToken);
468.
469.         string salt = Utility.GenerateSalt();

```

```

470.         string pass = Utility.EncodePassword(pswdToken, salt);
471.         if (pass.Length > 256)
472.         {
473.             throw new DgException(SR.GetString(SR.Password_is_invalid));
474.         }
475.
476.         user.Login = logOnToken;
477.         user.Pass = pass;
478.         user.Salt = salt;
479.         #endregion
480.
481.         user.Isactive = true;
482.         user.Islockedout = false;
483.         user.Totallogins = 0;
484.         user.Lastlogin = null;
485.         user.DirName = System.IO.Path.GetRandomFileName();
486.         user.CreateBy = user.UpdateBy = accessToken.UserId;
487.         user.CreateDt = user.UpdateDt = Utility.Now();
488.
489.         return DigitDLL.CreateUser(accessToken, user);
490.     }
491.     /// <summary>
492.     ///
493.     /// </summary>
494.     /// <param name="accessToken">Αντιπροσωπεύει τον χρήστη (authorization context) που καλεί
495.     αυτή τη μέθοδο.</param>
496.     /// <param name="user"></param>
497.     /// <returns></returns>
498.     public DgUser UpdateUser(DgAccessToken accessToken, DgUser user)
499.     {
500.         #region input parameters validation
501.         if (user == null) throw new ArgumentNullException("user");
502.         user.ValidateInstance();
503.         #endregion
504.
505.         #region SecurityLayer
506.         if (accessToken.IsSysAdmin == false)
507.         {
508.             throw new DgException("Only a SysAdmin can do that!");
509.         }
510.         #endregion
511.
512.         var existingUser = DigitDLL.GetUserById(accessToken, user.UserId);
513.         if (existingUser == null) throw new DgException(SR.GetString(SR.There_is_no_item_with_
514. id, "User", user.UserId));
515.         if (BuiltinUsers.IsBuiltin(existingUser))
516.         {
517.             throw new DgException(SR.GetString(SR.You_cannot_update_the_builtin_entity, "SystemUser", string.Format(CultureInfo.InvariantCulture, "{0}, {1}", user.Lastname, user.Firstname)));
518.         }
519.
520.         #region Έλεγχος του email του χρήστη μας:
521.         if (string.IsNullOrEmpty(user.Email))
522.         {
523.             throw new DgException("Users are required to have an email address!");
524.         }
525.         user.Email = user.Email.ToLowerInvariant();
526.         if (!Utility.EmailIsValid(user.Email))
527.         {

```

```

528.         throw new DgException(string.Format("Email address '{0}' is in use!", user.Email))
529.     }
530.
531.     var _user = DigitDLL.GetUserByEmail(accessToken, user.Email);
532.     if (_user != null && _user.UserId != user.UserId)
533.     {
534.         throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Email", user.Email
535.     ));
536.     }
537.
538.
539.     #region Ελεγχος του logOnToken του χρήστη μας:
540.     if (string.IsNullOrEmpty(user.Login))
541.     {
542.         throw new DgException("Users are required to have a username!");
543.     }
544.     user.Login = user.Login.Trim().ToLowerInvariant();
545.     if (user.Login.Length < 6)
546.     {
547.         throw new DgException("Username must be greater than 6 characters");
548.     }
549.     _user = DigitDLL.GetUserByLogin(accessToken, user.Login);
550.     if (_user != null && _user.UserId != user.UserId)
551.     {
552.         throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Username", user.Lo
553.     gin));
554.     }
555.     #endregion
556.
557.     existingUser.Login = user.Login;
558.     existingUser.Firstname = user.Firstname;
559.     existingUser.Lastname = user.Lastname;
560.     existingUser.Email = user.Email;
561.     existingUser.Telephone = user.Telephone;
562.     existingUser.Isactive = user.Isactive;
563.     existingUser.Issysadmin = user.Issysadmin;
564.     existingUser.Mngarticles = user.Mngarticles;
565.     existingUser.Mngpublications = user.Mngpublications;
566.     existingUser.Mngtutorials = user.Mngtutorials;
567.     existingUser.Mnggalleries = user.Mnggalleries;
568.     existingUser.Avatar = user.Avatar;
569.
570.     existingUser.UpdateBy = accessToken.UserId;
571.     existingUser.UpdateDt = Utility.Now();
572.
573.     return DigitDLL.UpdateUser(accessToken, existingUser);
574. }
575. /// <summary>
576. ///
577. /// </summary>
578. /// <param name="accessToken">Αντιπροσωπεύει τον χρήστη (authorization context) που καλεί
579. αυτή την μέθοδο.</param>
580. /// <param name="m_userId"></param>
581. public void DeleteUser(DgAccessToken accessToken, Int32 userId)
582. {
583.     /*
584.     * We have to exclude the case that a user is trying to delete itself.
585.     * It is not allowed
586.     */
587.     if (accessToken.UserId == userId)
588.     {

```

```

587.         throw new DgException("A user cannot delete itself!");
588.     }
589.
590.     var user = DigitDLL.GetUserById(accessToken, userId);
591.     if (user == null) throw new Exception(SR.GetString(SR.There_is_no_item_with_id, "user"
, userId));
592.
593.
594.     #region SecurityLayer
595.     if (accessToken.IsSysAdmin == false)
596.     {
597.         throw new DgException("Only a SysAdmin can do that!");
598.     }
599.     #endregion
600.
601.     if (BuiltinUsers.IsBuiltin(user))
602.     {
603.         throw new DgException(SR.GetString(SR.You_cannot_delete_the_builtin_entity, "user"
, string.Format(CultureInfo.InvariantCulture, "{0}, {1}", user.Lastname, user.Firstname)));
604.     }
605.
606.
607.     /*Δεν μπορούμε να διαγράψουμε έναν χρήστη εάν είναι supervisor σε πλαίσιο:*/
608.     var conts = DigitDLL.IsUserContSupervisor(accessToken, user.UserId);
609.     if (conts > 0)
610.     {
611.         if(conts == 1)
612.         {
613.             throw new DgException("You cannot delete the user. He is supervisor for one co
nt!");
614.         }
615.         else
616.         {
617.             throw new DgException("You cannot delete the user. He is supervisor for more t
han one conts!");
618.         }
619.     }
620.
621.
622.
623.     DigitDLL.DeleteUser(accessToken, user.UserId);
624. }
625. #endregion
626.
627. public DgvUserGrant GetUserGrantById_View(DgAccessToken accessToken, Int32 usergrantid)
628. {
629.     if (accessToken == null) throw new ArgumentNullException("accesstoken");
630.
631.     #region SecurityLayer
632.     #endregion
633.
634.     return DigitDLL.GetUserGrantById_View(accessToken, usergrantid);
635. }
636.
637. #region DgFile
638.
639.     /// <summary>
640.     /// Αυτό είναι το Root directory, μέσα στο οποίο η digit αποθηκεύει τα αρχεία περιεχομένου
641.
642.     /// </summary>
643.     private string RootDirectory
644.     {
        get

```

```

645.         {
646.             return digitConfiguration.USERFILESPATH;
647.         }
648.     }
649.
650.     string _GetFilePath(DgFile file)
651.     {
652.         return Path.Combine(Path.Combine(RootDirectory, file.Filepath), file.Mngfilename);
653.     }
654.     DgFile _UploadFile(DgAccessToken accessToken, DgFile digitFile, System.Byte[] buffer)
655.     {
656.         string inventoryPath = Path.Combine(RootDirectory, digitFile.Filepath);
657.         //Εάν δεν υπάρχει το directory το δημιουργούμε
658.         if (!Directory.Exists(inventoryPath))
659.         {
660.             Directory.CreateDirectory(inventoryPath);
661.         }
662.
663.
664.
665.         #region Πρέπει να σιγουρευτούμε ότι το managed όνομα είναι μοναδικό!
666.         int counter = 1;
667.         digitFile.Mngfilename = digitFile.Orgfilename;
668.         FileInfo file = new FileInfo(_GetFilePath(digitFile));
669.         while (file.Exists)
670.         {
671.             digitFile.Mngfilename = Path.GetFileNameWithoutExtension(digitFile.Orgfilename) +
672.             "_" + counter.ToString(CultureInfo.InvariantCulture) + Path.GetExtension(digitFile.Orgfilename);
673.             counter++;
674.             file = new FileInfo(_GetFilePath(digitFile));
675.         }
676.         #endregion
677.         /*
678.          * 0 τόπος με την οποία καλούμε την FileStream, διαγράφει προυπάρχον αρχείο με το ίδιο
679.          όνομα!
680.          */
681.         using (FileStream fstream = new FileStream(_GetFilePath(digitFile), FileMode.Create))
682.         {
683.             using (BinaryWriter bw = new BinaryWriter(fstream))
684.             {
685.                 bw.Write(buffer);
686.             }
687.
688.             digitFile.CreateBy = digitFile.UpdateBy = accessToken.UserId;
689.             digitFile.CreateDt = digitFile.UpdateDt = Utility.Now();
690.
691.             return DigitDLL.CreateFile(accessToken, digitFile);
692.         }
693.         void _DeleteFile(DgAccessToken accessToken, DgFile digitFile)
694.         {
695.             FileInfo file = new FileInfo(_GetFilePath(digitFile));
696.             if (file.Exists)
697.             {
698.                 file.Delete();
699.             }
700.
701.             DigitDLL.DeleteFile(accessToken, digitFile.FileId);
702.         }
703.
704.

```

```

705.     public DgFile GetFileById(DgAccessToken accessToken, Int32 fileId)
706.     {
707.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
708.
709.         #region SecurityLayer
710.         #endregion
711.
712.         return DigitDLL.GetFileById(accessToken, fileId);
713.     }
714.
715.     public IList<DgFile> GetFiles(DgAccessToken accessToken, string whereClause = null, string
orderByClause = "order by Name")
716.     {
717.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
718.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
719.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClause");
720.
721.         #region SecurityLayer
722.         #endregion
723.
724.         return DigitDLL.GetFiles(accessToken, whereClause, orderByClause);
725.     }
726.
727.
728.     public IList<DgFile> GetFiles(DgAccessToken accessToken, int pageIndex, int pageSize, ref
int totalRows, string whereClause = null, string orderByClause = "order by Name")
729.     {
730.         if (accessToken == null) throw new ArgumentNullException("accessToken");
731.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
732.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClause");
733.
734.         #region SecurityLayer
735.         #endregion
736.
737.
738.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
739.         int startRowIndex = ((pageIndex - 1) * pageSize);
740.
741.         return DigitDLL.GetFiles(accessToken, startRowIndex, pageSize, ref totalRows, whereCla
use, orderByClause);
742.     }
743.
744.     public DgFile UpdateFile(DgAccessToken accessToken, DgFile file)
745.     {
746.         #region input parameters validation
747.         if (accessToken == null) throw new ArgumentNullException("accessToken");
748.         if (file == null) throw new ArgumentNullException("file");
749.         file.ValidateInstance();
750.         #endregion
751.
752.         #region SecurityLayer
753.
754.         #endregion
755.
756.         throw new NotImplementedException();
757.     }
758.
759.     #endregion
760.
761.     #region DgCountry
762.     public DgCountry GetCountryById(DgAccessToken accessToken, Int32 countryId)

```

```

763.     {
764.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
765.
766.         #region SecurityLayer
767.         #endregion
768.
769.         return DigitDLL.GetCountryById(accessToken, countryId);
770.     }
771.
772.     public IList<DgCountry> GetCountries(DgAccessToken accessToken, string whereClause, string
orderByClause)
773.     {
774.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
775.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
776.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClause");
777.
778.         #region SecurityLayer
779.         #endregion
780.
781.         return DigitDLL.GetCountries(accessToken, whereClause, orderByClause);
782.     }
783.
784.
785.     public IList<DgCountry> GetCountries(DgAccessToken accessToken, int pageIndex, int pageSiz
e, ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
786.     {
787.         if (accessToken == null) throw new ArgumentNullException("accessToken");
788.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
789.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClause");
790.
791.         #region SecurityLayer
792.
793.         #endregion
794.
795.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
796.         int startRowIndex = ((pageIndex - 1) * pageSize);
797.
798.         return DigitDLL.GetCountries(accessToken, startRowIndex, pageSize, ref totalRows, wher
eClause, orderByClause);
799.     }
800.
801.     public DgCountry CreateCountry(DgAccessToken accessToken, DgCountry country)
802.     {
803.         #region input parameters validation
804.         if (accessToken == null) throw new ArgumentNullException("accessToken");
805.         if (country == null) throw new ArgumentNullException("country");
806.         country.ValidateInstance();
807.         #endregion
808.
809.         #region SecurityLayer
810.         if (accessToken.IsSysAdmin == false)
811.         {
812.             throw new DgException("Only a SysAdmin can do that!");
813.         }
814.         #endregion
815.
816.         /*Check Name uniqueness*/
817.         var existingRecord = DigitDLL.GetCountryByLongName(accessToken, country.LongName);
818.         if (existingRecord != null)
819.         {

```

```

820.         throw new DgException(SR.GetString(SR.Value_is_already_in_use, "LongName", country
      .LongName));
821.     }
822.
823.     country.CreateBy = country.UpdateBy = accessToken.UserId;
824.     country.CreateDt = country.UpdateDt = Utility.Now();
825.
826.     return DigitDLL.CreateCountry(accessToken, country);
827. }
828.
829. public DgCountry UpdateCountry(DgAccessToken accessToken, DgCountry country)
830. {
831.     #region input parameters validation
832.     if (accessToken == null) throw new ArgumentNullException("accessToken");
833.     if (country == null) throw new ArgumentNullException("country");
834.     country.ValidateInstance();
835.     #endregion
836.
837.     #region SecurityLayer
838.     if (accessToken.IsSysAdmin == false)
839.     {
840.         throw new DgException("Only a SysAdmin can do that!");
841.     }
842.     #endregion
843.
844.
845.     var existingRecord = DigitDLL.GetCountryByLongName(accessToken, country.LongName);
846.     if (existingRecord != null && existingRecord.CountryId != country.CountryId)
847.     {
848.         throw new DgException(SR.GetString(SR.Value_is_already_in_use, "LongName", country
      .LongName));
849.     }
850.     if (existingRecord == null)
851.     {
852.         existingRecord = DigitDLL.GetCountryById(accessToken, country.CountryId);
853.     }
854.     if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_item_wit
      h_id, "country", country.CountryId));
855.
856.
857.     existingRecord.CountryId = country.CountryId;
858.     existingRecord.LongName = country.LongName;
859.     existingRecord.ShortName = country.ShortName;
860.
861.     existingRecord.UpdateBy = accessToken.UserId;
862.     existingRecord.UpdateDt = Utility.Now();
863.
864.     return DigitDLL.UpdateCountry(accessToken, existingRecord);
865. }
866.
867. public void DeleteCountry(DgAccessToken accessToken, Int32 countryId)
868. {
869.     if (accessToken == null) throw new ArgumentNullException("accessToken");
870.     var existingRecord = DigitDLL.GetCountryById(accessToken, countryId);
871.     if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_item_wit
      h_id, "Country", countryId));
872.
873.     #region SecurityLayer
874.
875.     #endregion
876.
877.     //Can be deleted?
878.     DigitDLL.CanDeleteCountry(accessToken, existingRecord.CountryId);

```



```

879.
880.
881.     DigitDLL.DeleteCountry(accessToken, existingRecord.CountryId);
882.     }
883.     #endregion
884.
885.
886.     #region SiteEnv
887.     public DgSiteClimate GetSiteClimateById(DgAccessToken accessToken, Int16 climateId)
888.     {
889.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
890.
891.         #region SecurityLayer
892.         #endregion
893.
894.         return DigitDLL.GetSiteClimateById(accessToken, climateId);
895.     }
896.
897.     public IList<DgSiteClimate> GetSiteClimates(DgAccessToken accessToken, string whereClause
= null, string orderByClause = "order by name")
898.     {
899.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
900.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
901.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClause");
902.
903.         #region SecurityLayer
904.         #endregion
905.
906.         return DigitDLL.GetSiteClimates(accessToken, whereClause, orderByClause);
907.     }
908.
909.
910.     public IList<DgSiteClimate> GetSiteClimates(DgAccessToken accessToken, int pageIndex, int
pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
911.     {
912.         if (accessToken == null) throw new ArgumentNullException("accessToken");
913.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
914.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClause");
915.
916.         #region SecurityLayer
917.         #endregion
918.
919.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
920.         int startRowIndex = ((pageIndex - 1) * pageSize);
921.
922.         return DigitDLL.GetSiteClimates(accessToken, startRowIndex, pageSize, ref totalRows, w
hereClause, orderByClause);
923.     }
924.
925.
926.     public DgSiteClimate CreateSiteClimate(DgAccessToken accessToken, DgSiteClimate siteClimat
e)
927.     {
928.         #region input parameters validation
929.         if (accessToken == null) throw new ArgumentNullException("accessToken");
930.         if (siteClimate == null) throw new ArgumentNullException("siteClimate");
931.         siteClimate.ValidateInstance();
932.         #endregion
933.
934.         #region SecurityLayer
935.         if (accessToken.IsSysAdmin == false)

```

```

936.     {
937.         throw new DgException("Only a SysAdmin can do that!");
938.     }
939.     #endregion
940.
941.
942.     /*Check Name uniqueness*/
943.     var existingRecord = DigitDLL.GetSiteClimateByName(accessToken, siteClimate.Name);
944.     if (existingRecord != null)
945.     {
946.         throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", siteClimate
947.         .Name));
948.     }
949.     siteClimate.CreateBy = siteClimate.UpdateBy = accessToken.UserId;
950.     siteClimate.Createdt = siteClimate.Updatedt = Utility.Now();
951.
952.     return DigitDLL.CreateSiteClimate(accessToken, siteClimate);
953. }
954.
955. public DgSiteClimate UpdateSiteClimate(DgAccessToken accessToken, DgSiteClimate siteClimate)
956. {
957.     #region input parameters validation
958.     if (accessToken == null) throw new ArgumentNullException("accessToken");
959.     if (siteClimate == null) throw new ArgumentNullException("siteClimate");
960.     siteClimate.ValidateInstance();
961.     #endregion
962.
963.     #region SecurityLayer
964.     if (accessToken.IsSysAdmin == false)
965.     {
966.         throw new DgException("Only a SysAdmin can do that!");
967.     }
968.     #endregion
969.
970.
971.     var existingRecord = DigitDLL.GetSiteClimateByName(accessToken, siteClimate.Name);
972.     if (existingRecord != null && existingRecord.ClimateId != siteClimate.ClimateId)
973.     {
974.         throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", siteClimate
975.         .Name));
976.     }
977.     if (existingRecord == null)
978.     {
979.         existingRecord = DigitDLL.GetSiteClimateById(accessToken, siteClimate.ClimateId);
980.     }
981.     if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "siteClimate", siteClimate.ClimateId));
982.
983.     existingRecord.Name = siteClimate.Name;
984.     existingRecord.Description = siteClimate.Description;
985.     existingRecord.UpdateBy = accessToken.UserId;
986.     existingRecord.Updatedt = Utility.Now();
987.
988.     return DigitDLL.UpdateSiteClimate(accessToken, existingRecord);
989. }
990.
991. public void DeleteSiteClimate(DgAccessToken accessToken, Int16 climateId)
992. {
993.     if (accessToken == null) throw new ArgumentNullException("accessToken");

```

```

994.         var existingRecord = DigitDLL.GetSiteClimateById(accessToken, climateId);
995.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_item_wit
    h_id, "SiteClimate", climateId));
996.
997.         #region SecurityLayer
998.
999.         #endregion
1000.
1001.         //Can be deleted?
1002.         DigitDLL.CanDeleteSiteClimate(accessToken, existingRecord.ClimateId);
1003.
1004.         //Πραγματοποιούμε την διαγραφή:
1005.         DigitDLL.DeleteSiteClimate(accessToken, existingRecord.ClimateId);
1006.     }
1007.     #endregion
1008.
1009.
1010.
1011.     #region SiteEnv
1012.     public DgSiteEnv GetSiteEnvById(DgAccessToken accessToken, Int16 environmentId)
1013.     {
1014.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
1015.
1016.         #region SecurityLayer
1017.         #endregion
1018.
1019.         return DigitDLL.GetSiteEnvById(accessToken, environmentId);
1020.     }
1021.
1022.     public IList<DgSiteEnv> GetSiteEnvs(DgAccessToken accessToken, string whereClause = n
    ull, string orderByClause = "order by name")
1023.     {
1024.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
1025.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
        ;
1026.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
    se");
1027.
1028.         #region SecurityLayer
1029.         #endregion
1030.
1031.         return DigitDLL.GetSiteEnvs(accessToken, whereClause, orderByClause);
1032.     }
1033.
1034.
1035.     public IList<DgSiteEnv> GetSiteEnvs(DgAccessToken accessToken, int pageIndex, int pag
    eSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
1036.     {
1037.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1038.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
        ;
1039.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
    se");
1040.
1041.         #region SecurityLayer
1042.
1043.         #endregion
1044.
1045.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
1046.         int startRowIndex = ((pageIndex - 1) * pageSize);
1047.
1048.         return DigitDLL.GetSiteEnvs(accessToken, startRowIndex, pageSize, ref totalRows,
        whereClause, orderByClause);

```

```

1049.     }
1050.
1051.     public DgSiteEnv CreateSiteEnv(DgAccessToken accessToken, DgSiteEnv siteEnv)
1052.     {
1053.         #region input parameters validation
1054.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1055.         if (siteEnv == null) throw new ArgumentNullException("siteEnv");
1056.         siteEnv.ValidateInstance();
1057.         #endregion
1058.
1059.         #region SecurityLayer
1060.         if (accessToken.IsSysAdmin == false)
1061.         {
1062.             throw new DgException("Only a SysAdmin can do that!");
1063.         }
1064.         #endregion
1065.
1066.
1067.         /*Check Name uniqueness*/
1068.         var existingRecord = DigitDLL.GetSiteEnvByName(accessToken, siteEnv.Name);
1069.         if (existingRecord != null)
1070.         {
1071.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", siteEnv
1072. Name));
1073.         }
1074.
1075.         siteEnv.CreateBy = siteEnv.UpdateBy = accessToken.UserId;
1076.         siteEnv.CreateDt = siteEnv.UpdateDt = Utility.Now();
1077.
1078.         return DigitDLL.CreateSiteEnv(accessToken, siteEnv);
1079.     }
1080.     public DgSiteEnv UpdateSiteEnv(DgAccessToken accessToken, DgSiteEnv siteEnv)
1081.     {
1082.         #region input parameters validation
1083.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1084.         if (siteEnv == null) throw new ArgumentNullException("siteEnv");
1085.         siteEnv.ValidateInstance();
1086.         #endregion
1087.
1088.         #region SecurityLayer
1089.         if (accessToken.IsSysAdmin == false)
1090.         {
1091.             throw new DgException("Only a SysAdmin can do that!");
1092.         }
1093.         #endregion
1094.
1095.
1096.         var existingRecord = DigitDLL.GetSiteEnvByName(accessToken, siteEnv.Name);
1097.         if (existingRecord != null && existingRecord.EnvironmentId != siteEnv.Environment
1098. Id)
1099.         {
1100.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", siteEnv
1101. Name));
1102.         }
1103.         if (existingRecord == null)
1104.         {
1105.             existingRecord = DigitDLL.GetSiteEnvById(accessToken, siteEnv.EnvironmentId);
1106.         }
1107.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
1108. m_with_id, "siteEnv", siteEnv.EnvironmentId));
1109.
1110.

```

```

1107.
1108.         existingRecord.Name = siteEnv.Name;
1109.         existingRecord.Description = siteEnv.Description;
1110.         existingRecord.UpdateBy = accessToken.UserId;
1111.         existingRecord.UpdateDt = Utility.Now();
1112.
1113.         return DigitDLL.UpdateSiteEnv(accessToken, existingRecord);
1114.     }
1115.
1116.     public void DeleteSiteEnv(DgAccessToken accessToken, Int16 environmentId)
1117.     {
1118.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1119.         var existingRecord = DigitDLL.GetSiteEnvById(accessToken, environmentId);
1120.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "SiteEnv", environmentId));
1121.
1122.         #region SecurityLayer
1123.
1124.         #endregion
1125.
1126.         //Can be deleted?
1127.         DigitDLL.CanDeleteSiteEnv(accessToken, existingRecord.EnvironmentId);
1128.
1129.         //Πραγματοποιούμε την διαγραφή:
1130.         DigitDLL.DeleteSiteEnv(accessToken, existingRecord.EnvironmentId);
1131.     }
1132.     #endregion
1133.
1134.     #region SiteForm
1135.     public DgSiteForm GetSiteFormById(DgAccessToken accessToken, Int16 formId)
1136.     {
1137.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
1138.
1139.         #region SecurityLayer
1140.         #endregion
1141.
1142.         return DigitDLL.GetSiteFormById(accessToken, formId);
1143.     }
1144.
1145.     public IList<DgSiteForm> GetSiteForms(DgAccessToken accessToken, string whereClause =
null, string orderByClause = "order by Name")
1146.     {
1147.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
1148.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
1149.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
1150.
1151.         #region SecurityLayer
1152.         #endregion
1153.
1154.         return DigitDLL.GetSiteForms(accessToken, whereClause, orderByClause);
1155.     }
1156.
1157.
1158.     public IList<DgSiteForm> GetSiteForms(DgAccessToken accessToken, int pageIndex, int p
ageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
1159.     {
1160.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1161.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
1162.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");

```

```

1163.
1164.         #region SecurityLayer
1165.
1166.         #endregion
1167.
1168.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
1169.         int startRowIndex = ((pageIndex - 1) * pageSize);
1170.
1171.         return DigitDLL.GetSiteForms(accessToken, startRowIndex, pageSize, ref totalRows,
whereClause, orderByClause);
1172.     }
1173.
1174.     public DgSiteForm CreateSiteForm(DgAccessToken accessToken, DgSiteForm siteForm)
1175.     {
1176.         #region input parameters validation
1177.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1178.         if (siteForm == null) throw new ArgumentNullException("siteForm");
1179.         siteForm.ValidateInstance();
1180.         #endregion
1181.
1182.         #region SecurityLayer
1183.         if (accessToken.IsSysAdmin == false)
1184.         {
1185.             throw new DgException("Only a SysAdmin can do that!");
1186.         }
1187.         #endregion
1188.
1189.         /*Check Name uniqueness*/
1190.         var existingRecord = DigitDLL.GetSiteFormByName(accessToken, siteForm.Name);
1191.         if (existingRecord != null)
1192.         {
1193.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", siteFo
rm.Name));
1194.         }
1195.
1196.         siteForm.CreateBy = siteForm.UpdateBy = accessToken.UserId;
1197.         siteForm.Createdt = siteForm.UpdateDt = Utility.Now();
1198.
1199.         return DigitDLL.CreateSiteForm(accessToken, siteForm);
1200.     }
1201.
1202.     public DgSiteForm UpdateSiteForm(DgAccessToken accessToken, DgSiteForm siteForm)
1203.     {
1204.         #region input parameters validation
1205.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1206.         if (siteForm == null) throw new ArgumentNullException("siteForm");
1207.         siteForm.ValidateInstance();
1208.         #endregion
1209.
1210.         #region SecurityLayer
1211.         if (accessToken.IsSysAdmin == false)
1212.         {
1213.             throw new DgException("Only a SysAdmin can do that!");
1214.         }
1215.         #endregion
1216.
1217.
1218.         var existingRecord = DigitDLL.GetSiteFormByName(accessToken, siteForm.Name);
1219.         if (existingRecord != null && existingRecord.FormId != siteForm.FormId)
1220.         {
1221.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", siteFo
rm.Name));
1222.         }

```

```

1223.         if (existingRecord == null)
1224.         {
1225.             existingRecord = DigitDLL.GetSiteFormById(accessToken, siteForm.FormId);
1226.         }
1227.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "siteForm", siteForm.FormId));
1228.
1229.
1230.         existingRecord.Name = siteForm.Name;
1231.         existingRecord.Description = siteForm.Description;
1232.         existingRecord.UpdateBy = accessToken.UserId;
1233.         existingRecord.UpdateDt = Utility.Now();
1234.
1235.         return DigitDLL.UpdateSiteForm(accessToken, existingRecord);
1236.     }
1237.
1238.     public void DeleteSiteForm(DgAccessToken accessToken, Int16 formId)
1239.     {
1240.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1241.         var existingRecord = DigitDLL.GetSiteFormById(accessToken, formId);
1242.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "SiteForm", formId));
1243.
1244.         #region SecurityLayer
1245.
1246.         #endregion
1247.
1248.         //Can be deleted?
1249.         DigitDLL.CanDeleteSiteForm(accessToken, existingRecord.FormId);
1250.
1251.
1252.         DigitDLL.DeleteSiteForm(accessToken, existingRecord.FormId);
1253.     }
1254.     #endregion
1255.
1256.     #region SiteExcType
1257.     public DgSiteExcType GetSiteExcTypeById(DgAccessToken accessToken, Int16 exctypeId)
1258.     {
1259.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
1260.
1261.         #region SecurityLayer
1262.         #endregion
1263.
1264.         return DigitDLL.GetSiteExcTypeById(accessToken, exctypeId);
1265.     }
1266.
1267.     public IList<DgSiteExcType> GetSiteExcTypes(DgAccessToken accessToken, string whereCl
ause = null, string orderByClause = "order by Name")
1268.     {
1269.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
1270.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
1271.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
1272.
1273.         #region SecurityLayer
1274.         #endregion
1275.
1276.         return DigitDLL.GetSiteExcTypes(accessToken, whereClause, orderByClause);
1277.     }
1278.
1279.

```

```

1280.         public IList<DgSiteExcType> GetSiteExcTypes(DgAccessToken accessToken, int pageIndex,
1281.             int pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name
1282.             ")
1283.         {
1284.             if (accessToken == null) throw new ArgumentNullException("accessToken");
1285.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
1286.             ;
1287.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
1288.             se");
1289.             #region SecurityLayer
1290.             #endregion
1291.             Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
1292.             int startRowIndex = ((pageIndex - 1) * pageSize);
1293.             return DigitDLL.GetSiteExcTypes(accessToken, startRowIndex, pageSize, ref totalRo
1294.             ws, whereClause, orderByClause);
1295.         }
1296.         public DgSiteExcType CreateSiteExcType(DgAccessToken accessToken, DgSiteExcType siteE
1297.             xcType)
1298.         {
1299.             #region input parameters validation
1300.             if (accessToken == null) throw new ArgumentNullException("accessToken");
1301.             if (siteExcType == null) throw new ArgumentNullException("siteExcType");
1302.             siteExcType.ValidateInstance();
1303.             #endregion
1304.             #region SecurityLayer
1305.             if (accessToken.IsSysAdmin == false)
1306.             {
1307.                 throw new DgException("Only a SysAdmin can do that!");
1308.             }
1309.             #endregion
1310.             /*Check Name uniqueness*/
1311.             var existingRecord = DigitDLL.GetSiteExcTypeByName(accessToken, siteExcType.Name)
1312.             ;
1313.             if (existingRecord != null)
1314.             {
1315.                 throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", siteEx
1316.                 cType.Name));
1317.             }
1318.             siteExcType.CreateBy = siteExcType.UpdateBy = accessToken.UserId;
1319.             siteExcType.CreateDt = siteExcType.UpdateDt = Utility.Now();
1320.             return DigitDLL.CreateSiteExcType(accessToken, siteExcType);
1321.         }
1322.         public DgSiteExcType UpdateSiteExcType(DgAccessToken accessToken, DgSiteExcType siteE
1323.             xcType)
1324.         {
1325.             #region input parameters validation
1326.             if (accessToken == null) throw new ArgumentNullException("accessToken");
1327.             if (siteExcType == null) throw new ArgumentNullException("siteExcType");
1328.             siteExcType.ValidateInstance();
1329.             #endregion
1330.             #region SecurityLayer
1331.             if (accessToken.IsSysAdmin == false)

```



```

1334.         {
1335.             throw new DgException("Only a SysAdmin can do that!");
1336.         }
1337.         #endregion
1338.
1339.
1340.         var existingRecord = DigitDLL.GetSiteExcTypeByName(accessToken, siteExcType.Name)
1341.         ;
1342.         if (existingRecord != null && existingRecord.ExctypeId != siteExcType.ExctypeId)
1343.         {
1344.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", siteExcType.Name));
1345.         }
1346.         if (existingRecord == null)
1347.         {
1348.             existingRecord = DigitDLL.GetSiteExcTypeById(accessToken, siteExcType.ExctypeId);
1349.         }
1350.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "siteExcType", siteExcType.ExctypeId));
1351.
1352.         existingRecord.ExctypeId = siteExcType.ExctypeId;
1353.         existingRecord.Name = siteExcType.Name;
1354.         existingRecord.Description = siteExcType.Description;
1355.
1356.         existingRecord.UpdateBy = accessToken.UserId;
1357.         existingRecord.UpdateDt = Utility.Now();
1358.
1359.         return DigitDLL.UpdateSiteExcType(accessToken, existingRecord);
1360.     }
1361.
1362.     public void DeleteSiteExcType(DgAccessToken accessToken, Int16 exctypeId)
1363.     {
1364.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1365.         var existingRecord = DigitDLL.GetSiteExcTypeById(accessToken, exctypeId);
1366.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "SiteExcType", exctypeId));
1367.
1368.         #region SecurityLayer
1369.
1370.         #endregion
1371.
1372.         //Can be deleted?
1373.         DigitDLL.CanDeleteSiteExcType(accessToken, existingRecord.ExctypeId);
1374.
1375.         DigitDLL.DeleteSiteExcType(accessToken, existingRecord.ExctypeId);
1376.     }
1377.     #endregion
1378.
1379.     #region DgSiteExcTech
1380.     public DgSiteExcTech GetSiteExcTechById(DgAccessToken accessToken, Int16 exctechId)
1381.     {
1382.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
1383.
1384.         #region SecurityLayer
1385.         #endregion
1386.
1387.         return DigitDLL.GetSiteExcTechById(accessToken, exctechId);
1388.     }
1389.
1390.

```

```

1391.         public IList<DgSiteExcTech> GetSiteExcTechs(DgAccessToken accessToken, string whereCl
ause = null, string orderByClause = "order by Name")
1392.         {
1393.             if (accessToken == null) throw new ArgumentNullException("accesstoken");
1394.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
1395.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
1396.
1397.             #region SecurityLayer
1398.             #endregion
1399.
1400.             return DigitDLL.GetSiteExcTechs(accessToken, whereClause, orderByClause);
1401.         }
1402.
1403.
1404.         public IList<DgSiteExcTech> GetSiteExcTechs(DgAccessToken accessToken, int pageIndex,
int pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name
")
1405.         {
1406.             if (accessToken == null) throw new ArgumentNullException("accessToken");
1407.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
1408.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
1409.
1410.             #region SecurityLayer
1411.             #endregion
1412.
1413.             Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
1414.             int startRowIndex = ((pageIndex - 1) * pageSize);
1415.
1416.             return DigitDLL.GetSiteExcTechs(accessToken, startRowIndex, pageSize, ref totalRo
ws, whereClause, orderByClause);
1417.         }
1418.     }
1419.
1420.     public DgSiteExcTech CreateSiteExcTech(DgAccessToken accessToken, DgSiteExcTech siteE
xcTech)
1421.     {
1422.         #region input parameters validation
1423.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1424.         if (siteExcTech == null) throw new ArgumentNullException("siteExcTech");
1425.         siteExcTech.ValidateInstance();
1426.         #endregion
1427.
1428.         #region SecurityLayer
1429.         if (accessToken.IsSysAdmin == false)
1430.         {
1431.             throw new DgException("Only a SysAdmin can do that!");
1432.         }
1433.         #endregion
1434.
1435.         /*Check Name uniqueness*/
1436.         var existingRecord = DigitDLL.GetSiteExcTechByName(accessToken, siteExcTech.Name)
;
1437.         if (existingRecord != null)
1438.         {
1439.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", siteEx
cTech.Name));
1440.         }
1441.
1442.         siteExcTech.CreateBy = siteExcTech.UpdateBy = accessToken.UserId;

```

```

1443.             siteExcTech.CreateDt = siteExcTech.UpdateDt = Utility.Now();
1444.
1445.             return DigitDLL.CreateSiteExcTech(accessToken, siteExcTech);
1446.         }
1447.
1448.         public DgSiteExcTech UpdateSiteExcTech(DgAccessToken accessToken, DgSiteExcTech siteExcTech)
1449.         {
1450.             #region input parameters validation
1451.             if (accessToken == null) throw new ArgumentNullException("accessToken");
1452.             if (siteExcTech == null) throw new ArgumentNullException("siteExcTech");
1453.             siteExcTech.ValidateInstance();
1454.             #endregion
1455.
1456.             #region SecurityLayer
1457.             if (accessToken.IsSysAdmin == false)
1458.             {
1459.                 throw new DgException("Only a SysAdmin can do that!");
1460.             }
1461.             #endregion
1462.
1463.
1464.             var existingRecord = DigitDLL.GetSiteExcTechByName(accessToken, siteExcTech.Name)
1465.             ;
1466.             if (existingRecord != null && existingRecord.ExcotechId != siteExcTech.ExcotechId)
1467.             {
1468.                 throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", siteExcTech.Name));
1469.             }
1470.             if (existingRecord == null)
1471.             {
1472.                 existingRecord = DigitDLL.GetSiteExcTechById(accessToken, siteExcTech.ExcotechId);
1473.             }
1474.             if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "siteExcTech", siteExcTech.ExcotechId));
1475.
1476.             existingRecord.ExcotechId = siteExcTech.ExcotechId;
1477.             existingRecord.Name = siteExcTech.Name;
1478.             existingRecord.Description = siteExcTech.Description;
1479.
1480.             existingRecord.UpdateBy = accessToken.UserId;
1481.             existingRecord.UpdateDt = Utility.Now();
1482.
1483.             return DigitDLL.UpdateSiteExcTech(accessToken, existingRecord);
1484.         }
1485.
1486.         public void DeleteSiteExcTech(DgAccessToken accessToken, Int16 excotechId)
1487.         {
1488.             if (accessToken == null) throw new ArgumentNullException("accessToken");
1489.             var existingRecord = DigitDLL.GetSiteExcTechById(accessToken, excotechId);
1490.             if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "SiteExcTech", excotechId));
1491.
1492.             #region SecurityLayer
1493.
1494.             #endregion
1495.
1496.             //Can be deleted?
1497.             DigitDLL.CanDeleteSiteExcTech(accessToken, existingRecord.ExcotechId);
1498.

```

```

1499.
1500.         DigitDLL.DeleteSiteExcTech(accessToken, existingRecord.ExctechId);
1501.     }
1502.     #endregion
1503.
1504.     #region SiteInvMethod
1505.     public DgSiteInvMethod GetSiteInvMethodById(DgAccessToken accessToken, Int16 invmetho
dId)
1506.     {
1507.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
1508.
1509.         #region SecurityLayer
1510.         #endregion
1511.
1512.         return DigitDLL.GetSiteInvMethodById(accessToken, invmethodId);
1513.     }
1514.
1515.     public IList<DgSiteInvMethod> GetSiteInvMethods(DgAccessToken accessToken, string whe
reClause = null, string orderByClause = "order by Name")
1516.     {
1517.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
1518.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
1519.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
1520.
1521.         #region SecurityLayer
1522.         #endregion
1523.
1524.         return DigitDLL.GetSiteInvMethods(accessToken, whereClause, orderByClause);
1525.     }
1526.
1527.
1528.     public IList<DgSiteInvMethod> GetSiteInvMethods(DgAccessToken accessToken, int pageIn
dex, int pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by
Name")
1529.     {
1530.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1531.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
1532.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
1533.
1534.         #region SecurityLayer
1535.
1536.         #endregion
1537.
1538.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
1539.         int startRowIndex = ((pageIndex - 1) * pageSize);
1540.
1541.         return DigitDLL.GetSiteInvMethods(accessToken, startRowIndex, pageSize, ref total
Rows, whereClause, orderByClause);
1542.     }
1543.
1544.     public DgSiteInvMethod CreateSiteInvMethod(DgAccessToken accessToken, DgSiteInvMethod
siteInvMethod)
1545.     {
1546.         #region input parameters validation
1547.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1548.         if (siteInvMethod == null) throw new ArgumentNullException("siteInvMethod");
1549.         siteInvMethod.ValidateInstance();
1550.         #endregion
1551.

```

```

1552.         #region SecurityLayer
1553.         if (accessToken.IsSysAdmin == false)
1554.         {
1555.             throw new DgException("Only a SysAdmin can do that!");
1556.         }
1557.         #endregion
1558.
1559.         /*Check Name uniqueness*/
1560.         var existingRecord = DigitDLL.GetSiteInvMethodByName(accessToken, siteInvMethod.N
ame);
1561.         if (existingRecord != null)
1562.         {
1563.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", siteIn
vMethod.Name));
1564.         }
1565.
1566.         siteInvMethod.CreateBy = siteInvMethod.UpdateBy = accessToken.UserId;
1567.         siteInvMethod.CreateDt = siteInvMethod.UpdateDt = Utility.Now();
1568.
1569.         return DigitDLL.CreateSiteInvMethod(accessToken, siteInvMethod);
1570.     }
1571.
1572.     public DgSiteInvMethod UpdateSiteInvMethod(DgAccessToken accessToken, DgSiteInvMethod
siteInvMethod)
1573.     {
1574.         #region input parameters validation
1575.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1576.         if (siteInvMethod == null) throw new ArgumentNullException("siteInvMethod");
1577.         siteInvMethod.ValidateInstance();
1578.         #endregion
1579.
1580.         #region SecurityLayer
1581.         if (accessToken.IsSysAdmin == false)
1582.         {
1583.             throw new DgException("Only a SysAdmin can do that!");
1584.         }
1585.         #endregion
1586.
1587.
1588.         var existingRecord = DigitDLL.GetSiteInvMethodByName(accessToken, siteInvMethod.N
ame);
1589.         if (existingRecord != null && existingRecord.InvmethodId != siteInvMethod.Invmeth
odId)
1590.         {
1591.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", siteIn
vMethod.Name));
1592.         }
1593.         if (existingRecord == null)
1594.         {
1595.             existingRecord = DigitDLL.GetSiteInvMethodById(accessToken, siteInvMethod.Inv
methodId);
1596.         }
1597.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "siteInvMethod", siteInvMethod.InvmethodId));
1598.
1599.
1600.         existingRecord.InvmethodId = siteInvMethod.InvmethodId;
1601.         existingRecord.Name = siteInvMethod.Name;
1602.         existingRecord.Description = siteInvMethod.Description;
1603.
1604.         existingRecord.UpdateBy = accessToken.UserId;
1605.         existingRecord.UpdateDt = Utility.Now();
1606.

```

```

1607.         return DigitDLL.UpdateSiteInvMethod(accessToken, existingRecord);
1608.     }
1609.
1610.     public void DeleteSiteInvMethod(DgAccessToken accessToken, Int16 invmethodId)
1611.     {
1612.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1613.         var existingRecord = DigitDLL.GetSiteInvMethodById(accessToken, invmethodId);
1614.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "SiteInvMethod", invmethodId));
1615.
1616.         #region SecurityLayer
1617.
1618.         #endregion
1619.
1620.         //Can be deleted?
1621.         DigitDLL.CanDeleteSiteInvMethod(accessToken, existingRecord.InvmethodId);
1622.
1623.
1624.         DigitDLL.DeleteSiteInvMethod(accessToken, existingRecord.InvmethodId);
1625.     }
1626.     #endregion
1627.
1628.
1629.     #region DgContCategory
1630.     public DgContCategory GetContCategoryById(DgAccessToken accessToken, Int16 categoryId
)
1631.     {
1632.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
1633.
1634.         #region SecurityLayer
1635.         #endregion
1636.
1637.         return DigitDLL.GetContCategoryById(accessToken, categoryId);
1638.     }
1639.
1640.     public IList<DgContCategory> GetContCategories(DgAccessToken accessToken, string wher
eClause = null, string orderByClause = "order by Name")
1641.     {
1642.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
1643.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
1644.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
1645.
1646.         #region SecurityLayer
1647.         #endregion
1648.
1649.         return DigitDLL.GetContCategories(accessToken, whereClause, orderByClause);
1650.     }
1651.
1652.
1653.     public IList<DgContCategory> GetContCategories(DgAccessToken accessToken, int pageInd
ex, int pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by N
ame")
1654.     {
1655.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1656.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
1657.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
1658.
1659.         #region SecurityLayer
1660.

```

```

1661.             #endregion
1662.
1663.             Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
1664.             int startRowIndex = ((pageIndex - 1) * pageSize);
1665.
1666.             return DigitDLL.GetContCategories(accessToken, startRowIndex, pageSize, ref total
1667. Rows, whereClause, orderByClause);
1668.         }
1669.         public DgContCategory CreateContCategory(DgAccessToken accessToken, DgContCategory co
1670. ntCategory)
1671.         {
1672.             #region input parameters validation
1673.             if (accessToken == null) throw new ArgumentNullException("accessToken");
1674.             if (contCategory == null) throw new ArgumentNullException("contCategory");
1675.             contCategory.ValidateInstance();
1676.             #endregion
1677.
1678.             #region SecurityLayer
1679.             if (accessToken.IsSysAdmin == false)
1680.             {
1681.                 throw new DgException("Only a SysAdmin can do that!");
1682.             }
1683.             #endregion
1684.
1685.             /*Check Name uniqueness*/
1686.             var existingRecord = DigitDLL.GetContCategoryByName(accessToken, contCategory.Nam
1687. e);
1688.             if (existingRecord != null)
1689.             {
1690.                 throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contCa
1691. tegory.Name));
1692.             }
1693.
1694.             contCategory.CreateBy = contCategory.UpdateBy = accessToken.UserId;
1695.             contCategory.CreateDt = contCategory.UpdateDt = Utility.Now();
1696.
1697.             return DigitDLL.CreateContCategory(accessToken, contCategory);
1698.         }
1699.         public DgContCategory UpdateContCategory(DgAccessToken accessToken, DgContCategory co
1700. ntCategory)
1701.         {
1702.             #region input parameters validation
1703.             if (accessToken == null) throw new ArgumentNullException("accessToken");
1704.             if (contCategory == null) throw new ArgumentNullException("contCategory");
1705.             contCategory.ValidateInstance();
1706.             #endregion
1707.
1708.             #region SecurityLayer
1709.             if (accessToken.IsSysAdmin == false)
1710.             {
1711.                 throw new DgException("Only a SysAdmin can do that!");
1712.             }
1713.             #endregion
1714.
1715.             var existingRecord = DigitDLL.GetContCategoryByName(accessToken, contCategory.Nam
1716. e);
1717.             if (existingRecord != null && existingRecord.CategoryId != contCategory.CategoryI
1718. d)
1719.             {

```

```

1716.         throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contCa
    tegrity.Name));
1717.     }
1718.     if (existingRecord == null)
1719.     {
1720.         existingRecord = DigitDLL.GetContCategoryById(accessToken, contCategory.Categ
    oryId);
1721.     }
1722.     if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
    m_with_id, "contCategory", contCategory.CategoryId));
1723.
1724.
1725.         existingRecord.CategoryId = contCategory.CategoryId;
1726.         existingRecord.Name = contCategory.Name;
1727.         existingRecord.Description = contCategory.Description;
1728.
1729.         existingRecord.UpdateBy = accessToken.UserId;
1730.         existingRecord.UpdateDt = Utility.Now();
1731.
1732.         return DigitDLL.UpdateContCategory(accessToken, existingRecord);
1733.     }
1734.
1735.     public void DeleteContCategory(DgAccessToken accessToken, Int16 categoryId)
1736.     {
1737.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1738.         var existingRecord = DigitDLL.GetContCategoryById(accessToken, categoryId);
1739.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
    m_with_id, "ContCategory", categoryId));
1740.
1741.         #region SecurityLayer
1742.
1743.         #endregion
1744.
1745.         //Can be deleted?
1746.         DigitDLL.CanDeleteContCategory(accessToken, existingRecord.CategoryId);
1747.
1748.
1749.         DigitDLL.DeleteContCategory(accessToken, existingRecord.CategoryId);
1750.     }
1751.     #endregion
1752.
1753.     #region DgContConsist
1754.     public DgContConsist GetContConsistById(DgAccessToken accessToken, Int16 consistId)
1755.     {
1756.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
1757.
1758.         #region SecurityLayer
1759.         #endregion
1760.
1761.         return DigitDLL.GetContConsistById(accessToken, consistId);
1762.     }
1763.
1764.     public IList<DgContConsist> GetContConsists(DgAccessToken accessToken, string whereCl
    ause = null, string orderByClause = "order by Name")
1765.     {
1766.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
1767.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
    ;
1768.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
    se");
1769.
1770.         #region SecurityLayer
1771.         #endregion

```



```

1772.
1773.         return DigitDLL.GetContConsists(accessToken, whereClause, orderByClause);
1774.     }
1775.
1776.
1777.     public IList<DgContConsist> GetContConsists(DgAccessToken accessToken, int pageIndex,
1778.         int pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name
1779.         ")
1780.     {
1781.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1782.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
1783.         ;
1784.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
1785.         se");
1786.
1787.         #region SecurityLayer
1788.
1789.         #endregion
1790.
1791.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
1792.         int startRowIndex = ((pageIndex - 1) * pageSize);
1793.
1794.         return DigitDLL.GetContConsists(accessToken, startRowIndex, pageSize, ref totalRo
1795.         ws, whereClause, orderByClause);
1796.     }
1797.
1798.     public DgContConsist CreateContConsist(DgAccessToken accessToken, DgContConsist contC
1799.     onsist)
1800.     {
1801.         #region input parameters validation
1802.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1803.         if (contConsist == null) throw new ArgumentNullException("contConsist");
1804.         contConsist.ValidateInstance();
1805.         #endregion
1806.
1807.         #region SecurityLayer
1808.         if (accessToken.IsSysAdmin == false)
1809.         {
1810.             throw new DgException("Only a SysAdmin can do that!");
1811.         }
1812.         #endregion
1813.
1814.         /*Check Name uniqueness*/
1815.         var existingRecord = DigitDLL.GetContConsistByName(accessToken, contConsist.Name)
1816.         ;
1817.         if (existingRecord != null)
1818.         {
1819.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contCo
1820.             nsist.Name));
1821.         }
1822.
1823.         contConsist.CreateBy = contConsist.UpdateBy = accessToken.UserId;
1824.         contConsist.CreateDt = contConsist.UpdateDt = Utility.Now();
1825.
1826.         return DigitDLL.CreateContConsist(accessToken, contConsist);
1827.     }
1828.
1829.     public DgContConsist UpdateContConsist(DgAccessToken accessToken, DgContConsist contC
1830.     onsist)
1831.     {
1832.         #region input parameters validation
1833.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1834.         if (contConsist == null) throw new ArgumentNullException("contConsist");

```

```

1826.         contConsist.ValidateInstance();
1827.         #endregion
1828.
1829.         #region SecurityLayer
1830.         if (accessToken.IsSysAdmin == false)
1831.         {
1832.             throw new DgException("Only a SysAdmin can do that!");
1833.         }
1834.         #endregion
1835.
1836.
1837.         var existingRecord = DigitDLL.GetContConsistByName(accessToken, contConsist.Name)
1838.         ;
1839.         if (existingRecord != null && existingRecord.ConsistId != contConsist.ConsistId)
1840.         {
1841.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contCo
1842. nsist.Name));
1843.         }
1844.         if (existingRecord == null)
1845.         {
1846.             existingRecord = DigitDLL.GetContConsistById(accessToken, contConsist.Consist
1847. Id);
1848.         }
1849.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
1850 m_with_id, "contConsist", contConsist.ConsistId));
1851.
1852.         existingRecord.ConsistId = contConsist.ConsistId;
1853.         existingRecord.Name = contConsist.Name;
1854.         existingRecord.Description = contConsist.Description;
1855.
1856.         existingRecord.UpdateBy = accessToken.UserId;
1857.         existingRecord.UpdateDt = Utility.Now();
1858.
1859.         return DigitDLL.UpdateContConsist(accessToken, existingRecord);
1860.     }
1861.
1862.     public void DeleteContConsist(DgAccessToken accessToken, Int16 consistId)
1863.     {
1864.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1865.         var existingRecord = DigitDLL.GetContConsistById(accessToken, consistId);
1866.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
1867. m_with_id, "ContConsist", consistId));
1868.
1869.         #region SecurityLayer
1870.
1871.         #endregion
1872.
1873.         //Can be deleted?
1874.         DigitDLL.CanDeleteContConsist(accessToken, existingRecord.ConsistId);
1875.
1876.         DigitDLL.DeleteContConsist(accessToken, existingRecord.ConsistId);
1877.     }
1878.     #endregion
1879.
1880.     #region DgContCorse
1881.     public DgContCorse GetContCorseById(DgAccessToken accessToken, Int16 corceId)
1882.     {
1883.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
1884.
1885.         #region SecurityLayer

```

```

1883.             #endregion
1884.
1885.             return DigitDLL.GetContCorseById(accessToken, corceId);
1886.         }
1887.
1888.         public IList<DgContCorse> GetContCorses(DgAccessToken accessToken, string whereClause
= null, string orderByClause = "order by Name")
1889.         {
1890.             if (accessToken == null) throw new ArgumentNullException("accesstoken");
1891.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
1892.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
1893.
1894.             #region SecurityLayer
1895.             #endregion
1896.
1897.             return DigitDLL.GetContCorses(accessToken, whereClause, orderByClause);
1898.         }
1899.
1900.
1901.         public IList<DgContCorse> GetContCorses(DgAccessToken accessToken, int pageIndex, int
pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
1902.         {
1903.             if (accessToken == null) throw new ArgumentNullException("accessToken");
1904.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
1905.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
1906.
1907.             #region SecurityLayer
1908.             #endregion
1909.
1910.             Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
1911.             int startRowIndex = ((pageIndex - 1) * pageSize);
1912.
1913.             return DigitDLL.GetContCorses(accessToken, startRowIndex, pageSize, ref totalRows
, whereClause, orderByClause);
1914.         }
1915.
1916.         public DgContCorse CreateContCorse(DgAccessToken accessToken, DgContCorse contCorse)
1917.         {
1918.             #region input parameters validation
1919.             if (accessToken == null) throw new ArgumentNullException("accessToken");
1920.             if (contCorse == null) throw new ArgumentNullException("contCorse");
1921.             contCorse.ValidateInstance();
1922.             #endregion
1923.
1924.             #region SecurityLayer
1925.             if (accessToken.IsSysAdmin == false)
1926.             {
1927.                 throw new DgException("Only a SysAdmin can do that!");
1928.             }
1929.             #endregion
1930.
1931.             /*Check Name uniqueness*/
1932.             var existingRecord = DigitDLL.GetContCorseByName(accessToken, contCorse.Name);
1933.             if (existingRecord != null)
1934.             {
1935.                 throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contCo
rse.Name));
1936.

```

```

1937.         }
1938.
1939.         contCorse.CreateBy = contCorse.UpdateBy = accessToken.UserId;
1940.         contCorse.CreateDt = contCorse.UpdateDt = Utility.Now();
1941.
1942.         return DigitDLL.CreateContCorse(accessToken, contCorse);
1943.     }
1944.
1945.     public DgContCorse UpdateContCorse(DgAccessToken accessToken, DgContCorse contCorse)
1946.     {
1947.         #region input parameters validation
1948.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1949.         if (contCorse == null) throw new ArgumentNullException("contCorse");
1950.         contCorse.ValidateInstance();
1951.         #endregion
1952.
1953.         #region SecurityLayer
1954.         if (accessToken.IsSysAdmin == false)
1955.         {
1956.             throw new DgException("Only a SysAdmin can do that!");
1957.         }
1958.         #endregion
1959.
1960.
1961.         var existingRecord = DigitDLL.GetContCorseByName(accessToken, contCorse.Name);
1962.         if (existingRecord != null && existingRecord.CorceId != contCorse.CorceId)
1963.         {
1964.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contCo
1965. rse.Name));
1966.         }
1967.         if (existingRecord == null)
1968.         {
1969.             existingRecord = DigitDLL.GetContCorseById(accessToken, contCorse.CorceId);
1970.         }
1971.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
1972. m_with_id, "contCorse", contCorse.CorceId));
1973.
1974.         existingRecord.CorceId = contCorse.CorceId;
1975.         existingRecord.Name = contCorse.Name;
1976.         existingRecord.Description = contCorse.Description;
1977.
1978.         existingRecord.UpdateBy = accessToken.UserId;
1979.         existingRecord.UpdateDt = Utility.Now();
1980.
1981.         return DigitDLL.UpdateContCorse(accessToken, existingRecord);
1982.     }
1983.
1984.     public void DeleteContCorse(DgAccessToken accessToken, Int16 corceId)
1985.     {
1986.         if (accessToken == null) throw new ArgumentNullException("accessToken");
1987.         var existingRecord = DigitDLL.GetContCorseById(accessToken, corceId);
1988.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
1989. m_with_id, "ContCorse", corceId));
1990.
1991.         #region SecurityLayer
1992.
1993.         //Can be deleted?
1994.         DigitDLL.CanDeleteContCorse(accessToken, existingRecord.CorceId);
1995.

```

```

1996.
1997.         DigitDLL.DeleteContCorse(accessToken, existingRecord.CorceId);
1998.     }
1999.     #endregion
2000.
2001.     #region DgContEdgeDef
2002.     public DgContEdgeDef GetContEdgeDefById(DgAccessToken accessToken, Int16 edgedefId)
2003.     {
2004.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
2005.
2006.         #region SecurityLayer
2007.         #endregion
2008.
2009.         return DigitDLL.GetContEdgeDefById(accessToken, edgedefId);
2010.     }
2011.
2012.     public IList<DgContEdgeDef> GetContEdgeDefs(DgAccessToken accessToken, string whereCl
ause = null, string orderByClause = "order by Name")
2013.     {
2014.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
2015.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
2016.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
2017.
2018.         #region SecurityLayer
2019.         #endregion
2020.
2021.         return DigitDLL.GetContEdgeDefs(accessToken, whereClause, orderByClause);
2022.     }
2023.
2024.
2025.     public IList<DgContEdgeDef> GetContEdgeDefs(DgAccessToken accessToken, int pageIndex,
int pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name
")
2026.     {
2027.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2028.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
2029.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
2030.
2031.         #region SecurityLayer
2032.         #endregion
2033.
2034.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
2035.         int startRowIndex = ((pageIndex - 1) * pageSize);
2036.
2037.         return DigitDLL.GetContEdgeDefs(accessToken, startRowIndex, pageSize, ref totalRo
ws, whereClause, orderByClause);
2038.     }
2039.
2040.
2041.     public DgContEdgeDef CreateContEdgeDef(DgAccessToken accessToken, DgContEdgeDef contE
dgeDef)
2042.     {
2043.         #region input parameters validation
2044.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2045.         if (contEdgeDef == null) throw new ArgumentNullException("contEdgeDef");
2046.         contEdgeDef.ValidateInstance();
2047.         #endregion
2048.
2049.         #region SecurityLayer

```

```

2050.         if (accessToken.IsSysAdmin == false)
2051.         {
2052.             throw new DgException("Only a SysAdmin can do that!");
2053.         }
2054.         #endregion
2055.
2056.         /*Check Name uniqueness*/
2057.         var existingRecord = DigitDLL.GetContEdgeDefByName(accessToken, contEdgeDef.Name)
2058.         ;
2059.         if (existingRecord != null)
2060.         {
2061.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contEd
2062. geDef.Name));
2063.         }
2064.         contEdgeDef.CreateBy = contEdgeDef.UpdateBy = accessToken.UserId;
2065.         contEdgeDef.CreateDt = contEdgeDef.UpdateDt = Utility.Now();
2066.         return DigitDLL.CreateContEdgeDef(accessToken, contEdgeDef);
2067.     }
2068.
2069.     public DgContEdgeDef UpdateContEdgeDef(DgAccessToken accessToken, DgContEdgeDef contE
2070. dgeDef)
2071.     {
2072.         #region input parameters validation
2073.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2074.         if (contEdgeDef == null) throw new ArgumentNullException("contEdgeDef");
2075.         contEdgeDef.ValidateInstance();
2076.         #endregion
2077.
2078.         #region SecurityLayer
2079.         if (accessToken.IsSysAdmin == false)
2080.         {
2081.             throw new DgException("Only a SysAdmin can do that!");
2082.         }
2083.         #endregion
2084.
2085.         var existingRecord = DigitDLL.GetContEdgeDefByName(accessToken, contEdgeDef.Name)
2086.         ;
2087.         if (existingRecord != null && existingRecord.EdgedefId != contEdgeDef.EdgedefId)
2088.         {
2089.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contEd
2090. geDef.Name));
2091.         }
2092.         if (existingRecord == null)
2093.         {
2094.             existingRecord = DigitDLL.GetContEdgeDefById(accessToken, contEdgeDef.Edgedef
2095. Id);
2096.         }
2097.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
2098. m_with_id, "contEdgeDef", contEdgeDef.EdgedefId));
2099.
2100.         existingRecord.EdgedefId = contEdgeDef.EdgedefId;
2101.         existingRecord.Name = contEdgeDef.Name;
2102.         existingRecord.Description = contEdgeDef.Description;
2103.
2104.         existingRecord.UpdateBy = accessToken.UserId;
2105.         existingRecord.UpdateDt = Utility.Now();
2106.
2107.         return DigitDLL.UpdateContEdgeDef(accessToken, existingRecord);

```

```

2105.         }
2106.
2107.         public void DeleteContEdgeDef(DgAccessToken accessToken, Int16 edgedefId)
2108.         {
2109.             if (accessToken == null) throw new ArgumentNullException("accessToken");
2110.             var existingRecord = DigitDLL.GetContEdgeDefById(accessToken, edgedefId);
2111.             if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "ContEdgeDef", edgedefId));
2112.
2113.             #region SecurityLayer
2114.
2115.             #endregion
2116.
2117.             //Can be deleted?
2118.             DigitDLL.CanDeleteContEdgeDef(accessToken, existingRecord.EdgedefId);
2119.
2120.
2121.             DigitDLL.DeleteContEdgeDef(accessToken, existingRecord.EdgedefId);
2122.         }
2123.         #endregion
2124.
2125.         #region DgContGrade
2126.         public DgContGrade GetContGradeById(DgAccessToken accessToken, Int16 gradeId)
2127.         {
2128.             if (accessToken == null) throw new ArgumentNullException("accesstoken");
2129.
2130.             #region SecurityLayer
2131.             #endregion
2132.
2133.             return DigitDLL.GetContGradeById(accessToken, gradeId);
2134.         }
2135.
2136.         public IList<DgContGrade> GetContGrades(DgAccessToken accessToken, string whereClause
= null, string orderByClause = "order by Name")
2137.         {
2138.             if (accessToken == null) throw new ArgumentNullException("accesstoken");
2139.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
2140.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
2141.
2142.             #region SecurityLayer
2143.             #endregion
2144.
2145.             return DigitDLL.GetContGrades(accessToken, whereClause, orderByClause);
2146.         }
2147.
2148.
2149.         public IList<DgContGrade> GetContGrades(DgAccessToken accessToken, int pageIndex, int
pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
2150.         {
2151.             if (accessToken == null) throw new ArgumentNullException("accessToken");
2152.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
2153.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
2154.
2155.             #region SecurityLayer
2156.
2157.             #endregion
2158.
2159.             Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
2160.             int startRowIndex = ((pageIndex - 1) * pageSize);

```

```

2161.
2162.         return DigitDLL.GetContGrades(accessToken, startRowIndex, pageSize, ref totalRows
, whereClause, orderByClause);
2163.     }
2164.
2165.     public DgContGrade CreateContGrade(DgAccessToken accessToken, DgContGrade contGrade)
2166.     {
2167.         #region input parameters validation
2168.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2169.         if (contGrade == null) throw new ArgumentNullException("contGrade");
2170.         contGrade.ValidateInstance();
2171.         #endregion
2172.
2173.         #region SecurityLayer
2174.         if (accessToken.IsSysAdmin == false)
2175.         {
2176.             throw new DgException("Only a SysAdmin can do that!");
2177.         }
2178.         #endregion
2179.
2180.         /*Check Name uniqueness*/
2181.         var existingRecord = DigitDLL.GetContGradeByName(accessToken, contGrade.Name);
2182.         if (existingRecord != null)
2183.         {
2184.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contGr
ade.Name));
2185.         }
2186.
2187.         contGrade.CreateBy = contGrade.UpdateBy = accessToken.UserId;
2188.         contGrade.CreateDt = contGrade.UpdateDt = Utility.Now();
2189.
2190.         return DigitDLL.CreateContGrade(accessToken, contGrade);
2191.     }
2192.
2193.     public DgContGrade UpdateContGrade(DgAccessToken accessToken, DgContGrade contGrade)
2194.     {
2195.         #region input parameters validation
2196.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2197.         if (contGrade == null) throw new ArgumentNullException("contGrade");
2198.         contGrade.ValidateInstance();
2199.         #endregion
2200.
2201.         #region SecurityLayer
2202.         if (accessToken.IsSysAdmin == false)
2203.         {
2204.             throw new DgException("Only a SysAdmin can do that!");
2205.         }
2206.         #endregion
2207.
2208.
2209.         var existingRecord = DigitDLL.GetContGradeByName(accessToken, contGrade.Name);
2210.         if (existingRecord != null && existingRecord.GradeId != contGrade.GradeId)
2211.         {
2212.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contGr
ade.Name));
2213.         }
2214.         if (existingRecord == null)
2215.         {
2216.             existingRecord = DigitDLL.GetContGradeById(accessToken, contGrade.GradeId);
2217.         }

```



```

2218.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "contGrade", contGrade.GradeId));
2219.
2220.
2221.         existingRecord.GradeId = contGrade.GradeId;
2222.         existingRecord.Name = contGrade.Name;
2223.         existingRecord.Description = contGrade.Description;
2224.
2225.         existingRecord.UpdateBy = accessToken.UserId;
2226.         existingRecord.UpdateDt = Utility.Now();
2227.
2228.         return DigitDLL.UpdateContGrade(accessToken, existingRecord);
2229.     }
2230.
2231.     public void DeleteContGrade(DgAccessToken accessToken, Int16 gradeId)
2232.     {
2233.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2234.         var existingRecord = DigitDLL.GetContGradeById(accessToken, gradeId);
2235.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "ContGrade", gradeId));
2236.
2237.         #region SecurityLayer
2238.
2239.         #endregion
2240.
2241.         //Can be deleted?
2242.         DigitDLL.CanDeleteContGrade(accessToken, existingRecord.GradeId);
2243.
2244.         DigitDLL.DeleteContGrade(accessToken, existingRecord.GradeId);
2245.     }
2246.     #endregion
2247.
2248.     #region DgContGrid
2249.     public DgContGrid GetContGridById(DgAccessToken accessToken, Int32 gridId)
2250.     {
2251.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
2252.
2253.         #region SecurityLayer
2254.         #endregion
2255.
2256.         return DigitDLL.GetContGridById(accessToken, gridId);
2257.     }
2258.
2259.     public IList<DgContGrid> GetContGrids(DgAccessToken accessToken, string whereClause =
null, string orderByClause = "order by Name")
2260.     {
2261.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
2262.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
2263.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
2264.
2265.         #region SecurityLayer
2266.         #endregion
2267.
2268.         return DigitDLL.GetContGrids(accessToken, whereClause, orderByClause);
2269.     }
2270.
2271.
2272.     public IList<DgContGrid> GetContGrids(DgAccessToken accessToken, int pageIndex, int p
ageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
2273.     {
2274.         if (accessToken == null) throw new ArgumentNullException("accessToken");

```

```

2275.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
2276.         ;
2277.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClause");
2278.         #region SecurityLayer
2279.
2280.         #endregion
2281.
2282.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
2283.         int startRowIndex = ((pageIndex - 1) * pageSize);
2284.
2285.         return DigitDLL.GetContGrids(accessToken, startRowIndex, pageSize, ref totalRows,
whereClause, orderByClause);
2286.     }
2287.
2288.     public DgContGrid CreateContGrid(DgAccessToken accessToken, DgContGrid contGrid)
2289.     {
2290.         #region input parameters validation
2291.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2292.         if (contGrid == null) throw new ArgumentNullException("contGrid");
2293.         contGrid.ValidateInstance();
2294.         #endregion
2295.
2296.         #region SecurityLayer
2297.         if (accessToken.IsSysAdmin == false)
2298.         {
2299.             throw new DgException("Only a SysAdmin can do that!");
2300.         }
2301.         #endregion
2302.
2303.         /*Check Name uniqueness*/
2304.         var existingRecord = DigitDLL.GetContGridByName(accessToken, contGrid.Name);
2305.         if (existingRecord != null)
2306.         {
2307.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contGrid.Name));
2308.         }
2309.
2310.         contGrid.CreateBy = contGrid.UpdateBy = accessToken.UserId;
2311.         contGrid.CreatedDt = contGrid.UpdatedDt = Utility.Now();
2312.
2313.         return DigitDLL.CreateContGrid(accessToken, contGrid);
2314.     }
2315.
2316.     public DgContGrid UpdateContGrid(DgAccessToken accessToken, DgContGrid contGrid)
2317.     {
2318.         #region input parameters validation
2319.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2320.         if (contGrid == null) throw new ArgumentNullException("contGrid");
2321.         contGrid.ValidateInstance();
2322.         #endregion
2323.
2324.         #region SecurityLayer
2325.         if (accessToken.IsSysAdmin == false)
2326.         {
2327.             throw new DgException("Only a SysAdmin can do that!");
2328.         }
2329.         #endregion
2330.
2331.
2332.         var existingRecord = DigitDLL.GetContGridByName(accessToken, contGrid.Name);
2333.         if (existingRecord != null && existingRecord.GridId != contGrid.GridId)

```

```

2334.         {
2335.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contGr
id.Name));
2336.         }
2337.         if (existingRecord == null)
2338.         {
2339.             existingRecord = DigitDLL.GetContGridById(accessToken, contGrid.GridId);
2340.         }
2341.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "contGrid", contGrid.GridId));
2342.
2343.
2344.             existingRecord.GridId = contGrid.GridId;
2345.             existingRecord.Name = contGrid.Name;
2346.             existingRecord.Description = contGrid.Description;
2347.
2348.             existingRecord.UpdateBy = accessToken.UserId;
2349.             existingRecord.UpdateDt = Utility.Now();
2350.
2351.             return DigitDLL.UpdateContGrid(accessToken, existingRecord);
2352.         }
2353.
2354.         public void DeleteContGrid(DgAccessToken accessToken, Int32 gridId)
2355.         {
2356.             if (accessToken == null) throw new ArgumentNullException("accessToken");
2357.             var existingRecord = DigitDLL.GetContGridById(accessToken, gridId);
2358.             if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "ContGrid", gridId));
2359.
2360.             #region SecurityLayer
2361.
2362.             #endregion
2363.
2364.             //Can be deleted?
2365.             DigitDLL.CanDeleteContGrid(accessToken, existingRecord.GridId);
2366.
2367.
2368.             DigitDLL.DeleteContGrid(accessToken, existingRecord.GridId);
2369.         }
2370.         #endregion
2371.
2372.         #region DgContMethod
2373.         public DgContMethod GetContMethodById(DgAccessToken accessToken, Int16 methodId)
2374.         {
2375.             if (accessToken == null) throw new ArgumentNullException("accesstoken");
2376.
2377.             #region SecurityLayer
2378.             #endregion
2379.
2380.             return DigitDLL.GetContMethodById(accessToken, methodId);
2381.         }
2382.
2383.         public IList<DgContMethod> GetContMethods(DgAccessToken accessToken, string whereClau
se = null, string orderByClause = "order by Name")
2384.         {
2385.             if (accessToken == null) throw new ArgumentNullException("accesstoken");
2386.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
2387.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
2388.
2389.             #region SecurityLayer
2390.             #endregion

```

```

2391.
2392.         return DigitDLL.GetContMethods(accessToken, whereClause, orderByClause);
2393.     }
2394.
2395.
2396.     public IList<DgContMethod> GetContMethods(DgAccessToken accessToken, int pageIndex, int
        pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
2397.     {
2398.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2399.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
        ;
2400.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
        se");
2401.
2402.         #region SecurityLayer
2403.
2404.         #endregion
2405.
2406.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
2407.         int startRowIndex = ((pageIndex - 1) * pageSize);
2408.
2409.         return DigitDLL.GetContMethods(accessToken, startRowIndex, pageSize, ref totalRow
        s, whereClause, orderByClause);
2410.     }
2411.
2412.     public DgContMethod CreateContMethod(DgAccessToken accessToken, DgContMethod contMeth
        od)
2413.     {
2414.         #region input parameters validation
2415.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2416.         if (contMethod == null) throw new ArgumentNullException("contMethod");
2417.         contMethod.ValidateInstance();
2418.         #endregion
2419.
2420.         #region SecurityLayer
2421.         if (accessToken.IsSysAdmin == false)
2422.         {
2423.             throw new DgException("Only a SysAdmin can do that!");
2424.         }
2425.         #endregion
2426.
2427.         /*Check Name uniqueness*/
2428.         var existingRecord = DigitDLL.GetContMethodByName(accessToken, contMethod.Name);
2429.
2430.         if (existingRecord != null)
2431.         {
2432.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contMe
        thod.Name));
2433.         }
2434.
2435.         contMethod.CreateBy = contMethod.UpdateBy = accessToken.UserId;
2436.         contMethod.CreateDt = contMethod.UpdateDt = Utility.Now();
2437.
2438.         return DigitDLL.CreateContMethod(accessToken, contMethod);
2439.     }
2440.     public DgContMethod UpdateContMethod(DgAccessToken accessToken, DgContMethod contMeth
        od)
2441.     {
2442.         #region input parameters validation
2443.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2444.         if (contMethod == null) throw new ArgumentNullException("contMethod");

```

```

2445.         contMethod.ValidateInstance();
2446.     #endregion
2447.
2448.     #region SecurityLayer
2449.     if (accessToken.IsSysAdmin == false)
2450.     {
2451.         throw new DgException("Only a SysAdmin can do that!");
2452.     }
2453.     #endregion
2454.
2455.
2456.     var existingRecord = DigitDLL.GetContMethodByName(accessToken, contMethod.Name);
2457.
2458.     if (existingRecord != null && existingRecord.MethodId != contMethod.MethodId)
2459.     {
2460.         throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contMe
2461. thod.Name));
2462.     }
2463.     if (existingRecord == null)
2464.     {
2465.         existingRecord = DigitDLL.GetContMethodById(accessToken, contMethod.MethodId)
2466. ;
2467.     }
2468.     if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
2469. m_with_id, "contMethod", contMethod.MethodId));
2470.
2471.     existingRecord.MethodId = contMethod.MethodId;
2472.     existingRecord.Name = contMethod.Name;
2473.     existingRecord.Description = contMethod.Description;
2474.
2475.     existingRecord.UpdateBy = accessToken.UserId;
2476.     existingRecord.UpdateDt = Utility.Now();
2477.
2478.     return DigitDLL.UpdateContMethod(accessToken, existingRecord);
2479. }
2480.
2481. public void DeleteContMethod(DgAccessToken accessToken, Int16 methodId)
2482. {
2483.     if (accessToken == null) throw new ArgumentNullException("accessToken");
2484.     var existingRecord = DigitDLL.GetContMethodById(accessToken, methodId);
2485.     if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
2486. m_with_id, "ContMethod", methodId));
2487.
2488.     #region SecurityLayer
2489.     #endregion
2490.
2491.     //Can be deleted?
2492.     DigitDLL.CanDeleteContMethod(accessToken, existingRecord.MethodId);
2493.
2494.     DigitDLL.DeleteContMethod(accessToken, existingRecord.MethodId);
2495. }
2496. #endregion
2497.
2498. #region DgContProfile
2499. public DgContProfile GetContProfileById(DgAccessToken accessToken, Int16 profileId)
2500. {
2501.     if (accessToken == null) throw new ArgumentNullException("accesstoken");
2502.
2503.     #region SecurityLayer
2504.     #endregion

```

```

2503.
2504.         return DigitDLL.GetContProfileById(accessToken, profileId);
2505.     }
2506.
2507.     public IList<DgContProfile> GetContProfiles(DgAccessToken accessToken, string whereCl
ause = null, string orderByClause = "order by Name")
2508.     {
2509.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
2510.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
2511.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
2512.
2513.         #region SecurityLayer
2514.         #endregion
2515.
2516.         return DigitDLL.GetContProfiles(accessToken, whereClause, orderByClause);
2517.     }
2518.
2519.
2520.     public IList<DgContProfile> GetContProfiles(DgAccessToken accessToken, int pageIndex,
int pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name
")
2521.     {
2522.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2523.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
2524.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
2525.
2526.         #region SecurityLayer
2527.         #endregion
2528.
2529.
2530.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
2531.         int startRowIndex = ((pageIndex - 1) * pageSize);
2532.
2533.         return DigitDLL.GetContProfiles(accessToken, startRowIndex, pageSize, ref totalRo
ws, whereClause, orderByClause);
2534.     }
2535.
2536.     public DgContProfile CreateContProfile(DgAccessToken accessToken, DgContProfile contP
rofile)
2537.     {
2538.         #region input parameters validation
2539.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2540.         if (contProfile == null) throw new ArgumentNullException("contProfile");
2541.         contProfile.ValidateInstance();
2542.         #endregion
2543.
2544.         #region SecurityLayer
2545.         if (accessToken.IsSysAdmin == false)
2546.         {
2547.             throw new DgException("Only a SysAdmin can do that!");
2548.         }
2549.         #endregion
2550.
2551.         /*Check Name uniqueness*/
2552.         var existingRecord = DigitDLL.GetContProfileByName(accessToken, contProfile.Name)
;
2553.         if (existingRecord != null)
2554.         {

```

```

2555.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contPr
ofile.Name));
2556.         }
2557.
2558.         contProfile.CreateBy = contProfile.UpdateBy = accessToken.UserId;
2559.         contProfile.CreateDt = contProfile.UpdateDt = Utility.Now();
2560.
2561.         return DigitDLL.CreateContProfile(accessToken, contProfile);
2562.     }
2563.
2564.     public DgContProfile UpdateContProfile(DgAccessToken accessToken, DgContProfile contP
rofile)
2565.     {
2566.         #region input parameters validation
2567.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2568.         if (contProfile == null) throw new ArgumentNullException("contProfile");
2569.         contProfile.ValidateInstance();
2570.         #endregion
2571.
2572.         #region SecurityLayer
2573.         if (accessToken.IsSysAdmin == false)
2574.         {
2575.             throw new DgException("Only a SysAdmin can do that!");
2576.         }
2577.         #endregion
2578.
2579.
2580.         var existingRecord = DigitDLL.GetContProfileByName(accessToken, contProfile.Name)
;
2581.         if (existingRecord != null && existingRecord.ProfileId != contProfile.ProfileId)
2582.         {
2583.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contPr
ofile.Name));
2584.         }
2585.         if (existingRecord == null)
2586.         {
2587.             existingRecord = DigitDLL.GetContProfileById(accessToken, contProfile.Profile
Id);
2588.         }
2589.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "contProfile", contProfile.ProfileId));
2590.
2591.
2592.         existingRecord.ProfileId = contProfile.ProfileId;
2593.         existingRecord.Name = contProfile.Name;
2594.         existingRecord.Description = contProfile.Description;
2595.
2596.         existingRecord.UpdateBy = accessToken.UserId;
2597.         existingRecord.UpdateDt = Utility.Now();
2598.
2599.         return DigitDLL.UpdateContProfile(accessToken, existingRecord);
2600.     }
2601.
2602.     public void DeleteContProfile(DgAccessToken accessToken, Int16 profileId)
2603.     {
2604.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2605.         var existingRecord = DigitDLL.GetContProfileById(accessToken, profileId);
2606.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "ContProfile", profileId));
2607.
2608.         #region SecurityLayer
2609.

```

```

2610.         #endregion
2611.
2612.         //Can be deleted?
2613.         DigitDLL.CanDeleteContProfile(accessToken, existingRecord.ProfileId);
2614.
2615.
2616.         DigitDLL.DeleteContProfile(accessToken, existingRecord.ProfileId);
2617.     }
2618.     #endregion
2619.
2620.     #region DgContSlope
2621.     public DgContSlope GetContSlopeById(DgAccessToken accessToken, Int16 slopeId)
2622.     {
2623.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
2624.
2625.         #region SecurityLayer
2626.         #endregion
2627.
2628.         return DigitDLL.GetContSlopeById(accessToken, slopeId);
2629.     }
2630.
2631.     public IList<DgContSlope> GetContSlopes(DgAccessToken accessToken, string whereClause
= null, string orderByClause = "order by Name")
2632.     {
2633.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
2634.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
2635.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
2636.
2637.         #region SecurityLayer
2638.         #endregion
2639.
2640.         return DigitDLL.GetContSlopes(accessToken, whereClause, orderByClause);
2641.     }
2642.
2643.
2644.     public IList<DgContSlope> GetContSlopes(DgAccessToken accessToken, int pageIndex, int
pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
2645.     {
2646.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2647.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
2648.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
2649.
2650.         #region SecurityLayer
2651.         #endregion
2652.
2653.
2654.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
2655.         int startRowIndex = ((pageIndex - 1) * pageSize);
2656.
2657.         return DigitDLL.GetContSlopes(accessToken, startRowIndex, pageSize, ref totalRows
, whereClause, orderByClause);
2658.     }
2659.
2660.     public DgContSlope CreateContSlope(DgAccessToken accessToken, DgContSlope contSlope)
2661.     {
2662.         #region input parameters validation
2663.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2664.         if (contSlope == null) throw new ArgumentNullException("contSlope");

```



```

2665.         contSlope.ValidateInstance();
2666.         #endregion
2667.
2668.         #region SecurityLayer
2669.         if (accessToken.IsSysAdmin == false)
2670.         {
2671.             throw new DgException("Only a SysAdmin can do that!");
2672.         }
2673.         #endregion
2674.
2675.         /*Check Name uniqueness*/
2676.         var existingRecord = DigitDLL.GetContSlopeByName(accessToken, contSlope.Name);
2677.         if (existingRecord != null)
2678.         {
2679.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contSl
2680. ope.Name));
2681.         }
2682.         contSlope.CreateBy = contSlope.UpdateBy = accessToken.UserId;
2683.         contSlope.CreateDt = contSlope.UpdateDt = Utility.Now();
2684.
2685.         return DigitDLL.CreateContSlope(accessToken, contSlope);
2686.     }
2687.
2688.     public DgContSlope UpdateContSlope(DgAccessToken accessToken, DgContSlope contSlope)
2689.     {
2690.         #region input parameters validation
2691.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2692.         if (contSlope == null) throw new ArgumentNullException("contSlope");
2693.         contSlope.ValidateInstance();
2694.         #endregion
2695.
2696.         #region SecurityLayer
2697.         if (accessToken.IsSysAdmin == false)
2698.         {
2699.             throw new DgException("Only a SysAdmin can do that!");
2700.         }
2701.         #endregion
2702.
2703.
2704.         var existingRecord = DigitDLL.GetContSlopeByName(accessToken, contSlope.Name);
2705.         if (existingRecord != null && existingRecord.SlopeId != contSlope.SlopeId)
2706.         {
2707.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contSl
2708. ope.Name));
2709.         }
2710.         if (existingRecord == null)
2711.         {
2712.             existingRecord = DigitDLL.GetContSlopeById(accessToken, contSlope.SlopeId);
2713.         }
2714.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
2715. m_with_id, "contSlope", contSlope.SlopeId));
2716.
2717.         existingRecord.SlopeId = contSlope.SlopeId;
2718.         existingRecord.Name = contSlope.Name;
2719.         existingRecord.Description = contSlope.Description;
2720.
2721.         existingRecord.UpdateBy = accessToken.UserId;
2722.         existingRecord.UpdateDt = Utility.Now();
2723.
2724.         return DigitDLL.UpdateContSlope(accessToken, existingRecord);

```

```

2724.     }
2725.
2726.     public void DeleteContSlope(DgAccessToken accessToken, Int16 slopeId)
2727.     {
2728.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2729.         var existingRecord = DigitDLL.GetContSlopeById(accessToken, slopeId);
2730.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "ContSlope", slopeId));
2731.
2732.         #region SecurityLayer
2733.
2734.         #endregion
2735.
2736.         //Can be deleted?
2737.         DigitDLL.CanDeleteContSlope(accessToken, existingRecord.SlopeId);
2738.
2739.
2740.         DigitDLL.DeleteContSlope(accessToken, existingRecord.SlopeId);
2741.     }
2742.     #endregion
2743.
2744.     #region DgContTexture
2745.     public DgContTexture GetContTextureById(DgAccessToken accessToken, Int16 textureId)
2746.     {
2747.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
2748.
2749.         #region SecurityLayer
2750.         #endregion
2751.
2752.         return DigitDLL.GetContTextureById(accessToken, textureId);
2753.     }
2754.
2755.     public IList<DgContTexture> GetContTextures(DgAccessToken accessToken, string whereCl
ause = null, string orderByClause = "order by Name")
2756.     {
2757.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
2758.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
2759.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
2760.
2761.         #region SecurityLayer
2762.         #endregion
2763.
2764.         return DigitDLL.GetContTextures(accessToken, whereClause, orderByClause);
2765.     }
2766.
2767.
2768.     public IList<DgContTexture> GetContTextures(DgAccessToken accessToken, int pageIndex,
int pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name
")
2769.     {
2770.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2771.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
2772.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
2773.
2774.         #region SecurityLayer
2775.         #endregion
2776.
2777.
2778.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);

```

```

2779.             int startRowIndex = ((pageIndex - 1) * pageSize);
2780.
2781.             return DigitDLL.GetContTextures(accessToken, startRowIndex, pageSize, ref totalRo
ws, whereClause, orderByClause);
2782.         }
2783.
2784.         public DgContTexture CreateContTexture(DgAccessToken accessToken, DgContTexture contT
exture)
2785.         {
2786.             #region input parameters validation
2787.             if (accessToken == null) throw new ArgumentNullException("accessToken");
2788.             if (contTexture == null) throw new ArgumentNullException("contTexture");
2789.             contTexture.ValidateInstance();
2790.             #endregion
2791.
2792.             #region SecurityLayer
2793.             if (accessToken.IsSysAdmin == false)
2794.             {
2795.                 throw new DgException("Only a SysAdmin can do that!");
2796.             }
2797.             #endregion
2798.
2799.             /*Check Name uniqueness*/
2800.             var existingRecord = DigitDLL.GetContTextureByName(accessToken, contTexture.Name)
;
2801.             if (existingRecord != null)
2802.             {
2803.                 throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contTe
xture.Name));
2804.             }
2805.
2806.             contTexture.CreateBy = contTexture.UpdateBy = accessToken.UserId;
2807.             contTexture.CreateDt = contTexture.UpdateDt = Utility.Now();
2808.
2809.             return DigitDLL.CreateContTexture(accessToken, contTexture);
2810.         }
2811.
2812.         public DgContTexture UpdateContTexture(DgAccessToken accessToken, DgContTexture contT
exture)
2813.         {
2814.             #region input parameters validation
2815.             if (accessToken == null) throw new ArgumentNullException("accessToken");
2816.             if (contTexture == null) throw new ArgumentNullException("contTexture");
2817.             contTexture.ValidateInstance();
2818.             #endregion
2819.
2820.             #region SecurityLayer
2821.             if (accessToken.IsSysAdmin == false)
2822.             {
2823.                 throw new DgException("Only a SysAdmin can do that!");
2824.             }
2825.             #endregion
2826.
2827.
2828.             var existingRecord = DigitDLL.GetContTextureByName(accessToken, contTexture.Name)
;
2829.             if (existingRecord != null && existingRecord.TextureId != contTexture.TextureId)
2830.             {
2831.                 throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contTe
xture.Name));
2832.             }
2833.             if (existingRecord == null)

```

```

2834.         {
2835.             existingRecord = DigitDLL.GetContTextureById(accessToken, contTexture.Texture
Id);
2836.         }
2837.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "contTexture", contTexture.TextureId));
2838.
2839.
2840.         existingRecord.TextureId = contTexture.TextureId;
2841.         existingRecord.Name = contTexture.Name;
2842.         existingRecord.Description = contTexture.Description;
2843.
2844.         existingRecord.UpdateBy = accessToken.UserId;
2845.         existingRecord.UpdateDt = Utility.Now();
2846.
2847.         return DigitDLL.UpdateContTexture(accessToken, existingRecord);
2848.     }
2849.
2850.     public void DeleteContTexture(DgAccessToken accessToken, Int16 textureId)
2851.     {
2852.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2853.         var existingRecord = DigitDLL.GetContTextureById(accessToken, textureId);
2854.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "ContTexture", textureId));
2855.
2856.         #region SecurityLayer
2857.
2858.         #endregion
2859.
2860.         //Can be deleted?
2861.         DigitDLL.CanDeleteContTexture(accessToken, existingRecord.TextureId);
2862.
2863.
2864.         DigitDLL.DeleteContTexture(accessToken, existingRecord.TextureId);
2865.     }
2866.     #endregion
2867.
2868.     #region DgContType
2869.     public DgContType GetContTypeById(DgAccessToken accessToken, Int16 typeId)
2870.     {
2871.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
2872.
2873.         #region SecurityLayer
2874.         #endregion
2875.
2876.         return DigitDLL.GetContTypeById(accessToken, typeId);
2877.     }
2878.
2879.     public IList<DgContType> GetContTypes(DgAccessToken accessToken, string whereClause =
null, string orderByClause = "order by Name")
2880.     {
2881.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
2882.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
2883.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
2884.
2885.         #region SecurityLayer
2886.         #endregion
2887.
2888.         return DigitDLL.GetContTypes(accessToken, whereClause, orderByClause);
2889.     }
2890.

```

```

2891.
2892.     public IList<DgContType> GetContTypes(DgAccessToken accessToken, int pageIndex, int p
    ageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
2893.     {
2894.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2895.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
    ;
2896.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
    se");
2897.
2898.         #region SecurityLayer
2899.
2900.         #endregion
2901.
2902.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
2903.         int startIndex = ((pageIndex - 1) * pageSize);
2904.
2905.         return DigitDLL.GetContTypes(accessToken, startIndex, pageSize, ref totalRows,
    whereClause, orderByClause);
2906.     }
2907.
2908.     public DgContType CreateContType(DgAccessToken accessToken, DgContType contType)
2909.     {
2910.         #region input parameters validation
2911.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2912.         if (contType == null) throw new ArgumentNullException("contType");
2913.         contType.ValidateInstance();
2914.         #endregion
2915.
2916.         #region SecurityLayer
2917.         if (accessToken.IsSysAdmin == false)
2918.         {
2919.             throw new DgException("Only a SysAdmin can do that!");
2920.         }
2921.         #endregion
2922.
2923.         /*Check Name uniqueness*/
2924.         var existingRecord = DigitDLL.GetContTypeByName(accessToken, contType.Name);
2925.         if (existingRecord != null)
2926.         {
2927.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contTy
    pe.Name));
2928.         }
2929.
2930.         contType.CreateBy = contType.UpdateBy = accessToken.UserId;
2931.         contType.CreateDt = contType.UpdateDt = Utility.Now();
2932.
2933.         return DigitDLL.CreateContType(accessToken, contType);
2934.     }
2935.
2936.     public DgContType UpdateContType(DgAccessToken accessToken, DgContType contType)
2937.     {
2938.         #region input parameters validation
2939.         if (accessToken == null) throw new ArgumentNullException("accessToken");
2940.         if (contType == null) throw new ArgumentNullException("contType");
2941.         contType.ValidateInstance();
2942.         #endregion
2943.
2944.         #region SecurityLayer
2945.         if (accessToken.IsSysAdmin == false)
2946.         {
2947.             throw new DgException("Only a SysAdmin can do that!");
2948.         }

```

```

2949.             #endregion
2950.
2951.
2952.             var existingRecord = DigitDLL.GetContTypeByName(accessToken, contType.Name);
2953.             if (existingRecord != null && existingRecord.TypeId != contType.TypeId)
2954.             {
2955.                 throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contTy
pe.Name));
2956.             }
2957.             if (existingRecord == null)
2958.             {
2959.                 existingRecord = DigitDLL.GetContTypeById(accessToken, contType.TypeId);
2960.             }
2961.             if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "contType", contType.TypeId));
2962.
2963.
2964.             existingRecord.TypeId = contType.TypeId;
2965.             existingRecord.Name = contType.Name;
2966.             existingRecord.Description = contType.Description;
2967.
2968.             existingRecord.UpdateBy = accessToken.UserId;
2969.             existingRecord.UpdateDt = Utility.Now();
2970.
2971.             return DigitDLL.UpdateContType(accessToken, existingRecord);
2972.         }
2973.
2974.         public void DeleteContType(DgAccessToken accessToken, Int16 typeId)
2975.         {
2976.             if (accessToken == null) throw new ArgumentNullException("accessToken");
2977.             var existingRecord = DigitDLL.GetContTypeById(accessToken, typeId);
2978.             if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "ContType", typeId));
2979.
2980.             #region SecurityLayer
2981.             if (accessToken.IsSysAdmin == false)
2982.             {
2983.                 throw new DgException("Only a SysAdmin can do that!");
2984.             }
2985.             #endregion
2986.
2987.             //Can be deleted?
2988.             DigitDLL.CanDeleteContType(accessToken, existingRecord.TypeId);
2989.
2990.
2991.             DigitDLL.DeleteContType(accessToken, existingRecord.TypeId);
2992.         }
2993.         #endregion
2994.
2995.
2996.
2997.         #region DgContFindCategory
2998.         public DgContFindCategory GetContFindCategoryById(DgAccessToken accessToken, Int16 ca
tegoryId)
2999.         {
3000.             if (accessToken == null) throw new ArgumentNullException("accesstoken");
3001.
3002.             #region SecurityLayer
3003.             #endregion
3004.
3005.             return DigitDLL.GetContFindCategoryById(accessToken, categoryId);
3006.         }
3007.

```

```

3008.         public IList<DgContFindCategory> GetContFindCategories(DgAccessToken accessToken, string
           whereClause=null, string orderByClause=null)
3009.         {
3010.             if (accessToken == null) throw new ArgumentNullException("accesstoken");
3011.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
           ;
3012.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClause");
3013.
3014.             #region SecurityLayer
3015.             #endregion
3016.
3017.             return DigitDLL.GetContFindCategories(accessToken, whereClause, orderByClause);
3018.         }
3019.
3020.
3021.         public IList<DgContFindCategory> GetContFindCategories(DgAccessToken accessToken, int
           pageIndex, int pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order
           by Name")
3022.         {
3023.             if (accessToken == null) throw new ArgumentNullException("accessToken");
3024.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
           ;
3025.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClause");
3026.
3027.             #region SecurityLayer
3028.             #endregion
3029.
3030.
3031.             Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
3032.             int startIndex = ((pageIndex - 1) * pageSize);
3033.
3034.             return DigitDLL.GetContFindCategories(accessToken, startIndex, pageSize, ref totalRows, whereClause, orderByClause);
3035.         }
3036.
3037.         public DgContFindCategory CreateContFindCategory(DgAccessToken accessToken, DgContFindCategory contFindCategory)
3038.         {
3039.             #region input parameters validation
3040.             if (accessToken == null) throw new ArgumentNullException("accessToken");
3041.             if (contFindCategory == null) throw new ArgumentNullException("contFindCategory");
           ;
3042.             contFindCategory.ValidateInstance();
3043.             #endregion
3044.
3045.             #region SecurityLayer
3046.             if (accessToken.IsSysAdmin == false)
3047.             {
3048.                 throw new DgException("Only a SysAdmin can do that!");
3049.             }
3050.             #endregion
3051.
3052.             /*Check Name uniqueness*/
3053.             var existingRecord = DigitDLL.GetContFindCategoryByName(accessToken, contFindCategory.Name);
3054.             if (existingRecord != null)
3055.             {
3056.                 throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contFindCategory.Name));
3057.             }
3058.

```

```

3059.         contFindCategory.CreateBy = contFindCategory.UpdateBy = accessToken.UserId;
3060.         contFindCategory.Createdt = contFindCategory.UpdateDt = Utility.Now();
3061.
3062.         return DigitDLL.CreateContFindCategory(accessToken, contFindCategory);
3063.     }
3064.
3065.     public DgContFindCategory UpdateContFindCategory(DgAccessToken accessToken, DgContFin
dCategory contFindCategory)
3066.     {
3067.         #region input parameters validation
3068.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3069.         if (contFindCategory == null) throw new ArgumentNullException("contFindCategory")
;
3070.         contFindCategory.ValidateInstance();
3071.         #endregion
3072.
3073.         #region SecurityLayer
3074.         if (accessToken.IsSysAdmin == false)
3075.         {
3076.             throw new DgException("Only a SysAdmin can do that!");
3077.         }
3078.         #endregion
3079.
3080.
3081.         var existingRecord = DigitDLL.GetContFindCategoryByName(accessToken, contFindCate
gory.Name);
3082.         if (existingRecord != null && existingRecord.CategoryId != contFindCategory.Categ
oryId)
3083.         {
3084.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contFi
ndCategory.Name));
3085.         }
3086.         if (existingRecord == null)
3087.         {
3088.             existingRecord = DigitDLL.GetContFindCategoryById(accessToken, contFindCatego
ry.CategoryId);
3089.         }
3090.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "contFindCategory", contFindCategory.CategoryId));
3091.
3092.
3093.         existingRecord.CategoryId = contFindCategory.CategoryId;
3094.         existingRecord.Name = contFindCategory.Name;
3095.         existingRecord.Description = contFindCategory.Description;
3096.
3097.         existingRecord.UpdateBy = accessToken.UserId;
3098.         existingRecord.UpdateDt = Utility.Now();
3099.
3100.         return DigitDLL.UpdateContFindCategory(accessToken, existingRecord);
3101.     }
3102.
3103.     public void DeleteContFindCategory(DgAccessToken accessToken, Int16 categoryId)
3104.     {
3105.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3106.         var existingRecord = DigitDLL.GetContFindCategoryById(accessToken, categoryId);
3107.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "ContFindCategory", categoryId));
3108.
3109.         #region SecurityLayer
3110.         if (accessToken.IsSysAdmin == false)
3111.         {
3112.             throw new DgException("Only a SysAdmin can do that!");
3113.         }

```



```

3114.         #endregion
3115.
3116.         //Can be deleted?
3117.         DigitDLL.CanDeleteContFindCategory(accessToken, existingRecord.CategoryId);
3118.
3119.
3120.         DigitDLL.DeleteContFindCategory(accessToken, existingRecord.CategoryId);
3121.     }
3122.     #endregion
3123.
3124.     #region DgContFindGroup
3125.     public DgContFindGroup GetContFindGroupId(DgAccessToken accessToken, Int16 grouping
3126. Id)
3127.     {
3128.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
3129.
3130.         #region SecurityLayer
3131.         #endregion
3132.         return DigitDLL.GetContFindGroupId(accessToken, groupingId);
3133.     }
3134.
3135.     public IList<DgContFindGroup> GetContFindGroups(DgAccessToken accessToken, string whe
3136. reClause = null, string orderByClause = null)
3137.     {
3138.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
3139.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
3140. ;
3141.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
3142. se");
3143.
3144.         #region SecurityLayer
3145.         #endregion
3146.         return DigitDLL.GetContFindGroups(accessToken, whereClause, orderByClause);
3147.     }
3148.
3149.     public IList<DgContFindGroup> GetContFindGroups(DgAccessToken accessToken, int pageIn
3150. dex, int pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by
3151. Name")
3152.     {
3153.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3154.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
3155. ;
3156.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
3157. se");
3158.
3159.         #region SecurityLayer
3160.         #endregion
3161.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
3162.         int startRowIndex = ((pageIndex - 1) * pageSize);
3163.
3164.         return DigitDLL.GetContFindGroups(accessToken, startRowIndex, pageSize, ref total
3165. Rows, whereClause, orderByClause);
3166.     }
3167.
3168.     public DgContFindGroup CreateContFindGroup(DgAccessToken accessToken, DgContFindGroup
3169. contFindGroup)
3170.     {
3171.         #region input parameters validation

```

```

3167.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3168.         if (contFindGroup == null) throw new ArgumentNullException("contFindGroup");
3169.         contFindGroup.ValidateInstance();
3170.         #endregion
3171.
3172.         #region SecurityLayer
3173.         if (accessToken.IsSysAdmin == false)
3174.         {
3175.             throw new DgException("Only a SysAdmin can do that!");
3176.         }
3177.         #endregion
3178.
3179.         /*Check Name uniqueness*/
3180.         var existingRecord = DigitDLL.GetContFindGroupByName(accessToken, contFindGroup.N
ame);
3181.         if (existingRecord != null)
3182.         {
3183.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contFi
ndGroup.Name));
3184.         }
3185.
3186.         contFindGroup.CreateBy = contFindGroup.UpdateBy = accessToken.UserId;
3187.         contFindGroup.CreateDt = contFindGroup.UpdateDt = Utility.Now();
3188.
3189.         return DigitDLL.CreateContFindGroup(accessToken, contFindGroup);
3190.     }
3191.
3192.     public DgContFindGroup UpdateContFindGroup(DgAccessToken accessToken, DgContFindGroup
contFindGroup)
3193.     {
3194.         #region input parameters validation
3195.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3196.         if (contFindGroup == null) throw new ArgumentNullException("contFindGroup");
3197.         contFindGroup.ValidateInstance();
3198.         #endregion
3199.
3200.         #region SecurityLayer
3201.         if (accessToken.IsSysAdmin == false)
3202.         {
3203.             throw new DgException("Only a SysAdmin can do that!");
3204.         }
3205.         #endregion
3206.
3207.
3208.         var existingRecord = DigitDLL.GetContFindGroupByName(accessToken, contFindGroup.N
ame);
3209.         if (existingRecord != null && existingRecord.GroupingId != contFindGroup.Grouping
Id)
3210.         {
3211.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contFi
ndGroup.Name));
3212.         }
3213.         if (existingRecord == null)
3214.         {
3215.             existingRecord = DigitDLL.GetContFindGroupById(accessToken, contFindGroup.Gro
upingId);
3216.         }
3217.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "contFindGroup", contFindGroup.GroupingId));
3218.
3219.
3220.         existingRecord.GroupingId = contFindGroup.GroupingId;
3221.         existingRecord.Name = contFindGroup.Name;

```

```

3222.         existingRecord.Description = contFindGroup.Description;
3223.
3224.         existingRecord.UpdateBy = accessToken.UserId;
3225.         existingRecord.UpdateDt = Utility.Now();
3226.
3227.         return DigitDLL.UpdateContFindGroup(accessToken, existingRecord);
3228.     }
3229.
3230.     public void DeleteContFindGroup(DgAccessToken accessToken, Int16 groupingId)
3231.     {
3232.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3233.         var existingRecord = DigitDLL.GetContFindGroupById(accessToken, groupingId);
3234.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "ContFindGroup", groupingId));
3235.
3236.         #region SecurityLayer
3237.         if (accessToken.IsSysAdmin == false)
3238.         {
3239.             throw new DgException("Only a SysAdmin can do that!");
3240.         }
3241.         #endregion
3242.
3243.         //Can be deleted?
3244.         DigitDLL.CanDeleteContFindGroup(accessToken, existingRecord.GroupingId);
3245.
3246.         DigitDLL.DeleteContFindGroup(accessToken, existingRecord.GroupingId);
3247.     }
3248.     #endregion
3249.
3250.
3251.     #region DgContFindType & DgContFindSubtype
3252.     public DgContFindType GetContFindTypeById(DgAccessToken accessToken, Int16 findTypeId
)
3253.     {
3254.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
3255.
3256.         #region SecurityLayer
3257.         #endregion
3258.
3259.         return DigitDLL.GetContFindTypeById(accessToken, findTypeId);
3260.     }
3261.
3262.     public IList<DgContFindType> GetContFindTypes(DgAccessToken accessToken, string where
Clause = null, string orderByClause = null)
3263.     {
3264.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
3265.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
3266.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
3267.
3268.         #region SecurityLayer
3269.         #endregion
3270.
3271.         return DigitDLL.GetContFindTypes(accessToken, whereClause, orderByClause);
3272.     }
3273.
3274.     public IList<DgContFindType> GetContFindTypes(DgAccessToken accessToken, int pageInde
x, int pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Na
me")
3275.     {
3276.         if (accessToken == null) throw new ArgumentNullException("accessToken");

```

```

3277.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
3278.         ;
3279.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClause");
3280.         #region SecurityLayer
3281.
3282.         #endregion
3283.
3284.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
3285.         int startRowIndex = ((pageIndex - 1) * pageSize);
3286.
3287.         return DigitDLL.GetContFindTypes(accessToken, startRowIndex, pageSize, ref totalRows, whereClause, orderByClause);
3288.     }
3289.
3290.     public DgContFindType CreateContFindType(DgAccessToken accessToken, DgContFindType contFindType)
3291.     {
3292.         #region input parameters validation
3293.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3294.         if (contFindType == null) throw new ArgumentNullException("contFindType");
3295.         contFindType.ValidateInstance();
3296.         #endregion
3297.
3298.         #region SecurityLayer
3299.         if (accessToken.IsSysAdmin == false)
3300.         {
3301.             throw new DgException("Only a SysAdmin can do that!");
3302.         }
3303.         #endregion
3304.
3305.         /*Check Name uniqueness*/
3306.         var existingRecord = DigitDLL.GetContFindTypeByName(accessToken, contFindType.Name);
3307.         if (existingRecord != null)
3308.         {
3309.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contFindType.Name));
3310.         }
3311.
3312.         contFindType.CreateBy = contFindType.UpdateBy = accessToken.UserId;
3313.         contFindType.CreateDt = contFindType.UpdateDt = Utility.Now();
3314.
3315.         return DigitDLL.CreateContFindType(accessToken, contFindType);
3316.     }
3317.
3318.     public DgContFindType UpdateContFindType(DgAccessToken accessToken, DgContFindType contFindType)
3319.     {
3320.         #region input parameters validation
3321.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3322.         if (contFindType == null) throw new ArgumentNullException("contFindType");
3323.         contFindType.ValidateInstance();
3324.         #endregion
3325.
3326.         #region SecurityLayer
3327.         if (accessToken.IsSysAdmin == false)
3328.         {
3329.             throw new DgException("Only a SysAdmin can do that!");
3330.         }
3331.         #endregion
3332.

```

```

3333.
3334.         var existingRecord = DigitDLL.GetContFindTypeByName(accessToken, contFindType.Nam
e);
3335.         if (existingRecord != null && existingRecord.FindTypeId != contFindType.FindTypeI
d)
3336.         {
3337.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", contFi
ndType.Name));
3338.         }
3339.         if (existingRecord == null)
3340.         {
3341.             existingRecord = DigitDLL.GetContFindTypeById(accessToken, contFindType.FindT
ypeId);
3342.         }
3343.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "contFindType", contFindType.FindTypeId));
3344.
3345.
3346.         existingRecord.Code = contFindType.Code;
3347.         existingRecord.Name = contFindType.Name;
3348.         existingRecord.Description = contFindType.Description;
3349.
3350.         existingRecord.UpdateBy = accessToken.UserId;
3351.         existingRecord.UpdateDt = Utility.Now();
3352.
3353.         return DigitDLL.UpdateContFindType(accessToken, existingRecord);
3354.     }
3355.
3356.     public void DeleteContFindType(DgAccessToken accessToken, Int16 findTypeId)
3357.     {
3358.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3359.         var existingRecord = DigitDLL.GetContFindTypeById(accessToken, findTypeId);
3360.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "ContFindType", findTypeId));
3361.
3362.         #region SecurityLayer
3363.         if (accessToken.IsSysAdmin == false)
3364.         {
3365.             throw new DgException("Only a SysAdmin can do that!");
3366.         }
3367.         #endregion
3368.
3369.         //Can be deleted?
3370.         DigitDLL.CanDeleteContFindType(accessToken, existingRecord.FindTypeId);
3371.
3372.
3373.         DigitDLL.DeleteContFindType(accessToken, existingRecord.FindTypeId);
3374.     }
3375.
3376.
3377.     public DgContFindSubtype GetContFindSubtypeById(DgAccessToken accessToken, Int16 find
TypeId)
3378.     {
3379.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
3380.
3381.         #region SecurityLayer
3382.         #endregion
3383.
3384.         return DigitDLL.GetContFindSubtypeById(accessToken, findTypeId);
3385.     }
3386.     public IList<DgContFindSubtype> GetContFindSubtypes(DgAccessToken accessToken, Int16
findType, string whereClause = null, string orderByClause = null)
3387.     {

```

```

3388.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
3389.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
3390.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
3391.
3392.         #region SecurityLayer
3393.         #endregion
3394.
3395.         return DigitDLL.GetContFindSubtypes(accessToken, findType, whereClause, orderByCl
ause);
3396.     }
3397.     public IList<DgContFindSubtype> GetContFindSubtypes(DgAccessToken accessToken, Int16
findType, int pageIndex, int pageSize, ref int totalRows, string whereClause = null, string orderB
yClause = null)
3398.     {
3399.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3400.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
3401.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
3402.
3403.         #region SecurityLayer
3404.         #endregion
3405.
3406.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
3407.         int startRowIndex = ((pageIndex - 1) * pageSize);
3408.
3409.         return DigitDLL.GetContFindSubtypes(accessToken, findType, startRowIndex, pageSiz
e, ref totalRows, whereClause, orderByClause);
3410.     }
3411. }
3412.
3413.     public DgContFindSubtype CreateContFindSubtype(DgAccessToken accessToken, DgContFindS
ubtype cntFindSubtype)
3414.     {
3415.         #region input parameters validation
3416.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3417.         if (cntFindSubtype == null) throw new ArgumentNullException("cntFindSubtype");
3418.         cntFindSubtype.ValidateInstance();
3419.         #endregion
3420.
3421.         #region SecurityLayer
3422.         if (accessToken.IsSysAdmin == false)
3423.         {
3424.             throw new DgException("Only a SysAdmin can do that!");
3425.         }
3426.         #endregion
3427.
3428.         /*Υπάρχει το ContFindType*/
3429.         var contFindType = DigitDLL.GetContFindTypeById(accessToken, cntFindSubtype.FindT
ype);
3430.         if (contFindType == null)
3431.         {
3432.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContFindType
", cntFindSubtype.FindType));
3433.         }
3434.
3435.         /*Check Name uniqueness*/
3436.         var existingRecord = DigitDLL.GetContFindSubtypeByName(accessToken, cntFindSubtyp
e.FindType, cntFindSubtype.Name);
3437.         if (existingRecord != null)
3438.         {

```

```

3439.         throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", cntFin
dSubtype.Name));
3440.     }
3441.
3442.     cntFindSubtype.CreateBy = cntFindSubtype.UpdateBy = accessToken.UserId;
3443.     cntFindSubtype.CreateDt = cntFindSubtype.UpdateDt = Utility.Now();
3444.
3445.     return DigitDLL.CreateContFindSubtype(accessToken, cntFindSubtype);
3446. }
3447. public DgContFindSubtype UpdateContFindSubtype(DgAccessToken accessToken, DgContFindS
ubtype cntFindSubtype)
3448. {
3449.     #region input parameters validation
3450.     if (accessToken == null) throw new ArgumentNullException("accessToken");
3451.     if (cntFindSubtype == null) throw new ArgumentNullException("cntFindSubtype");
3452.     cntFindSubtype.ValidateInstance();
3453.     #endregion
3454.
3455.     #region SecurityLayer
3456.     if (accessToken.IsSysAdmin == false)
3457.     {
3458.         throw new DgException("Only a SysAdmin can do that!");
3459.     }
3460.     #endregion
3461.
3462.
3463.     var existingRecord = DigitDLL.GetContFindSubtypeByName(accessToken, cntFindSubtyp
e.FindType, cntFindSubtype.Name);
3464.     if (existingRecord != null && existingRecord.SubTypeId != cntFindSubtype.SubTypeI
d)
3465.     {
3466.         throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", cntFin
dSubtype.Name));
3467.     }
3468.     if (existingRecord == null)
3469.     {
3470.         existingRecord = DigitDLL.GetContFindSubtypeById(accessToken, cntFindSubtype.
SubTypeId);
3471.     }
3472.     if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "ContFindSubtype", cntFindSubtype.SubTypeId));
3473.
3474.
3475.     existingRecord.Code = cntFindSubtype.Code;
3476.     existingRecord.Name = cntFindSubtype.Name;
3477.     existingRecord.Description = cntFindSubtype.Description;
3478.
3479.     existingRecord.UpdateBy = accessToken.UserId;
3480.     existingRecord.UpdateDt = Utility.Now();
3481.
3482.     return DigitDLL.UpdateContFindSubtype(accessToken, cntFindSubtype);
3483. }
3484. public void DeleteContFindSubtype(DgAccessToken accessToken, Int16 subtypeid)
3485. {
3486.     if (accessToken == null) throw new ArgumentNullException("accessToken");
3487.     var existingRecord = DigitDLL.GetContFindSubtypeById(accessToken, subtypeid);
3488.     if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "ContFindSubtype", subtypeid));
3489.
3490.     #region SecurityLayer
3491.     if (accessToken.IsSysAdmin == false)
3492.     {
3493.         throw new DgException("Only a SysAdmin can do that!");

```

```

3494.         }
3495.         #endregion
3496.
3497.         //Can be deleted?
3498.         DigitDLL.CanDeleteContFindSubtype(accessToken, existingRecord.SubTypeId);
3499.
3500.
3501.         DigitDLL.DeleteContFindSubtype(accessToken, existingRecord.SubTypeId);
3502.     }
3503.
3504.     #endregion
3505.
3506.
3507.     #region DgStratumColor
3508.     public DgStratumColor GetStratumColorById(DgAccessToken accessToken, Int16 colorId)
3509.     {
3510.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
3511.
3512.         #region SecurityLayer
3513.         #endregion
3514.
3515.         return DigitDLL.GetStratumColorById(accessToken, colorId);
3516.     }
3517.
3518.     public IList<DgStratumColor> GetStratumColors(DgAccessToken accessToken, string where
    Clause = null, string orderByClause = null)
3519.     {
3520.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
3521.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
    ;
3522.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
    se");
3523.
3524.         #region SecurityLayer
3525.         #endregion
3526.
3527.         return DigitDLL.GetStratumColors(accessToken, whereClause, orderByClause);
3528.     }
3529.
3530.
3531.     public IList<DgStratumColor> GetStratumColors(DgAccessToken accessToken, int pageInde
    x, int pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Na
    me")
3532.     {
3533.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3534.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
    ;
3535.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
    se");
3536.
3537.         #region SecurityLayer
3538.         #endregion
3539.
3540.
3541.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
3542.         int startRowIndex = ((pageIndex - 1) * pageSize);
3543.
3544.         return DigitDLL.GetStratumColors(accessToken, startRowIndex, pageSize, ref totalR
    ows, whereClause, orderByClause);
3545.     }
3546.
3547.     public DgStratumColor CreateStratumColor(DgAccessToken accessToken, DgStratumColor st
    ratumColor)

```



```

3548.         {
3549.             #region input parameters validation
3550.             if (accessToken == null) throw new ArgumentNullException("accessToken");
3551.             if (stratumColor == null) throw new ArgumentNullException("stratumColor");
3552.             stratumColor.ValidateInstance();
3553.             #endregion
3554.
3555.             #region SecurityLayer
3556.             if (accessToken.IsSysAdmin == false)
3557.             {
3558.                 throw new DgException("Only a SysAdmin can do that!");
3559.             }
3560.             #endregion
3561.
3562.             /*Check Name uniqueness*/
3563.             var existingRecord = DigitDLL.GetStratumColorByName(accessToken, stratumColor.Name);
3564.             if (existingRecord != null)
3565.             {
3566.                 throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", stratumColor.Name));
3567.             }
3568.
3569.             stratumColor.CreateBy = stratumColor.UpdateBy = accessToken.UserId;
3570.             stratumColor.Createdt = stratumColor.Updatedt = Utility.Now();
3571.
3572.             return DigitDLL.CreateStratumColor(accessToken, stratumColor);
3573.         }
3574.
3575.         public DgStratumColor UpdateStratumColor(DgAccessToken accessToken, DgStratumColor stratumColor)
3576.         {
3577.             #region input parameters validation
3578.             if (accessToken == null) throw new ArgumentNullException("accessToken");
3579.             if (stratumColor == null) throw new ArgumentNullException("stratumColor");
3580.             stratumColor.ValidateInstance();
3581.             #endregion
3582.
3583.             #region SecurityLayer
3584.             if (accessToken.IsSysAdmin == false)
3585.             {
3586.                 throw new DgException("Only a SysAdmin can do that!");
3587.             }
3588.             #endregion
3589.
3590.
3591.             var existingRecord = DigitDLL.GetStratumColorByName(accessToken, stratumColor.Name);
3592.             if (existingRecord != null && existingRecord.ColorId != stratumColor.ColorId)
3593.             {
3594.                 throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", stratumColor.Name));
3595.             }
3596.             if (existingRecord == null)
3597.             {
3598.                 existingRecord = DigitDLL.GetStratumColorById(accessToken, stratumColor.ColorId);
3599.             }
3600.             if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "stratumColor", stratumColor.ColorId));
3601.
3602.
3603.             existingRecord.ColorId = stratumColor.ColorId;

```

```

3604.         existingRecord.Name = stratumColor.Name;
3605.         existingRecord.Description = stratumColor.Description;
3606.
3607.         existingRecord.UpdateBy = accessToken.UserId;
3608.         existingRecord.UpdateDt = Utility.Now();
3609.
3610.         return DigitDLL.UpdateStratumColor(accessToken, existingRecord);
3611.     }
3612.
3613.     public void DeleteStratumColor(DgAccessToken accessToken, Int16 colorId)
3614.     {
3615.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3616.         var existingRecord = DigitDLL.GetStratumColorById(accessToken, colorId);
3617.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "StratumColor", colorId));
3618.
3619.         #region SecurityLayer
3620.         if (accessToken.IsSysAdmin == false)
3621.         {
3622.             throw new DgException("Only a SysAdmin can do that!");
3623.         }
3624.         #endregion
3625.
3626.         //Can be deleted?
3627.         DigitDLL.CanDeleteStratumColor(accessToken, existingRecord.ColorId);
3628.
3629.
3630.         DigitDLL.DeleteStratumColor(accessToken, existingRecord.ColorId);
3631.     }
3632. #endregion
3633.
3634. #region DgDatingType
3635. public DgDatingType GetDatingTypeById(DgAccessToken accessToken, Int16 typeId)
3636. {
3637.     if (accessToken == null) throw new ArgumentNullException("accesstoken");
3638.
3639.     #region SecurityLayer
3640.     #endregion
3641.
3642.     return DigitDLL.GetDatingTypeById(accessToken, typeId);
3643. }
3644.
3645. public IList<DgDatingType> GetDatingTypes(DgAccessToken accessToken, string whereClau
se = null, string orderByClause = null)
3646. {
3647.     if (accessToken == null) throw new ArgumentNullException("accesstoken");
3648.     Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
3649.     Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
3650.
3651.     #region SecurityLayer
3652.     #endregion
3653.
3654.     return DigitDLL.GetDatingTypes(accessToken, whereClause, orderByClause);
3655. }
3656.
3657.
3658. public IList<DgDatingType> GetDatingTypes(DgAccessToken accessToken, int pageIndex, i
nt pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
3659. {
3660.     if (accessToken == null) throw new ArgumentNullException("accessToken");

```

```

3661.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
3662.         ;
3663.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClause");
3664.         #region SecurityLayer
3665.
3666.         #endregion
3667.
3668.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
3669.         int startRowIndex = ((pageIndex - 1) * pageSize);
3670.
3671.         return DigitDLL.GetDatingTypes(accessToken, startRowIndex, pageSize, ref totalRows, whereClause, orderByClause);
3672.     }
3673.
3674.     public DgDatingType CreateDatingType(DgAccessToken accessToken, DgDatingType datingType)
3675.     {
3676.         #region input parameters validation
3677.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3678.         if (datingType == null) throw new ArgumentNullException("datingType");
3679.         datingType.ValidateInstance();
3680.         #endregion
3681.
3682.         #region SecurityLayer
3683.         if (accessToken.IsSysAdmin == false)
3684.         {
3685.             throw new DgException("Only a SysAdmin can do that!");
3686.         }
3687.         #endregion
3688.
3689.         /*Check Name uniqueness*/
3690.         var existingRecord = DigitDLL.GetDatingTypeByName(accessToken, datingType.Name);
3691.
3692.         if (existingRecord != null)
3693.         {
3694.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", datingType.Name));
3695.         }
3696.
3697.         datingType.CreateBy = datingType.UpdateBy = accessToken.UserId;
3698.         datingType.CreateDt = datingType.UpdateDt = Utility.Now();
3699.
3700.         return DigitDLL.CreateDatingType(accessToken, datingType);
3701.     }
3702.     public DgDatingType UpdateDatingType(DgAccessToken accessToken, DgDatingType datingType)
3703.     {
3704.         #region input parameters validation
3705.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3706.         if (datingType == null) throw new ArgumentNullException("datingType");
3707.         datingType.ValidateInstance();
3708.         #endregion
3709.
3710.         #region SecurityLayer
3711.         if (accessToken.IsSysAdmin == false)
3712.         {
3713.             throw new DgException("Only a SysAdmin can do that!");
3714.         }
3715.         #endregion
3716.

```

```

3717.
3718.         var existingRecord = DigitDLL.GetDatingTypeByName(accessToken, datingType.Name);
3719.         if (existingRecord != null && existingRecord.TypeId != datingType.TypeId)
3720.         {
3721.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", dating
Type.Name));
3722.         }
3723.         if (existingRecord == null)
3724.         {
3725.             existingRecord = DigitDLL.GetDatingTypeById(accessToken, datingType.TypeId);
3726.         }
3727.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "datingType", datingType.TypeId));
3728.
3729.
3730.         existingRecord.TypeId = datingType.TypeId;
3731.         existingRecord.Name = datingType.Name;
3732.         existingRecord.Description = datingType.Description;
3733.
3734.         existingRecord.UpdateBy = accessToken.UserId;
3735.         existingRecord.UpdateDt = Utility.Now();
3736.
3737.         return DigitDLL.UpdateDatingType(accessToken, existingRecord);
3738.     }
3739.
3740.     public void DeleteDatingType(DgAccessToken accessToken, Int16 typeId)
3741.     {
3742.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3743.         var existingRecord = DigitDLL.GetDatingTypeById(accessToken, typeId);
3744.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "DatingType", typeId));
3745.
3746.         #region SecurityLayer
3747.         if (accessToken.IsSysAdmin == false)
3748.         {
3749.             throw new DgException("Only a SysAdmin can do that!");
3750.         }
3751.         #endregion
3752.
3753.         //Can be deleted?
3754.         DigitDLL.CanDeleteDatingType(accessToken, existingRecord.TypeId);
3755.
3756.
3757.         DigitDLL.DeleteDatingType(accessToken, existingRecord.TypeId);
3758.     }
3759.     #endregion
3760.
3761.     #region DgDatingMethod
3762.     public DgDatingMethod GetDatingMethodById(DgAccessToken accessToken, Int16 methodId)
3763.     {
3764.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
3765.
3766.         #region SecurityLayer
3767.         #endregion
3768.
3769.         return DigitDLL.GetDatingMethodById(accessToken, methodId);
3770.     }
3771.
3772.     public IList<DgDatingMethod> GetDatingMethods(DgAccessToken accessToken, string where
Clause = null, string orderByClause=null)

```

```

3773.         {
3774.             if (accessToken == null) throw new ArgumentNullException("accesstoken");
3775.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
3776.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
3777.
3778.             #region SecurityLayer
3779.             #endregion
3780.
3781.             return DigitDLL.GetDatingMethods(accessToken, whereClause, orderByClause);
3782.         }
3783.
3784.
3785.         public IList<DgDatingMethod> GetDatingMethods(DgAccessToken accessToken, int pageInde
x, int pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Na
me")
3786.         {
3787.             if (accessToken == null) throw new ArgumentNullException("accessToken");
3788.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
3789.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
3790.
3791.             #region SecurityLayer
3792.
3793.             #endregion
3794.
3795.             Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
3796.             int startRowIndex = ((pageIndex - 1) * pageSize);
3797.
3798.             return DigitDLL.GetDatingMethods(accessToken, startRowIndex, pageSize, ref totalR
ows, whereClause, orderByClause);
3799.         }
3800.
3801.         public DgDatingMethod CreateDatingMethod(DgAccessToken accessToken, DgDatingMethod da
tingMethod)
3802.         {
3803.             #region input parameters validation
3804.             if (accessToken == null) throw new ArgumentNullException("accessToken");
3805.             if (datingMethod == null) throw new ArgumentNullException("datingMethod");
3806.             datingMethod.ValidateInstance();
3807.             #endregion
3808.
3809.             #region SecurityLayer
3810.             if (accessToken.IsSysAdmin == false)
3811.             {
3812.                 throw new DgException("Only a SysAdmin can do that!");
3813.             }
3814.             #endregion
3815.
3816.             /*Check Name uniqueness*/
3817.             var existingRecord = DigitDLL.GetDatingMethodByName(accessToken, datingMethod.Nam
e);
3818.             if (existingRecord != null)
3819.             {
3820.                 throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", dating
Method.Name));
3821.             }
3822.
3823.             datingMethod.CreateBy = datingMethod.UpdateBy = accessToken.UserId;
3824.             datingMethod.CreateDt = datingMethod.UpdateDt = Utility.Now();
3825.

```

```

3826.         return DigitDLL.CreateDatingMethod(accessToken, datingMethod);
3827.     }
3828.
3829.     public DgDatingMethod UpdateDatingMethod(DgAccessToken accessToken, DgDatingMethod da
tingMethod)
3830.     {
3831.         #region input parameters validation
3832.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3833.         if (datingMethod == null) throw new ArgumentNullException("datingMethod");
3834.         datingMethod.ValidateInstance();
3835.         #endregion
3836.
3837.         #region SecurityLayer
3838.         if (accessToken.IsSysAdmin == false)
3839.         {
3840.             throw new DgException("Only a SysAdmin can do that!");
3841.         }
3842.         #endregion
3843.
3844.
3845.         var existingRecord = DigitDLL.GetDatingMethodByName(accessToken, datingMethod.Nam
e);
3846.         if (existingRecord != null && existingRecord.MethodId != datingMethod.MethodId)
3847.         {
3848.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", dating
Method.Name));
3849.         }
3850.         if (existingRecord == null)
3851.         {
3852.             existingRecord = DigitDLL.GetDatingMethodById(accessToken, datingMethod.Metho
dId);
3853.         }
3854.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "datingMethod", datingMethod.MethodId));
3855.
3856.
3857.         existingRecord.MethodId = datingMethod.MethodId;
3858.         existingRecord.Name = datingMethod.Name;
3859.         existingRecord.Description = datingMethod.Description;
3860.
3861.         existingRecord.UpdateBy = accessToken.UserId;
3862.         existingRecord.UpdateDt = Utility.Now();
3863.
3864.         return DigitDLL.UpdateDatingMethod(accessToken, existingRecord);
3865.     }
3866.
3867.     public void DeleteDatingMethod(DgAccessToken accessToken, Int16 methodId)
3868.     {
3869.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3870.         var existingRecord = DigitDLL.GetDatingMethodById(accessToken, methodId);
3871.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "DatingMethod", methodId));
3872.
3873.         #region SecurityLayer
3874.         if (accessToken.IsSysAdmin == false)
3875.         {
3876.             throw new DgException("Only a SysAdmin can do that!");
3877.         }
3878.         #endregion
3879.
3880.         //Can be deleted?
3881.         DigitDLL.CanDeleteDatingMethod(accessToken, existingRecord.MethodId);
3882.

```

```

3883.
3884.         DigitDLL.DeleteDatingMethod(accessToken, existingRecord.MethodId);
3885.     }
3886.     #endregion
3887.
3888.     #region DgDating
3889.     public DgDating GetDatingById(DgAccessToken accessToken, Int16 datingId)
3890.     {
3891.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
3892.
3893.         #region SecurityLayer
3894.         #endregion
3895.
3896.         return DigitDLL.GetDatingById(accessToken, datingId);
3897.     }
3898.
3899.     public IList<DgDating> GetDatings(DgAccessToken accessToken, string whereClause = nul
1, string orderByClause = null)
3900.     {
3901.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
3902.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
3903.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
3904.
3905.         #region SecurityLayer
3906.         #endregion
3907.
3908.         return DigitDLL.GetDatings(accessToken, whereClause, orderByClause);
3909.     }
3910.
3911.
3912.     public IList<DgDating> GetDatings(DgAccessToken accessToken, int pageIndex, int pageS
ize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
3913.     {
3914.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3915.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
3916.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
3917.
3918.         #region SecurityLayer
3919.         #endregion
3920.
3921.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
3922.         int startRowIndex = ((pageIndex - 1) * pageSize);
3923.
3924.         return DigitDLL.GetDatings(accessToken, startRowIndex, pageSize, ref totalRows, w
hereClause, orderByClause);
3925.     }
3926.
3927.
3928.     public DgDating CreateDating(DgAccessToken accessToken, DgDating dating)
3929.     {
3930.         #region input parameters validation
3931.         if (accessToken == null) throw new ArgumentNullException("accessToken");
3932.         if (dating == null) throw new ArgumentNullException("dating");
3933.         dating.ValidateInstance();
3934.         #endregion
3935.
3936.         #region SecurityLayer
3937.         if (accessToken.IsSysAdmin == false)
3938.         {

```

```

3939.             throw new DgException("Only a SysAdmin can do that!");
3940.         }
3941.     #endregion
3942.
3943.     /*Check Name uniqueness*/
3944.     var existingRecord = DigitDLL.GetDatingByName(accessToken, dating.Name);
3945.     if (existingRecord != null)
3946.     {
3947.         throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", dating
3948. .Name));
3949.     }
3950.     dating.CreateBy = dating.UpdateBy = accessToken.UserId;
3951.     dating.CreateDt = dating.UpdateDt = Utility.Now();
3952.
3953.     return DigitDLL.CreateDating(accessToken, dating);
3954. }
3955.
3956. public DgDating UpdateDating(DgAccessToken accessToken, DgDating dating)
3957. {
3958.     #region input parameters validation
3959.     if (accessToken == null) throw new ArgumentNullException("accessToken");
3960.     if (dating == null) throw new ArgumentNullException("dating");
3961.     dating.ValidateInstance();
3962.     #endregion
3963.
3964.     #region SecurityLayer
3965.     if (accessToken.IsSysAdmin == false)
3966.     {
3967.         throw new DgException("Only a SysAdmin can do that!");
3968.     }
3969.     #endregion
3970.
3971.
3972.     var existingRecord = DigitDLL.GetDatingByName(accessToken, dating.Name);
3973.     if (existingRecord != null && existingRecord.DatingId != dating.DatingId)
3974.     {
3975.         throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", dating
3976. .Name));
3977.     }
3978.     if (existingRecord == null)
3979.     {
3980.         existingRecord = DigitDLL.GetDatingById(accessToken, dating.DatingId);
3981.     }
3982.     if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
3983. m_with_id, "dating", dating.DatingId));
3984.
3985.     existingRecord.DatingId = dating.DatingId;
3986.     existingRecord.Name = dating.Name;
3987.     existingRecord.Description = dating.Description;
3988.
3989.     existingRecord.UpdateBy = accessToken.UserId;
3990.     existingRecord.UpdateDt = Utility.Now();
3991.
3992.     return DigitDLL.UpdateDating(accessToken, existingRecord);
3993. }
3994.
3995. public void DeleteDating(DgAccessToken accessToken, Int16 datingId)
3996. {
3997.     if (accessToken == null) throw new ArgumentNullException("accessToken");
3998.     var existingRecord = DigitDLL.GetDatingById(accessToken, datingId);

```



```

3998.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "Dating", datingId));
3999.
4000.         #region SecurityLayer
4001.         if (accessToken.IsSysAdmin == false)
4002.         {
4003.             throw new DgException("Only a SysAdmin can do that!");
4004.         }
4005.         #endregion
4006.
4007.         //Can be deleted?
4008.         DigitDLL.CanDeleteDating(accessToken, existingRecord.DatingId);
4009.
4010.
4011.         DigitDLL.DeleteDating(accessToken, existingRecord.DatingId);
4012.     }
4013.     #endregion
4014.
4015.
4016.     #region DgGaleryCategory
4017.     public DgGaleryCategory GetGaleryCategoryById(DgAccessToken accessToken, Int16 catego
ryId)
4018.     {
4019.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
4020.
4021.         #region SecurityLayer
4022.         #endregion
4023.
4024.         return DigitDLL.GetGaleryCategoryById(accessToken, categoryId);
4025.     }
4026.
4027.
4028.     public IList<DgGaleryCategory> GetGaleryCategories(DgAccessToken accessToken, string
whereClause, string orderByClause)
4029.     {
4030.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
4031.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
4032.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
4033.
4034.         #region SecurityLayer
4035.         #endregion
4036.
4037.         return DigitDLL.GetGaleryCategories(accessToken, whereClause, orderByClause);
4038.     }
4039.
4040.
4041.     public IList<DgGaleryCategory> GetGaleryCategories(DgAccessToken accessToken, int pag
eIndex, int pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order
by Name")
4042.     {
4043.         if (accessToken == null) throw new ArgumentNullException("accessToken");
4044.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
4045.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
4046.
4047.         #region SecurityLayer
4048.
4049.         #endregion
4050.
4051.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);

```

```

4052.         int startRowIndex = ((pageIndex - 1) * pageSize);
4053.
4054.         return DigitDLL.GetGaleryCategories(accessToken, startRowIndex, pageSize, ref totalRows, whereClause, orderByClause);
4055.     }
4056.
4057.     public DgGaleryCategory CreateGaleryCategory(DgAccessToken accessToken, DgGaleryCategory category)
4058.     {
4059.         #region input parameters validation
4060.         if (accessToken == null) throw new ArgumentNullException("accessToken");
4061.         if (category == null) throw new ArgumentNullException("category");
4062.         category.ValidateInstance();
4063.         #endregion
4064.
4065.         #region SecurityLayer
4066.         if (accessToken.IsSysAdmin == false)
4067.         {
4068.             throw new DgException("Only a SysAdmin can do that!");
4069.         }
4070.         #endregion
4071.
4072.         /*Check Name uniqueness*/
4073.         var existingRecord = DigitDLL.GetGaleryCategoryByName(accessToken, category.Name);
4074.         ;
4075.         if (existingRecord != null)
4076.         {
4077.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", category.Name));
4078.         }
4079.         category.CreateBy = category.UpdateBy = accessToken.UserId;
4080.         category.Createdt = category.UpdateDt = Utility.Now();
4081.
4082.         return DigitDLL.CreateGaleryCategory(accessToken, category);
4083.     }
4084.
4085.     public DgGaleryCategory UpdateGaleryCategory(DgAccessToken accessToken, DgGaleryCategory category)
4086.     {
4087.         #region input parameters validation
4088.         if (accessToken == null) throw new ArgumentNullException("accessToken");
4089.         if (category == null) throw new ArgumentNullException("category");
4090.         category.ValidateInstance();
4091.         #endregion
4092.
4093.         #region SecurityLayer
4094.         if (accessToken.IsSysAdmin == false)
4095.         {
4096.             throw new DgException("Only a SysAdmin can do that!");
4097.         }
4098.         #endregion
4099.
4100.
4101.         var existingRecord = DigitDLL.GetGaleryCategoryByName(accessToken, category.Name);
4102.         ;
4103.         if (existingRecord != null && existingRecord.CategoryId != category.CategoryId)
4104.         {
4105.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", category.Name));
4106.         }
4107.         if (existingRecord == null)
4108.         {

```

```

4108.             existingRecord = DigitDLL.GetGaleryCategoryById(accessToken, category.Categor
yId);
4109.         }
4110.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "category", category.CategoryId));
4111.
4112.
4113.         existingRecord.Name = category.Name;
4114.         existingRecord.Description = category.Description;
4115.
4116.         existingRecord.UpdateBy = accessToken.UserId;
4117.         existingRecord.UpdateDt = Utility.Now();
4118.
4119.         return DigitDLL.UpdateGaleryCategory(accessToken, existingRecord);
4120.     }
4121.
4122.     public void DeleteGaleryCategory(DgAccessToken accessToken, Int16 categoryId)
4123.     {
4124.         if (accessToken == null) throw new ArgumentNullException("accessToken");
4125.         var existingRecord = DigitDLL.GetGaleryCategoryById(accessToken, categoryId);
4126.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "GaleryCategory", categoryId));
4127.
4128.         #region SecurityLayer
4129.         if (accessToken.IsSysAdmin == false)
4130.         {
4131.             throw new DgException("Only a SysAdmin can do that!");
4132.         }
4133.         #endregion
4134.
4135.         //Can be deleted?
4136.         DigitDLL.CanDeleteGaleryCategory(accessToken, existingRecord.CategoryId);
4137.
4138.
4139.         DigitDLL.DeleteGaleryCategory(accessToken, existingRecord.CategoryId);
4140.     }
4141.     #endregion
4142.
4143.
4144.
4145.     #region DgGeosite
4146.     public DgGeosite GetGeositeById(DgAccessToken accessToken, Int32 geositeId)
4147.     {
4148.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
4149.
4150.         #region SecurityLayer
4151.         #endregion
4152.
4153.         return DigitDLL.GetGeositeById(accessToken, geositeId);
4154.     }
4155.
4156.     public IList<DgGeosite> GetGeosites(DgAccessToken accessToken, string whereClause = n
ull, string orderByClause = "order by name")
4157.     {
4158.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
4159.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
4160.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
4161.
4162.         #region SecurityLayer
4163.         #endregion
4164.

```

```

4165.         return DigitDLL.GetGeosites(accessToken, whereClause, orderByClause);
4166.     }
4167.
4168.     public IList<DgGeosite> GetGeosites(DgAccessToken accessToken, int pageIndex, int pag
4169. eSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
4170.     {
4171.         if (accessToken == null) throw new ArgumentNullException("accessToken");
4172.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
4173. ;
4174.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
4175. se");
4176.
4177.         #region SecurityLayer
4178.         #endregion
4179.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
4180.         int startRowIndex = ((pageIndex - 1) * pageSize);
4181.         return DigitDLL.GetGeosites(accessToken, startRowIndex, pageSize, ref totalRows,
4182. whereClause, orderByClause);
4183.     }
4184.
4185.     void _ReverseGeocoding(DgAccessToken accessToken, DgGeosite geosite)
4186.     {
4187.         if (!geosite.Latitude.HasValue || !geosite.Longitude.HasValue)
4188.         {
4189.             geosite.Latitude = Default_Latitude;
4190.             geosite.Longitude = Default_Longitude;
4191.             geosite.MapZoomLevel = 4;
4192.             geosite.Country = BuiltinCountries.Unknown.CountryId;
4193.             return;
4194.         }
4195.
4196.         try
4197.         {
4198.             var apiKey = ConfigurationManager.AppSettings["GoogleAPIkey"];
4199.             var lat = geosite.Latitude.Value.ToString(CultureInfo.InvariantCulture);
4200.             var lng = geosite.Longitude.Value.ToString(CultureInfo.InvariantCulture);
4201.             var url = string.Format("https://maps.googleapis.com/maps/api/geocode/xml?key
4202. ={0}&latlng={1},{2}&language=el&result_type=country&sensor=false", apiKey, lat, lng);
4203.
4204.             var request = WebRequest.Create(url);
4205.             var response = (HttpWebResponse)request.GetResponse();
4206.             XDocument xdoc = XDocument.Load(response.GetResponseStream());
4207.
4208.             XElement status = xdoc.Element("GeocodeResponse").Element("status");
4209.             if (status.Value == "OK")
4210.             {
4211.                 var address_component = xdoc.Element("GeocodeResponse").Element("result")
4212. .Element("address_component");
4213.                 var long_name = address_component.Element("long_name").Value;
4214.                 var short_name = address_component.Element("short_name").Value;
4215.
4216.                 var country = DigitDLL.GetCountryByLongName(accessToken, long_name);
4217.                 if (country == null)
4218.                 {
4219.                     country = CreateCountry(accessToken, new DgCountry { LongName = long_
4220. name, ShortName = short_name });
4221.                 }
4222.                 geosite.Country = country.CountryId;

```

```

4221.         }
4222.     }
4223.     catch (Exception ex)
4224.     {
4225.         geosite.Latitude = Default_Latitude;
4226.         geosite.Longitude = Default_Longitude;
4227.         geosite.Country = BuiltinCountries.Unknown.CountryId;
4228.     }
4229. }
4230.
4231. public DgGeosite CreateGeosite(DgAccessToken accessToken, DgGeosite geosite)
4232. {
4233.     #region input parameters validation
4234.     if (accessToken == null) throw new ArgumentNullException("accessToken");
4235.     if (geosite == null) throw new ArgumentNullException("geosite");
4236.     geosite.ValidateInstance();
4237.     #endregion
4238.
4239.     #region SecurityLayer
4240.     if (accessToken.IsSysAdmin == false)
4241.     {
4242.         throw new DgException("Requires SysAdmin permission!");
4243.     }
4244.     #endregion
4245.
4246.     /*Check Name uniqueness*/
4247.     var existingRecord = DigitDLL.GetGeositeByName(accessToken, geosite.Name);
4248.     if (existingRecord != null)
4249.     {
4250.         throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", geosite.Name));
4251.     }
4252.
4253.     /*Τώρα θα φτιάξουμε το dirName του geosite*/
4254.     geosite.DirName = Utility.ConvertToGreeklsh(geosite.Name);
4255.     int numOfTries = 1;
4256.     while (DigitDLL.GetGeositeByDirName(accessToken, geosite.DirName) != null)
4257.     {
4258.         geosite.DirName = Utility.ConvertToGreeklsh(geosite.Name) + string.Format("_{0:D2}", numOfTries);
4259.     }
4260.
4261.     #region Reverse Geocoding (Address Lookup)
4262.     _ReverseGeocoding(accessToken, geosite);
4263.     #endregion
4264.
4265.
4266.     geosite.CreateBy = geosite.UpdateBy = accessToken.UserId;
4267.     geosite.CreateDt = geosite.UpdateDt = Utility.Now();
4268.
4269.     return DigitDLL.CreateGeosite(accessToken, geosite);
4270. }
4271.
4272. public DgGeosite UpdateGeosite(DgAccessToken accessToken, DgGeosite geosite)
4273. {
4274.     #region input parameters validation
4275.     if (accessToken == null) throw new ArgumentNullException("accessToken");
4276.     if (geosite == null) throw new ArgumentNullException("geosite");
4277.     geosite.ValidateInstance();
4278.     #endregion
4279.
4280.     #region SecurityLayer
4281.     if (accessToken.IsSysAdmin == false)

```

```

4282.         {
4283.             throw new DgException("Requires SysAdmin permission!");
4284.         }
4285.         #endregion
4286.
4287.         var existingRecord = DigitDLL.GetGeositeByName(accessToken, geosite.Name);
4288.         if (existingRecord != null && existingRecord.GeositeId != geosite.GeositeId)
4289.         {
4290.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", geosit
4291. e.Name));
4292.         }
4293.         if (existingRecord == null)
4294.         {
4295.             existingRecord = DigitDLL.GetGeositeById(accessToken, geosite.GeositeId);
4296.         }
4297.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
4298. m_with_id, "geosite", geosite.GeositeId));
4299.         if (existingRecord.Latitude != geosite.Latitude || existingRecord.Longitude != ge
4300. osite.Longitude)
4301.         {
4302.             #region Reverse Geocoding (Address Lookup)
4303.             _ReverseGeocoding(accessToken, geosite);
4304.             #endregion
4305.         }
4306.         existingRecord.Name = geosite.Name;
4307.         existingRecord.Summary = geosite.Summary;
4308.         existingRecord.Description = geosite.Description;
4309.         existingRecord.Latitude = geosite.Latitude;
4310.         existingRecord.Longitude = geosite.Longitude;
4311.         existingRecord.MapZoomLevel = geosite.MapZoomLevel;
4312.         existingRecord.UpdateBy = accessToken.UserId;
4313.         existingRecord.UpdateDt = Utility.Now();
4314.
4315.         return DigitDLL.UpdateGeosite(accessToken, existingRecord);
4316.     }
4317.
4318.     public void DeleteGeosite(DgAccessToken accessToken, Int32 geositeId)
4319.     {
4320.         if (accessToken == null) throw new ArgumentNullException("accessToken");
4321.         var existingRecord = DigitDLL.GetGeositeById(accessToken, geositeId);
4322.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
4323. m_with_id, "GeoSite", geositeId));
4324.         #region SecurityLayer
4325.         if (accessToken.IsSysAdmin == false)
4326.         {
4327.             throw new DgException("Requires SysAdmin permission!");
4328.         }
4329.         #endregion
4330.
4331.         //Can be deleted?
4332.         DigitDLL.CanDeleteGeosite(accessToken, existingRecord.GeositeId);
4333.
4334.
4335.         //Πρέπει να διαγράψουμε όλα τα συσχετιζόμενα αρχεία του:
4336.         var files = GetGeositeImages(accessToken, existingRecord.GeositeId, DgImageCatego
4337. ry.All);
4338.         foreach (var _file in files)
4339.         {
4340.             _DeleteFile(accessToken, _file);

```

```

4340.     }
4341.     //Διαγράφουμε το directory:
4342.     string inventoryPath = Path.Combine(RootDirectory, existingRecord.DirName);
4343.     if (Directory.Exists(inventoryPath))
4344.     {
4345.         Directory.Delete(inventoryPath, true);
4346.     }
4347.
4348.
4349.     //Πραγματοποιούμε την διαγραφή:
4350.     DigitDLL.DeleteGeosite(accessToken, existingRecord.GeositeId);
4351.     }
4352.
4353.
4354.
4355.     public IList<DgFile> GetGeositeImages(DgAccessToken accessToken, Int32 geositeId, DgI
imageCategory? imgCategory)
4356.     {
4357.         var orderByClause = "order by createDt asc";
4358.
4359.         if (imgCategory == null || imgCategory == DgImageCategory.All)
4360.         {
4361.             var whereClause = string.Format("where ownername='geosites' and ownerId={0}",
geositeId);
4362.             return DigitDLL.GetFiles(accessToken, whereClause, orderByClause);
4363.         }
4364.         else
4365.         {
4366.             var whereClause = string.Format("where ownername='geosites' and ownerId={0} a
nd imagecategory={1}", geositeId, (byte)imgCategory);
4367.             return DigitDLL.GetFiles(accessToken, whereClause, orderByClause);
4368.         }
4369.     }
4370.     public DgFile UploadGeositeImage(DgAccessToken accessToken, Int32 geositeId, System.B
yte[] buffer, string mimeType, string fileName, DgImageCategory imgCategory)
4371.     {
4372.         if (accessToken == null) throw new ArgumentNullException("accessToken");
4373.         if (buffer.Length == 0) throw new ArgumentException("buffer cannot be empty!");
4374.         if (String.IsNullOrEmpty(mimeType)) throw new ArgumentException("mimeType cannot
be empty!");
4375.         if (String.IsNullOrEmpty(fileName)) throw new ArgumentException("fileName cannot
be empty!");
4376.
4377.
4378.         #region SecurityLayer
4379.         if (accessToken.IsSysAdmin == false)
4380.         {
4381.             throw new DgException("Requires SysAdmin permission!");
4382.         }
4383.         #endregion
4384.
4385.         //Βρίσκουμε τον owner:
4386.         var geosite = GetGeositeById(accessToken, geositeId);
4387.         if (geosite == null)
4388.         {
4389.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Geosite", ge
ositeId));
4390.         }
4391.         if (string.IsNullOrEmpty(geosite.DirName))
4392.         {
4393.             throw new DgException("Invalid geosite.DirName");
4394.         }
4395.

```

```

4396.
4397.         DgFile digitFile = new DgFile();
4398.         digitFile.OwnerName = "geosites";
4399.         digitFile.OwnerId = geosite.GeositeId;
4400.         digitFile.Filetype = DgFileType.Image;
4401.         digitFile.Imagecategory = imgCategory;
4402.         digitFile.Orgfilename = fileName;
4403.         digitFile.Filepath = geosite.DirName;
4404.
4405.
4406.         return _UploadFile(accessToken, digitFile, buffer);
4407.     }
4408.     public void DeleteGeositeFile(DgAccessToken accessToken, Int32 geositeId, Int32 fileId)
4409.     {
4410.         if (accessToken == null) throw new ArgumentNullException("accessToken");
4411.
4412.         #region SecurityLayer
4413.         if (accessToken.IsSysAdmin == false)
4414.         {
4415.             throw new DgException("Requires SysAdmin permission!");
4416.         }
4417.         #endregion
4418.
4419.         var geosite = GetGeositeById(accessToken, geositeId);
4420.         if (geosite == null)
4421.         {
4422.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Geosite", geositeId));
4423.         }
4424.
4425.         var file = DigitDLL.GetFileById(accessToken, "geosites", geosite.GeositeId, fileId);
4426.         if (file == null)
4427.         {
4428.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "File", fileId));
4429.         }
4430.
4431.         _DeleteFile(accessToken, file);
4432.     }
4433.     #endregion
4434.
4435.
4436.     #region DgSite
4437.     public DgSite GetSiteById(DgAccessToken accessToken, Int32 siteId)
4438.     {
4439.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
4440.
4441.         #region SecurityLayer
4442.         #endregion
4443.
4444.         return DigitDLL.GetSiteById(accessToken, siteId);
4445.     }
4446.
4447.     public IList<DgSite> GetSites(DgAccessToken accessToken, string whereClause = null, string orderByClause = "order by name")
4448.     {
4449.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
4450.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
4451.         ;
4451.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClause");

```



```

4452.
4453.         #region SecurityLayer
4454.         #endregion
4455.
4456.         return DigitDLL.GetSites(accessToken, whereClause, orderByClause);
4457.     }
4458.
4459.     public IList<DgSite> GetSites(DgAccessToken accessToken, int pageIndex, int pageSize,
4460.         ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
4461.     {
4462.         if (accessToken == null) throw new ArgumentNullException("accessToken");
4463.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
4464.         ;
4465.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
4466. se");
4467.
4468.         #region SecurityLayer
4469.         #endregion
4470.
4471.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
4472.         int startRowIndex = ((pageIndex - 1) * pageSize);
4473.
4474.         return DigitDLL.GetSites(accessToken, startRowIndex, pageSize, ref totalRows, whe
4475. reClause, orderByClause);
4476.     }
4477.     public IList<DgvSite> GetSites_View(DgAccessToken accessToken, int pageIndex, int pag
4478. eSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
4479.     {
4480.         if (accessToken == null) throw new ArgumentNullException("accessToken");
4481.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
4482.         ;
4483.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
4484. se");
4485.
4486.         #region SecurityLayer
4487.         #endregion
4488.
4489.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
4490.         int startRowIndex = ((pageIndex - 1) * pageSize);
4491.
4492.         return DigitDLL.GetSites_View(accessToken, startRowIndex, pageSize, ref totalRows
4493. , whereClause, orderByClause);
4494.     }
4495.
4496.     void _checkSitePermissionsForWork(DgAccessToken accessToken, DgSite site)
4497.     {
4498.         bool hasPermissions = false;
4499.         if (accessToken.IsSysAdmin == true)
4500.         {
4501.             hasPermissions = true;
4502.         }
4503.         if (hasPermissions == false)
4504.         {
4505.             foreach (var grant in accessToken.UserGrants)
4506.             {
4507.                 if (grant.Site == site.SiteId)
4508.                 {
4509.                     hasPermissions = true;
4510.                     break;

```

```

4507.         }
4508.     }
4509. }
4510.     if (hasPermissions == false)
4511.     {
4512.         throw new DgException(string.Format("You do not have permission on Site '{0}'
!\"", site.Name));
4513.     }
4514. }
4515.
4516.     public DgSite CreateSite(DgAccessToken accessToken, DgSite site)
4517.     {
4518.         #region input parameters validation
4519.         if (accessToken == null) throw new ArgumentNullException("accessToken");
4520.         if (site == null) throw new ArgumentNullException("site");
4521.         site.ValidateInstance();
4522.         #endregion
4523.
4524.         #region SecurityLayer
4525.         if (accessToken.IsSysAdmin == false)
4526.         {
4527.             throw new DgException("Requires SysAdmin permission!");
4528.         }
4529.         #endregion
4530.
4531.         /*Έλεγχος του Geosite*/
4532.         var geosite = DigitDLL.GetGeositeById(accessToken, site.Geosite);
4533.         if (geosite == null)
4534.         {
4535.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Geosite", si
te.Geosite));
4536.         }
4537.
4538.         /*Check Name uniqueness*/
4539.         var existingRecord = DigitDLL.GetSiteByName(accessToken, site.Name);
4540.         if (existingRecord != null)
4541.         {
4542.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", site.N
ame));
4543.         }
4544.
4545.         /*Τώρα θα φτιάξουμε το dirName του site*/
4546.         site.DirName = Utility.ConvertToGreeklish(site.Name);
4547.         int numOfTries = 1;
4548.         while (DigitDLL.GetSiteByDirName(accessToken, site.DirName) != null)
4549.         {
4550.             site.DirName = Utility.ConvertToGreeklish(site.Name) + string.Format("_{0:D2}
", numOfTries);
4551.         }
4552.
4553.         #region LatLong
4554.         if (!site.Latitude.HasValue || !site.Longitude.HasValue)
4555.         {
4556.             site.Latitude = null;
4557.             site.Longitude = null;
4558.             site.MapZoomLevel = 4;
4559.         }
4560.         #endregion
4561.
4562.         //έλεγχος του env
4563.         if (site.Env.HasValue)
4564.         {
4565.             var siteenv = DigitDLL.GetSiteEnvById(accessToken, site.Env.Value);

```

```

4566.         if (siteenv == null)
4567.         {
4568.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "SiteEnv"
, site.Env.Value));
4569.         }
4570.     }
4571.     //ελεγχος του form
4572.     if (site.Form.HasValue)
4573.     {
4574.         var siteform = DigitDLL.GetSiteFormById(accessToken, site.Form.Value);
4575.         if (siteform == null)
4576.         {
4577.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "SiteForm
", site.Form.Value));
4578.         }
4579.     }
4580.     //ελεγχος του exctype
4581.     if (site.Exctype.HasValue)
4582.     {
4583.         var exctype = DigitDLL.GetSiteExcTypeById(accessToken, site.Exctype.Value);
4584.         if (exctype == null)
4585.         {
4586.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "SiteExcT
ype", site.Exctype.Value));
4587.         }
4588.     }
4589.     //ελεγχος του exctech
4590.     if (site.Exctech.HasValue)
4591.     {
4592.         var exctech = DigitDLL.GetSiteExcTechById(accessToken, site.Exctech.Value);
4593.         if (exctech == null)
4594.         {
4595.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "SiteExcT
ech", site.Exctech.Value));
4596.         }
4597.     }
4598.     //ελεγχος του invmethod
4599.     if (site.Invmethod.HasValue)
4600.     {
4601.         var invmethod = DigitDLL.GetSiteInvMethodById(accessToken, site.Invmethod.Val
ue);
4602.         if (invmethod == null)
4603.         {
4604.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "SiteInvM
ethod", site.Invmethod.Value));
4605.         }
4606.     }
4607.     //ελεγχος του climate
4608.     if (site.Climate.HasValue)
4609.     {
4610.         var invmethod = DigitDLL.GetSiteClimateById(accessToken, site.Climate.Value);
4611.         if (invmethod == null)
4612.         {
4613.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "SiteClim
ate", site.Climate.Value));
4614.         }
4615.     }
4616.     //
4617.     site.ShowOrder = 0;
4618.     site.IsActive = false;
4619.
4620.     site.CreateBy = site.UpdateBy = accessToken.UserId;

```

```

4621.         site.CreateDt = site.UpdateDt = Utility.Now();
4622.
4623.         return DigitDLL.CreateSite(accessToken, site);
4624.     }
4625.
4626.     public DgSite UpdateSite(DgAccessToken accessToken, DgSite site)
4627.     {
4628.         #region input parameters validation
4629.         if (accessToken == null) throw new ArgumentNullException("accessToken");
4630.         if (site == null) throw new ArgumentNullException("site");
4631.         site.ValidateInstance();
4632.         #endregion
4633.
4634.         #region SecurityLayer
4635.         if (accessToken.IsSysAdmin == false)
4636.         {
4637.             throw new DgException("Requires SysAdmin permission!");
4638.         }
4639.         #endregion
4640.
4641.         var existingRecord = DigitDLL.GetSiteByName(accessToken, site.Name);
4642.         if (existingRecord != null && existingRecord.SiteId != site.SiteId)
4643.         {
4644.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", site.N
ame));
4645.         }
4646.         if (existingRecord == null)
4647.         {
4648.             existingRecord = DigitDLL.GetSiteById(accessToken, site.SiteId);
4649.         }
4650.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "site", site.SiteId));
4651.
4652.
4653.         /*Ενα site δεν μπορεί να αλλάξει geosite*/
4654.         if (existingRecord.Geosite != site.Geosite)
4655.         {
4656.             throw new DgException("Geosite is invalid");
4657.         }
4658.
4659.         #region LatLong
4660.         if (!site.Latitude.HasValue || !site.Longitude.HasValue)
4661.         {
4662.             site.Latitude = null;
4663.             site.Longitude = null;
4664.             site.MapZoomLevel = 4;
4665.         }
4666.         #endregion
4667.
4668.
4669.         //ελεγχος του env
4670.         if (existingRecord.Env != site.Env)
4671.         {
4672.             if (site.Env.HasValue)
4673.             {
4674.                 var siteenv = DigitDLL.GetSiteEnvById(accessToken, site.Env.Value);
4675.                 if (siteenv == null)
4676.                 {
4677.                     throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Site
Env", site.Env.Value));
4678.                 }
4679.             }
4680.         }

```

```

4681.         //ελεγχος του form
4682.         if (existingRecord.Form != site.Form)
4683.         {
4684.             if (site.Form.HasValue)
4685.             {
4686.                 var siteform = DigitDLL.GetSiteFormById(accessToken, site.Form.Value);
4687.                 if (siteform == null)
4688.                 {
4689.                     throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Site
Form", site.Form.Value));
4690.                 }
4691.             }
4692.         }
4693.         //ελεγχος του exctype
4694.         if (existingRecord.Exctype != site.Exctype)
4695.         {
4696.             if (site.Exctype.HasValue)
4697.             {
4698.                 var exctype = DigitDLL.GetSiteExcTypeById(accessToken, site.Exctype.Value
);
4699.                 if (exctype == null)
4700.                 {
4701.                     throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Site
ExcType", site.Exctype.Value));
4702.                 }
4703.             }
4704.         }
4705.         //ελεγχος του exctech
4706.         if (existingRecord.Exctech != site.Exctech)
4707.         {
4708.             if (site.Exctech.HasValue)
4709.             {
4710.                 var exctech = DigitDLL.GetSiteExcTechById(accessToken, site.Exctech.Value
);
4711.                 if (exctech == null)
4712.                 {
4713.                     throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Site
ExcTech", site.Exctech.Value));
4714.                 }
4715.             }
4716.         }
4717.         //ελεγχος του invmethod
4718.         if (existingRecord.Invmethod != site.Invmethod)
4719.         {
4720.             if (site.Invmethod.HasValue)
4721.             {
4722.                 var invmethod = DigitDLL.GetSiteInvMethodById(accessToken, site.Invmethod
.Value);
4723.                 if (invmethod == null)
4724.                 {
4725.                     throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Site
InvMethod", site.Invmethod.Value));
4726.                 }
4727.             }
4728.         }
4729.         //ελεγχος του climate
4730.         if (existingRecord.Climate != site.Climate)
4731.         {
4732.             if (site.Climate.HasValue)
4733.             {
4734.                 var climate = DigitDLL.GetSiteClimateById(accessToken, site.Climate.Value
);
4735.                 if (climate == null)

```

```

4736.         {
4737.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Site
Climate", site.Climate.Value));
4738.         }
4739.     }
4740. }
4741.
4742.     //Geosite
4743.     existingRecord.Name = site.Name;
4744.     existingRecord.Summary = site.Summary;
4745.     existingRecord.Description = site.Description;
4746.     existingRecord.Latitude = site.Latitude;
4747.     existingRecord.Longitude = site.Longitude;
4748.     existingRecord.MapZoomLevel = site.MapZoomLevel;
4749.     existingRecord.ShowOrder = site.ShowOrder;
4750.     existingRecord.IsActive = site.IsActive;
4751.     existingRecord.Env = site.Env;
4752.     existingRecord.Form = site.Form;
4753.     existingRecord.Exctype = site.Exctype;
4754.     existingRecord.Exctech = site.Exctech;
4755.     existingRecord.Invmethod = site.Invmethod;
4756.     existingRecord.Climate = site.Climate;
4757.
4758.     existingRecord.UpdateBy = accessToken.UserId;
4759.     existingRecord.UpdateDt = Utility.Now();
4760.
4761.     return DigitDLL.UpdateSite(accessToken, existingRecord);
4762. }
4763.
4764.     public void DeleteSite(DgAccessToken accessToken, Int32 siteid)
4765.     {
4766.         if (accessToken == null) throw new ArgumentNullException("accessToken");
4767.         var existingRecord = DigitDLL.GetSiteById(accessToken, siteid);
4768.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "Site", siteid));
4769.
4770.         #region SecurityLayer
4771.         if (accessToken.IsSysAdmin == false)
4772.         {
4773.             throw new DgException("Requires SysAdmin permission!");
4774.         }
4775.         #endregion
4776.
4777.
4778.
4779.         //Can be deleted?
4780.         DigitDLL.CanDeleteSite(accessToken, existingRecord.SiteId);
4781.
4782.
4783.         #region βρισκουμε geosite:
4784.         var geosite = GetGeositeById(accessToken, existingRecord.Geosite);
4785.         if (geosite == null)
4786.         {
4787.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Geosite", ex
istingRecord.Geosite));
4788.         }
4789.         if (string.IsNullOrEmpty(geosite.DirName))
4790.         {
4791.             throw new DgException("Invalid geosite.DirName");
4792.         }
4793.         #endregion
4794.
4795.

```

```

4796.         //Πρέπει να διαγράψουμε όλα τα usergrants
4797.         var grants = DigitDLL.GetUserGrantsForSite_View(accessToken, existingRecord.SiteId);
4798.         foreach (var grant in grants)
4799.         {
4800.             DigitDLL.DeleteUserGrant(accessToken, grant.Usergrantid);
4801.         }
4802.         //Πρέπει να διαγράψουμε όλα τα συσχετιζόμενα αρχεία του:
4803.         var files = GetSiteImages(accessToken, existingRecord.SiteId, DgImageCategory.All);
4804.         foreach (var _file in files)
4805.         {
4806.             _DeleteFile(accessToken, _file);
4807.         }
4808.         //Διαγράφουμε το directory:
4809.         string inventoryPath = Path.Combine(RootDirectory, geosite.DirName, existingRecord.DirName);
4810.         if (Directory.Exists(inventoryPath))
4811.         {
4812.             Directory.Delete(inventoryPath, true);
4813.         }
4814.
4815.
4816.         //Πραγματοποιούμε την διαγραφή:
4817.         DigitDLL.DeleteSite(accessToken, existingRecord.SiteId);
4818.     }
4819.
4820.
4821.
4822.
4823.     public IList<DgFile> GetSiteImages(DgAccessToken accessToken, Int32 siteid, DgImageCategory? imgCategory)
4824.     {
4825.         var orderByClause = "order by createdt asc";
4826.
4827.         if (imgCategory == null || imgCategory == DgImageCategory.All)
4828.         {
4829.             var whereClause = string.Format("where ownername='sites' and ownerId={0}", siteid);
4830.             return DigitDLL.GetFiles(accessToken, whereClause, orderByClause);
4831.         }
4832.         else
4833.         {
4834.             var whereClause = string.Format("where ownername='sites' and ownerId={0} and imagecategory={1}", siteid, (byte)imgCategory);
4835.             return DigitDLL.GetFiles(accessToken, whereClause, orderByClause);
4836.         }
4837.     }
4838.     public DgFile UploadSiteImage(DgAccessToken accessToken, Int32 siteId, System.Byte[] buffer, string mimeType, string fileName, DgImageCategory imgCategory)
4839.     {
4840.         if (accessToken == null) throw new ArgumentNullException("accessToken");
4841.         if (buffer.Length == 0) throw new ArgumentException("buffer cannot be empty!");
4842.         if (String.IsNullOrEmpty(mimeType)) throw new ArgumentException("mimeType cannot be empty!");
4843.         if (String.IsNullOrEmpty(fileName)) throw new ArgumentException("fileName cannot be empty!");
4844.
4845.
4846.         #region SecurityLayer
4847.         if (accessToken.IsSysAdmin == false)
4848.         {
4849.             throw new DgException("Requires SysAdmin permission!");

```

```

4850.         }
4851.         #endregion
4852.
4853.         //Βρίσκουμε τον owner:
4854.         var site = GetSiteById(accessToken, siteId);
4855.         if (site == null)
4856.         {
4857.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Site", siteI
d));
4858.         }
4859.         if (string.IsNullOrEmpty(site.DirName))
4860.         {
4861.             throw new DgException("Invalid Site.DirName");
4862.         }
4863.
4864.         //Βρίσκουμε το geosite στο οποίο ανήκει αυτό το site:
4865.         var geosite = GetGeositeById(accessToken, site.Geosite);
4866.         if (geosite == null)
4867.         {
4868.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Geosite", si
te.Geosite));
4869.         }
4870.         if (string.IsNullOrEmpty(geosite.DirName))
4871.         {
4872.             throw new DgException("Invalid Geosite.DirName");
4873.         }
4874.
4875.         DgFile digitFile = new DgFile();
4876.         digitFile.OwnerName = "sites";
4877.         digitFile.OwnerId = site.SiteId;
4878.         digitFile.Filetype = DgFileType.Image;
4879.         digitFile.Imagecategory = imgCategory;
4880.         digitFile.Orgfilename = fileName;
4881.         digitFile.Filepath = Path.Combine(geosite.DirName, site.DirName);
4882.
4883.
4884.         return _UploadFile(accessToken, digitFile, buffer);
4885.     }
4886.     public void DeleteSiteFile(DgAccessToken accessToken, Int32 siteId, Int32 fileId)
4887.     {
4888.         if (accessToken == null) throw new ArgumentNullException("accessToken");
4889.
4890.         #region SecurityLayer
4891.         if (accessToken.IsSysAdmin == false)
4892.         {
4893.             throw new DgException("Requires SysAdmin permission!");
4894.         }
4895.         #endregion
4896.
4897.         var site = GetSiteById(accessToken, siteId);
4898.         if (site == null)
4899.         {
4900.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Site", siteI
d));
4901.         }
4902.         if (string.IsNullOrEmpty(site.DirName))
4903.         {
4904.             throw new DgException("Invalid Site.DirName");
4905.         }
4906.
4907.         var file = DigitDLL.GetFileById(accessToken, "sites", site.SiteId, fileId);
4908.         if (file == null)
4909.         {

```



```

4910.         throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "File", fileId));
4911.     }
4912.
4913.     _DeleteFile(accessToken, file);
4914. }
4915.
4916.     /// <summary>
4917.     ///
4918.     /// </summary>
4919.     /// <param name="accessToken"></param>
4920.     /// <param name="userId"></param>
4921.     /// <param name="siteId"></param>
4922.     /// <returns></returns>
4923. public DgUserGrant GetUserGrantForSite(DgAccessToken accessToken, Int32 userId, Int32
siteId)
4924. {
4925.     if (accessToken == null) throw new ArgumentNullException("accessToken");
4926.
4927.     return DigitDLL.GetUserGrantForSite(accessToken, siteId, userId);
4928. }
4929.     /// <summary>
4930.     ///
4931.     /// </summary>
4932.     /// <param name="accessToken"></param>
4933.     /// <param name="userId"></param>
4934.     /// <param name="siteid"></param>
4935.     /// <param name="siteRole"></param>
4936.     /// <returns></returns>
4937. public DgUserGrant GrantSiteToUser(DgAccessToken accessToken, Int32 userId, Int32 sit
eid, DgEntityRole siteRole)
4938. {
4939.     if (accessToken == null) throw new ArgumentNullException("accessToken");
4940.
4941.     #region SecurityLayer
4942.     if (accessToken.IsSysAdmin == false)
4943.     {
4944.         throw new DgException("Requires SysAdmin permission!");
4945.     }
4946.     #endregion
4947.
4948.     var site = DigitDLL.GetSiteById(accessToken, siteid);
4949.     if (site == null) throw new DgException(SR.GetString(SR.There_is_no_item_with_id,
"Site", siteid));
4950.
4951.     var user = DigitDLL.GetUserById(accessToken, userId);
4952.     if (user == null) throw new DgException(SR.GetString(SR.There_is_no_item_with_id,
"User", userId));
4953.
4954.
4955.     /*Μήπως είναι ήδη συνδεδεμένος ο χρήστης με αυτό το site:*/
4956.     var existingGrant = DigitDLL.GetUserGrantForSite(accessToken, siteid, userId);
4957.     if (existingGrant != null)
4958.     {
4959.         if (existingGrant.SiteRole != siteRole)
4960.         {
4961.             existingGrant.SiteRole = siteRole;
4962.             existingGrant.UpdateBy = accessToken.UserId;
4963.             existingGrant.UpdateDt = Utility.Now();
4964.
4965.             return DigitDLL.UpdateUserGrant(accessToken, existingGrant);
4966.         }
4967.     }
else

```

```

4968.         {
4969.             return existingGrant;
4970.         }
4971.     }
4972.     else
4973.     {
4974.         DgUserGrant usersite = new DgUserGrant();
4975.         usersite.User = user.UserId;
4976.         usersite.Site = site.SiteId;
4977.         usersite.SiteRole = siteRole;
4978.
4979.         usersite.CreateBy = usersite.UpdateBy = accessToken.UserId;
4980.         usersite.Createdt = usersite.UpdateDt = Utility.Now();
4981.
4982.
4983.         return DigitDLL.CreateUserGrant(accessToken, usersite);
4984.     }
4985. }
4986. /// <summary>
4987. ///
4988. /// </summary>
4989. /// <param name="accessToken"></param>
4990. /// <param name="userId"></param>
4991. /// <param name="siteid"></param>
4992. public void RevokeSiteFromUser(DgAccessToken accessToken, Int32 userId, Int32 siteid)
4993. {
4994.     if (accessToken == null) throw new ArgumentNullException("accessToken");
4995.
4996.     #region SecurityLayer
4997.     if (accessToken.IsSysAdmin == false)
4998.     {
4999.         throw new DgException("Requires SysAdmin permission!");
5000.     }
5001.     #endregion
5002.
5003.
5004.     var existingGrant = DigitDLL.GetUserGrantForSite(accessToken, siteid, userId);
5005.     if (existingGrant != null)
5006.     {
5007.         DigitDLL.DeleteUserGrant(accessToken, existingGrant.Usergrantid);
5008.     }
5009. }
5010.
5011. /// <summary>
5012. /// Επιτρέπει τους χρήστες που έχουν δικαιώματα επάνω στο συγκεκριμένο site:
5013. /// </summary>
5014. /// <param name="accessToken"></param>
5015. /// <param name="siteid"></param>
5016. /// <returns></returns>
5017. public IList<DgvUserGrant> GetUserGrantsForSite(DgAccessToken accessToken, Int32 site
id)
5018. {
5019.     if (accessToken == null) throw new ArgumentNullException("accessToken");
5020.
5021.     #region SecurityLayer
5022.
5023.     #endregion
5024.
5025.     return DigitDLL.GetUserGrantsForSite_View(accessToken, siteid);
5026. }
5027. #endregion
5028.

```

```

5029.
5030.         #region DgPhase
5031.         public DgPhase GetPhaseById(DgAccessToken accessToken, Int32 phaseId)
5032.         {
5033.             if (accessToken == null) throw new ArgumentNullException("accesstoken");
5034.
5035.             #region SecurityLayer
5036.             #endregion
5037.
5038.             return DigitDLL.GetPhaseById(accessToken, phaseId);
5039.         }
5040.
5041.         public IList<DgPhase> GetPhases(DgAccessToken accessToken, string whereClause = null,
5042. string orderByClause = "order by name")
5043.         {
5044.             if (accessToken == null) throw new ArgumentNullException("accesstoken");
5045.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
5046.             ;
5047.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
5048. se");
5049.
5050.             #region SecurityLayer
5051.             #endregion
5052.
5053.             return DigitDLL.GetPhases(accessToken, whereClause, orderByClause);
5054.         }
5055.
5056.         public IList<DgvPhase> GetPhases_View(DgAccessToken accessToken, string whereClause =
5057. null, string orderByClause = "order by name")
5058.         {
5059.             if (accessToken == null) throw new ArgumentNullException("accesstoken");
5060.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
5061.             ;
5062.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
5063. se");
5064.
5065.             #region SecurityLayer
5066.             #endregion
5067.
5068.             return DigitDLL.GetPhases_View(accessToken, whereClause, orderByClause);
5069.         }
5070.
5071.         public IList<DgPhase> GetPhases(DgAccessToken accessToken, int pageIndex, int pageSiz
5072. e, ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
5073.         {
5074.             if (accessToken == null) throw new ArgumentNullException("accessToken");
5075.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
5076.             ;
5077.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
5078. se");
5079.
5080.             #region SecurityLayer
5081.             #endregion
5082.
5083.             Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
5084.             int startRowIndex = ((pageIndex - 1) * pageSize);
5085.
5086.             return DigitDLL.GetPhases(accessToken, startRowIndex, pageSize, ref totalRows, wh
5087. ereClause, orderByClause);
5088.         }

```

```

5081.         public IList<DgvPhase> GetPhases_View(DgAccessToken accessToken, int pageIndex, int p
ageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
5082.     {
5083.         if (accessToken == null) throw new ArgumentNullException("accessToken");
5084.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
5085.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
5086.
5087.         #region SecurityLayer
5088.
5089.         #endregion
5090.
5091.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
5092.         int startIndex = ((pageIndex - 1) * pageSize);
5093.
5094.         return DigitDLL.GetPhases_View(accessToken, startIndex, pageSize, ref totalRow
s, whereClause, orderByClause);
5095.     }
5096.
5097.     void _checkPhasePermissionsForWork(DgAccessToken accessToken, DgPhase phase)
5098.     {
5099.         bool hasPermissions = false;
5100.         if (accessToken.IsSysAdmin == true)
5101.         {
5102.             hasPermissions = true;
5103.         }
5104.         if (hasPermissions == false)
5105.         {
5106.             foreach (var grant in accessToken.UserGrants)
5107.             {
5108.                 if (grant.Phase == phase.PhaseId || grant.Site == phase.Site)
5109.                 {
5110.                     hasPermissions = true;
5111.                     break;
5112.                 }
5113.             }
5114.         }
5115.         if (hasPermissions == false)
5116.         {
5117.             throw new DgException(string.Format("You do not have permission on Phase '{0}
'!", phase.Name));
5118.         }
5119.     }
5120.
5121.     public DgPhase CreatePhase(DgAccessToken accessToken, DgPhase phase)
5122.     {
5123.         #region input parameters validation
5124.         if (accessToken == null) throw new ArgumentNullException("accessToken");
5125.         if (phase == null) throw new ArgumentNullException("site");
5126.         phase.ValidateInstance();
5127.         #endregion
5128.
5129.         #region SecurityLayer
5130.         if (accessToken.IsSysAdmin == false)
5131.         {
5132.             throw new DgException("Requires SysAdmin permission!");
5133.         }
5134.         #endregion
5135.
5136.         /*Ελεγχος του Site*/
5137.         var site = DigitDLL.GetSiteById(accessToken, phase.Site);
5138.         if (site == null)

```

```

5139.         {
5140.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Site", phase
5141. .Site));
5142.         }
5143.         /*Check Name uniqueness*/
5144.         var existingRecord = DigitDLL.GetPhaseByName(accessToken, site.SiteId, phase.Name
5145. );
5146.         if (existingRecord != null)
5147.         {
5148.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", phase.
5149. Name));
5150.         }
5151.         /*Τώρα θα φτιάξουμε το dirName του site*/
5152.         phase.DirName = Utility.ConvertToGreeklish(phase.Name);
5153.         int numOfTries = 1;
5154.         while (DigitDLL.GetPhaseByDirName(accessToken, site.SiteId, phase.DirName) != nul
5155. 1)
5156.         {
5157.             phase.DirName = Utility.ConvertToGreeklish(phase.Name) + string.Format("_{0:D
5158. 2}", numOfTries);
5159.         }
5160.         phase.ShowOrder = 0;
5161.         phase.IsActive = site.IsActive;
5162.         phase.CreateBy = phase.UpdateBy = accessToken.UserId;
5163.         phase.Createdt = phase.UpdateDt = Utility.Now();
5164.         return DigitDLL.CreatePhase(accessToken, phase);
5165.     }
5166.
5167.     public DgPhase UpdatePhase(DgAccessToken accessToken, DgPhase phase)
5168.     {
5169.         #region input parameters validation
5170.         if (accessToken == null) throw new ArgumentNullException("accessToken");
5171.         if (phase == null) throw new ArgumentNullException("site");
5172.         phase.ValidateInstance();
5173.         #endregion
5174.
5175.         #region SecurityLayer
5176.         if (accessToken.IsSysAdmin == false)
5177.         {
5178.             throw new DgException("Requires SysAdmin permission!");
5179.         }
5180.         #endregion
5181.
5182.         var existingRecord = DigitDLL.GetPhaseByName(accessToken, phase.Site, phase.Name)
5183. ;
5184.         if (existingRecord != null && existingRecord.PhaseId != phase.PhaseId)
5185.         {
5186.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", phase.
5187. Name));
5188.         }
5189.         if (existingRecord == null)
5190.         {
5191.             existingRecord = DigitDLL.GetPhaseById(accessToken, phase.PhaseId);
5192.         }
5193.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
5194. m_with_id, "phase", phase.PhaseId));

```

```

5194.
5195.         /*Ενα phase δεν μπορεί να αλλάξει site*/
5196.         if (existingRecord.Site != phase.Site)
5197.         {
5198.             throw new DgException("Site is invalid");
5199.         }
5200.
5201.
5202.
5203.
5204.         //Site
5205.         existingRecord.Name = phase.Name;
5206.         existingRecord.Summary = phase.Summary;
5207.         existingRecord.Description = phase.Description;
5208.         existingRecord.Start = phase.Start;
5209.         existingRecord.End = phase.End;
5210.         existingRecord.ShowOrder = phase.ShowOrder;
5211.         existingRecord.IsActive = phase.IsActive;
5212.         existingRecord.DirName = phase.DirName;
5213.
5214.         existingRecord.UpdateBy = accessToken.UserId;
5215.         existingRecord.UpdateDt = Utility.Now();
5216.
5217.         return DigitDLL.UpdatePhase(accessToken, existingRecord);
5218.     }
5219.
5220.     public void DeletePhase(DgAccessToken accessToken, Int32 phaseId)
5221.     {
5222.         if (accessToken == null) throw new ArgumentNullException("accessToken");
5223.         var existingRecord = DigitDLL.GetPhaseById(accessToken, phaseId);
5224.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "Phase", phaseId));
5225.
5226.         #region SecurityLayer
5227.         if (accessToken.IsSysAdmin == false)
5228.         {
5229.             throw new DgException("Requires SysAdmin permission!");
5230.         }
5231.         #endregion
5232.
5233.
5234.
5235.         //Can be deleted?
5236.         DigitDLL.CanDeletePhase(accessToken, existingRecord.PhaseId);
5237.
5238.         #region βρίσκουμε Site and Geosite:
5239.         var site = GetSiteById(accessToken, existingRecord.Site);
5240.         if (site == null)
5241.         {
5242.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Site", exist
ingRecord.Site));
5243.         }
5244.         if (string.IsNullOrEmpty(site.DirName))
5245.         {
5246.             throw new DgException("Invalid Site.DirName");
5247.         }
5248.
5249.         var geosite = GetGeositeById(accessToken, site.Geosite);
5250.         if (geosite == null)
5251.         {
5252.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Geosite", si
te.Geosite));
5253.         }

```

```

5254.         if (string.IsNullOrEmpty(geosite.DirName))
5255.         {
5256.             throw new DgException("Invalid Geosite.DirName");
5257.         }
5258.         #endregion
5259.
5260.
5261.         //Πρέπει να διαγράψουμε όλα τα usergrants
5262.         var grants = DigitDLL.GetUserGrantsForPhase_View(accessToken, existingRecord.PhaseId);
5263.         foreach (var grant in grants)
5264.         {
5265.             DigitDLL.DeleteUserGrant(accessToken, grant.Usergrantid);
5266.         }
5267.         //Πρέπει να διαγράψουμε όλα τα συσχετιζόμενα αρχεία του:
5268.         var files = GetPhaseImages(accessToken, existingRecord.PhaseId, DgImageCategory.All);
5269.         foreach (var _file in files)
5270.         {
5271.             _DeleteFile(accessToken, _file);
5272.         }
5273.         //Διαγράφουμε το directory:
5274.         string inventoryPath = Path.Combine(RootDirectory, geosite.DirName, site.DirName, existingRecord.DirName);
5275.         if (Directory.Exists(inventoryPath))
5276.         {
5277.             Directory.Delete(inventoryPath, true);
5278.         }
5279.
5280.
5281.         //Πραγματοποιούμε την διαγραφή:
5282.         DigitDLL.DeletePhase(accessToken, existingRecord.PhaseId);
5283.     }
5284.
5285.
5286.
5287.
5288.     public IList<DgFile> GetPhaseImages(DgAccessToken accessToken, Int32 phaseId, DgImageCategory? imgCategory)
5289.     {
5290.         var orderByClause = "order by createdt asc";
5291.
5292.         if (imgCategory == null || imgCategory == DgImageCategory.All)
5293.         {
5294.             var whereClause = string.Format("where ownername='phases' and ownerId={0}", phaseId);
5295.             return DigitDLL.GetFiles(accessToken, whereClause, orderByClause);
5296.         }
5297.         else
5298.         {
5299.             var whereClause = string.Format("where ownername='phases' and ownerId={0} and imagecategory={1}", phaseId, (byte)imgCategory);
5300.             return DigitDLL.GetFiles(accessToken, whereClause, orderByClause);
5301.         }
5302.     }
5303.     public DgFile UploadPhaseImage(DgAccessToken accessToken, Int32 phaseId, System.Byte[] buffer, string mimeType, string fileName, DgImageCategory imgCategory)
5304.     {
5305.         if (accessToken == null) throw new ArgumentNullException("accessToken");
5306.         if (buffer.Length == 0) throw new ArgumentException("buffer cannot be empty!");
5307.         if (String.IsNullOrEmpty(mimeType)) throw new ArgumentException("mimeType cannot be empty!");

```

```

5308.         if (String.IsNullOrEmpty(fileName)) throw new ArgumentException("fileName cannot
        be empty!");
5309.
5310.
5311.         #region SecurityLayer
5312.         if (accessToken.IsSysAdmin == false)
5313.         {
5314.             throw new DgException("Requires SysAdmin permission!");
5315.         }
5316.         #endregion
5317.
5318.         //Βρίσκουμε τον owner:
5319.         var phase = GetPhaseById(accessToken, phaseId);
5320.         if (phase == null)
5321.         {
5322.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Phase", phaseId));
5323.         }
5324.         if (string.IsNullOrEmpty(phase.DirName))
5325.         {
5326.             throw new DgException("Invalid Phase.DirName");
5327.         }
5328.
5329.         #region βρίσκουμε Site and Geosite:
5330.         var site = GetSiteById(accessToken, phase.Site);
5331.         if (site == null)
5332.         {
5333.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Site", phase.Site));
5334.         }
5335.         if (string.IsNullOrEmpty(site.DirName))
5336.         {
5337.             throw new DgException("Invalid Site.DirName");
5338.         }
5339.
5340.         var geosite = GetGeositeById(accessToken, site.Geosite);
5341.         if (geosite == null)
5342.         {
5343.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Geosite", site.Geosite));
5344.         }
5345.         if (string.IsNullOrEmpty(geosite.DirName))
5346.         {
5347.             throw new DgException("Invalid Geosite.DirName");
5348.         }
5349.         #endregion
5350.
5351.         DgFile digitFile = new DgFile();
5352.         digitFile.OwnerName = "phases";
5353.         digitFile.OwnerId = phase.PhaseId;
5354.         digitFile.Filetype = DgFileType.Image;
5355.         digitFile.Imagecategory = imgCategory;
5356.         digitFile.Orgfilename = fileName;
5357.         digitFile.Filepath = Path.Combine(geosite.DirName, site.DirName, phase.DirName);
5358.
5359.
5360.         return UploadFile(accessToken, digitFile, buffer);
5361.     }
5362.     public void DeletePhaseFile(DgAccessToken accessToken, Int32 phaseId, Int32 fileId)
5363.     {
5364.         if (accessToken == null) throw new ArgumentNullException("accessToken");
5365.

```



```

5366.         #region SecurityLayer
5367.         if (accessToken.IsSysAdmin == false)
5368.         {
5369.             throw new DgException("Requires SysAdmin permission!");
5370.         }
5371.         #endregion
5372.
5373.         var phase = GetPhaseById(accessToken, phaseId);
5374.         if (phase == null)
5375.         {
5376.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Phase", phaseId));
5377.         }
5378.
5379.         var file = DigitDLL.GetFileById(accessToken, "phases", phase.PhaseId, fileId);
5380.         if (file == null)
5381.         {
5382.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "File", fileId));
5383.         }
5384.
5385.         _DeleteFile(accessToken, file);
5386.     }
5387.
5388.
5389.     /// <summary>
5390.     ///
5391.     /// </summary>
5392.     /// <param name="accessToken"></param>
5393.     /// <param name="userId"></param>
5394.     /// <param name="phaseId"></param>
5395.     /// <returns></returns>
5396.     public DgUserGrant GetUserGrantForPhase(DgAccessToken accessToken, Int32 userId, Int32
2 phaseId)
5397.     {
5398.         if (accessToken == null) throw new ArgumentNullException("accessToken");
5399.
5400.         return DigitDLL.GetUserGrantForPhase(accessToken, phaseId, userId);
5401.     }
5402.     /// <summary>
5403.     ///
5404.     /// </summary>
5405.     /// <param name="accessToken"></param>
5406.     /// <param name="userId"></param>
5407.     /// <param name="phaseId"></param>
5408.     /// <param name="phaseRole"></param>
5409.     /// <returns></returns>
5410.     public DgUserGrant GrantPhaseToUser(DgAccessToken accessToken, Int32 userId, Int32 phaseId, DgEntityRole phaseRole)
5411.     {
5412.         if (accessToken == null) throw new ArgumentNullException("accessToken");
5413.
5414.         #region SecurityLayer
5415.         if (accessToken.IsSysAdmin == false)
5416.         {
5417.             throw new DgException("Requires SysAdmin permission!");
5418.         }
5419.         #endregion
5420.
5421.         var phase = DigitDLL.GetPhaseById(accessToken, phaseId);
5422.         if (phase == null) throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Phase", phaseId));
5423.

```

```

5424.         var user = DigitDLL.GetUserById(accessToken, userId);
5425.         if (user == null) throw new DgException(SR.GetString(SR.There_is_no_item_with_id,
"User", userId));
5426.
5427.
5428.         /*Μήπως είναι ήδη συνδεδεμένος ο χρήστης με αυτό το site:*/
5429.         var existingGrant = DigitDLL.GetUserGrantForPhase(accessToken, phaseId, userId);

5430.         if (existingGrant != null)
5431.         {
5432.             if (existingGrant.PhaseRole != phaseRole)
5433.             {
5434.                 existingGrant.PhaseRole = phaseRole;
5435.                 existingGrant.UpdateBy = accessToken.UserId;
5436.                 existingGrant.UpdatedDt = Utility.Now();
5437.
5438.                 return DigitDLL.UpdateUserGrant(accessToken, existingGrant);
5439.             }
5440.             else
5441.             {
5442.                 return existingGrant;
5443.             }
5444.         }
5445.         else
5446.         {
5447.             DgUserGrant usersite = new DgUserGrant();
5448.             usersite.User = user.UserId;
5449.             usersite.Phase = phase.PhaseId;
5450.             usersite.PhaseRole = phaseRole;
5451.
5452.             usersite.CreateBy = usersite.UpdateBy = accessToken.UserId;
5453.             usersite.CreatedDt = usersite.UpdatedDt = Utility.Now();
5454.
5455.
5456.             return DigitDLL.CreateUserGrant(accessToken, usersite);
5457.         }
5458.     }
5459.     /// <summary>
5460.     ///
5461.     /// </summary>
5462.     /// <param name="accessToken"></param>
5463.     /// <param name="userId"></param>
5464.     /// <param name="phaseId"></param>
5465.     public void RevokePhaseFromUser(DgAccessToken accessToken, Int32 userId, Int32 phaseI
d)
5466.     {
5467.         if (accessToken == null) throw new ArgumentNullException("accessToken");
5468.
5469.         #region SecurityLayer
5470.         if (accessToken.IsSysAdmin == false)
5471.         {
5472.             throw new DgException("Requires SysAdmin permission!");
5473.         }
5474.         #endregion
5475.
5476.
5477.         var existingGrant = DigitDLL.GetUserGrantForPhase(accessToken, phaseId, userId);
5478.
5479.         if (existingGrant != null)
5480.         {
5481.             DigitDLL.DeleteUserGrant(accessToken, existingGrant.Usergrantid);
5482.         }
5482.     }

```

```

5483.
5484.     /// <summary>
5485.     /// Επιτρέπει τους χρήστες που έχουν δικαιώματα επάνω στην συγκεκριμένη φάση:
5486.     /// </summary>
5487.     /// <param name="accessToken"></param>
5488.     /// <param name="phaseId"></param>
5489.     /// <returns></returns>
5490.     public IList<DgvUserGrant> GetUserGrantsForPhase(DgAccessToken accessToken, Int32 pha
seId)
5491.     {
5492.         if (accessToken == null) throw new ArgumentNullException("accessToken");
5493.
5494.         #region SecurityLayer
5495.
5496.         #endregion
5497.
5498.         return DigitDLL.GetUserGrantsForPhase_View(accessToken, phaseId);
5499.     }
5500.     #endregion
5501.
5502.     #region DgCont
5503.     public DgCont GetContById(DgAccessToken accessToken, Int32 contId)
5504.     {
5505.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
5506.
5507.         #region SecurityLayer
5508.         #endregion
5509.
5510.         return DigitDLL.GetContById(accessToken, contId);
5511.     }
5512.
5513.     public IList<DgCont> GetConts(DgAccessToken accessToken, string whereClause = null, s
tring orderByClause = "order by name")
5514.     {
5515.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
5516.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
5517.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
5518.
5519.         #region SecurityLayer
5520.         #endregion
5521.
5522.         return DigitDLL.GetConts(accessToken, whereClause, orderByClause);
5523.     }
5524.
5525.
5526.
5527.     /// <summary>
5528.     /// Επιτρέπει όλα τα ορατά conts για τον χρήστη μας, στα οποία μπορεί να εργαστεί
5529.     /// </summary>
5530.     /// <param name="accessToken"></param>
5531.     /// <param name="userId"></param>
5532.     /// <returns></returns>
5533.     public IList<DgwSite> GetVisibleHierarchyForUser(DgAccessToken accessToken, int userI
d)
5534.     {
5535.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
5536.
5537.         #region SecurityLayer
5538.         #endregion
5539.
5540.         return DigitDLL.GetVisibleHierarchyForUser(accessToken, userId);

```

```

5541.         }
5542.
5543.
5544.         public IList<DgCont> GetConts(DgAccessToken accessToken, int pageIndex, int pageSize,
5545.             ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
5546.         {
5547.             if (accessToken == null) throw new ArgumentNullException("accessToken");
5548.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
5549.             ;
5550.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
5551. se");
5552.
5553.             #region SecurityLayer
5554.
5555.             #endregion
5556.
5557.             Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
5558.             int startRowIndex = ((pageIndex - 1) * pageSize);
5559.
5560.             return DigitDLL.GetConts(accessToken, startRowIndex, pageSize, ref totalRows, whe
5561. reClause, orderByClause);
5562.         }
5563.         public IList<DgvCont> GetConts_View(DgAccessToken accessToken, int pageIndex, int pag
5564. eSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
5565.         {
5566.             if (accessToken == null) throw new ArgumentNullException("accessToken");
5567.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
5568.             ;
5569.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
5570. se");
5571.
5572.             #region SecurityLayer
5573.
5574.             #endregion
5575.
5576.             Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
5577.             int startRowIndex = ((pageIndex - 1) * pageSize);
5578.
5579.             return DigitDLL.GetConts_View(accessToken, startRowIndex, pageSize, ref totalRows
5580. , whereClause, orderByClause);
5581.         }
5582.
5583.         void _checkCont(DgAccessToken accessToken, DgCont cont, DgCont existingCont)
5584.         {
5585.             /*Ελέγχουμε τον supervisor:*/
5586.             var supervisor = DigitDLL.GetUserById(accessToken, cont.Supervisor);
5587.             if (supervisor == null)
5588.             {
5589.                 throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "User", cont.
5590. Supervisor));
5591.             }
5592.
5593.             #region LatLong
5594.             if (!cont.Latitude.HasValue || !cont.Longitude.HasValue)
5595.             {
5596.                 cont.Latitude = null;
5597.                 cont.Longitude = null;
5598.                 cont.MapZoomLevel = 4;
5599.             }
5600.             #endregion
5601.
5602.
5603.
5604.
5605.
5606.
5607.
5608.
5609.
5610.
5611.
5612.
5613.
5614.
5615.
5616.
5617.
5618.
5619.
5620.
5621.
5622.
5623.
5624.
5625.
5626.
5627.
5628.
5629.
5630.
5631.
5632.
5633.
5634.
5635.
5636.
5637.
5638.
5639.
5640.
5641.
5642.
5643.
5644.
5645.
5646.
5647.
5648.
5649.
5650.
5651.
5652.
5653.
5654.
5655.
5656.
5657.
5658.
5659.
5660.
5661.
5662.
5663.
5664.
5665.
5666.
5667.
5668.
5669.
5670.
5671.
5672.
5673.
5674.
5675.
5676.
5677.
5678.
5679.
5680.
5681.
5682.
5683.
5684.
5685.
5686.
5687.
5688.
5689.
5690.
5691.
5692.
5693.
5694.
5695.
5696.
5697.
5698.
5699.
5700.
5701.
5702.
5703.
5704.
5705.
5706.
5707.
5708.
5709.
5710.
5711.
5712.
5713.
5714.
5715.
5716.
5717.
5718.
5719.
5720.
5721.
5722.
5723.
5724.
5725.
5726.
5727.
5728.
5729.
5730.
5731.
5732.
5733.
5734.
5735.
5736.
5737.
5738.
5739.
5740.
5741.
5742.
5743.
5744.
5745.
5746.
5747.
5748.
5749.
5750.
5751.
5752.
5753.
5754.
5755.
5756.
5757.
5758.
5759.
5760.
5761.
5762.
5763.
5764.
5765.
5766.
5767.
5768.
5769.
5770.
5771.
5772.
5773.
5774.
5775.
5776.
5777.
5778.
5779.
5780.
5781.
5782.
5783.
5784.
5785.
5786.
5787.
5788.
5789.
5790.
5791.
5792.
5793.
5794.
5795.
5796.
5797.
5798.
5799.
5800.
5801.
5802.
5803.
5804.
5805.
5806.
5807.
5808.
5809.
5810.
5811.
5812.
5813.
5814.
5815.
5816.
5817.
5818.
5819.
5820.
5821.
5822.
5823.
5824.
5825.
5826.
5827.
5828.
5829.
5830.
5831.
5832.
5833.
5834.
5835.
5836.
5837.
5838.
5839.
5840.
5841.
5842.
5843.
5844.
5845.
5846.
5847.
5848.
5849.
5850.
5851.
5852.
5853.
5854.
5855.
5856.
5857.
5858.
5859.
5860.
5861.
5862.
5863.
5864.
5865.
5866.
5867.
5868.
5869.
5870.
5871.
5872.
5873.
5874.
5875.
5876.
5877.
5878.
5879.
5880.
5881.
5882.
5883.
5884.
5885.
5886.
5887.
5888.
5889.
5890.
5891.
5892.
5893.
5894.
5895.
5896.
5897.
5898.
5899.
5900.

```

```

5595.
5596.
5597.         //ελεγχος του conttype
5598.         var conttype = DigitDLL.GetContTypeById(accessToken, cont.Conttype);
5599.         if (conttype == null)
5600.         {
5601.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContType", c
ont.Conttype));
5602.         }
5603.         //ελεγχος του contmethod
5604.         var contmethod = DigitDLL.GetContMethodById(accessToken, cont.Contmethod);
5605.         if (contmethod == null)
5606.         {
5607.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContMethod",
cont.Contmethod));
5608.         }
5609.         //ελεγχος του contcategory
5610.         var contcategory = DigitDLL.GetContCategoryById(accessToken, cont.Contcategory);
5611.
5612.         if (contcategory == null)
5613.         {
5614.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContCategory
", cont.Contcategory));
5615.         }
5616.         //ελεγχος του contgrade
5617.         var contgrade = DigitDLL.GetContGradeById(accessToken, cont.Contgrade);
5618.         if (contgrade == null)
5619.         {
5620.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContGrade",
cont.Contgrade));
5621.         }
5622.         //ελεγχος του contslope
5623.         var contslope = DigitDLL.GetContSlopeById(accessToken, cont.Contslope);
5624.         if (contslope == null)
5625.         {
5626.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContSlope",
cont.Contslope));
5627.         }
5628.         //ελεγχος του contprofile
5629.         var contprofile = DigitDLL.GetContProfileById(accessToken, cont.Contprofile);
5630.         if (contprofile == null)
5631.         {
5632.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContProfile"
, cont.Contprofile));
5633.         }
5634.         //ελεγχος του contedgedef
5635.         var contedgedef = DigitDLL.GetContEdgeDefById(accessToken, cont.Contedgedef);
5636.         if (contedgedef == null)
5637.         {
5638.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContEdgeDef"
, cont.Contedgedef));
5639.         }
5640.         //ελεγχος του contconsist
5641.         var contconsist = DigitDLL.GetContConsistById(accessToken, cont.Contconsist);
5642.         if (contconsist == null)
5643.         {
5644.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContConsist"
, cont.Contconsist));
5645.         }
5646.         //ελεγχος του conttexture
5647.         var conttexture = DigitDLL.GetContTextureById(accessToken, cont.Conttexture);
5648.         if (conttexture == null)
5649.         {

```

```

5649.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContTexture"
, cont.Conttexture));
5650.         }
5651.         //ελεγχος του contcorse
5652.         var contcorse = DigitDLL.GetContCorseById(accessToken, cont.Contcorse);
5653.         if (contcorse == null)
5654.         {
5655.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContCorse",
cont.Contcorse));
5656.         }
5657.         //ελεγχος του contgrid
5658.         var contgrid = DigitDLL.GetContGridById(accessToken, cont.Contgrid);
5659.         if (contgrid == null)
5660.         {
5661.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContGrid", c
ont.Contgrid));
5662.         }
5663.
5664.
5665.
5666.         if (cont.Height.HasValue && cont.Height.Value < 0)
5667.         {
5668.             throw new DgException("Height is invalid!");
5669.         }
5670.         if (cont.Width.HasValue && cont.Width.Value < 0)
5671.         {
5672.             throw new DgException("Width is invalid!");
5673.         }
5674.         if (cont.Length.HasValue && cont.Length.Value < 0)
5675.         {
5676.             throw new DgException("Length is invalid!");
5677.         }
5678.         if (cont.Mindepth.HasValue && cont.Mindepth.Value < 0)
5679.         {
5680.             throw new DgException("Mindepth is invalid!");
5681.         }
5682.         if (cont.Maxdepth.HasValue && cont.Maxdepth.Value < 0)
5683.         {
5684.             throw new DgException("Maxdepth is invalid!");
5685.         }
5686.         if (cont.Volume.HasValue && cont.Volume.Value < 0)
5687.         {
5688.             throw new DgException("Volume is invalid!");
5689.         }
5690.         if (cont.Arbitrary)
5691.         {
5692.             if(cont.Length.HasValue == false)
5693.             {
5694.                 throw new DgException("The cont is arbitrary, but length is undefined!");
5695.             }
5696.         }
5697.     }
5698.     void _checkContPermissionsForWork(DgAccessToken accessToken, DgCont cont)
5699.     {
5700.         bool hasPermissions = false;
5701.         if (accessToken.IsSysAdmin == true)
5702.         {
5703.             hasPermissions = true;
5704.         }
5705.         if (cont.Supervisor == accessToken.UserId)
5706.         {
5707.             hasPermissions = true;

```

```

5708.     }
5709.     if(hasPermissions == false)
5710.     {
5711.         foreach (var grant in accessToken.UserGrants)
5712.         {
5713.             if (grant.Cont == cont.ContId || grant.Phase == cont.Phase || grant.Site
== cont.Site)
5714.             {
5715.                 hasPermissions = true;
5716.                 break;
5717.             }
5718.         }
5719.     }
5720.     if (hasPermissions == false)
5721.     {
5722.         throw new DgException(string.Format("You do not have permission on cont '{0}'
!", cont.Name));
5723.     }
5724. }
5725.
5726.
5727.     public DgCont CreateCont(DgAccessToken accessToken, DgCont cont)
5728.     {
5729.         #region input parameters validation
5730.         if (accessToken == null) throw new ArgumentNullException("accessToken");
5731.         if (cont == null) throw new ArgumentNullException("cont");
5732.         cont.ValidateInstance();
5733.         #endregion
5734.
5735.         #region SecurityLayer
5736.         if (accessToken.IsSysAdmin == false)
5737.         {
5738.             throw new DgException("Requires SysAdmin permission!");
5739.         }
5740.         #endregion
5741.
5742.         /*Ελεγχος του phase*/
5743.         var phase = DigitDLL.GetPhaseById(accessToken, cont.Phase);
5744.         if (phase == null)
5745.         {
5746.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Phase", phas
e.Site));
5747.         }
5748.         cont.Site = phase.Site;
5749.
5750.         /*Check Name uniqueness*/
5751.         var existingRecord = DigitDLL.GetContByName(accessToken, phase.PhaseId, cont.Name
);
5752.         if (existingRecord != null)
5753.         {
5754.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", cont.N
ame));
5755.         }
5756.
5757.         /*Τώρα θα φτιάξουμε το dirName του site*/
5758.         cont.DirName = Utility.ConvertToGreeklish(cont.Name);
5759.         int numOfTries = 1;
5760.         while (DigitDLL.GetContByDirName(accessToken, phase.PhaseId, cont.DirName) != nul
l)
5761.         {
5762.             cont.DirName = Utility.ConvertToGreeklish(cont.Name) + string.Format("_{0:D2}
", numOfTries);
5763.         }

```

```

5764.
5765.         _checkCont(accessToken, cont, null);
5766.
5767.         cont.ShowOrder = 0;
5768.
5769.         cont.CreateBy = cont.UpdateBy = accessToken.UserId;
5770.         cont.CreateDt = cont.UpdateDt = Utility.Now();
5771.
5772.         return DigitDLL.CreateCont(accessToken, cont);
5773.     }
5774.
5775.     public DgCont UpdateCont(DgAccessToken accessToken, DgCont cont)
5776.     {
5777.         #region input parameters validation
5778.         if (accessToken == null) throw new ArgumentNullException("accessToken");
5779.         if (cont == null) throw new ArgumentNullException("cont");
5780.         cont.ValidateInstance();
5781.         #endregion
5782.
5783.         #region SecurityLayer
5784.         if (accessToken.IsSysAdmin == false)
5785.         {
5786.             throw new DgException("Requires SysAdmin permission!");
5787.         }
5788.         #endregion
5789.
5790.
5791.         var existingRecord = DigitDLL.GetContById(accessToken, cont.ContId);
5792.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "cont", cont.ContId));
5793.
5794.         if (existingRecord.Phase != cont.Phase)
5795.         {
5796.             throw new DgException("A cont cannot change phase!");
5797.         }
5798.
5799.         if (existingRecord.Name != cont.Name )
5800.         {
5801.             var _existingRecordName = DigitDLL.GetContByName(accessToken, cont.Phase, con
t.Name);
5802.             if (_existingRecordName != null && _existingRecordName.ContId != cont.ContId)
5803.             {
5804.                 throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", co
nt.Name));
5805.             }
5806.         }
5807.
5808.
5809.         _checkCont(accessToken, cont, existingRecord);
5810.
5811.         //Site
5812.         //Phase
5813.         existingRecord.Name = cont.Name;
5814.         existingRecord.Summary = cont.Summary;
5815.         existingRecord.Description = cont.Description;
5816.         existingRecord.Arbitrary = cont.Arbitrary;
5817.         existingRecord.Supervisor = cont.Supervisor;
5818.         existingRecord.Height = cont.Height;
5819.         existingRecord.Width = cont.Width;
5820.         existingRecord.Length = cont.Length;
5821.         existingRecord.Mindepth = cont.Mindepth;
5822.         existingRecord.Maxdepth = cont.Maxdepth;

```



```

5823.         existingRecord.Volume = cont.Volume;
5824.         existingRecord.Latitude = cont.Latitude;
5825.         existingRecord.Longitude = cont.Longitude;
5826.         existingRecord.MapZoomLevel = cont.MapZoomLevel;
5827.         existingRecord.ShowOrder = cont.ShowOrder;
5828.
5829.
5830.         existingRecord.Conttype = cont.Conttype;
5831.         existingRecord.Contmethod = cont.Contmethod;
5832.         existingRecord.Contcategory = cont.Contcategory;
5833.         existingRecord.Contgrade = cont.Contgrade;
5834.         existingRecord.Contslope = cont.Contslope;
5835.         existingRecord.Contprofile = cont.Contprofile;
5836.         existingRecord.Contedgedef = cont.Contedgedef;
5837.         existingRecord.Contconsist = cont.Contconsist;
5838.         existingRecord.Conttexture = cont.Conttexture;
5839.         existingRecord.Contcorce = cont.Contcorce;
5840.         existingRecord.Contgrid = cont.Contgrid;
5841.
5842.         existingRecord.DirName = cont.DirName;
5843.
5844.         existingRecord.UpdateBy = accessToken.UserId;
5845.         existingRecord.UpdateDt = Utility.Now();
5846.
5847.         return DigitDLL.UpdateCont(accessToken, existingRecord);
5848.     }
5849.
5850.     public void DeleteCont(DgAccessToken accessToken, Int32 contId)
5851.     {
5852.         if (accessToken == null) throw new ArgumentNullException("accessToken");
5853.         var existingRecord = DigitDLL.GetContById(accessToken, contId);
5854.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "cont", contId));
5855.
5856.         #region SecurityLayer
5857.         if (accessToken.IsSysAdmin == false)
5858.         {
5859.             throw new DgException("Requires SysAdmin permission!");
5860.         }
5861.         #endregion
5862.
5863.
5864.         //Can be deleted?
5865.         DigitDLL.CanDeleteCont(accessToken, existingRecord.ContId);
5866.
5867.         #region βρίσκουμε phase, site and geosite:
5868.         //Βρίσκουμε την φάση
5869.         var phase = GetPhaseById(accessToken, existingRecord.Phase);
5870.         if (phase == null)
5871.         {
5872.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Phase", exis
tingRecord.Phase));
5873.         }
5874.         if (string.IsNullOrEmpty(phase.DirName))
5875.         {
5876.             throw new DgException("Invalid Phase.DirName");
5877.         }
5878.         //Βρίσκουμε το site:
5879.         var site = GetSiteById(accessToken, phase.Site);
5880.         if (site == null)
5881.         {
5882.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Site", phase
.Site));

```

```

5883.         }
5884.         if (string.IsNullOrEmpty(site.DirName))
5885.         {
5886.             throw new DgException("Invalid Site.DirName");
5887.         }
5888.         //Βρίσκουμε το geosite στο οποίο ανήκει αυτό το site:
5889.         var geosite = GetGeositeById(accessToken, site.Geosite);
5890.         if (geosite == null)
5891.         {
5892.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Geosite", si
te.Geosite));
5893.         }
5894.         if (string.IsNullOrEmpty(geosite.DirName))
5895.         {
5896.             throw new DgException("Invalid geosite.DirName");
5897.         }
5898.         #endregion
5899.
5900.
5901.         //Διαγράφουμε όλα τα stratums:
5902.         var stratums = DigitDLL.GetStratums(accessToken, existingRecord.ContId, null, nul
l);
5903.         foreach(var item in stratums)
5904.         {
5905.             DigitDLL.DeleteStratum(accessToken, item.StratumId);
5906.         }
5907.         //Πρέπει να διαγράψουμε όλα τα usergrants
5908.         var grants = DigitDLL.GetUserGrantsForCont_View(accessToken, existingRecord.ContI
d);
5909.         foreach(var grant in grants)
5910.         {
5911.             DigitDLL.DeleteUserGrant(accessToken, grant.Usergrantid);
5912.         }
5913.         //Πρέπει να διαγράψουμε όλα τα συσχετιζόμενα αρχεία του:
5914.         var files = GetContImages(accessToken, existingRecord.ContId, DgImageCategory.All
);
5915.         foreach (var _file in files)
5916.         {
5917.             _DeleteFile(accessToken, _file);
5918.         }
5919.         //Διαγράφουμε το directory:
5920.         string inventoryPath = Path.Combine(RootDirectory, geosite.DirName, site.DirName,
phase.DirName, existingRecord.DirName);
5921.         if (Directory.Exists(inventoryPath))
5922.         {
5923.             Directory.Delete(inventoryPath, true);
5924.         }
5925.
5926.
5927.         //Πραγματοποιούμε την διαγραφή (substrarums/stratums/cont):
5928.         DigitDLL.DeleteCont(accessToken, existingRecord.ContId);
5929.     }
5930.
5931.
5932.
5933.
5934.     public IList<DgFile> GetContImages(DgAccessToken accessToken, Int32 contId, DgImageCa
tegory? imgCategory)
5935.     {
5936.         var orderByClause = "order by createdt asc";
5937.
5938.         if (imgCategory == null || imgCategory == DgImageCategory.All)
5939.         {

```

```

5940.         var whereClause = string.Format("where ownername='conts' and ownerId={0}", co
ntId);
5941.         return DigitDLL.GetFiles(accessToken, whereClause, orderByClause);
5942.     }
5943.     else
5944.     {
5945.         var whereClause = string.Format("where ownername='conts' and ownerId={0} and
imagecategory={1}", contId, (byte)imgCategory);
5946.         return DigitDLL.GetFiles(accessToken, whereClause, orderByClause);
5947.     }
5948. }
5949. public DgFile UploadContImage(DgAccessToken accessToken, Int32 contId, System.Byte[]
buffer, string mimeType, string fileName, DgImageCategory imgCategory)
5950. {
5951.     if (accessToken == null) throw new ArgumentNullException("accessToken");
5952.     if (buffer.Length == 0) throw new ArgumentException("buffer cannot be empty!");
5953.     if (String.IsNullOrEmpty(mimeType)) throw new ArgumentException("mimeType cannot
be empty!");
5954.     if (String.IsNullOrEmpty(fileName)) throw new ArgumentException("fileName cannot
be empty!");
5955.
5956.
5957.     #region SecurityLayer
5958.     if (accessToken.IsSysAdmin == false)
5959.     {
5960.         throw new DgException("Requires SysAdmin permission!");
5961.     }
5962.     #endregion
5963.
5964.
5965.     //Βρίσκουμε το πλαίσιο
5966.     var cont = GetContById(accessToken, contId);
5967.     if (cont == null)
5968.     {
5969.         throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Cont", contI
d));
5970.     }
5971.     if (string.IsNullOrEmpty(cont.DirName))
5972.     {
5973.         throw new DgException("Invalid Cont.DirName");
5974.     }
5975.     #region βρίσκουμε phase, site and geosite:
5976.     //Μετά βρίσκουμε την φάση στην οποία ανήκει το πλαίσιο:
5977.     var phase = GetPhaseById(accessToken, cont.Phase);
5978.     if (phase == null)
5979.     {
5980.         throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Phase", cont
.Phase));
5981.     }
5982.     if (string.IsNullOrEmpty(phase.DirName))
5983.     {
5984.         throw new DgException("Invalid Phase.DirName");
5985.     }
5986.     //Μετά βρίσκουμε το site στο οποίο ανήκει η φάση:
5987.     var site = DigitDLL.GetSiteById(accessToken, phase.Site);
5988.     if (site == null)
5989.     {
5990.         throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Site", phase
.Site));
5991.     }
5992.     if (string.IsNullOrEmpty(site.DirName))
5993.     {
5994.         throw new DgException("Invalid Site.DirName");

```

```

5995.         }
5996.         //Βρίσκουμε το geosite στο οποίο ανήκει αυτό το site:
5997.         var geosite = DigitDLL.GetGeositeById(accessToken, site.Geosite);
5998.         if (geosite == null)
5999.         {
6000.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Geosite", si
te.Geosite));
6001.         }
6002.         if (string.IsNullOrEmpty(geosite.DirName))
6003.         {
6004.             throw new DgException("Invalid geosite.DirName");
6005.         }
6006.         #endregion
6007.
6008.
6009.         DgFile digitFile = new DgFile();
6010.         digitFile.OwnerName = "conts";
6011.         digitFile.OwnerId = cont.ContId;
6012.         digitFile.Filetype = DgFileType.Image;
6013.         digitFile.Imagecategory = imgCategory;
6014.         digitFile.Orgfilename = fileName;
6015.         digitFile.Filepath = Path.Combine(geosite.DirName, site.DirName, phase.DirName, c
ont.DirName);
6016.
6017.
6018.         return _UploadFile(accessToken, digitFile, buffer);
6019.     }
6020.     public void DeleteContFile(DgAccessToken accessToken, Int32 contId, Int32 fileId)
6021.     {
6022.         if (accessToken == null) throw new ArgumentNullException("accessToken");
6023.
6024.
6025.         #region SecurityLayer
6026.         if (accessToken.IsSysAdmin == false)
6027.         {
6028.             throw new DgException("Requires SysAdmin permission!");
6029.         }
6030.         #endregion
6031.
6032.         var cont = GetContById(accessToken, contId);
6033.         if (cont == null)
6034.         {
6035.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Cont", contI
d));
6036.         }
6037.
6038.         var file = DigitDLL.GetFileById(accessToken, "conts", cont.ContId, fileId);
6039.         if (file == null)
6040.         {
6041.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "File", fileI
d));
6042.         }
6043.
6044.         _DeleteFile(accessToken, file);
6045.     }
6046.
6047.
6048.
6049.     /// <summary>
6050.     ///
6051.     /// </summary>
6052.     /// <param name="accessToken"></param>
6053.     /// <param name="userId"></param>

```

```

6054.         /// <param name="contId"></param>
6055.         /// <returns></returns>
6056.         public DgUserGrant GetUserGrantForCont(DgAccessToken accessToken, Int32 userId, Int32
contId)
6057.         {
6058.             if (accessToken == null) throw new ArgumentNullException("accessToken");
6059.
6060.             return DigitDLL.GetUserGrantForCont(accessToken, contId, userId);
6061.         }
6062.         /// <summary>
6063.         ///
6064.         /// </summary>
6065.         /// <param name="accessToken"></param>
6066.         /// <param name="userId"></param>
6067.         /// <param name="contId"></param>
6068.         /// <param name="contRole"></param>
6069.         /// <returns></returns>
6070.         public DgUserGrant GrantContToUser(DgAccessToken accessToken, Int32 userId, Int32 con
tId, DgEntityRole contRole)
6071.         {
6072.             if (accessToken == null) throw new ArgumentNullException("accessToken");
6073.
6074.             #region SecurityLayer
6075.             if (accessToken.IsSysAdmin == false)
6076.             {
6077.                 throw new DgException("Requires SysAdmin permission!");
6078.             }
6079.             #endregion
6080.
6081.             var cont = DigitDLL.GetContById(accessToken, contId);
6082.             if (cont == null) throw new DgException(SR.GetString(SR.There_is_no_item_with_id,
"Cont", contId));
6083.
6084.             var user = DigitDLL.GetUserById(accessToken, userId);
6085.             if (user == null) throw new DgException(SR.GetString(SR.There_is_no_item_with_id,
"User", userId));
6086.
6087.
6088.             /*Μήπως είναι ήδη συνδεδεμένος ο χρήστης με αυτό το site:*/
6089.             var existingGrant = DigitDLL.GetUserGrantForCont(accessToken, contId, userId);
6090.             if (existingGrant != null)
6091.             {
6092.                 if (existingGrant.ContRole != contRole)
6093.                 {
6094.                     existingGrant.ContRole = contRole;
6095.                     existingGrant.UpdateBy = accessToken.UserId;
6096.                     existingGrant.UpdateDt = Utility.Now();
6097.
6098.                     return DigitDLL.UpdateUserGrant(accessToken, existingGrant);
6099.                 }
6100.                 else
6101.                 {
6102.                     return existingGrant;
6103.                 }
6104.             }
6105.             else
6106.             {
6107.                 DgUserGrant usersite = new DgUserGrant();
6108.                 usersite.User = user.UserId;
6109.                 usersite.Cont = cont.ContId;
6110.                 usersite.ContRole = contRole;
6111.
6112.                 usersite.CreateBy = usersite.UpdateBy = accessToken.UserId;

```

```

6113.             usersite.CreatedDt = usersite.UpdateDt = Utility.Now();
6114.
6115.
6116.             return DigitDLL.CreateUserGrant(accessToken, usersite);
6117.         }
6118.     }
6119.     /// <summary>
6120.     ///
6121.     /// </summary>
6122.     /// <param name="accessToken"></param>
6123.     /// <param name="userId"></param>
6124.     /// <param name="contId"></param>
6125.     public void RevokeContFromUser(DgAccessToken accessToken, Int32 userId, Int32 contId)
6126.     {
6127.         if (accessToken == null) throw new ArgumentNullException("accessToken");
6128.
6129.         #region SecurityLayer
6130.         if (accessToken.IsSysAdmin == false)
6131.         {
6132.             throw new DgException("Requires SysAdmin permission!");
6133.         }
6134.         #endregion
6135.
6136.
6137.         var existingGrant = DigitDLL.GetUserGrantForCont(accessToken, contId, userId);
6138.         if (existingGrant != null)
6139.         {
6140.             DigitDLL.DeleteUserGrant(accessToken, existingGrant.Usergrantid);
6141.         }
6142.     }
6143.
6144.     /// <summary>
6145.     /// Επιτρέπει τους χρήστες που έχουν δικαιώματα επάνω στην συγκεκριμένη φάση:
6146.     /// </summary>
6147.     /// <param name="accessToken"></param>
6148.     /// <param name="contId"></param>
6149.     /// <returns></returns>
6150.     public IList<DgvUserGrant> GetUserGrantsForCont(DgAccessToken accessToken, Int32 cont
    Id)
6151.     {
6152.         if (accessToken == null) throw new ArgumentNullException("accessToken");
6153.
6154.         #region SecurityLayer
6155.
6156.         #endregion
6157.
6158.         return DigitDLL.GetUserGrantsForCont_View(accessToken, contId);
6159.     }
6160.     #endregion
6161.
6162.
6163.     #region DgStratum
6164.     public DgStratum GetStratumById(DgAccessToken accessToken, Int32 stratumId)
6165.     {
6166.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
6167.
6168.         #region SecurityLayer
6169.         #endregion
6170.
6171.         return DigitDLL.GetStratumById(accessToken, stratumId);
6172.     }
6173.

```

```

6174.         /// <summary>
6175.         /// Επιστρέφει τα stratums για το συγκεκριμένο πλαίσιο
6176.         /// </summary>
6177.         /// <param name="accessToken"></param>
6178.         /// <param name="contid"></param>
6179.         /// <param name="whereClause"></param>
6180.         /// <param name="orderByClause"></param>
6181.         /// <returns></returns>
6182.         public IList<DgStratum> GetStratums(DgAccessToken accessToken, Int32 contid, string w
hereClause = null, string orderByClause = "order by Name")
6183.         {
6184.             if (accessToken == null) throw new ArgumentNullException("accesstoken");
6185.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
6186.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
6187.
6188.             #region SecurityLayer
6189.             #endregion
6190.
6191.             return DigitDLL.GetStratums(accessToken, contid, whereClause, orderByClause);
6192.         }
6193.
6194.         /// <summary>
6195.         /// Επιστρέφει τα stratums για το συγκεκριμένο πλαίσιο
6196.         /// </summary>
6197.         /// <param name="accessToken"></param>
6198.         /// <param name="contid"></param>
6199.         /// <param name="pageIndex"></param>
6200.         /// <param name="pageSize"></param>
6201.         /// <param name="totalRows"></param>
6202.         /// <param name="whereClause"></param>
6203.         /// <param name="orderByClause"></param>
6204.         /// <returns></returns>
6205.         public IList<DgStratum> GetStratums(DgAccessToken accessToken, Int32 contid, int page
Index, int pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order b
y Name")
6206.         {
6207.             if (accessToken == null) throw new ArgumentNullException("accessToken");
6208.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
6209.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
6210.
6211.             #region SecurityLayer
6212.
6213.             #endregion
6214.
6215.             Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
6216.             int startRowIndex = ((pageIndex - 1) * pageSize);
6217.
6218.             return DigitDLL.GetStratums(accessToken, contid, startRowIndex, pageSize, ref tot
alRows, whereClause, orderByClause);
6219.         }
6220.
6221.         /// <summary>
6222.         /// Διαβάζει όλα τα Stratums για το συγκεκριμένο πλαίσιο (contid).
6223.         /// Παράλληλα για κάθε stratum διαβάζει όλα τα substratums αυτού!
6224.         /// </summary>
6225.         /// <param name="accessToken"></param>
6226.         /// <param name="contid"></param>
6227.         /// <returns></returns>
6228.         public IList<DgStratum> GetStratumsTree(DgAccessToken accessToken, Int32 contid)

```

```

6229.         {
6230.             if (accessToken == null) throw new ArgumentNullException("accessToken");
6231.
6232.             #region SecurityLayer
6233.
6234.             #endregion
6235.
6236.             return DigitDLL.GetStratumsTree(accessToken, contid);
6237.         }
6238.
6239.         public DgStratum CreateStratum(DgAccessToken accessToken, DgStratum stratum)
6240.         {
6241.             #region input parameters validation
6242.             if (accessToken == null) throw new ArgumentNullException("accessToken");
6243.             if (stratum == null) throw new ArgumentNullException("stratum");
6244.             stratum.ValidateInstance();
6245.             #endregion
6246.
6247.             #region SecurityLayer
6248.             if (accessToken.IsSysAdmin == false)
6249.             {
6250.                 throw new DgException("Requires SysAdmin permission!");
6251.             }
6252.             #endregion
6253.
6254.             /**/
6255.             var cont = DigitDLL.GetContById(accessToken, stratum.Cont);
6256.             if(cont == null)
6257.             {
6258.                 throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Cont", stratum.Cont));
6259.             }
6260.             /*Check Name uniqueness*/
6261.             var existingRecord = DigitDLL.GetStratumByName(accessToken, stratum.Cont, stratum.Name);
6262.             if (existingRecord != null)
6263.             {
6264.                 throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", stratum.Name));
6265.             }
6266.
6267.             stratum.CreateBy = stratum.UpdateBy = accessToken.UserId;
6268.             stratum.CreateDt = stratum.UpdateDt = Utility.Now();
6269.
6270.             return DigitDLL.CreateStratum(accessToken, stratum);
6271.         }
6272.
6273.         public DgStratum UpdateStratum(DgAccessToken accessToken, DgStratum stratum)
6274.         {
6275.             #region input parameters validation
6276.             if (accessToken == null) throw new ArgumentNullException("accessToken");
6277.             if (stratum == null) throw new ArgumentNullException("stratum");
6278.             stratum.ValidateInstance();
6279.             #endregion
6280.
6281.             #region SecurityLayer
6282.             if (accessToken.IsSysAdmin == false)
6283.             {
6284.                 throw new DgException("Requires SysAdmin permission!");
6285.             }
6286.             #endregion
6287.
6288.

```



```

6289.         var existingRecord = DigitDLL.GetStratumById(accessToken, stratum.StratumId);
6290.         if (existingRecord == null)
6291.         {
6292.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "stratum", stratum.StratumId));
6293.         }
6294.         if(existingRecord.Cont != stratum.Cont)
6295.         {
6296.             throw new DgException("Cannot change cont");
6297.         }
6298.
6299.         var _existingRecord = DigitDLL.GetStratumByName(accessToken, stratum.Cont, stratum.Name);
6300.         if (_existingRecord != null && _existingRecord.StratumId != stratum.StratumId)
6301.         {
6302.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", stratum.Name));
6303.         }
6304.
6305.
6306.         existingRecord.Name = stratum.Name;
6307.         existingRecord.Description = stratum.Description;
6308.         existingRecord.Colour = stratum.Colour;
6309.
6310.         existingRecord.UpdateBy = accessToken.UserId;
6311.         existingRecord.UpdateDt = Utility.Now();
6312.
6313.         return DigitDLL.UpdateStratum(accessToken, existingRecord);
6314.     }
6315.
6316.     public void DeleteStratum(DgAccessToken accessToken, Int32 stratumId)
6317.     {
6318.         if (accessToken == null) throw new ArgumentNullException("accessToken");
6319.         var existingRecord = DigitDLL.GetStratumById(accessToken, stratumId);
6320.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Stratum", stratumId));
6321.
6322.         #region SecurityLayer
6323.         if (accessToken.IsSysAdmin == false)
6324.         {
6325.             throw new DgException("Requires SysAdmin permission!");
6326.         }
6327.         #endregion
6328.
6329.         //Can be deleted?
6330.         DigitDLL.CanDeleteStratum(accessToken, existingRecord.StratumId);
6331.
6332.         DigitDLL.DeleteStratum(accessToken, existingRecord.StratumId);
6333.     }
6334.     #endregion
6335.
6336.     #region DgSubStratum
6337.     public DgSubStratum GetSubStratumById(DgAccessToken accessToken, Int32 substratumId)
6338.     {
6339.         if (accessToken == null) throw new ArgumentNullException("accessToken");
6340.
6341.         #region SecurityLayer
6342.         #endregion
6343.
6344.         return DigitDLL.GetSubStratumById(accessToken, substratumId);
6345.     }
6346.

```

```

6347.
6348.         public IList<DgSubStratum> GetSubStratums(DgAccessToken accessToken, int stratum, string whereClause, string orderByClause)
6349.         {
6350.             if (accessToken == null) throw new ArgumentNullException("accessToken");
6351.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
6352.             ;
6353.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClause");
6354.             #region SecurityLayer
6355.             #endregion
6356.
6357.             return DigitDLL.GetSubStratums(accessToken, stratum, whereClause, orderByClause);
6358.         }
6359.
6360.
6361.         public IList<DgSubStratum> GetSubStratums(DgAccessToken accessToken, int stratum, int pageIndex, int pageSize, ref int totalRows, string whereClause = null, string orderByClause = "order by Name")
6362.         {
6363.             if (accessToken == null) throw new ArgumentNullException("accessToken");
6364.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause");
6365.             ;
6366.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClause");
6367.             #region SecurityLayer
6368.             #endregion
6369.
6370.             Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
6371.             int startRowIndex = ((pageIndex - 1) * pageSize);
6372.
6373.             return DigitDLL.GetSubStratums(accessToken, stratum, startRowIndex, pageSize, ref totalRows, whereClause, orderByClause);
6374.         }
6375.
6376.
6377.         public DgSubStratum CreateSubStratum(DgAccessToken accessToken, DgSubStratum subStratum)
6378.         {
6379.             #region input parameters validation
6380.             if (accessToken == null) throw new ArgumentNullException("accessToken");
6381.             if (subStratum == null) throw new ArgumentNullException("subStratum");
6382.             subStratum.ValidateInstance();
6383.             #endregion
6384.
6385.             #region SecurityLayer
6386.             if (accessToken.IsSysAdmin == false)
6387.             {
6388.                 throw new DgException("Requires SysAdmin permission!");
6389.             }
6390.             #endregion
6391.
6392.
6393.             var stratum = DigitDLL.GetStratumById(accessToken, subStratum.Stratum);
6394.             if(stratum == null)
6395.             {
6396.                 throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Stratum", subStratum.Stratum));
6397.             }
6398.             /*Check Name uniqueness*/

```

```

6399.         var existingRecord = DigitDLL.GetSubStratumByName(accessToken, subStratum.Stratum
, subStratum.Name);
6400.         if (existingRecord != null)
6401.         {
6402.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", subStr
atum.Name));
6403.         }
6404.
6405.         subStratum.CreateBy = subStratum.UpdateBy = accessToken.UserId;
6406.         subStratum.CreateDt = subStratum.UpdateDt = Utility.Now();
6407.
6408.         return DigitDLL.CreateSubStratum(accessToken, subStratum);
6409.     }
6410.
6411.     public DgSubStratum UpdateSubStratum(DgAccessToken accessToken, DgSubStratum subStrat
um)
6412.     {
6413.         #region input parameters validation
6414.         if (accessToken == null) throw new ArgumentNullException("accessToken");
6415.         if (subStratum == null) throw new ArgumentNullException("subStratum");
6416.         subStratum.ValidateInstance();
6417.         #endregion
6418.
6419.         #region SecurityLayer
6420.         if (accessToken.IsSysAdmin == false)
6421.         {
6422.             throw new DgException("Requires SysAdmin permission!");
6423.         }
6424.         #endregion
6425.
6426.
6427.         var existingRecord = DigitDLL.GetSubStratumById(accessToken, subStratum.Substratu
mId);
6428.         if (existingRecord == null)
6429.         {
6430.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "SubStratum",
subStratum.SubstratumId));
6431.         }
6432.         if(existingRecord.Stratum != subStratum.Stratum)
6433.         {
6434.             throw new DgException("Cannot change stratum");
6435.         }
6436.
6437.         var _existingRecord = DigitDLL.GetSubStratumByName(accessToken, subStratum.Stratu
m, subStratum.Name);
6438.         if (_existingRecord != null && _existingRecord.SubstratumId != subStratum.Substra
tumId)
6439.         {
6440.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Name", subStr
atum.Name));
6441.         }
6442.
6443.
6444.         existingRecord.SubstratumId = subStratum.SubstratumId;
6445.         existingRecord.Name = subStratum.Name;
6446.         existingRecord.Description = subStratum.Description;
6447.
6448.         existingRecord.UpdateBy = accessToken.UserId;
6449.         existingRecord.UpdateDt = Utility.Now();
6450.
6451.         return DigitDLL.UpdateSubStratum(accessToken, existingRecord);
6452.     }
6453.

```

```

6454.         public void DeleteSubStratum(DgAccessToken accessToken, Int32 substratumId)
6455.         {
6456.             if (accessToken == null) throw new ArgumentNullException("accessToken");
6457.             var existingRecord = DigitDLL.GetSubStratumById(accessToken, substratumId);
6458.             if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "SubStratum", substratumId));
6459.
6460.             #region SecurityLayer
6461.             if (accessToken.IsSysAdmin == false)
6462.             {
6463.                 throw new DgException("Requires SysAdmin permission!");
6464.             }
6465.             #endregion
6466.
6467.             //Can be deleted?
6468.             DigitDLL.CanDeleteSubStratum(accessToken, existingRecord.SubstratumId);
6469.
6470.
6471.             DigitDLL.DeleteSubStratum(accessToken, existingRecord.SubstratumId);
6472.         }
6473.         #endregion
6474.
6475.
6476.         #region DgDiaryEntry & DgContFind
6477.
6478.         public DgDiaryEntry GetDiaryEntryById(DgAccessToken accessToken, Int32 diaryId)
6479.         {
6480.             if (accessToken == null) throw new ArgumentNullException("accesstoken");
6481.
6482.             #region SecurityLayer
6483.             #endregion
6484.
6485.             return DigitDLL.GetDiaryEntryById(accessToken, diaryId);
6486.         }
6487.
6488.         /// <summary>
6489.         ///
6490.         /// </summary>
6491.         /// <param name="accessToken"></param>
6492.         /// <param name="contid"></param>
6493.         /// <param name="startRowIndex"></param>
6494.         /// <param name="maximumRows"></param>
6495.         /// <param name="totalRows"></param>
6496.         /// <param name="whereClause"></param>
6497.         /// <param name="orderByClause"></param>
6498.         /// <returns></returns>
6499.         public IList<DgvDiaryEntry> GetDiaryEntriesForCont(DgAccessToken accessToken, Int32 c
ontid, int pageIndex, int pageSize, ref int totalRows, string whereClause = null, string orderByCl
ause = "order by createDt Desc")
6500.         {
6501.             if (accessToken == null) throw new ArgumentNullException("accesstoken");
6502.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
6503.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
6504.
6505.             #region SecurityLayer
6506.             #endregion
6507.
6508.             Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
6509.             int startRowIndex = ((pageIndex - 1) * pageSize);
6510.

```

```

6511.         return DigitDLL.GetDiaryEntriesForCont(accessToken, contid, startRowIndex, pageSi
ze, ref totalRows, whereClause, orderByClause);
6512.     }
6513.
6514.     /// <summary>
6515.     /// Επιστρέφει τις τελευταίες εγγραφές του ημερολογίου που μπορεί να βλέπει ο τρέχων
χρήστης.
6516.     /// <para>0 τρέχων χρήστης μπορεί να βλέπει εγγραφές ημερολογίου που αφορούν τα πλαίσια
τα στα οποία έχει "δικαιώματα"</para>
6517.     /// </summary>
6518.     /// <param name="accessToken"></param>
6519.     /// <param name="topRecords"></param>
6520.     /// <returns></returns>
6521.     public IList<DgvDiaryEntry> GetLatestVisibleDiaryEntriesForUser(DgAccessToken accessT
oken, int topRecords)
6522.     {
6523.         if (accessToken == null) throw new ArgumentNullException("accessToken");
6524.         if(topRecords <= 0)
6525.         {
6526.             throw new ArgumentOutOfRangeException("topRecords");
6527.         }
6528.
6529.         return DigitDLL.GetLatestVisibleDiaryEntriesForUser(accessToken, topRecords);
6530.     }
6531.
6532.     public DgDiaryEntry CreateDiaryEntry(DgAccessToken accessToken, DgDiaryEntry entry)
6533.     {
6534.         #region input parameters validation
6535.         if (accessToken == null) throw new ArgumentNullException("accessToken");
6536.         if (entry == null) throw new ArgumentNullException("entry");
6537.         entry.ValidateInstance();
6538.         #endregion
6539.
6540.
6541.         /*Πρέπει να βρούμε το πλαίσιο στο οποίο ανήκει η εγγραφή μας:*/
6542.         var cont = DigitDLL.GetContById(accessToken, entry.Cont);
6543.         if (cont == null)
6544.         {
6545.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Cont", entry
.Cont));
6546.         }
6547.         entry.Phase = cont.Phase;
6548.         entry.Site = cont.Site;
6549.
6550.
6551.         #region Εχει δικαιομα ο χρήστης μας να προσθέτει εγγραφές στο ημερολόγιο σε αυτό
το πλαίσιο?
6552.         _checkContPermissionsForWork(accessToken, cont);
6553.         #endregion
6554.
6555.
6556.
6557.
6558.         entry.CreateBy = entry.UpdateBy = accessToken.UserId;
6559.         entry.CreateDt = entry.UpdateDt = Utility.Now();
6560.
6561.         /*Τώρα θα φτιάξουμε το dirName της εγγραφής μας:*/
6562.         entry.DirName = string.Format("entry_{0}", entry.CreateDt.ToString("yyyyMMddHHmm
s"));
6563.         int numOfTries = 1;
6564.         while (DigitDLL.GetDiaryEntryByDirName(accessToken, entry.Cont, entry.DirName) !=
null)
6565.         {

```

```

6566.         cont.DirName = string.Format("entry_{0}", entry.CreateDt.ToString("yyyyMMddHH
mmss")) + string.Format("_{0:D2}", numOfTries);
6567.     }
6568.
6569.
6570.         return DigitDLL.CreateDiaryEntry(accessToken, entry);
6571.     }
6572.
6573.     public DgDiaryEntry UpdateDiaryEntry(DgAccessToken accessToken, DgDiaryEntry entry)
6574.     {
6575.         #region input parameters validation
6576.         if (accessToken == null) throw new ArgumentNullException("accessToken");
6577.         if (entry == null) throw new ArgumentNullException("entry");
6578.         entry.ValidateInstance();
6579.         #endregion
6580.
6581.         /*Πρέπει να βρούμε το πλαίσιο στο οποίο ανήκει η εγγραφή μας:*/
6582.         var cont = DigitDLL.GetContById(accessToken, entry.Cont);
6583.         if (cont == null)
6584.         {
6585.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Cont", entry
.Cont));
6586.         }
6587.
6588.
6589.         #region Έχει δικαίωμα ο χρήστης μας να προσθέτει εγγραφές στο ημερολόγιο σε αυτό
το πλαίσιο?
6590.         _checkContPermissionsForWork(accessToken, cont);
6591.         #endregion
6592.
6593.
6594.         var existingRecord = DigitDLL.GetDiaryEntryById(accessToken, entry.DiaryId);
6595.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "Diary", entry.DiaryId));
6596.
6597.         existingRecord.Title = entry.Title;
6598.         existingRecord.Body = entry.Body;
6599.
6600.         existingRecord.UpdateBy = accessToken.UserId;
6601.         existingRecord.UpdateDt = Utility.Now();
6602.
6603.         return DigitDLL.UpdateDiaryEntry(accessToken, existingRecord);
6604.     }
6605.
6606.     public void DeleteDiaryEntry(DgAccessToken accessToken, Int32 diaryId)
6607.     {
6608.         if (accessToken == null) throw new ArgumentNullException("accessToken");
6609.         var existingRecord = DigitDLL.GetDiaryEntryById(accessToken, diaryId);
6610.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "Diary", diaryId));
6611.
6612.
6613.         /*Πρέπει να βρούμε το πλαίσιο στο οποίο ανήκει η εγγραφή μας:*/
6614.         var cont = DigitDLL.GetContById(accessToken, existingRecord.Cont);
6615.         if (cont == null)
6616.         {
6617.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Cont", exist
ingRecord.Cont));
6618.         }
6619.
6620.         #region Έχει δικαίωμα ο χρήστης μας να προσθέτει εγγραφές στο ημερολόγιο σε αυτό
το πλαίσιο?
6621.         _checkContPermissionsForWork(accessToken, cont);

```

```

6622.         #endregion
6623.
6624.
6625.         //Can be deleted?
6626.         DigitDLL.CanDeleteDiaryEntry(accessToken, existingRecord.DiaryId);
6627.
6628.
6629.
6630.         #region διαβάζουμε phase, site & geosite:
6631.         //Μετά βρίσκουμε την φάση στην οποία ανήκει το πλαίσιο:
6632.         var phase = DigitDLL.GetPhaseById(accessToken, cont.Phase);
6633.         if (phase == null)
6634.         {
6635.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Phase", cont
6636. .Phase));
6636.         }
6637.         if (string.IsNullOrEmpty(phase.DirName))
6638.         {
6639.             throw new DgException("Invalid phase.DirName");
6640.         }
6641.         //Μετά βρίσκουμε το site στο οποίο ανήκει η φάση:
6642.         var site = DigitDLL.GetSiteById(accessToken, phase.Site);
6643.         if (site == null)
6644.         {
6645.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Site", phase
6646. .Site));
6646.         }
6647.         if (string.IsNullOrEmpty(site.DirName))
6648.         {
6649.             throw new DgException("Invalid site.DirName");
6650.         }
6651.         //Βρίσκουμε το geosite στο οποίο ανήκει αυτό το site:
6652.         var geosite = DigitDLL.GetGeositeById(accessToken, site.Geosite);
6653.         if (geosite == null)
6654.         {
6655.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Geosite", si
6656. te.Geosite));
6656.         }
6657.         if (string.IsNullOrEmpty(geosite.DirName))
6658.         {
6659.             throw new DgException("Invalid geosite.DirName");
6660.         }
6661.         #endregion
6662.
6663.
6664.         //Πρέπει να διαγράψουμε όλα τα συσχετιζόμενα αρχεία του:
6665.         var files = GetDiaryImages(accessToken, existingRecord.DiaryId, DgImageCategory.A
6666. ll);
6666.         foreach (var _file in files)
6667.         {
6668.             _DeleteFile(accessToken, _file);
6669.         }
6670.         //Διαγράφουμε το directory:
6671.         string inventoryPath = Path.Combine(RootDirectory, geosite.DirName, site.DirName,
6672. phase.DirName, cont.DirName, existingRecord.DirName);
6672.         if (Directory.Exists(inventoryPath))
6673.         {
6674.             Directory.Delete(inventoryPath, true);
6675.         }
6676.
6677.
6678.
6679.

```

```

6680.
6681.         //Πραγματοποιούμε την διαγραφή (substrarums/stratums/cont):
6682.         DigitDLL.DeleteDiaryEntry(accessToken, existingRecord.DiaryId);
6683.     }
6684.
6685.     public IList<DgFile> GetDiaryImages(DgAccessToken accessToken, Int32 diaryId, DgImage
        Category? imgCategory)
6686.     {
6687.         var orderByClause = "order by createdt asc";
6688.
6689.         if (imgCategory == null || imgCategory == DgImageCategory.All)
6690.         {
6691.             var whereClause = string.Format("where ownername='diary' and ownerId={0}", di
                aryId);
6692.             return DigitDLL.GetFiles(accessToken, whereClause, orderByClause);
6693.         }
6694.         else
6695.         {
6696.             var whereClause = string.Format("where ownername='diary' and ownerId={0} and
                imagecategory={1}", diaryId, (byte)imgCategory);
6697.             return DigitDLL.GetFiles(accessToken, whereClause, orderByClause);
6698.         }
6699.     }
6700.     public DgFile UploadDiaryImage(DgAccessToken accessToken, Int32 diaryId, System.Byte[
        ] buffer, string mimeType, string fileName, DgImageCategory imgCategory)
6701.     {
6702.         if (accessToken == null) throw new ArgumentNullException("accessToken");
6703.         if (buffer.Length == 0) throw new ArgumentException("buffer cannot be empty!");
6704.         if (String.IsNullOrEmpty(mimeType)) throw new ArgumentException("mimeType cannot
                be empty!");
6705.         if (String.IsNullOrEmpty(fileName)) throw new ArgumentException("fileName cannot
                be empty!");
6706.
6707.
6708.         //Βρίσκουμε, καταρχάς το εύρημα:
6709.         var diary = DigitDLL.GetDiaryEntryById(accessToken, diaryId);
6710.         if (diary == null)
6711.         {
6712.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Diary", diar
                yId));
6713.         }
6714.         //Μετά βρίσκουμε το πλαίσιο στο οποίο ανήκει το εύρημα:
6715.         var cont = DigitDLL.GetContById(accessToken, diary.Cont);
6716.         if (cont == null)
6717.         {
6718.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Cont", diary
                .Cont));
6719.         }
6720.         if (string.IsNullOrWhiteSpace(cont.DirName))
6721.         {
6722.             throw new DgException("Invalid Cont.DirName");
6723.         }
6724.
6725.
6726.         #region Έχει δικαίωμα ο χρήστης μας να χειρίζεται ευρήματα σε αυτό το πλαίσιο?
6727.         _checkContPermissionsForWork(accessToken, cont);
6728.         #endregion
6729.
6730.
6731.         #region διαβάζουμε phase, site & geosite:
6732.         //Μετά βρίσκουμε την φάση στην οποία ανήκει το πλαίσιο:
6733.         var phase = DigitDLL.GetPhaseById(accessToken, cont.Phase);
6734.         if (phase == null)

```



```

6735.         {
6736.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Phase", cont
.Phase));
6737.         }
6738.         if (string.IsNullOrEmpty(phase.DirName))
6739.         {
6740.             throw new DgException("Invalid phase.DirName");
6741.         }
6742.         //Μετά βρίσκουμε το site στο οποίο ανήκει η φάση:
6743.         var site = DigitDLL.GetSiteById(accessToken, phase.Site);
6744.         if (site == null)
6745.         {
6746.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Site", phase
.Site));
6747.         }
6748.         if (string.IsNullOrEmpty(site.DirName))
6749.         {
6750.             throw new DgException("Invalid Site.DirName");
6751.         }
6752.         //Βρίσκουμε το geosite στο οποίο ανήκει αυτό το site:
6753.         var geosite = DigitDLL.GetGeositeById(accessToken, site.Geosite);
6754.         if (geosite == null)
6755.         {
6756.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Geosite", si
te.Geosite));
6757.         }
6758.         if (string.IsNullOrEmpty(geosite.DirName))
6759.         {
6760.             throw new DgException("Invalid geosite.DirName");
6761.         }
6762.         #endregion
6763.
6764.
6765.         DgFile digitFile = new DgFile();
6766.         digitFile.OwnerName = "diary";
6767.         digitFile.OwnerId = diary.DiaryId;
6768.         digitFile.Filetype = DgFileType.Image;
6769.         digitFile.Imagecategory = imgCategory;
6770.         digitFile.Orgfilename = fileName;
6771.         digitFile.Filepath = Path.Combine(geosite.DirName, site.DirName, phase.DirName, c
ont.DirName, diary.DirName);
6772.
6773.
6774.         return _UploadFile(accessToken, digitFile, buffer);
6775.     }
6776.     public void DeleteDiaryFile(DgAccessToken accessToken, Int32 diaryId, Int32 fileId)
6777.     {
6778.         if (accessToken == null) throw new ArgumentNullException("accessToken");
6779.
6780.
6781.         //Βρίσκουμε, καταρχάς το εύρημα:
6782.         var diary = DigitDLL.GetDiaryEntryById(accessToken, diaryId);
6783.         if (diary == null)
6784.         {
6785.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Diary", diar
yId));
6786.         }
6787.         //Μετά βρίσκουμε το πλαίσιο στο οποίο ανήκει το εύρημα:
6788.         var cont = DigitDLL.GetContById(accessToken, diary.Cont);
6789.         if (cont == null)
6790.         {
6791.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Cont", diary
.Cont));

```

```

6792.         }
6793.
6794.
6795.         #region Εχει δικαιωμα ο χρήστης μας να χειρίζεται εγγραφές ημερολογίου σε αυτό το
πλαίσιο?
6796.         _checkContPermissionsForWork(accessToken, cont);
6797.         #endregion
6798.
6799.
6800.         //Βρίσκουμε το αρχείο
6801.         var file = DigitDLL.GetFileById(accessToken, "diary", diary.DiaryId, fileId);
6802.         if (file == null)
6803.         {
6804.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "File", fileId));
6805.         }
6806.
6807.         _DeleteFile(accessToken, file);
6808.     }
6809.
6810.
6811.
6812.
6813.
6814.     public DgContFind GetContFindById(DgAccessToken accessToken, Int32 contfindid)
6815.     {
6816.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
6817.
6818.         #region SecurityLayer
6819.         #endregion
6820.
6821.         return DigitDLL.GetContFindById(accessToken, contfindid);
6822.     }
6823.
6824.     /// <summary>
6825.     /// Επιστρέφει τα ευρήματα που ανήκουν στο συγκεκριμένο πλαίσιο (contid)
6826.     /// </summary>
6827.     /// <param name="accessToken"></param>
6828.     /// <param name="contid"></param>
6829.     /// <param name="whereClause"></param>
6830.     /// <param name="orderByClause"></param>
6831.     /// <returns></returns>
6832.     public IList<DgContFind> GetContFindsForCont(DgAccessToken accessToken, Int32 contid,
string whereClause = null, string orderByClause="order by code")
6833.     {
6834.         if (accessToken == null) throw new ArgumentNullException("accesstoken");
6835.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
6836.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
6837.
6838.         #region SecurityLayer
6839.         #endregion
6840.
6841.         return DigitDLL.GetContFindsForCont(accessToken, contid, whereClause, orderByClau
se);
6842.     }
6843.     /// <summary>
6844.     /// Επιστρέφει τα ευρήματα που ανήκουν στο συγκεκριμένο πλαίσιο (contid)
6845.     /// </summary>
6846.     /// <param name="accessToken"></param>
6847.     /// <param name="contid"></param>
6848.     /// <param name="pageIndex"></param>

```

```

6849.         /// <param name="pageSize"></param>
6850.         /// <param name="totalRows"></param>
6851.         /// <param name="whereClause"></param>
6852.         /// <param name="orderByClause"></param>
6853.         /// <returns></returns>
6854.         public IList<DgContFind> GetContFindsForCont(DgAccessToken accessToken, Int32 contid,
int pageIndex, int pageSize, ref int totalRows, string whereClause = null, string orderByClause =
"order by code")
6855.         {
6856.             if (accessToken == null) throw new ArgumentNullException("accessToken");
6857.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
6858.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
6859.
6860.             #region SecurityLayer
6861.
6862.             #endregion
6863.
6864.             Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
6865.             int startRowIndex = ((pageIndex - 1) * pageSize);
6866.
6867.             return DigitDLL.GetContFindsForCont(accessToken, contid, startRowIndex, pageSize,
ref totalRows, whereClause, orderByClause);
6868.         }
6869.         /// <summary>
6870.         /// Επιστρέφει τα ευρήματα που ανήκουν στο συγκεκριμένο πλαίσιο (contid)
6871.         /// </summary>
6872.         /// <param name="accessToken"></param>
6873.         /// <param name="contid"></param>
6874.         /// <param name="pageIndex"></param>
6875.         /// <param name="pageSize"></param>
6876.         /// <param name="totalRows"></param>
6877.         /// <param name="whereClause"></param>
6878.         /// <param name="orderByClause"></param>
6879.         /// <returns></returns>
6880.         public IList<DgvContFind> GetContFindsForCont_View(DgAccessToken accessToken, Int32 c
ontid, int pageIndex, int pageSize, ref int totalRows, string whereClause = null, string orderByCl
ause = "order by code")
6881.         {
6882.             if (accessToken == null) throw new ArgumentNullException("accessToken");
6883.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
6884.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
6885.
6886.             #region SecurityLayer
6887.
6888.             #endregion
6889.
6890.             Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
6891.             int startRowIndex = ((pageIndex - 1) * pageSize);
6892.
6893.             return DigitDLL.GetContFindsForCont_View(accessToken, contid, startRowIndex, page
Size, ref totalRows, whereClause, orderByClause);
6894.         }
6895.
6896.
6897.         /// <summary>
6898.         /// Επιστρέφει τα ευρήματα που ανήκουν στο συγκεκριμένο diary entry
6899.         /// </summary>
6900.         /// <param name="accessToken"></param>
6901.         /// <param name="diaryId"></param>

```

```

6902.         /// <param name="whereClause"></param>
6903.         /// <param name="orderByClause"></param>
6904.         /// <returns></returns>
6905.         public IList<DgContFind> GetContFindsForDiaryEntry(DgAccessToken accessToken, Int32 d
diaryId, string whereClause = null, string orderByClause = "order by code")
6906.         {
6907.             if (accessToken == null) throw new ArgumentNullException("accesstoken");
6908.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
6909.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
6910.
6911.             #region SecurityLayer
6912.             #endregion
6913.
6914.             return DigitDLL.GetContFindsForDiaryEntry(accessToken, diaryId, whereClause, orde
rByClause);
6915.         }
6916.         /// <summary>
6917.         /// Επιστρέφει τα ευρήματα που ανήκουν στο συγκεκριμένο diary entry
6918.         /// </summary>
6919.         /// <param name="accessToken"></param>
6920.         /// <param name="diaryId"></param>
6921.         /// <param name="pageIndex"></param>
6922.         /// <param name="pageSize"></param>
6923.         /// <param name="totalRows"></param>
6924.         /// <param name="whereClause"></param>
6925.         /// <param name="orderByClause"></param>
6926.         /// <returns></returns>
6927.         public IList<DgContFind> GetContFindsForDiaryEntry(DgAccessToken accessToken, Int32 d
diaryId, int pageIndex, int pageSize, ref int totalRows, string whereClause = null, string orderByC
lause = "order by code")
6928.         {
6929.             if (accessToken == null) throw new ArgumentNullException("accessToken");
6930.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
6931.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
6932.
6933.             #region SecurityLayer
6934.             #endregion
6935.
6936.             Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
6937.             int startRowIndex = ((pageIndex - 1) * pageSize);
6938.
6939.             return DigitDLL.GetContFindsForDiaryEntry(accessToken, diaryId, startRowIndex, pa
geSize, ref totalRows, whereClause, orderByClause);
6940.         }
6941.         /// <summary>
6942.         /// Επιστρέφει τα ευρήματα που ανήκουν στο συγκεκριμένο diary entry
6943.         /// </summary>
6944.         /// <param name="accessToken"></param>
6945.         /// <param name="diaryId"></param>
6946.         /// <param name="pageIndex"></param>
6947.         /// <param name="pageSize"></param>
6948.         /// <param name="totalRows"></param>
6949.         /// <param name="whereClause"></param>
6950.         /// <param name="orderByClause"></param>
6951.         /// <returns></returns>
6952.         public IList<DgvContFind> GetContFindsForDiaryEntry_View(DgAccessToken accessToken, I
nt32 diaryId, int pageIndex, int pageSize, ref int totalRows, string whereClause = null, string or
derByClause = "order by code")

```

```

6954.         {
6955.             if (accessToken == null) throw new ArgumentNullException("accessToken");
6956.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
6957.             ;
6958.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
6959. se");
6960.             #region SecurityLayer
6961.             #endregion
6962.             Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
6963.             int startRowIndex = ((pageIndex - 1) * pageSize);
6964.             return DigitDLL.GetContFindsForDiaryEntry_View(accessToken, diaryId, startRowInde
6965. x, pageSize, ref totalRows, whereClause, orderByClause);
6966.         }
6967.     }
6968.
6969.
6970.
6971.     void _checkContFind(DgAccessToken accessToken, DgContFind contFind, DgContFind existi
6972. ngFinding = null, bool isNew = false)
6973.     {
6974.         //type
6975.         var findtype = DigitDLL.GetContFindTypeById(accessToken, contFind.Type);
6976.         if (findtype == null)
6977.         {
6978.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContFindType
6979. ", contFind.Type));
6980.         }
6981.         //subtype
6982.         if(contFind.Subtype.HasValue)
6983.         {
6984.             var subtype = DigitDLL.GetContFindSubtypeById(accessToken, contFind.Subtype.V
6985. alue);
6986.             if (subtype == null)
6987.             {
6988.                 throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContFind
6989. Subtype", contFind.Subtype.Value));
6990.             }
6991.             if (subtype.FindType != findtype.FindTypeId)
6992.             {
6993.                 throw new DgException("ContFindType,ContFindSubtype Invalid!");
6994.             }
6995.         }
6996.         //category
6997.         var findcategory = DigitDLL.GetContFindCategoryById(accessToken, contFind.Categor
6998. y);
6999.         if (findcategory == null)
7000.         {
7001.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContFindCate
7002. gory", contFind.Category));
7003.         }
7004.         //grouping
7005.         var grouping = DigitDLL.GetContFindGroupById(accessToken, contFind.Grouping);
7006.         if (grouping == null)
7007.         {
7008.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContFindGrou
7009. p", contFind.Grouping));
7010.         }
7011.         //weight

```

```

7007.
7008.         //stratum
7009.         var stratum = DigitDLL.GetStratumById(accessToken, contFind.Stratum);
7010.         if (stratum == null)
7011.         {
7012.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Stratum", co
7013. ntFind.Stratum));
7014.         }
7015.         //substratum
7016.         if (contFind.Substratum.HasValue)
7017.         {
7018.             var substratum = DigitDLL.GetSubStratumById(accessToken, contFind.Substratum.
7019. Value);
7020.             if (substratum == null)
7021.             {
7022.                 throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "SubStrat
7023. um", contFind.Substratum.Value));
7024.             }
7025.             if (substratum.Stratum != stratum.StratumId)
7026.             {
7027.                 throw new DgException("Stratum,substratum Invalid!");
7028.             }
7029.         }
7030.         //dating
7031.         var dating = DigitDLL.GetDatingById(accessToken, contFind.Dating);
7032.         if (dating == null)
7033.         {
7034.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Dating", con
7035. tFind.Dating));
7036.         }
7037.         //datingtype
7038.         var datingtype = DigitDLL.GetDatingTypeById(accessToken, contFind.Datingtype);
7039.         if (datingtype == null)
7040.         {
7041.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "DatingType",
7042. contFind.Datingtype));
7043.         }
7044.         //datingmethod
7045.         var datingmethod = DigitDLL.GetDatingMethodById(accessToken, contFind.Datingmetho
7046. d);
7047.         if (datingmethod == null)
7048.         {
7049.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "DatingMethod
7050. ", contFind.Datingmethod));
7051.         }
7052.     }
7053. }
7054.
7055.     public DgContFind CreateContFind(DgAccessToken accessToken, DgContFind contFind)
7056.     {
7057.         #region input parameters validation
7058.         if (accessToken == null) throw new ArgumentNullException("accessToken");
7059.         if (contFind == null) throw new ArgumentNullException("contFind");
7060.         contFind.ValidateInstance();
7061.         #endregion
7062.
7063.         /*Πρέπει να βρούμε το diary στο οποίο ανηκει το εύρημα:*/
7064.         var diary = DigitDLL.GetDiaryEntryById(accessToken, contFind.Diary);
7065.         if (diary == null)
7066.         {
7067.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Diary", cont
7068. Find.Diary));

```

```

7062.         }
7063.
7064.         /*Πρέπει να βρούμε το πλαίσιο στο οποίο ανήκει το εύρημα:*/
7065.         var cont = DigitDLL.GetContById(accessToken, contFind.Cont);
7066.         if (cont == null)
7067.         {
7068.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Cont", contF
ind.Cont));
7069.         }
7070.         contFind.Phase = cont.Phase;
7071.
7072.
7073.         #region Έχει δικαίωμα ο χρήστης μας να προσθέτει ευρήματα σε αυτό το πλαίσιο?
7074.         _checkContPermissionsForWork(accessToken, cont);
7075.         #endregion
7076.
7077.         _checkContFind(accessToken, contFind, isNew: true);
7078.
7079.
7080.         contFind.CreateBy = contFind.UpdateBy = accessToken.UserId;
7081.         contFind.CreateDt = contFind.UpdateDt = Utility.Now();
7082.
7083.         return DigitDLL.CreateContFind(accessToken, contFind);
7084.     }
7085.
7086.     public DgContFind UpdateContFind(DgAccessToken accessToken, DgContFind contFind)
7087.     {
7088.         #region input parameters validation
7089.         if (accessToken == null) throw new ArgumentNullException("accessToken");
7090.         if (contFind == null) throw new ArgumentNullException("contFind");
7091.         contFind.ValidateInstance();
7092.         #endregion
7093.
7094.         /*Πρέπει να βρούμε το πλαίσιο στο οποίο ανήκει το εύρημα:*/
7095.         var cont = DigitDLL.GetContById(accessToken, contFind.Cont);
7096.         if (cont == null)
7097.         {
7098.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Cont", contF
ind.Cont));
7099.         }
7100.
7101.
7102.         #region Έχει δικαίωμα ο χρήστης μας να χειρίζεται ευρήματα σε αυτό το πλαίσιο?
7103.         _checkContPermissionsForWork(accessToken, cont);
7104.         #endregion
7105.
7106.
7107.         var existingRecord = DigitDLL.GetContFindById(accessToken, contFind.Contfindid);
7108.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "ContFind", contFind.Contfindid));
7109.
7110.         _checkContFind(accessToken, contFind);
7111.
7112.         existingRecord.Name = contFind.Name;
7113.         existingRecord.Summary = contFind.Summary;
7114.         existingRecord.Description = contFind.Description;
7115.         existingRecord.Code = contFind.Code;
7116.         existingRecord.Catalogue = contFind.Catalogue;
7117.         existingRecord.North = contFind.North;
7118.         existingRecord.East = contFind.East;
7119.         existingRecord.Type = contFind.Type;
7120.         existingRecord.Subtype = contFind.Subtype;

```

```

7121.
7122.         existingRecord.Category = contFind.Category;
7123.         existingRecord.Grouping = contFind.Grouping;
7124.         existingRecord.Weight = contFind.Weight;
7125.         existingRecord.Stratum = contFind.Stratum;
7126.         existingRecord.Substratum = contFind.Substratum;
7127.         existingRecord.Dating = contFind.Dating;
7128.         existingRecord.Datingtype = contFind.Datingtype;
7129.         existingRecord.Datingmethod = contFind.Datingmethod;
7130.
7131.         existingRecord.UpdateBy = accessToken.UserId;
7132.         existingRecord.UpdateDt = Utility.Now();
7133.
7134.         return DigitDLL.UpdateContFind(accessToken, existingRecord);
7135.     }
7136.
7137.     public void DeleteContFind(DgAccessToken accessToken, Int32 contfindid)
7138.     {
7139.         if (accessToken == null) throw new ArgumentNullException("accessToken");
7140.         var existingRecord = DigitDLL.GetContFindById(accessToken, contfindid);
7141.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContFind", contfindid));
7142.
7143.
7144.         #region Πρέπει να βρούμε το πλαίσιο στο οποίο ανήκει το εύρημα:
7145.         var cont = DigitDLL.GetContById(accessToken, existingRecord.Cont);
7146.         if (cont == null)
7147.         {
7148.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Cont", existingRecord.Cont));
7149.         }
7150.         if (string.IsNullOrEmpty(cont.DirName))
7151.         {
7152.             throw new DgException("Invalid Cont.DirName");
7153.         }
7154.         #endregion
7155.
7156.
7157.         #region Έχει δικαίωμα ο χρήστης μας να διαγράψει ευρήματα από αυτό το πλαίσιο?
7158.         _checkContPermissionsForWork(accessToken, cont);
7159.         #endregion
7160.
7161.
7162.         #region διαβάζουμε phase, site & geosite:
7163.         //Μετά βρίσκουμε την φάση στην οποία ανήκει το πλαίσιο:
7164.         var phase = DigitDLL.GetPhaseById(accessToken, cont.Phase);
7165.         if (phase == null)
7166.         {
7167.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Phase", cont.Phase));
7168.         }
7169.         if (string.IsNullOrEmpty(phase.DirName))
7170.         {
7171.             throw new DgException("Invalid phase.DirName");
7172.         }
7173.         //Μετά βρίσκουμε το site στο οποίο ανήκει η φάση:
7174.         var site = DigitDLL.GetSiteById(accessToken, phase.Site);
7175.         if (site == null)
7176.         {
7177.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Site", phase.Site));
7178.         }
7179.         if (string.IsNullOrEmpty(site.DirName))

```



```

7180.         {
7181.             throw new DgException("Invalid site.DirName");
7182.         }
7183.         //Βρίσκουμε το geosite στο οποίο ανήκει αυτό το site:
7184.         var geosite = DigitDLL.GetGeositeById(accessToken, site.Geosite);
7185.         if (geosite == null)
7186.         {
7187.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Geosite", si
te.Geosite));
7188.         }
7189.         if (string.IsNullOrEmpty(geosite.DirName))
7190.         {
7191.             throw new DgException("Invalid geosite.DirName");
7192.         }
7193.         #endregion
7194.
7195.
7196.         //Πρέπει να διαγράψουμε όλα τα συσχετιζόμενα αρχεία του:
7197.         var files = GetContFindImages(accessToken, existingRecord.Contfindid, DgImageCate
gory.All);
7198.         foreach (var _file in files)
7199.         {
7200.             _DeleteFile(accessToken, _file);
7201.         }
7202.         //Διαγράφουμε το directory:
7203.         string inventoryPath = Path.Combine(RootDirectory, geosite.DirName, site.DirName,
phase.DirName, cont.DirName, "findings", existingRecord.DirName);
7204.         if (Directory.Exists(inventoryPath))
7205.         {
7206.             Directory.Delete(inventoryPath, true);
7207.         }
7208.
7209.
7210.
7211.
7212.         //Πραγματοποιούμε την διαγραφή (substrarums/stratums/cont):
7213.         DigitDLL.DeleteContFind(accessToken, existingRecord.Contfindid);
7214.     }
7215.
7216.
7217.
7218.
7219.     public IList<DgFile> GetContFindImages(DgAccessToken accessToken, Int32 contfindid, D
gImageCategory? imgCategory)
7220.     {
7221.         var orderByClause = "order by createdt asc";
7222.
7223.         if (imgCategory == null || imgCategory == DgImageCategory.All)
7224.         {
7225.             var whereClause = string.Format("where ownername='contfinds' and ownerId={0}"
, contfindid);
7226.             return DigitDLL.GetFiles(accessToken, whereClause, orderByClause);
7227.         }
7228.         else
7229.         {
7230.             var whereClause = string.Format("where ownername='contfinds' and ownerId={0}
and imagecategory={1}", contfindid, (byte)imgCategory);
7231.             return DigitDLL.GetFiles(accessToken, whereClause, orderByClause);
7232.         }
7233.     }
7234.     public DgFile UploadContFindImage(DgAccessToken accessToken, Int32 contfindid, System
.Byte[] buffer, string mimeType, string fileName, DgImageCategory imgCategory)
7235.     {

```

```

7236.         if (accessToken == null) throw new ArgumentNullException("accessToken");
7237.         if (buffer.Length == 0) throw new ArgumentException("buffer cannot be empty!");
7238.         if (String.IsNullOrEmpty(mimeType)) throw new ArgumentException("mimeType cannot
be empty!");
7239.         if (String.IsNullOrEmpty(fileName)) throw new ArgumentException("fileName cannot
be empty!");
7240.
7241.
7242.         //Βρίσκουμε, καταρχάς το εύρημα:
7243.         var find = DigitDLL.GetContFindById(accessToken, contfindid);
7244.         if (find == null)
7245.         {
7246.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContFind", c
ontfindid));
7247.         }
7248.         //Μετά βρίσκουμε το πλαίσιο στο οποίο ανήκει το εύρημα:
7249.         var cont = DigitDLL.GetContById(accessToken, find.Cont);
7250.         if (cont == null)
7251.         {
7252.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Cont", find.
Cont));
7253.         }
7254.         if (string.IsNullOrEmpty(cont.DirName))
7255.         {
7256.             throw new DgException("Invalid Cont.DirName");
7257.         }
7258.
7259.
7260.         #region Έχει δικαίωμα ο χρήστης μας να χειρίζεται ευρήματα σε αυτό το πλαίσιο?
7261.         _checkContPermissionsForWork(accessToken, cont);
7262.         #endregion
7263.
7264.
7265.         #region διαβάζουμε phase, site & geosite:
7266.         //Μετά βρίσκουμε την φάση στην οποία ανήκει το πλαίσιο:
7267.         var phase = DigitDLL.GetPhaseById(accessToken, cont.Phase);
7268.         if (phase == null)
7269.         {
7270.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Phase", cont
.Phase));
7271.         }
7272.         if (string.IsNullOrEmpty(phase.DirName))
7273.         {
7274.             throw new DgException("Invalid phase.DirName");
7275.         }
7276.         //Μετά βρίσκουμε το site στο οποίο ανήκει η φάση:
7277.         var site = DigitDLL.GetSiteById(accessToken, phase.Site);
7278.         if (site == null)
7279.         {
7280.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Site", phase
.Site));
7281.         }
7282.         if (string.IsNullOrEmpty(site.DirName))
7283.         {
7284.             throw new DgException("Invalid Site.DirName");
7285.         }
7286.         //Βρίσκουμε το geosite στο οποίο ανήκει αυτό το site:
7287.         var geosite = DigitDLL.GetGeositeById(accessToken, site.Geosite);
7288.         if (geosite == null)
7289.         {
7290.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Geosite", si
te.Geosite));
7291.         }

```

```

7292.         if (string.IsNullOrEmpty(geosite.DirName))
7293.         {
7294.             throw new DgException("Invalid geosite.DirName");
7295.         }
7296.         #endregion
7297.
7298.
7299.         DgFile digitFile = new DgFile();
7300.         digitFile.OwnerName = "contfinds";
7301.         digitFile.OwnerId = find.Contfindid;
7302.         digitFile.Filetype = DgFileType.Image;
7303.         digitFile.Imagecategory = imgCategory;
7304.         digitFile.Orgfilename = fileName;
7305.         digitFile.Filepath = Path.Combine(geosite.DirName, site.DirName, phase.DirName, c
ont.DirName, "findings", find.DirName);
7306.
7307.
7308.         return _UploadFile(accessToken, digitFile, buffer);
7309.     }
7310.     public void DeleteContFindFile(DgAccessToken accessToken, Int32 contfindid, Int32 fil
eId)
7311.     {
7312.         if (accessToken == null) throw new ArgumentNullException("accessToken");
7313.
7314.
7315.         //Βρίσκουμε, καταρχάς το εύρημα:
7316.         var find = DigitDLL.GetContFindById(accessToken, contfindid);
7317.         if (find == null)
7318.         {
7319.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "ContFind", c
ontfindid));
7320.         }
7321.         //Μετά βρίσκουμε το πλαίσιο στο οποίο ανήκει το εύρημα:
7322.         var cont = DigitDLL.GetContById(accessToken, find.Cont);
7323.         if (cont == null)
7324.         {
7325.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Cont", find.
Cont));
7326.         }
7327.
7328.
7329.         #region Έχει δικαίωμα ο χρήστης μας να χειρίζεται ευρήματα σε αυτό το πλαίσιο?
7330.         _checkContPermissionsForWork(accessToken, cont);
7331.         #endregion
7332.
7333.
7334.         //Βρίσκουμε το αρχείο
7335.         var file = DigitDLL.GetFilesById(accessToken, "contfinds", find.Contfindid, fileId
);
7336.         if (file == null)
7337.         {
7338.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "File", fileI
d));
7339.         }
7340.
7341.         _DeleteFile(accessToken, file);
7342.     }
7343.
7344.     #endregion
7345.
7346.
7347.     #region DgArticle
7348.     public DgArticle GetArticleById(Int32 articleId)

```

```

7349.         {
7350.
7351.             #region SecurityLayer
7352.             #endregion
7353.
7354.             return DigitDLL.GetArticleById(articleId);
7355.         }
7356.
7357.         public DgArticle GetArticleByUrl(string url)
7358.         {
7359.             if(string.IsNullOrEmpty(url)) throw new ArgumentNullException("url");
7360.
7361.             #region SecurityLayer
7362.             #endregion
7363.
7364.             return DigitDLL.GetArticleByUrl(url.ToLowerInvariant());
7365.         }
7366.
7367.         public IList<DgArticle> GetArticles(int pageIndex, int pageSize, ref int totalRows, s
tring whereClause = null, string orderByClause = "order by createDt desc", bool fetchImages = fals
e)
7368.         {
7369.             Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
;
7370.             Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
se");
7371.
7372.             #region SecurityLayer
7373.             #endregion
7374.
7375.             Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
7376.             int startRowIndex = ((pageIndex - 1) * pageSize);
7377.
7378.
7379.             var articles = DigitDLL.GetArticles(startRowIndex, pageSize, ref totalRows, where
Clause, orderByClause);
7380.
7381.             if(fetchImages)
7382.             {
7383.                 foreach(var item in articles)
7384.                 {
7385.                     item.Images = GetArticleImages(item.ArticleId, DgImageCategory.Photo);
7386.                 }
7387.             }
7388.
7389.             return articles;
7390.         }
7391.
7392.         public DgArticle CreateArticle(DgAccessToken accessToken, DgArticle article)
7393.         {
7394.             #region input parameters validation
7395.             if (accessToken == null) throw new ArgumentNullException("accessToken");
7396.             if (article == null) throw new ArgumentNullException("article");
7397.             article.ValidateInstance();
7398.             #endregion
7399.
7400.             #region SecurityLayer
7401.             if (accessToken.IsSysAdmin == false && accessToken.Mngarticles == false)
7402.             {
7403.                 throw new DgException("Requires Mngarticles permission!");
7404.             }
7405.             #endregion
7406.

```

```

7407.
7408.         /*Ελεγχος του URL:*/
7409.         if(string.IsNullOrEmpty(article.Url))
7410.         {
7411.             article.Url = Utility.ConvertToGreeklish(article.Title);
7412.             int numOfTries = 1;
7413.             while(DigitDLL.GetArticleByUrl(article.Url) != null)
7414.             {
7415.                 article.Url = Utility.ConvertToGreeklish(article.Title) + string.Format(
7416.                     "_{0:D2}", numOfTries);
7417.             }
7418.             var existingRecord = DigitDLL.GetArticleByUrl(article.Url);
7419.             if(existingRecord != null)
7420.             {
7421.                 throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Url", article
7422.                     .Url));
7423.             }
7424.             article.CreateBy = article.UpdateBy = accessToken.UserId;
7425.             article.CreateDt = article.UpdateDt = Utility.Now();
7426.
7427.             return DigitDLL.CreateArticle(accessToken, article);
7428.         }
7429.
7430.     public DgArticle UpdateArticle(DgAccessToken accessToken, DgArticle article)
7431.     {
7432.         #region input parameters validation
7433.         if (accessToken == null) throw new ArgumentNullException("accessToken");
7434.         if (article == null) throw new ArgumentNullException("article");
7435.         article.ValidateInstance();
7436.         #endregion
7437.
7438.         #region SecurityLayer
7439.         if (accessToken.IsSysAdmin == false && accessToken.Mngarticles == false)
7440.         {
7441.             throw new DgException("Requires Mngarticles permission!");
7442.         }
7443.         #endregion
7444.
7445.
7446.         /*Ελεγχος του URL:*/
7447.         if (string.IsNullOrEmpty(article.Url))
7448.         {
7449.             article.Url = Utility.ConvertToGreeklish(article.Title);
7450.             int numOfTries = 1;
7451.             while (DigitDLL.GetArticleByUrl(article.Url) != null)
7452.             {
7453.                 article.Url = Utility.ConvertToGreeklish(article.Title) + string.Format(
7454.                     "_{0:D2}", numOfTries);
7455.             }
7456.             var existingRecord = DigitDLL.GetArticleByUrl(article.Url);
7457.             if (existingRecord != null && existingRecord.ArticleId != article.ArticleId)
7458.             {
7459.                 throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Url", article
7460.                     .Url));
7461.             }
7462.             {
7463.                 existingRecord = DigitDLL.GetArticleById(article.ArticleId);
7464.             }
7465.             if (existingRecord == null)

```

```

7466.         {
7467.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Article", ar
article.ArticleId));
7468.         }
7469.
7470.         existingRecord.Site = article.Site;
7471.         existingRecord.Phase = article.Phase;
7472.         existingRecord.Title = article.Title;
7473.         existingRecord.Summary = article.Summary;
7474.         existingRecord.Body = article.Body;
7475.         existingRecord.Isvisible = article.Isvisible;
7476.         existingRecord.Author = article.Author;
7477.         existingRecord.SourceName = article.SourceName;
7478.         existingRecord.SourceUrl = article.SourceUrl;
7479.         existingRecord.Url = article.Url;
7480.
7481.         existingRecord.UpdateBy = accessToken.UserId;
7482.         existingRecord.UpdateDt = Utility.Now();
7483.
7484.         return DigitDLL.UpdateArticle(accessToken, existingRecord);
7485.     }
7486.
7487.     public void DeleteArticle(DgAccessToken accessToken, Int32 articleId)
7488.     {
7489.         if (accessToken == null) throw new ArgumentNullException("accessToken");
7490.         var existingRecord = DigitDLL.GetArticleById(articleId);
7491.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
m_with_id, "Article", articleId));
7492.
7493.         #region SecurityLayer
7494.         if (accessToken.IsSysAdmin == false && accessToken.Mngarticles == false)
7495.         {
7496.             throw new DgException("Requires Mngarticles permission!");
7497.         }
7498.         #endregion
7499.
7500.
7501.         //Can be deleted?
7502.         DigitDLL.CanDeleteArticle(accessToken, existingRecord.ArticleId);
7503.
7504.
7505.         //Πρέπει να διαγράψουμε όλα τα συσχετιζόμενα αρχεία του:
7506.         var files = GetArticleImages(existingRecord.ArticleId, DgImageCategory.All);
7507.         foreach (var _file in files)
7508.         {
7509.             _DeleteFile(accessToken, _file);
7510.         }
7511.         //Διαγράφουμε το directory:
7512.         string inventoryPath = Path.Combine(RootDirectory, existingRecord.DirName);
7513.         if (Directory.Exists(inventoryPath))
7514.         {
7515.             Directory.Delete(inventoryPath, true);
7516.         }
7517.
7518.         //Πραγματοποιούμε την διαγραφή:
7519.         DigitDLL.DeleteArticle(accessToken, existingRecord.ArticleId);
7520.     }
7521.
7522.
7523.     public IList<DgFile> GetArticleImages(Int32 articleId, DgImageCategory? imgCategory)
7524.     {
7525.         var orderByClause = "order by createDt asc";

```

```

7526.
7527.         if (imgCategory == null || imgCategory == DgImageCategory.All)
7528.         {
7529.             var whereClause = string.Format("where ownername='articles' and ownerId={0}",
articleId);
7530.             return DigitDLL.GetFiles(null, whereClause, orderByClause);
7531.         }
7532.         else
7533.         {
7534.             var whereClause = string.Format("where ownername='articles' and ownerId={0} a
nd imagecategory={1}", articleId, (byte)imgCategory);
7535.             return DigitDLL.GetFiles(null, whereClause, orderByClause);
7536.         }
7537.     }
7538.     public DgFile UploadArticleImage(DgAccessToken accessToken, Int32 articleId, System.B
yte[] buffer, string mimeType, string fileName, DgImageCategory imgCategory)
7539.     {
7540.         if (accessToken == null) throw new ArgumentNullException("accessToken");
7541.         if (buffer.Length == 0) throw new ArgumentException("buffer cannot be empty!");
7542.         if (String.IsNullOrEmpty(mimeType)) throw new ArgumentException("mimeType cannot
be empty!");
7543.         if (String.IsNullOrEmpty(fileName)) throw new ArgumentException("fileName cannot
be empty!");
7544.
7545.
7546.         #region SecurityLayer
7547.         if (accessToken.IsSysAdmin == false && accessToken.Mngarticles == false)
7548.         {
7549.             throw new DgException("Requires Mngarticles permission!");
7550.         }
7551.         #endregion
7552.
7553.         //Βρίσκουμε το άρθρο στο οποίο ανήκει
7554.         var article = DigitDLL.GetArticleById(articleId);
7555.         if(article == null)
7556.         {
7557.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Article", ar
ticleId));
7558.         }
7559.
7560.
7561.         DgFile digitFile = new DgFile();
7562.         digitFile.OwnerName = "articles";
7563.         digitFile.OwnerId = article.ArticleId;
7564.         digitFile.Filetype = DgFileType.Image;
7565.         digitFile.Imagecategory = imgCategory;
7566.         digitFile.Orgfilename = fileName;
7567.         digitFile.Filepath = article.DirName;
7568.
7569.
7570.         return _UploadFile(accessToken, digitFile, buffer);
7571.     }
7572.     public void DeleteArticleFile(DgAccessToken accessToken, Int32 articleId, Int32 fileI
d)
7573.     {
7574.         if (accessToken == null) throw new ArgumentNullException("accessToken");
7575.
7576.
7577.         #region SecurityLayer
7578.         if (accessToken.IsSysAdmin == false && accessToken.Mngarticles == false)
7579.         {
7580.             throw new DgException("Requires Mngarticles permission!");
7581.         }

```

```

7582.         #endregion
7583.
7584.         var article = DigitDLL.GetArticleById(articleId);
7585.         if (article == null)
7586.         {
7587.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Article", ar
7588. ticleId));
7589.         }
7590.         var file = DigitDLL.GetFileById(accessToken, "articles", article.ArticleId, fileI
7591. d);
7592.         if (file == null)
7593.         {
7594.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "File", artic
7595. le.ArticleId));
7596.         }
7597.         _DeleteFile(accessToken, file);
7598.     }
7599.     #endregion
7600.
7601.
7602.     #region DgTutorial
7603.     public DgTutorial GetTutorialById(Int32 tutorialid)
7604.     {
7605.
7606.         #region SecurityLayer
7607.         #endregion
7608.
7609.         return DigitDLL.GetTutorialById(tutorialid);
7610.     }
7611.
7612.     public DgTutorial GetTutorialByUrl(string url)
7613.     {
7614.         if (string.IsNullOrEmpty(url)) throw new ArgumentNullException("url");
7615.
7616.         #region SecurityLayer
7617.         #endregion
7618.
7619.         return DigitDLL.GetTutorialByUrl(url.ToLowerInvariant());
7620.     }
7621.
7622.     public IList<DgTutorial> GetTutorials(int pageIndex, int pageSize, ref int totalRows,
7623. string whereClause = null, string orderByClause = "order by createdt desc", bool fetchImages = fa
7624. lse)
7625.     {
7626.         Utility.CheckParameter(ref whereClause, false, false, false, 2048, "whereClause")
7627. ;
7628.         Utility.CheckParameter(ref orderByClause, false, false, false, 1024, "orderByClau
7629. se");
7630.
7631.         #region SecurityLayer
7632.         #endregion
7633.
7634.         Utility.CheckPagingParameters(ref pageIndex, ref pageSize);
7635.         int startRowIndex = ((pageIndex - 1) * pageSize);
7636.
7637.         var tutorial = DigitDLL.GetTutorials(startRowIndex, pageSize, ref totalRows, wher
7638. eClause, orderByClause);
7639.
7640.         if (fetchImages)

```



```

7637.         {
7638.             foreach (var item in tutorial)
7639.             {
7640.                 item.Images = GetTutorialImages(item.Tutorialid, DgImageCategory.Photo);
7641.             }
7642.         }
7643.
7644.         return tutorial;
7645.     }
7646.
7647.     public DgTutorial CreateTutorial(DgAccessToken accessToken, DgTutorial tutorial)
7648.     {
7649.         #region input parameters validation
7650.         if (accessToken == null) throw new ArgumentNullException("accessToken");
7651.         if (tutorial == null) throw new ArgumentNullException("tutorial");
7652.         tutorial.ValidateInstance();
7653.         #endregion
7654.
7655.         #region SecurityLayer
7656.         if (accessToken.IsSysAdmin == false && accessToken.Mngtutorials == false)
7657.         {
7658.             throw new DgException("Requires Mngtutorials permission!");
7659.         }
7660.         #endregion
7661.
7662.         /*Ελεγχος του URL:*/
7663.         if (string.IsNullOrEmpty(tutorial.Url))
7664.         {
7665.             tutorial.Url = Utility.ConvertToGreeklis(tutorial.Title);
7666.             int numOfTries = 1;
7667.             while (DigitDLL.GetTutorialByUrl(tutorial.Url) != null)
7668.             {
7669.                 tutorial.Url = Utility.ConvertToGreeklis(tutorial.Title) + string.Format
7670.                 ("_{0:D2}", numOfTries);
7671.             }
7672.         }
7673.         var existingRecord = DigitDLL.GetTutorialByUrl(tutorial.Url);
7674.         if (existingRecord != null)
7675.         {
7676.             throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Url", tutorial
7677.             .Url));
7678.         }
7679.         tutorial.CreateBy = tutorial.UpdateBy = accessToken.UserId;
7680.         tutorial.CreateDt = tutorial.UpdateDt = Utility.Now();
7681.
7682.         return DigitDLL.CreateTutorial(accessToken, tutorial);
7683.     }
7684.
7685.     public DgTutorial UpdateTutorial(DgAccessToken accessToken, DgTutorial tutorial)
7686.     {
7687.         #region input parameters validation
7688.         if (accessToken == null) throw new ArgumentNullException("accessToken");
7689.         if (tutorial == null) throw new ArgumentNullException("article");
7690.         tutorial.ValidateInstance();
7691.         #endregion
7692.
7693.         #region SecurityLayer
7694.         if (accessToken.IsSysAdmin == false && accessToken.Mngtutorials == false)
7695.         {
7696.             throw new DgException("Requires Mngtutorials permission!");

```

```

7697.         }
7698.         #endregion
7699.
7700.
7701.         /*Ελεγχος του URL:*/
7702.         if (string.IsNullOrEmpty(tutorial.Url))
7703.         {
7704.             tutorial.Url = Utility.ConvertToGreeklish(tutorial.Title);
7705.             int numOfTries = 1;
7706.             while (DigitDLL.GetTutorialByUrl(tutorial.Url) != null)
7707.             {
7708.                 tutorial.Url = Utility.ConvertToGreeklish(tutorial.Title) + string.Format
7709.                 ("_{0:D2}", numOfTries);
7710.             }
7711.             var existingRecord = DigitDLL.GetTutorialByUrl(tutorial.Url);
7712.             if (existingRecord != null && existingRecord.Tutorialid != tutorial.Tutorialid)
7713.             {
7714.                 throw new DgException(SR.GetString(SR.Value_is_already_in_use, "Url", tutorial
7715.                 l.Url));
7716.             }
7717.             if (existingRecord == null)
7718.             {
7719.                 existingRecord = DigitDLL.GetTutorialById(tutorial.Tutorialid);
7720.             }
7721.             if (existingRecord == null)
7722.             {
7723.                 throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Tutorial", t
7724.                 utorial.Tutorialid));
7725.             }
7726.             existingRecord.Title = tutorial.Title;
7727.             existingRecord.Summary = tutorial.Summary;
7728.             existingRecord.Body = tutorial.Body;
7729.             existingRecord.Isvisible = tutorial.Isvisible;
7730.             existingRecord.Author = tutorial.Author;
7731.             existingRecord.SourceName = tutorial.SourceName;
7732.             existingRecord.SourceUrl = tutorial.SourceUrl;
7733.             existingRecord.Url = tutorial.Url;
7734.             existingRecord.UpdateBy = accessToken.UserId;
7735.             existingRecord.UpdateDt = Utility.Now();
7736.             return DigitDLL.UpdateTutorial(accessToken, existingRecord);
7737.         }
7738.     }
7739.
7740.     public void DeleteTutorial(DgAccessToken accessToken, Int32 tutorialid)
7741.     {
7742.         if (accessToken == null) throw new ArgumentNullException("accessToken");
7743.         var existingRecord = DigitDLL.GetTutorialById(tutorialid);
7744.         if (existingRecord == null) throw new DgException(SR.GetString(SR.There_is_no_ite
7745.         m_with_id, "Tutorial", tutorialid));
7746.         #region SecurityLayer
7747.         if (accessToken.IsSysAdmin == false && accessToken.Mngtutorials == false)
7748.         {
7749.             throw new DgException("Requires Mngtutorials permission!");
7750.         }
7751.         #endregion
7752.
7753.
7754.         //Can be deleted?
7755.         DigitDLL.CanDeleteTutorial(accessToken, existingRecord.Tutorialid);

```

```

7756.
7757.         //Πρέπει να διαγράψουμε όλα τα συσχετιζόμενα αρχεία του:
7758.         var photos = GetTutorialImages(existingRecord.Tutorialid, DgImageCategory.All);
7759.         foreach (var _file in photos)
7760.         {
7761.             _DeleteFile(accessToken, _file);
7762.         }
7763.         var files = GetTutorialAttachments(existingRecord.Tutorialid);
7764.         foreach (var _file in files)
7765.         {
7766.             _DeleteFile(accessToken, _file);
7767.         }
7768.         //Διαγράφουμε το directory:
7769.         string inventoryPath = Path.Combine(RootDirectory, "tutorials", existingRecord.Di
rName);
7770.         if (Directory.Exists(inventoryPath))
7771.         {
7772.             Directory.Delete(inventoryPath, true);
7773.         }
7774.
7775.         //Πραγματοποιούμε την διαγραφή:
7776.         DigitDLL.DeleteTutorial(accessToken, existingRecord.Tutorialid);
7777.     }
7778.
7779.
7780.     public IList<DgFile> GetTutorialImages(Int32 tutorialid, DgImageCategory? imgCategory
)
7781.     {
7782.         var orderByClause = "order by createdt asc";
7783.
7784.         if (imgCategory == null || imgCategory == DgImageCategory.All)
7785.         {
7786.             var whereClause = string.Format("where ownername='tutorials' and ownerId={0}
and filetype = {1}", tutorialid, (byte)DgFileType.Image);
7787.             return DigitDLL.GetFiles(null, whereClause, orderByClause);
7788.         }
7789.         else
7790.         {
7791.             var whereClause = string.Format("where ownername='tutorials' and ownerId={0}
and filetype = {1} and imagecategory={2}", tutorialid, (byte)DgFileType.Image, (byte)imgCategory);
7792.             return DigitDLL.GetFiles(null, whereClause, orderByClause);
7793.         }
7794.     }
7795.     public DgFile UploadTutorialImage(DgAccessToken accessToken, Int32 tutorialid, System
.Byte[] buffer, string mimeType, string fileName, DgImageCategory imgCategory)
7796.     {
7797.         if (accessToken == null) throw new ArgumentNullException("accessToken");
7798.         if (buffer.Length == 0) throw new ArgumentException("buffer cannot be empty!");
7799.         if (String.IsNullOrEmpty(mimeType)) throw new ArgumentException("mimeType cannot
be empty!");
7800.         if (String.IsNullOrEmpty(fileName)) throw new ArgumentException("fileName cannot
be empty!");
7801.
7802.
7803.         #region SecurityLayer
7804.         if (accessToken.IsSysAdmin == false && accessToken.Mngtutorials == false)
7805.         {
7806.             throw new DgException("Requires Mngtutorials permission!");
7807.         }
7808.         #endregion
7809.
7810.         //Βρίσκουμε το άρθρο στο οποίο ανήκει

```

```

7811.         var tutorial = DigitDLL.GetTutorialById(tutorialid);
7812.         if (tutorial == null)
7813.         {
7814.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Tutorial", t
utorialid));
7815.         }
7816.
7817.
7818.         DgFile digitFile = new DgFile();
7819.         digitFile.OwnerName = "tutorials";
7820.         digitFile.OwnerId = tutorial.Tutorialid;
7821.         digitFile.Filetype = DgFileType.Image;
7822.         digitFile.Imagecategory = imgCategory;
7823.         digitFile.Orgfilename = fileName;
7824.         digitFile.Filepath = Path.Combine("tutorials", tutorial.DirName);
7825.
7826.         return _UploadFile(accessToken, digitFile, buffer);
7827.     }
7828.
7829.     public IList<DgFile> GetTutorialAttachments(Int32 tutorialid)
7830.     {
7831.         var orderByClause = "order by createdt asc";
7832.
7833.         var whereClause = string.Format("where ownername='tutorials' and ownerId={0} and
filetype != {1}", tutorialid, (byte)DgFileType.Image);
7834.         return DigitDLL.GetFiles(null, whereClause, orderByClause);
7835.     }
7836.     public DgFile UploadTutorialAttachment(DgAccessToken accessToken, Int32 tutorialid, S
ystem.Byte[] buffer, string mimeType, string fileName)
7837.     {
7838.         if (accessToken == null) throw new ArgumentNullException("accessToken");
7839.         if (buffer.Length == 0) throw new ArgumentException("buffer cannot be empty!");
7840.         if (String.IsNullOrEmpty(mimeType)) throw new ArgumentException("mimeType cannot
be empty!");
7841.         if (String.IsNullOrEmpty(fileName)) throw new ArgumentException("fileName cannot
be empty!");
7842.
7843.
7844.         #region SecurityLayer
7845.         if (accessToken.IsSysAdmin == false && accessToken.Mngtutorials == false)
7846.         {
7847.             throw new DgException("Requires Mngtutorials permission!");
7848.         }
7849.         #endregion
7850.
7851.         //Βρίσκουμε το άρθρο στο οποίο ανήκει
7852.         var tutorial = DigitDLL.GetTutorialById(tutorialid);
7853.         if (tutorial == null)
7854.         {
7855.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Tutorial", t
utorialid));
7856.         }
7857.
7858.         /*
7859.          * Βρίσκουμε το DgFileType
7860.          */
7861.         DgFileType _Filetype = DgFileType.Unknown;
7862.         if (mimeType == "application/pdf")
7863.             _Filetype = DgFileType.PDF;
7864.         if (mimeType == "application/msword" || mimeType == "application/vnd.openxmlforma
ts-officedocument.wordprocessingml.document")
7865.             _Filetype = DgFileType.Word;

```

```

7866.         if (mimeType == "application/excel" || mimeType == "application/x-
excel" || mimeType == "application/x-msexcel" || mimeType == "application/vnd.ms-
excel" || mimeType == "application/vnd.openxmlformats-officedocument.spreadsheetml.sheet")
7867.             _FileType = DgFileType.Excel;
7868.
7869.
7870.             DgFile digitFile = new DgFile();
7871.             digitFile.OwnerName = "tutorials";
7872.             digitFile.OwnerId = tutorial.Tutorialid;
7873.             digitFile.Filetype = _FileType;
7874.             digitFile.Orgfilename = fileName;
7875.             digitFile.Filepath = Path.Combine("tutorials", tutorial.DirName);
7876.
7877.
7878.             return _UploadFile(accessToken, digitFile, buffer);
7879.         }
7880.
7881.     public void DeleteTutorialFile(DgAccessToken accessToken, Int32 tutorialid, Int32 fil
eId)
7882.     {
7883.         if (accessToken == null) throw new ArgumentNullException("accessToken");
7884.
7885.
7886.         #region SecurityLayer
7887.         if (accessToken.IsSysAdmin == false && accessToken.Mngtutorials == false)
7888.         {
7889.             throw new DgException("Requires Mngtutorials permission!");
7890.         }
7891.         #endregion
7892.
7893.         var tutorial = DigitDLL.GetTutorialById(tutorialid);
7894.         if (tutorial == null)
7895.         {
7896.             throw new DgException(SR.GetString(SR.There_is_no_item_with_id, "Tutorial", t
utorialid));
7897.         }
7898.
7899.         var file = DigitDLL.GetFileById(accessToken, "tutorials", tutorial.Tutorialid, fi
leId);
7900.         if (file == null)
7901.             throw new DgException("Unknown file");
7902.
7903.         _DeleteFile(accessToken, file);
7904.     }
7905.     #endregion
7906.
7907.
7908.     }
7909. }

```