ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

΄ΕΝΑ ΣΥΣΤΗΜΑ ΣΗΜΑΣΙΟΛΟΓΙΚΗΣ ΔΙΕΡΕΥΝΗΣΗΣ ΒΑΣΕΩΝ ΓΝΩΣΕΩΝ

Θανάσης Β. Χρυσός

Μεταπτυχιακή Εργασία

Ηράκλειο, Κρήτη
Φεβρουάριος 1999

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

΄ΕΝΑ ΣΥΣΤΗΜΑ ΣΗΜΑΣΙΟΛΟΓΙΚΗΣ ΔΙΕΡΕΥΝΗΣΗΣ ΒΑΣΕΩΝ ΓΝΩΣΕΩΝ

Εργασία που υποβλήθηκε
ως μερική απαίτηση για την απόκτηση του
ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΔΙΠΛΩΜΑΤΟΣ ΕΙΔΙΚΕΥΣΗΣ
Φεβρουάριος 1999

Συγγραφέας:

Θανάσης Β. Χρυσός
Τμήμα Επιστήμης Υπολογιστών

Εισηγητική Επιτροπή:

Πάνος Κωνσταντόπουλος, Καθηγητής, Επόπτης

Απόστολος Τραγανίτης, Αναπληρωτής Καθηγητής, Μέλος

Γιώργος Γεωργακόπουλος, Επίκουρος Καθηγητής, Μέλος

Δεκτή:

Πάνος Κωνσταντόπουλος
Πρόεδρος Επιτροπής Μεταπτυχιακών Σπουδών

Ένα Σύστημα Σημασιολογικής Διερεύνησης Βάσεων Γνώσεων

Θανάσης Β. Χρυσός

Μεταπτυχιακή Εργασία

Τμήμα Επιστήμης Υπολογιστών
Πανεπιστήμιο Κρήτης

ΠΕΡΙΛΗΨΗ

Στις βάσεις γνώσεων συναντάμε συχνά πολύπλοκες συσχετίσεις ανάμεσα στις οντότητες. Για την παρουσίασή τους στο χρήστη, οι πίνακες δεδομένων, που χρησιμοποιούνται για παράδειγμα στο σχεσιακό μοντέλο δεδομένων, δεν είναι κατάλληλοι. Ως πιο φυσική και αποδοτική μορφή παρουσίασης προκύπτουν οι γενικοί γράφοι.

Η παρούσα εργασία ασχολείται με τον προσδιορισμό μιας μεθοδολογίας για την εξερεύνηση βάσεων δεδομένων τύπου σημασιολογικού δικτύου. Προσδιορίζουμε τις λειτουργίες που μπορούν να χρησιμοποιηθούν στη διερεύνηση μιας βάσης, και τις οργανώνουμε σε ένα λειτουργικό πλαίσιο, ώστε να παρέχονται στο χρήστη κατά τρόπο διαισθητικά απλό και εύχρηστο.

Ξεχωρίζουμε τέσσερα επίπεδα λειτουργικότητας: το πρώτο σχετίζεται με την εκτέλεση επερωτήσεων, το δεύτερο με την αυτόματη σχεδίαση του γράφου, το τρίτο με την χρήση της σημασιολογικής και τοπολογικής πληροφορίας του γράφου και το τέταρτο με ζητήματα οπτικοποίησης του γράφου και διεπαφής χρήσης.

Παράλληλα, αναπτύσσουμε το Σύστημα Σημασιολογικής Διερεύνησης, που υλοποιεί την μεθοδολογία μας.

Επόπτης: Πάνος Κωνσταντόπουλος
Καθηγητής Επιστήμης Υπολογιστών
Πανεπιστήμιο Κρήτης

A Semantic Browsing System for Knowledge Bases

Thanassis V. Chryssos

Masters Thesis

Computer Science Deptartment,
University of Crete

ABSTRACT

In knowledge bases we often encounter complex relationships among the objects. For their presentation to the user, tables, such as those used for the relational data model, are not suitable. General graphs emerge as a more natural and effective form of presentation.

In the present work, we define a methodology for the exploration of semantic network data bases. We identify the methods that can be used in browsing a semantic-network data base, and we organize them in a functional context, so that the user easily perceives the whole process in an intuitive way.

We identify four layers of functionality: The first deals with query execution, the second with the automatic layout of the graph, the third with using the semantic and topological information in the graph, and the fourth with graph visualization and user interface issues.

Finally, we develop the Semantic Browser System, which implements our methodology.

Supervisor: Panos Constantopoulos
Professor of Computer Science
University of Crete

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κ. Πάνο Κωνσταντόπουλο, για τη δυνατότητα που μου έδωσε να ασχοληθώ με την περιοχή έρευνας που κέντριζε το ενδιαφέρον μου σε όλη τη διάρκεια των σπουδών μου, καθώς και για το ενδιαφέρον που έδειξε και μου ενέπνευσε για την εργασία μου.

Ευχαριστώ ακόμα τον δρ. Martin Dörr, για τις ιδέες και συμβουλές του, που ήταν μια ανεκτίμητη βοήθεια σε όλη τη διάρκεια της εργασίας μου.

Ιδιαίτερα ευχαριστώ πηγαίνουν στον κ. Στέλιο Ορφανουδάκη, για έναν ιδιαίτερο λόγο.

Πολλά μαζεμένα ευχαριστώ στους γονείς μου, για την γενναία τους οικονομική, και όχι μόνο, υποστήριξη καθ' όλη τη διάρκεια των σπουδών μου. Το ίδιο σε όλους τους φίλους μου, που τις έκαναν να είναι ένα όμορφο και αξέχαστο κομμάτι της ζωής μου.

Δε θα μπορούσα να μην αναφέρω το Noam Chomsky, τα πολιτικά βιβλία του οποίου ήταν αφορμή για το πρώτο μου βήμα έξω από την απάτη της αστικής ιδεολογίας, ώστε να αρχίσω να καταλαβαίνω πώς πραγματικά λειτουργεί ο κόσμος γύρω μας. Επίσης, το Σοσιαλιστικό Εργατικό Κόμμα και όλους τους Διεθνιστές Σοσιαλιστές σε όλον τον κόσμο, για την πάλη τους ενάντια στο απάνθρωπο καπιταλιστικό σύστημα και την προοπτική που μας δίνουν για μια ανθρώπινη κοινωνία, για την επανάσταση και το σοσιαλισμό.

Περιεχόμενα

Περίληψη **i**

Abstract **iii**

Ευχαριστίες **v**

Περιεχόμενα **vii**

Κατάλογος Σχημάτων **x**

1 Εισαγωγή **1**

1.1 Το περιεχόμενο της εργασίας 1

1.2 Οριοθέτηση του προβλήματος 2

1.3 Οργάνωση της εργασίας 3

2 Σημασιολογική Διερεύνηση **5**

2.1 Συστήματα Σημασιολογικής Διερεύνησης 5

2.1.1 Concept Base 6

2.1.2 GAIN 7

2.2 Εξερεύνηση Σημασιολογικών Δικτύων 8

3 Αυτόματη σχεδίαση γράφων **11**

3.1 Αλγόριθμοι αυτόματης σχεδίασης γράφων 11

3.2 Δηλωτικές προσεγγίσεις 15

3.3 Αυξητική σχεδίαση γράφων 16

3.4 Τριδιάστατη σχεδίαση 16
3.5 Ομαδοποίηση κόμβων 17

4 Λειτουργικότητα του Συστήματος Σημασιολογικής Διερεύνησης. 19
4.1 Εκτέλεση επερωτήσεων 20
4.1.1 Επέκταση ερώτησης 21
4.1.1.1 Είδη επερωτήσεων 22
4.1.1.2 Οπτικοποίηση της επέκτασης 22
4.1.1.3 Η λειτουργία επέκτασης στην πράξη 24
4.2 Χωροδιάταξη του γράφου απάντησης 25
4.2.1 Παράσταση των συσχετίσεων ως ακμών ή κόμβων 25
4.2.2 Συγχώνευση γράφων 27
4.2.3 Ανασχεδίαση του γράφου 28
4.2.4 Εξάλειψη επικαλύψεων 29
4.3 Σημασιολογική και τοπολογική πληροφορία στο γράφο 29
4.3.1 Κατηγοριοποίηση των συνδέσμων 29
4.3.2 Σύμπτυξη πληροφορίας σε έναν κόμβο 31
4.3.2.1 Ομαδοποίηση 31
4.3.2.2 Τοπολογική Σύμπτυξη 32
4.3.2.3 Σημασιολογική Σύμπτυξη 34
4.3.3 Στρώματα του γράφου 36
4.4 Λειτουργίες οπτικοποίησης και διεπαφής χρήσης 36
4.4.1 Οπτικοποίηση του γράφου 37
4.4.1.1 Οπτική κλιμάκωση του γράφου 37
4.4.1.2 Ολική Όψη του γράφου 37
4.4.1.3 Απόκρυψη και Σκίαση τμημάτων του γράφου 38
4.4.2 Διεπαφή χρήσης 38
4.4.2.1 Πολλαπλά παράθυρα γράφων 38
4.4.2.2 Αποθήκευση και Φόρτωμα του γράφου 39
4.5 Σύνοψη λειτουργιών 39

5 Αρχιτεκτονική του συστήματος **41**

5.1 Καθοριστικές απαιτήσεις 41

5.2 Αρχιτεκτονική 42

5.2.1 Το Εργοστάσιο Γράφων 42

5.2.2 Ο Σημασιολογικός Διερευνητής 46

6 Συμπεράσματα - Μελλοντικές Κατευθύνσεις **51**

6.1 Συμπεράσματα 51

6.2 Σύγκριση με το GAIN 52

6.3 Μελλοντικές Κατευθύνσεις 53

6.3.1 Σχεδίαση του γράφου 53

6.3.1.1 Δηλωτική σχεδίαση 53

6.3.1.2 Σχεδίαση βάσει παραδείγματος 54

6.3.1.3 Σχεδίαση με ομαδοποίηση 54

6.3.1.4 Εναλλακτικές μορφές παράστασης γράφων 55

6.3.1.5 Πολλαπλές όψεις 56

6.3.1.6 Αυτόματη επιλογή αλγορίθμου σχεδίασης 56

6.3.2 Άλλες επεκτάσεις 56

Βιβλιογραφία **58**

Κατάλογος σχημάτων

3.1 Χωροδιάταξη με βάση τον αλγ. Sugiyama 12

3.2 Ορθογωνική χωροδιάταξη 13

3.3 Χωροδιάταξη με βάση τον αλγόριθμο ελατηρίου του Kamada 14

3.4 Τρεις χωροδιατάξεις του γράφου $K_{3,3}$ 15

4.1 Η Διεπαφή Χρήσης του ΣΣΔ. 20

4.2 Επέκταση επερώτησης 23

4.3 Συσχετίσεις ανάμεσα σε όρους διαφόρων θησαυρών όρων. 26

4.4 Κόμβοι κατηγοριών 28

4.5 Σύμπτυξη κόμβων 31

4.6 Πριν την τοπολογική σύμπτυξη στον κόμβο <settlements by economic base>. 33

4.7 Μετά την τοπολογική σύμπτυξη στον κόμβο <settlements by economic base>. 34

4.8 Αμφισημία κατά τη σύμπτυξη και επίλυσή της 35

5.1 Αρχιτεκτονική του Ε.Γ. 42

5.2 Αρχιτεκτονική του Σ.Δ. 47

6.1 Σχέδιο εγκλεισμού 55

Κεφάλαιο 1

Εισαγωγή

1.1 Το περιεχόμενο της εργασίας

Στις βάσεις γνώσεων που βασίζονται στα σημασιολογικά δίκτυα, συναντάμε συχνά πολύπλοκες συσχετίσεις ανάμεσα στις οντότητες. Για την παρουσίασή τους στο χρήστη, παραστάσεις με πίνακες δεδομένων, που χρησιμοποιούνται για παράδειγμα στο σχεσιακό μοντέλο δεδομένων, ή με πλαίσια, όπως συνηθίζεται σε οντοκεντρικά συστήματα, δεν είναι κατάλληλες. Ως πιο φυσική και αποδοτική μορφή παρουσίασης προκύπτουν οι γενικοί γράφοι.

Σαν παραδείγματα αναφέρουμε βάσεις πολιτισμικής τεκμηρίωσης [12] [9], συστήματα διαχείρισης θησαυρών όρων [18], και συστήματα τεχνικής τεκμηρίωσης [3]. Στην πρώτη περίπτωση συναντάμε οντότητες με πολύ ποικίλες και μεταβαλλόμενες συσχετίσεις μεταξύ τους (μεγάλο ‘‘σχήμα’’), ενώ στις άλλες δύο προκύπτουν γράφοι που περιλαμβάνουν πολλές οντότητες, πυκνά συσχετισμένες (πολλά ‘‘δεδομένα’’).

Η παρούσα εργασία ασχολείται με τον προσδιορισμό μιας μεθοδολογίας για την εξερεύνηση σημασιολογικών βάσεων. Προσδιορίζουμε τις λειτουργίες που μπορούν να χρησιμοποιηθούν στη διερεύνηση μιας σημασιολογικής βάσης, και τις οργανώνουμε σε ένα λειτουργικό πλαίσιο, ώστε να παρέχονται στο χρήστη κατά τρόπο διαισθητικά απλό και εύχρηστο. Παράλληλα, αναπτύσουμε το *Σύστημα Σημασιολογικής Διερεύνησης*, που υλοποιεί την μεθοδολογία μας.

Στα επόμενα κεφάλαια μελετάμε τη διαδικασία διερεύνησης, και παρουσιάζουμε τη λειτουργικότητα που προτείνουμε για να την καλύψουμε και να την οργανώσουμε. Επίσης, παρουσιάζουμε την αρχιτεκτονική του συστήματος που υλοποιήσαμε.

1.2 Οριοθέτηση του προβλήματος

Στα σημασιολογικά δίκτυα, και ειδικότερα στο μοντέλο δεδομένων του Συστήματος Σημασιολογικού Ευρετηριασμού (ΣΣΕ) [13], δεν υπάρχει ένας σαφής διαχωρισμός ‘‘σχήματος’’ και ‘‘δεδομένων’’ [45]. Κατά συνέπεια, το σχήμα δεν μπορεί να χρησιμοποιηθεί για να προσδιορίσει επακριβώς τη διαδικασία αναζήτησης πληροφορίας. Ο χρήστης θα πρέπει κατά την αναζήτηση αφ’ ενός να δημιουργήσει στο μυαλό του την εικόνα του ‘‘σχήματος’’, ως ένα τμήμα της βάσης που λειτουργεί σαν ‘‘σχήμα’’ για τα ‘‘δεδομένα’’ που τον ενδιαφέρουν, και παράλληλα να αντιληφθεί και το περιεχόμενο των ίδιων των δεδομένων. Αυτό είναι ένα ‘‘τίμημα’’ που πληρώνουμε προκειμένου να έχουμε σχεδόν απεριόριστη ευελιξία για παράσταση γνώσης, κατά την ανάπτυξη και συντήρηση της βάσης μας. Τέτοια ευελιξία είναι απαραίτητη, δεδομένου ότι η γνώση μπορεί να μεταβάλλεται ή να συμπληρώνεται με το χρόνο. Η παραπάνω επιβάρυνση του χρήστη είναι, σε πολλές περιπτώσεις, σαφώς προτιμώτερη από την εναλλακτική προσέγγιση της υποχρέωσής του να μελετήσει και να μάθει εκ των προτέρων ολόκληρο το συγκεκριμένο σχήμα της βάσης, πριν μπορέσει να δει οποιαδήποτε δεδομένα.

Μια συνηθισμένη μορφή παράστασης σημασιολογικών δικτύων είναι, όπως προαναφέραμε, οι γράφοι. Κατά την διαδικασία εξερεύνησης της βάσης, προκύπτουν γράφοι με μεγάλη πολυπλοκότητα και ως προς το περιεχόμενό τους (πολλά διαφορετικά είδη οντοτήτων), και ως προς τη δομή τους (πολλές συσχετίσεις μεταξύ των οντοτήτων). Ως συνέπεια, ένα βασικό πρόβλημα που πρέπει να αντιμετωπίσουμε, είναι η σχεδίαση των γράφων κατά έναν τρόπο που θα επιτρέπει στο χρήστη να εντοπίζει εύκολα την πληροφορία που τον ενδιαφέρει.

Για τη σχεδίαση των γράφων, χρησιμοποιούμε έτοιμους αλγόριθμους σχεδίασης γράφων. Αρκετοί τέτοιοι αλγόριθμοι έχουν αναπτυχθεί από τη σχετική ερευνητική κοινότητα [48], [40], [16], με τον καθένα να έχει διαφορετικά χαρακτηριστικά ως προς την εμφάνιση του αποτελέσματος, και να είναι περισσότερο κατάλληλος από άλλους για ορισμένα είδη γράφων. Ακόμα όμως και με αυτούς τους αλγόριθμους, πολλοί από τους γράφους απάντησης είναι τόσο πολύπλοκοι που είναι αδύνατο να σχεδιαστούν αποτελεσματικά, είτε στο επίπεδο είτε ακόμα και σε τρεις διαστάσεις. Για το λόγο αυτό, θα πρέπει

1. Αφ’ ενός να επινοηθούν τρόποι για την μείωση της πολυπλοκότητας των γράφων, με κατά το δυνατόν αυτόματη προεπεξεργασία τους, και

2. αφ’ ετέρου να βρεθούν τρόποι που να βοηθούν το χρήστη σε μια ενεργό διαδικασία εξερεύνησης του γράφου ειδικά, και της βάσης γενικότερα. Αυτό μπορεί να θεωρηθεί ώς άλλος ένας, ‘‘δυναμικός’’ τρόπος για τη μείωση της πολυπλοκότητας της

πληροφορίας που εξερευνά ο χρήστης. Όπως επισημαίνουν ερευνητές στην περιοχή της αυτόματης σχεδίασης γράφων [20], *''η τριδιάστατη σχεδίαση γράφων μπορεί να είναι μια πολύ καλή ιδέα, εάν η τρίτη διάσταση είναι χρονική, όχι χωρική''*.

Για το πρώτο, την προεπεξεργασία του γράφου, συγκεντρώνουμε και προτείνουμε ιδέες που έχουν εμφανιστεί στη βιβλιογραφία. Η πιο πρόσφορη φαίνεται να είναι η ομαδοποίηση (clustering) υπογράφων βάση της συνεκτικότητάς τους, και η σχεδίαση κάθε υπογράφου και του ολικού γράφου χωριστά. Στην σχεδίαση του συστήματος έχει προβλεφθεί η δυνατότητα για προεπεξεργασία του γράφου. Στο σύστημά μας, έχουμε υλοποιήσει μόνο προεπεξεργασία για την ομαδοποίηση πολλών συνδέσμων με βάση την κατηγορία τους, ώστε να μπορούμε να τους παριστάνουμε με ένα μόνο κόμβο.

Έχουμε επικεντρωθεί περισσότερο στο δεύτερο, την ενεργό εξερεύνηση του γράφου. Σε αυτή τη διαδικασία, διακρίνουμε τέσσερα επίπεδα λειτουργικότητας. Το πρώτο έχει να κάνει με την *εκτέλεση επερωτήσεων* στη βάση πληροφοριών, και το δεύτερο με τη *χωροδιάταξη του γράφου*. Ένας γράφος σημασιολογικού δικτύου έχει εν γένει περισσότερη πληροφορία από έναν απλό τοπολογικό γράφο. Τη χρήση της *σημασιολογικής και της τοπολογικής πληροφορίας του γράφου απάντησης* την οργανώνουμε σε ένα τρίτο επίπεδο λειτουργικότητας. Τέλος, στο τέταρτο επίπεδο, ασχολούμαστε με *ζητήματα οπτικοποίησης του γράφου και διεπιφάνειας χρήσης*.

1.3 Οργάνωση της εργασίας

Στο δεύτερο κεφάλαιο εξετάζουμε τις ανάγκες οι οποίες προκύπτουν κατά τη σημασιολογική διερεύνηση, τον τρόπο με τον οποίο αντιμετωπίζονται σε σχετικά συστήματα, και προτείνουμε τη δική μας προσέγγιση.

Στο τρίτο κεφάλαιο κάνουμε μια επισκόπηση της ερευνητικής περιοχής που ασχολείται με την αυτόματη σχεδίαση γράφων, η οποία είναι κεντρικού ενδιαφέροντος στην προσέγγισή μας.

Στο τέταρτο κεφάλαιο παρουσιάζουμε την λειτουργικότητα που προτείνουμε για συστήματα σημασιολογικής διερεύνησης.

Στο πέμπτο κεφάλαιο παρουσιάζουμε την αρχιτεκτονική του συστήματος μας, και αναφερόμαστε στα βασικά ζητήματα υλοποίησης.

Τέλος, στο έκτο κεφάλαιο παρουσιάζουμε τα συμπεράσματα που αποκομίσαμε κατά τη διάρκεια της εργασίας μας, και προτείνουμε μελλοντικές κατευθύνσεις.

Κεφάλαιο 2

Σημασιολογική Διερεύνηση

Η διαδικασία της *διερεύνησης (browsing)*, έχει γίνει ένας ευρέως χρησιμοποιούμενος τρόπος αναζήτησης πληροφορίας τα τελευταία χρόνια, και χρησιμοποιείται κατά κόρον στο WWW.

Στην παρούσα εργασία, χρησιμοποιούμε τον όρο *Σημασιολογική Διερεύνηση (Semantic Browsing)* για να χαρακτηρίσουμε τη διαδικασία αναζήτησης πληροφοριών σε σημασιολογικά δίκτυα και βάσεις δεδομένων που σχετίζονται με αυτά. Ο όρος έχει αρχίσει να εμφανίζεται την τελευταία δεκαετία για να χαρακτηρίσει διάφορες μεθοδολογίες, κοινό χαρακτηριστικό των οποίων είναι ότι περιλαμβάνουν κάποια έννοια διερεύνησης. Δεν υπάρχει κάποιος αυστηρός ορισμός του: Γύρω από την διαδικασία της απλής διερεύνησης, που κατέχει συνήθως κεντρικό ρόλο στην όλη διαδικασία, διάφορες γενικότερες τεχνικές μπορούν να αναπτύσσονται, που λαμβάνουν υπ' όψη τον πλούτο των πληροφοριών που υπάρχει στα σημασιολογικά δίκτυα ή τη βάση πληροφοριών. Επιπλέον, η διερεύνηση μπορεί να συνδυάζεται με ποικίλες τεχνικές από την περιοχή της ανάκτησης πληροφοριών, ώστε να προκύπτει ένα πολύπλοκο υβριδικό σύστημα αναζήτησης πληροφοριών, στο οποίο είναι πια δύσκολο να διαχωριστεί μια καθαρή διαδικασία διερεύνησης.

Στο κεφάλαιο αυτό εμβαθύνουμε στη διαδικασία της σημασιολογικής διερεύνησης. Επικεντρωνόμαστε στις ανάγκες που προκύπτουν και τις δυνατότητες που ανοίγονται κατά τη σημασιολογική διερεύνηση, εξετάζουμε τον τρόπο με τον οποίο αντιμετωπίζονται σε σχετικά συστήματα, και προτείνουμε την δική μας προσέγγιση.

2.1 Συστήματα Σημασιολογικής Διερεύνησης

Στην ενότητα αυτή θα κάνουμε μια ανασκόπηση των συστημάτων και των μεθόδων

παρουσίασης πληροφορίας από σημασιολογικά δίκτυα. Θα εντοπίσουμε τις ιδέες στις οποίες βασίστηκε η εργασία μας, και θα δούμε πως οδηγηθήκαμε στη δική μας προσέγγιση, την οποία παρουσιάζουμε συγκροτημένα στην επόμενη ενότητα, 2.2.

Ειδικότερα, εξετάζουμε τις μεθόδους παρουσίασης πληροφορίας στην Concept Base [28], και το σύστημα GAIN [22] [4] για γραφική εξερεύνηση βάσεων του ΣΣΕ. Και στις δύο περιπτώσεις, το θεωρητικό υπόβαθρο των συστημάτων είναι η γλώσσα παράστασης γνώσης Telos [38].

2.1.1 Concept Base

Η Concept Base είναι ένα σύστημα διαχείρησης των βάσεων γνώσεων που βασίζεται στην οντοκεντρική γλώσσα παράστασης γνώσης Telos, και δίνει έμφαση σε επαγωγικές δυνατότητες. Η εξερεύνηση των περιεχομένων της βάσης γίνεται με διάφορους τρόπους. Αφ' ενός, παρέχεται μια ερωτηματική γλώσσα, την οποία μπορεί ο χρήστης να χρησιμοποιήσει για να φτιάξει ένα δικό του σύστημα παρουσίασης. Επιπλέον, δίνεται δυνατότητα για παρουσίαση ιδιοτήτων των αντικειμένων σε μορφή πινάκων, και σε μορφή μιας ιεραρχίας κειμένου. Τέλος, παρέχεται ένα υποσύστημα γραφικής διερεύνησης.

Το σύστημα γραφικής διερεύνησης είναι μάλλον επηρεασμένο από μια περισσότερο οντοκεντρική θεώρηση της βάσης και της διαδικασίας αναζήτησης πληροφορίας, παρά από μια θεώρηση σημασιολογικού δικτύου: παρέχει τη δυνατότητα στο χρήστη να εξετάσει τα χαρακτηριστικά *μιας* οντότητας τη φορά, αντί για την εξέταση μιας σημασιολογικής περιοχής. Η οντότητα εμφανίζεται ως ένας κόμβος σε ένα γραφικό παράθυρο. Ο χρήστης μπορεί να δει τα γνωρίσματά της, τις κλάσεις της, τις περιπτώσεις της κ.ο.κ. Οι νέες οντότητες και συσχετίσεις που αποτελούν την ζητούμενη πληροφορία, εμφανίζονται σαν νέοι κόμβοι και ακμές, που προστίθενται στον υπάρχοντα γράφο. Διαφορετικά χρώματα, σχήματα αλλά και εικονίδια χρησιμοποιούνται για να προσδώσουν επιπλέον πληροφορία στους κόμβους, ιδιαίτερα για να δηλώσουν τις κλάσεις στις οποίες ανήκουν. Εάν ο χρήστης το επιλέξει, νέα παράθυρα μπορούν να εμφανιστούν που να περιέχουν κάποια επιλεγμένη οντότητα, για περαιτέρω εξερεύνηση.

Η παραπάνω διαδικασία έχει ένα ενδιαφέρον στοιχείο: Ο χρήστης μπορεί σταδιακά να αποκτά περισσότερη γνώση για τη βάση του, ενώ έχει διαθέσιμη στο γράφο που επιθεωρεί και όλη την ''ιστορία'' της μέχρι στιγμής αναζήτησής του. Όμως, όπως επισημάναμε, είναι χρήσιμη μόνο για μια επιθεώρηση των σχέσεων ιδωμένων ως ''ιδιοτήτων'': μόνο οι άμεσες συσχετίσεις (ή κλάσεις ή υποκλάσεις ή περιπτώσεις) κάποιας οντότητας επιθεωρούνται σε κάθε βήμα, χωρίς να παρέχεται η δυνατότητα αναδρομικής διάσχισης, με αποτέλεσμα η δημιουργία ενός γράφου μιας περιοχής του σημασιολογικού δικτύου να είναι πολύ

επίπονη διαδικασία. Για παράδειγμα, ας θεωρήσουμε έναν γράφο με όλες τις οικογενειακές σχέσεις ανάμεσα σε ένα σύνολο συγγενικών ατόμων. Για να φτιάξει ο χρήστης τον πλήρη γράφο, θα πρέπει να επιθεωρήσει ένα-ένα όλα τα άτομα, για να δεί όλες τις μεταξύ τους συσχετίσεις.

Σημειώνουμε τέλος τη χρήση ενός τυποποιημένου συστήματος επιθεώρησης ιεραρχιών, που παρουσίαζει τις ιδιότητες οντοτήτων μέσω δυναμικά διαμορφωνόμενων λιστών.

2.1.2 GAIN

Το GAIN (Graphical Analysis Interface) είναι ένα σύστημα που παρέχει πρόσβαση σε βάσεις του ΣΣΕ (SIS -- Semantic Index System), ενός συστήματος διαχείρησης βάσεων δεδομένων τύπου σημασιολογικού δικτύου, που βασίζεται επίσης στην Telos. Παρέχει τη δυνατότητα στο χρήστη να επιθεωρήσει κάποια οντότητα ως αυτόνομο αντικείμενο, με παράσταση των ‘‘ιδιοτήτων’’ του σε μορφή πλαισίου (*Object Card*) ή γράφου μιας συγκεκριμένης μορφής (*Star View*). Ακόμα, μπορεί να επιθεωρήσει τα περιεχόμενα της βάσης σε μορφή πινάκων, με πιό εύχρηστο τρόπο από την Concept Base: οι πίνακες προκύπτουν από επερωτήσεις. Είναι εν γένει περισσότερο προσανατολισμένο στην εξερεύνηση της βάσης ως σημασιολογικό δίκτυο, από το προηγούμενο σύστημα.

Προσφέρει προκατασκευασμένες ερωτήσεις που εκτελούνται επάνω σε οντότητες της βάσης, το αποτέλεσμα των οποίων μπορεί να παρουσιάζεται σε γραφική μορφή. Οι ερωτήσεις επιτρέπουν αναδρομική διάσχιση του σημασιολογικού δικτύου, και μπορούν να επιστρέψουν ένα σύνολο από συσχετίσεις. Το σύνολο αυτό εμφανίζεται ως γράφος με την παράσταση των οντοτήτων που εμπλέκονται στις συσχετίσεις σε κόμβων και των συνδέσμων ως ακμών. Ένας αλγόριθμος αυτόματης σχεδίασης γράφου (της Μ. Καραβασίλη, [1]) εκτελείται, που σχεδιάζει τον γράφο σε μορφή κατευθυνόμενου ακυκλικού γράφου, και κατόπιν ο γράφος παρουσιάζεται στο χρήστη.

Ενώ ο χρήστης μπορεί να επιλέξει νέες οντότητες και να εκτελέσει καινούργιες επερώτησεις, η ιστορία της αναζήτησης είναι μόνο έμμεσα διαθέσιμη κάθε στιγμή: ο χρήστης μπορεί να επιλέξει κάποιαν προηγούμενη επερώτηση από μια λίστα με το ιστορικό των επερωτήσεων, αλλά όχι να βλέπει πληροφορία που είχε εντοπίσει παλιότερα ταυτόχρονα με πιο πρόσφατη πληροφορία.

Η μέθοδος του GAIN, με τις αναδρομικές προκατασκευασμένες ερωτήσεις και τον αυτόματο σχεδιασμό των γράφων είναι η βάση και της δικής μας προσέγγισης.

2.2 Εξερεύνηση Σημασιολογικών Δικτύων

Παρόλο που ιδέες από τα σημασιολογικά δίκτυα *(σ.δ.)* έχουν αρχίσει να εφαρμόζονται σε διάφορα συστήματα, και σχετικά μοντέλα δεδομένων γίνονται όλο και πιο διαδεδομένα, δεν υπάρχουν βάσεις δεδομένων ή γνώσεων που να τα έχουν ως υπόβαθρο, έξω από την ερευνητική κοινότητα. Η πιο σύγχρονη τεχνολογία βάσεων δεδομένων που αρχίζει να κυκλοφορεί ευρέως είναι οι οντοκεντρικές βάσεις, οι οποίες όμως έχουν αρκετά διαφορετικά χαρακτηριστικά από τα σημασιολογικά δίκτυα, όπως επισημάναμε. Ως εκ τούτου, δεν έχει αναπτυχθεί μια καθιερωμένη μεθοδολογία εξερεύνησης των σημασιολογικών δικτύων.

Μπορούμε, παρ' όλα αυτά, να διακρίνουμε αρκετές αποδοτικές τεχνικές που εφαρμόζονται σε περιπτώσεις οπτικοποίησης πληροφορίας, είτε σε συστήματα όπως αυτά που εξετάσαμε, είτε σε άλλα συστήματα σχεδίασης γράφων [36] [44], είτε ακόμα και κατά τη σχεδίαση με το χέρι, και να προσπαθήσουμε να τις συγκεντρώσουμε και να τις οργανώσουμε κατά συστηματικό τρόπο.

Αφήνουμε εξ αρχής την παράσταση πληφορορίας υπό τη μορφή κειμένου, πάνω στην οποία έχουν γίνει αρκετές εργασίες [5] [6], και ασχολούμαστε στην παρούσα εργασία με την παράσταση πληροφορίας υπό μορφή γράφου.

Η παράσταση ενός σημασιολογικού δικτύου σε μορφή γράφου προκύπτει φυσιολογικά: παριστάνουμε κάθε οντότητα με κόμβο και κάθε συσχέτιση με ακμή. Προφανώς η διαδικασία αυτή δεν είναι σκόπιμο να εφαρμοστεί σε ολόκληρο το σ.δ., λόγω του μεγέθους του. Επιπλέον, στις περισσότερες περιπτώσεις δεν μας ενδιαφέρει καν να κατανοήσουμε ολόκληρο το σ.δ., παρά μόνο ένα υποσύνολό του. Επομένως, πρέπει κατ' αρχήν να προμηθεύσουμε ένα μηχανισμό *επερώτησης*: Μια επερώτηση την ορίζουμε γενικά ως μια συνάρτηση που εφαρμοζόμενη πάνω σε ένα σ.δ. επιστρέφει ένα υποσύνολό του. Συνεπώς, οι επερωτήσεις μας επιτρέπουν να ασχολούμαστε κάθε φορά με ένα τμήμα του σ.δ.

Μια δυνατότητα που προκύπτει ως πολύ ελκυστική, είναι να μπορούμε να επεκτείνουμε το τμήμα του σ.δ. που επιθεωρούμε δυναμικά, με βάση τις επιπλέον ανάγκες για πληροφόρηση που δημιουργούνται κατά τη διαδικασία εξερεύνησης.

Από την άλλη πλευρά, πολλές φορές ζητούμενο είναι το αντίθετο: να περιοριστούμε σε ένα μικρότερο υποσύνολο του συνόλου της απάντησης που ήδη εξερευνούμε, για να έχουμε καλύτερη εποπτεία του. Σε αυτή μας την προσπάθεια, βοήθεια μπορεί να μας προσφέρει αφ' ενός η τοπολογία του γράφου, αφ' ετέρου η σημασιολογική πληροφορία που φέρουν τα αντικείμενα του σημασιολογικού δικτύου. Μπορούμε να χρησιμοποιήσουμε αυτές τις πληροφορίες και να εφαρμόσουμε διάφορες τεχνικές για την απλοποίηση του

γράφου.

Οι δυο παραπάνω βασικές ιδέες, της ελεγχόμενης επέκτασης από τη μια, και του ελεγχόμενου περιορισμού από την άλλη, αποτελούν τη βάση της προσέγγισής μας. Ξεκινώντας από τη χρήση αυτόματων τεχνικών για τη σχεδίαση των αποτελεσμάτων των επερωτήσεων ως γράφους, εξετάζουμε ένα πλαίσιο για *επεκτάσεις των επερωτήσεων* και *σύμπτυξη τμημάτων του γράφου*, ως μηχανισμούς εξερεύνησης σημασιολογικών δικτύων.

Ένα σημαντικό ζήτημα που καλούμαστε να αντιμετωπίσουμε, είναι αυτό του συνδυασμού των τεχνικών που χρησιμοποιούμε: Είναι κρίσιμο κάθε λειτουργία να είναι αυτόνομη και ανεξάρτητη από άλλες, ώστε να διευκολυνθεί ο συνδυασμός τους σε ένα σύστημα, και το σύστημα αυτό να είναι ανοικτό για επεκτάσεις.

Στο Κεφ. 4 παρουσιάζουμε αναλυτικά τη λειτουργικότητα στην οποία καταλήξαμε, καθώς και τη σκοπιμότητα πίσω από κάθε μας επιλογή. Περιοριζόμαστε στη λειτουργικότητα που ενσωματώνει το *Σύστημα Σημασιολογικής Διερεύνησης* που αναπτύξαμε, κάνοντας παράλληλα αναφορές στις κατευθύνσεις επέκτασης που θεωρούμε απαραίτητες, ώστε να δώσουμε τη συνολική εικόνα της αντίληψής μας για την όλη διαδικασιά διερεύνησης. Αφήνουμε το καθήκον της έκθεσης και ανάλυσης των επιπλέον πολύ χρήσιμων ιδεών που συναντήσαμε κατά τη διάρκεια της εργασίας μας αλλά δεν μπορέσαμε να υλοποιήσουμε στα πλαίσιά της, για το κεφάλαιο 6 της μελλοντικής εργασίας.

Κεφάλαιο 3

Αυτόματη σχεδίαση γράφων

Τα τελευταία χρόνια, με την μεγάλη αύξηση της υπολογιστικής ισχύος και των γραφικών δυνατοτήτων των υπολογιστών, έγινε δυνατή η χρήση γράφων για παράσταση και παρουσίαση πληροφορίας. Ένας γράφος όμως είναι χρήσιμος για παρουσίαση πληροφορίας μόνο όταν οι κόμβοι και οι ακμές του έχουν τοποθετηθεί χωροταξικά κατά κατάλληλο τρόπο, ώστε να είναι εύκολο στο χρήστη να αντιληφθεί τη δομή του. Έτσι, ιδιαίτερα τα τελευταία 10 χρόνια, η ερευνητική περιοχή που ασχολείται με το αυτόματη χωροδιάταξη[1] γράφων είναι πολύ ενεργός.

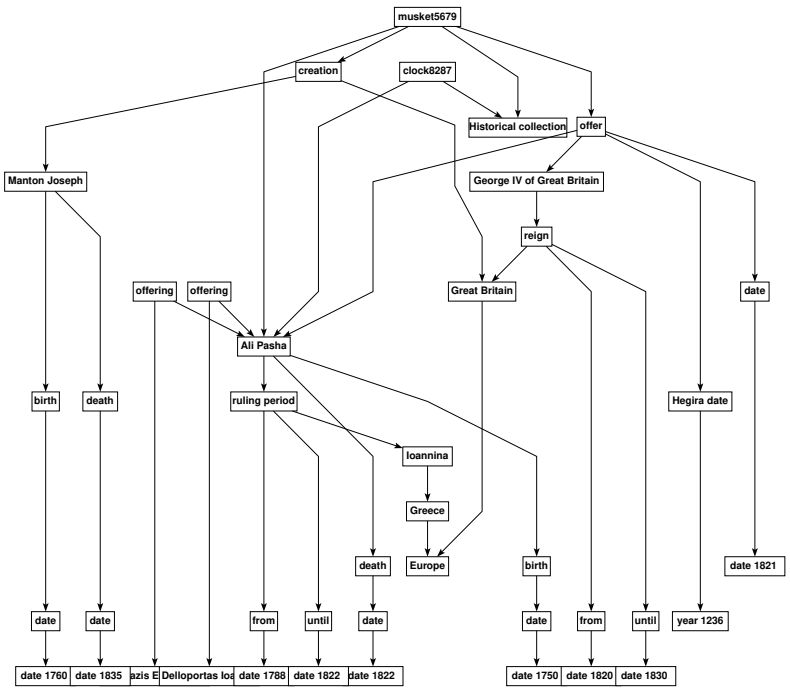
Στο κεφάλαιο αυτό κάνουμε μια επισκόπηση της αυτόματης σχεδίασης γράφων, κρίνοντας τις σχετικές προσεγγίσεις υπό τη σκοπιά της χρήσης τους για παρουσίαση πληροφορίας από βάσεις γνώσεων. Παρουσιάζουμε τους αλγόριθμους αυτόματης σχεδίασης, καθώς και εναλλακτικές μεθόδους που έχουν εμφανιστεί τελευταία.

3.1 Αλγόριθμοι αυτόματης σχεδίασης γράφων

Οι πρώτοι αλγόριθμοι για αυτόματη σχεδίαση γράφων αφορούσαν μόνο ιεραχικούς γράφους, και οι περισσότεροι βασίζονται στον αλγόριθμο του Sugiyama [29]: οι κόμβοι τοποθετούνται σε διαδοχικά επίπεδα, με βάση τη θέση τους στην ιεραρχία, και κατόπιν γίνεται μια αναδιάταξή τους σε κάθε επίπεδο ώστε να ελαχιστοποιηθούν οι διασταυρώσεις των ακμών. Ένα παράδειγμα φαίνεται στο σχ.3.1. Με επεκτάσεις αυτών των αλγορίθμων, εμφανίστηκαν αλγόριθμοι για κατευθυνόμενους ακυκλικούς γράφους (ΚΑΓ). Οι πρώτες προσεγγίσεις για τη σχεδίαση γενικών γράφων βασίζονταν στο σχεδιασμό τους ως

[1] Ο όρος *χωροδιάταξη* αντιστοιχεί στον αγγλικό όρο *layout*. Στην εργασία μας, θα χρησιμοποιούμε εναλλακτικά και τον όρο *σχεδίαση* ο οποίος είναι απλούστερος. Τον τελευταίο τον αποφεύγουμε μόνο σε μερικές περιπτώσεις, όπου μπορεί να παρεξηγηθεί ώς ‘‘ζωγράφισμα’’ του γράφου στην οθόνη.

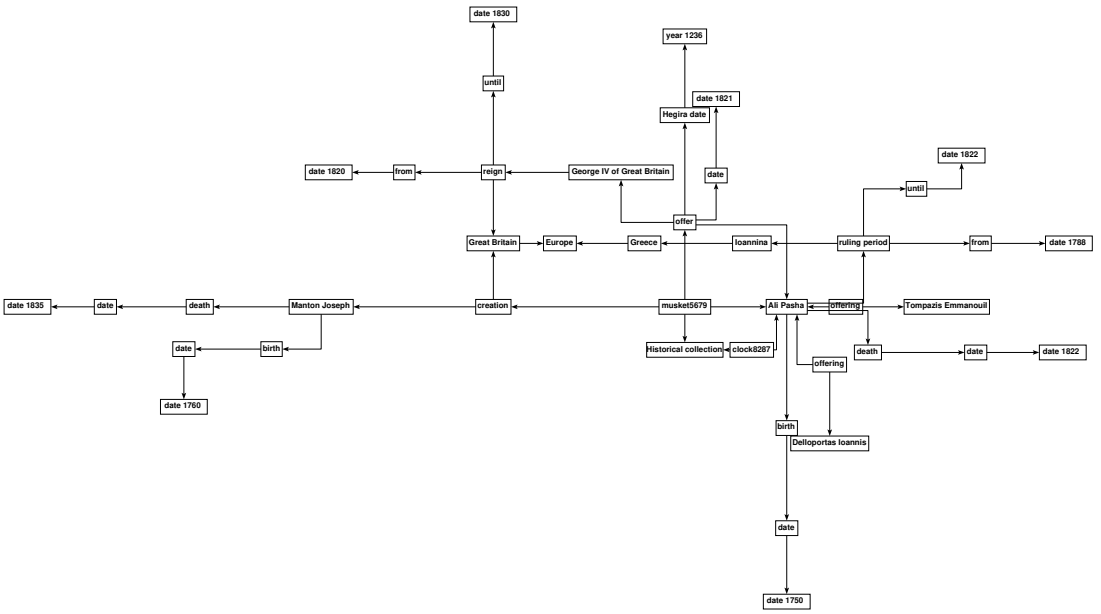
κατευθυνόμενων ακυκλικών γράφων, με επιλογή ενός υποσυνόλου των ακμών [1].



Σχήμα 3.1: Χωροδιάταξη με βάση τον αλγ. Sugiyama

Οι ανάγκες της σχεδίασης ολοκληρωμένων κυκλωμάτων οδήγησαν σε ανάπτυξη αλγορίθμων που παράγουν ‘‘ορθογωνικά’’ σχέδια . Σε ένα ορθογωνικό σχέδιο, οι ακμές σχεδιάζονται με κατευθύνσεις οριζόντιες ή κάθετες, και κατάλληλα λυγίσματα. Ο γράφος του σχ.3.1, με ορθογωνική χωροδιάταξη σχεδιάζεται όπως στο σχ.3.2.

Ο P. Eades το 1984 [19], εισήγαγε τη μέθοδο της χωροδιάταξης ενός γράφου βασιζόμενος σε ένα φυσικό μοντέλο: Οι κόμβοι θεωρούνται ως ομώνυμα φορτισμένοι δακτύλιοι που απωθούνται μεταξύ τους, με δύναμη σχετική με την απόστασή τους, ενώ οι ακμές ως ελατήρια που έλκουν τους κόμβους, με πιθανόν διαφορετικά μήκη και σταθερές ελατηρίου η κάθε μία. Διάφοροι τύποι έχουν εμφανιστεί σε παραλαγές του αρχικού αλγορίθμου [49] [32], για το μέτρο της ελκτικής και της απωστικής δύναμης. Εν γένει, η απωστική δύναμη είναι αντιστρόφως ανάλογη μιας δύναμης της ευκλείδιας απόστασης ανάμεσα στους κόμβους, ενώ η ελκτική δύναμη είναι ανάλογη μιας δύναμης της απόστασης. Τα αποτελέσματα των διαφόρων παραλλαγών είναι παρόμοια στις περισσότερες περιπτώσεις γράφων. Με μια τυχαία αρχική τοποθέτηση των κόμβων και προσομοίωση του φυσικού συστήματος, αυτό φτάνει μετά από κάποιο χρόνο σε μια από τις πιθανές θέσεις ισορροπίας του. Για οικονομία χρόνου, ο τερματισμός της προσομοίωσης συνήθως προσδιορίζεται ελαχιστοποιώντας μια συνάρτηση κόστους πάνω στις ασκούμενες δυνάμεις στο γράφο. Ο γράφος μας, με χρήση ενός τέτοιου αλγορίθμου, σχεδιάζεται όπως στο σχ.3.3.



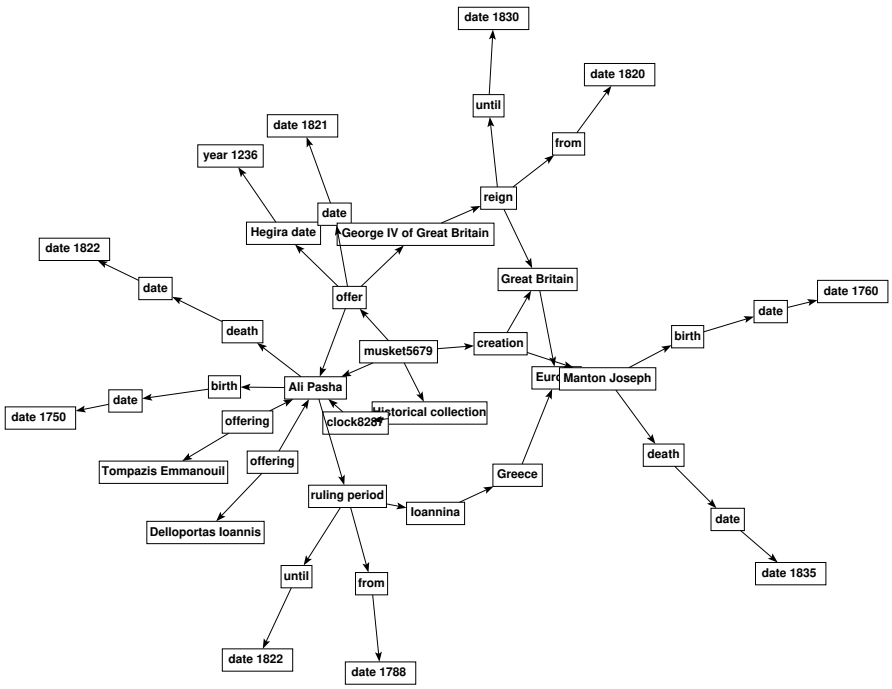
Σχήμα 3.2: Ορθογωνική χωροδιάταξη

Όλοι οι αλγόριθμοι αποσκοπούν στο να δημιουργήσουν έναν ‘‘ευπαρουσίαστο’’ γράφο[2]. Κάθε οικογένεια αλγορίθμων αποσκοπεί στο να τονίσει ορισμένα χαρακτηριστικά του γράφου. Έτσι, οι περισσότεροι από τους αλγόριθμους που ασχολούνται με κατευθυνόμενους γράφους οπτικοποιούν αποδοτικά γράφους με ιεραρχική δομή. Οι αλγόριθμοι ελατηρίων είναι καλοί στην οπτικοποίηση των συμμετριών του γράφου, και επιπλέον το μοντέλο τους των ελατηρίων μπορεί να χρησιμοποιηθεί και για ομαδοποίηση (clustering) των κόμβων του.

Εν γένει, οι στόχοι που προσπαθεί να επιτύχει ένας αλγόριθμος σχεδίασης γράφων είναι [7]:

- να κάνει εμφανείς τις συμμετρίες του γράφου
- να αποφύγει διασταυρώσεις των ακμών
- να αποφύγει λυγίσματα στις ακμές
- να κρατήσει τα μήκη των ακμών περίπου ίσα
- να μοιράσει ομοιόμορφα τους κόμβους στο χώρο
- να κάνει εμφανή τη ‘‘ροή’’ των ακμών, αν είναι κατευθυνόμενες

[2]Αυτό δεν είναι απόλυτα ακριβές: αρκετοί αλγόριθμοι για σχεδίαση ολοκληρωμένων κυκλωμάτων έχουν ως επιδίωξη την κατάλληλη τοποθέτηση στο επίπεδο για την καλύτερη δρομολόγηση των ακμών, και όχι την παρουσίαση ενός εύληπτου σχεδίου για το χρήστη. Με τέτοιους αλγόριθμους όμως δεν θα ασχοληθούμε στην παρούσα εργασία.



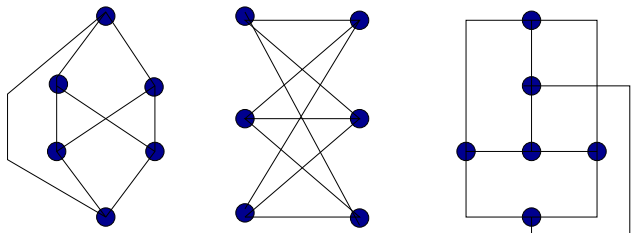
Σχήμα 3.3: Χωροδιάταξη με βάση τον αλγόριθμο ελατηρίου του Kamada

Από τεχνικής άποψης, μπορούμε να αξιολογήσουμε τους αλγόριθμους με βάση τα εξής μετρήσιμα κριτήρια: (πάντα σε σχέση με τον αριθμό των κόμβων και ακμών του γράφου)

- ελάχιστη απαιτούμενη περιοχή σχεδίασης
- μέγιστη ελάχιστη γωνία μεταξύ δύο ακμών:
  Μικρές γωνίες ανάμεσα σε ακμές του γράφου οδηγούν σε σχέδια δυσανάγνωστα.
- ελάχιστος αριθμός λυγισμάτων στις ακμές
- μέγεθος περιοχής σχεδίασης ενάντια στο λόγο μήκους/πλάτους περιοχής:
  Παρόλο που είναι κρίσιμο να περιορίσουμε το μέγεθος της περιοχής σχεδίασης, αυτό δεν είναι από μόνο του επαρκές: Για παρουσίαση σε μία οθόνη τερματικού, σχέδια π.χ. που εκτείνονται πολύ στο μήκος, δεν είναι κατάλληλα ακόμα και αν συνολικά το εμβαδό τους είναι μικρό.
- μέγεθος περιοχής σχεδίασης ενάντια στη μέγιστη ελαχίστη γωνία μεταξύ ακμών:
  Οι δυο παραπάνω επιδιώξεις είναι επίσης ανταγωνιστικές, όπως φαίνεται από το σχ.3.4(α).

Εκτός από τα παραπάνω κριτήρια, που αναφέρονται στο τελικό αποτέλεσμα του αλγορίθμου, κρίσιμος είναι για ορισμένες εφαρμογές (όπως και η δική μας), ο χρόνος που απαιτεί ο αλγόριθμος. Οι αλγόριθμοι που σχεδιάζουν τους γράφους σε μορφή δέντρων

ή ΚΑΓ, είναι εν γένει πολύ γρήγοροι, οι περισσότεροι χρειάζονται γραμμικό χρόνο. Οι ορθογωνικοί και οι αλγόριθμοι ελατηρίων χρειάζονται πολυωνυμικό χρόνο, συνήθως τετραγωνικό. Για ένα διαλογικό σύστημα, τέτοιοι αλγόριθμοι είναι πρακτικοί μόνο για γράφους με μερικές δεκάδες κόμβων, περίπτωση πάντως που είναι αρκετά τυπική.



Σχήμα 3.4: Τρεις χωροδιατάξεις του γράφου $K_{3,3}$

Στο σχήμα 3.4 βλέπουμε διάφορους δυνατούς τρόπους να σχεδιάσουμε το γράφο $K_{3,3}$[3]. Στο σχήμα αυτό παρατηρούμε ότι δεν είναι τετριμμένο να προκαθορίσουμε ποιο κριτήριο είναι σημαντικότερο. Το σχέδιο (β), παρόλο που έχει τις περισσότερες διασταυρώσεις ακμών, είναι μάλλον η πιο οικεία και εύληπτη παράσταση του $K_{3,3}$.

3.2 Δηλωτικές προσεγγίσεις

Για τις περιπτώσεις που υπάρχει σημασιολογική πληροφορία στο γράφο, θα ήταν χρήσιμο αυτή να μπορεί να χρησιμοποιηθεί για να καθορίσει τη χωροδιάταξή του. Η πληροφορία θα μπορούσε να εκφραστεί υπό τη μορφή περιορισμών, όπως π.χ. ‘‘τοποθέτησε τον σημαντικότερο κόμβο στο κέντρο’’. Όπως επισημαίνουν ερευνητές [47], ‘‘*ερευνητικές εργασίες έχουν δείξει ότι μόνο μια μάλλον περιορισμένη δυνατότητα για ικανοποίηση περιορισμών μπορεί να προστεθεί στους υπάρχοντες αλγόριθμους χωροδιάταξης*’’.

Από τότε, έχουν γίνει προσπάθειες να επινοηθούν τεχνικές σχεδίασης γράφων που να βασίζονται εξ αρχής σε δηλωτική πληροφορία με περιορισμούς. Έχουν αναπτυχθεί γλώσσες για προδιαγραφή σχεδίων (display specification) [46], γραμματικές για σχεδίαση γράφων [30], [51], [8], καθώς και συστήματα που βασίζονται σε οπτικούς περιορισμούς [14].

Οι παραπάνω προσεγγίσεις δεν έχουν ξεπεράσει το στάδιο της ερευνητικής εργασίας: υποφέρουν από πτώση της ποιότητας του σχεδίου για μεγάλους γράφους και απαιτούν συνήθως πολύ μεγάλο χρόνο εκτέλεσης [8].

[3] Δύο ομάδες κόμβων πλήρως συνδεδεμένες μεταξύ τους, με τρεις κόμβους η κάθε ομάδα

Οπωσδήποτε, η σχεδίαση γράφων βάσει δηλωτικών περιορισμών είναι μια ιδέα που ενθουσιάζει τους ερευνητές, δυστυχώς όμως έχουν γίνει πολύ μικρά βήματα για την αντιμετώπιση των προβλημάτων που θέτει. Χαρακτηριστικό είναι ότι συστήματα που βασίζονται σε δηλωτικούς περιορισμούς για τον προσδιορισμό της χωροδιάταξης, μπορούν μερικές φορές να ενσωματώνουν έναν έτοιμο αλγόριθμο ελατηρίου για γρήγορη σχεδίαση γράφων [46].

3.3 Αυξητική σχεδίαση γράφων

Σε αρκετές εφαρμογές προκύπτουν γράφοι που μεταβάλλονται με το χρόνο, με κόμβους και ακμές να προστίθενται ή να αφαιρούνται. Η πλειοψηφία των αλγορίθμων αυτόματης σχεδίασης είναι πολύ ευαίσθητοι σε αλλαγές: μια μικρή προσθήκη στο γράφο μπορεί να αλλάξει δραματικά το σχήμα του αποτελέσματος. Την τελευταία δεκαετία έχουν γίνει απόπειρες να αντιμετωπιστεί αυτό το πρόβλημα, με αλγόριθμους που λαμβάνουν υπ' όψιν την προηγούμενη χωροδιάταξη του γράφου, και προσπαθούν να τη διατηρήσουν κατά το δυνατόν σταθερή μετά από αλλαγές.

Σε τέτοιους αλγόριθμους, πρέπει να αναζητηθεί μια ισορροπία ανάμεσα στο χρόνο εκτέλεσης, τη βελτιστοποίηση των ιδιοτήτων του τελικού σχεδίου, και τη διατήρηση του αρχικού σχήματος, που είναι, όπως προκύπτει, ανταγωνιστικές επιδιώξεις.

Οι αλγόριθμοι που έχουν αναπτυχθεί μέχρι σήμερα έχουν ακόμα περιορισμένες δυνατότητες. Ορισμένοι αναφέρονται σε ειδικές κλάσεις γράφων, όπως π.χ. επίπεδους γράφους [10]. Άλλοι [39], είναι προσανατολισμένοι στο να χειρίζονται καλά αλλαγές του γράφου κατά ένα κόμβο ή ακμή τη φορά, όμως όταν ο γράφος αλλάζει κατά ένα μεγαλύτερο σύνολο ακμών και κόμβων, το τελικό αποτέλεσμα μπορεί να είναι αρκετά διαφορετικό από το αρχικό.

Τέλος, σημειώνουμε ότι τελευταία έχει εμφανιστεί και ένα εμπορικό πακέτο (το *Graph Layout Toolkit* από την Tom Sawyer Soft. [17]), που παρέχει αυξητικούς αλγόριθμους σχεδίασης.

3.4 Τριδιάστατη σχεδίαση

Αρκετές προσπάθειες έχουν γίνει τελευταία για χρήση της τρίτης διάστασης του χώρου για τη σχεδίαση γράφων [11]. Με αυτό τον τρόπο, ένας μεγαλύτερος βαθμός ελευθερίας στη σχεδίαση είναι δυνατός (π.χ. εξαλείφονται οι διασταυρώσεις των ακμών), όμως εισάγεται ένα μεγαλύτερο πρόβλημα προσανατολισμού του χρήστη. Με την υπάρχουσα

τεχνολογία επίδειξης τριδιάστατης πληροφορίας, αυτό το πρόβλημα δεν αντιμετωπίζεται ικανοποιητικά. Η διαπίστωση στο GD'96 [40], την οποία αναφέραμε και στην εισαγωγή ότι *''η τριδιάστατη σχεδίαση γράφων μπορεί να είναι μια πολύ καλή ιδέα, εάν η τρίτη διάσταση είναι χρονική, όχι χωρική''*, υποδεικνύει ότι τουλάχιστον με τα υπάρχοντα συστήματα, άλλες προσεγγίσεις ίσως είναι πιο άμεσα αποδοτικές από τη (χωρική) τριδιάστατη σχεδίαση.

Ένα σύστημα τριδιάστατης παράστασης για εξερεύνηση σημασιολογικών βάσεων, το οποίο μάλιστα χρησιμοποιείται για βάσεις του ΣΣΕ, είναι το *Virtual Data Browser* [23], το οποίο χρησιμοποιεί έναν αλγόριθμο ελατηρίου για τρεις διαστάσεις.

3.5 Ομαδοποίηση κόμβων

Μια ιδέα που μπορεί να αποδειχθεί πολύ αποτελεσματική είναι ο χωρισμός του γράφου σε υπογράφους βάσει της συνεκτικότητας του κάθε υπογράφου (clustering), και ο σχεδιασμός του κάθε υπογράφου χωριστά [17], [43]. Η προσέγγιση αυτή έχει πολλά πλεονεκτήματα: Κάθε υπογράφος ως μικρότερος σχεδιάζεται γρηγορότερα, ενώ η διαδικασία ομαδοποίησης του γράφου δεν παίρνει περισσότερο χρόνο από έναν αλγόριθμο ελατηρίου. Επιπλέον, για κάθε υπογράφο, μπορεί να επιλεχθεί διαφορετικός αλγόριθμος σχεδίασης, που να επιδεινύει καλύτερα την πιθανώς ειδική μορφή του. Για παράδειγμα, μπορεί η μορφή του συνολικού γράφου στην οποία καταλήγουμε να είναι ένας γενικός γράφος από συμπλέγματα (clusters), καθένα εκ των οποίων να είναι ένα απλό δέντρο.

Κεφάλαιο 4

Λειτουργικότητα του Συστήματος Σημασιολογικής Διερεύνησης.

Στο κεφάλαιο αυτό περιγράφουμε τις λειτουργίες που παρέχει το *Σύστημα Σημασιολογικής Διερεύνησης (ΣΣΔ).* Προσπαθούμε να παρουσιάσουμε παράλληλα το σκεπτικό που διαμόρφωσε την τελική λειτουργικότητα: εναλλακτικές ιδέες που εμφανίστηκαν ως ενδιαφέρουσες κατά τη διάρκεια της σχεδίασης και ανάπτυξης, καθώς και τους λόγους για τους οποίους προτιμήθηκε η μία ή η άλλη επιλογή. Παράλληλα, αναφέρουμε και τις σημαντικότερες από τις κατευθύνσεις επέκτασης της λειτουργικότητας που μας φάνηκαν ενδιαφέρουσες, αλλά δεν μπορέσαμε να υλοποιήσουμε λόγω χρονικών περιορισμών. Τέλος, εκθέτουμε τη γνώμη μας για τη χρησιμότητα της κάθε λειτουργίας στην πράξη. Εδώ σημειώνουμε ότι για να έχουμε μια αξιόπιστη εκτίμηση χρειάζεται μια αξιολόγηση του συστήματος από ανεξάρτητους χρήστες.

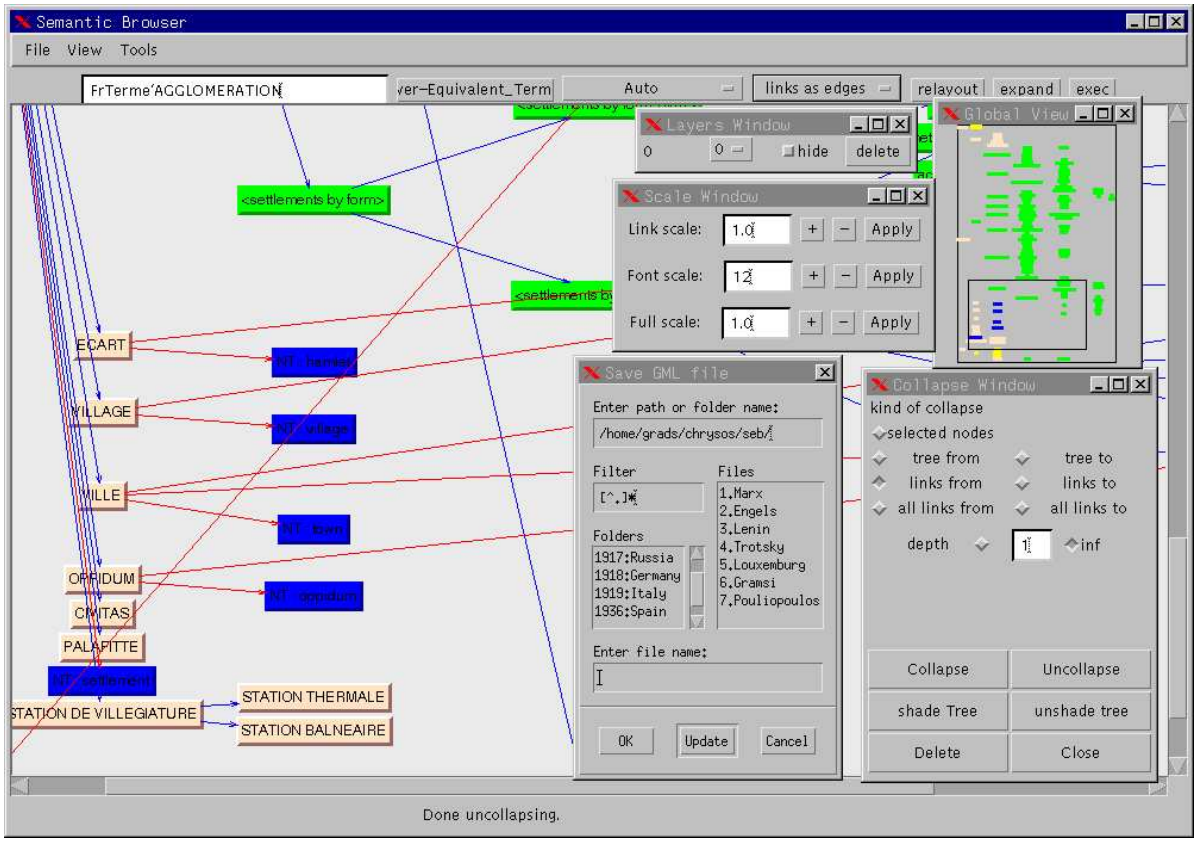
Επίπεδα λειτουργικότητας

Η διερεύνηση σημασιολογικών δικτύων είναι μια διαδικασία που ανοίγει δυνατότητες για πιο ενεργό και αποτελεσματική ανάμειξη του χρήστη στην ανεύρεση των απαιτούμενων πληροφοριών, από ότι τα παραδοσιακά συστήματα βάσεων δεδομένων. Στην εργασία μας, την αντιλαμβανόμαστε ως μια διαδικασία αποτελούμενη από ημι-αυτόματες λειτουργίες, που συνδυάζονται με την παρέμβαση και την καθοδήγηση του χρήστη ώστε να επιτύχει το στόχο της.

Όλο το σύστημα έχει σχεδιαστεί προσανατολισμένο σε αυτή τη θεώρηση. Η όλη διαδικασία μπορεί να χωριστεί σε τέσσερα επίπεδα λειτουργικότητας: Το πρώτο έχει να κάνει με την *εκτέλεση επερωτήσεων*. Το δεύτερο σχετίζεται με τη *σχεδίαση του γράφου*, το

τρίτο με *τη χρήση της σημασιολογικής και τοπολογικής πληροφορίας που περιέχεται στο γράφο απάντησης*, και το τέταρτο με *ζητήματα οπτικοποίησης του γράφου και ζητήματα της διεπιφάνειας χρήσης*.

Μια εικόνα της Διεπαφής Χρήσης του συστήματος, με τους πίνακες ελέγχου για τις διάφορες λειτουργίες, φαίνεται στο σχ. 4.1. Μπορούμε να δούμε τον έλεγχο εκτέλεσης επερώτησης και επιλογής αλγόριθμου χωροδιάταξης, αμέσως κάτω από το κυρίως μενού, τους πίνακες ελέγχου για την οπτική κλιμάκωση του γράφου και το χειρισμό των στρωμάτων του, την ολική όψη του γράφου, το παράθυρο ελέγχου για την σύμπτυξη υπογράφων, καθώς και την επιλογή για αποθήκευση του γράφου. Παρακάτω σε αυτό το κεφάλαιο θα μιλήσουμε αναλυτικότερα γι' αυτά.



Σχήμα 4.1: Η Διεπαφή Χρήσης του ΣΣΔ.

4.1 Εκτέλεση επερωτήσεων

Ένα σημασιολογικό δίκτυο εν γένει και μια βάση του ΣΣΕ ειδικότερα, περιέχουν συνήθως εκατοντάδες ή χιλιάδες οντότητες και συσχετίσεις. Για να περιορίσουμε την

έκταση της πληροφορίας που παρουσιάζουμε στο χρήστη, χρησιμοποιούμε προκατασκευασμένες επερωτήσεις, τις οποίες ο χρήστης εκτελεί επάνω σε μια οντότητα, που ονομάζουμε *''στόχο''* της επερώτησης. Περισσότερες πληροφορίες για το μηχανισμό των προκατασκευασμένων ερωτήσεων υπάρχουν στο [15].

Το αποτέλεσμα της επερώτησης παρουσιάζεται σε μορφή γράφου. (Στην πράξη είναι σύνηθες, αλλά όχι απαραίτητο, ο γράφος αυτός να είναι συνδεδεμένος.) Τους γράφους αυτούς, θα τους ονομάζουμε παρακάτω για συντομία ''γράφους απάντησης''. Αφού φτάσουμε σε ένα γράφο απάντησης, τον σχεδιάζουμε με χρήση ενός αλγορίθμου αυτόματης σχεδίασης γράφων, και τον παρουσιάζουμε στο χρήστη. Ο χρήστης μπορεί να εκτελέσει κατόπιν κάποια λειτουργία στο γράφο, ή να επιλέξει μια νέα οντότητα που τον ενδιαφέρει, και να εκτελέσει μια νέα επερώτηση.

Τα παραπάνω αποτελούν τη βασική διαδικασία που εκτελείται κατά τη χρήση του συστήματος.

Στη συνέχεια αυτής της ενότητας, θα παρουσιάσουμε τη λειτουργία της *''επέκτασης ερώτησης'' (query expansion)*, που επεκτείνει το παραπάνω βασικό σχήμα.

4.1.1 Επέκταση ερώτησης

Το βασικό σχήμα επερώτησης που είδαμε, είναι πολύ στατικό. Ο χρήστης επιθεωρεί έναν γράφο απάντησης, και κατόπιν προχωράει σε κάποιον νέο. Με αυτόν όμως τον τρόπο, η σχέση που είναι σύνηθες να υπάρχει ανάμεσα σε διαδοχικούς κατά τη διερεύνηση γράφους, δεν γίνεται σαφής και άμεσα αντιληπτή από το χρήστη.

Για παράδειγμα, έστω ότι ο χρήστης επιθεωρεί όλα τα ιστορικά στοιχεία που αναφέρονται *από* την οντότητα *Αλή Πασά*, και σε κάποια στιγμή θέλει να δεί όλα τα ιστορικά στοιχεία που σχετίζονται *με οποιονδήποτε τρόπο* με τον Αλή Πασά. Προφανώς, το σύνολο απάντησης στη δεύτερη περίπτωση περιλαμβάνει το σύνολο απάντησης στην πρώτη, ως υποσύνολό του. Θα θέλαμε αυτή η σχέση να γίνεται εμφανής στο χρήστη.

Οδηγούμαστε έτσι στην ιδέα της *''επέκτασης ερώτησης''*. Δεδομένου ενός συνόλου απάντησης, ο χρήστης μπορεί να εκτελέσει μια νέα επερώτηση (συνηθέστερα αλλά όχι απαραίτητα επάνω σε ένα στοιχείο του αρχικού συνόλου απάντησης), και τα αποτελέσματα της νέας επερώτησης να θεωρηθεί ότι *''επεκτείνουν''* το προηγούμενο αποτέλεσμα.

Μερικά σημαντικά ζητήματα προκύπτουν κατ' αυτή τη λειτουργία: Τί είδους επερωτήσεις είναι κατάλληλες για τέτοιες επεκτάσεις; Με ποιό τρόπο οπτικοποιούμε την επέκταση; Παρακάτω αντιμετωπίζουμε αυτά τα ζητήματα.

4.1.1.1 Είδη επερωτήσεων

Μια πρώτη ιδέα για επέκταση του γράφου απάντησης, είναι η προσέγγιση που είδαμε στην Concept Base, στην ενότητα 2.1.1: η επέκταση του γράφου αφορά συγκεκριμένες ιδιότητες των οντοτήτων που εμφανίζονται σε αυτόν. Μπορούμε λ.χ. να ρωτήσουμε για τις κλάσεις, τις υπερκλάσεις ή τις συσχετίσεις μιας οντότητας που έχουμε εντοπίσει.

Με μια δεύτερη σκέψη όμως, συμπεραίνουμε ότι δεν υπάρχει λόγος να περιοριστούμε σε τέτοιες ''απλές'' επεκτάσεις. Αντίθετα, κρίθηκε σκόπιμο *επέκταση να μπορεί να γίνει με βάση οποιαδήποτε προκατασκευασμένη επερώτηση.* Έτσι, ο γράφος μπορεί να επεκταθεί κατά οποιονδήποτε τρόπο μπορεί να φανταστεί χρήσιμο ο χρήστης (ή ο σχεδιαστής της βάσης). Οι περιπτώσεις των ''απλών'' επερωτήσεων για συγκεκριμένες ιδιότητες οντοτήτων, μπορούν να παρέχονται επίσης ως προκατασκευασμένες.

Άλλη σημαντική παρατήρηση είναι ότι είναι πιθανόν ο χρήστης να θέλει επιπλέον πληροφορίες για περισσότερα από ένα αντικείμενα που εμφανίζονται στο γράφο απάντησης. Θα ήταν χρήσιμο να μπορεί να γίνει επέκταση της ερώτησης επάνω σε όλα τα αντικείμενα ενδιαφέροντος ταυτόχρονα.

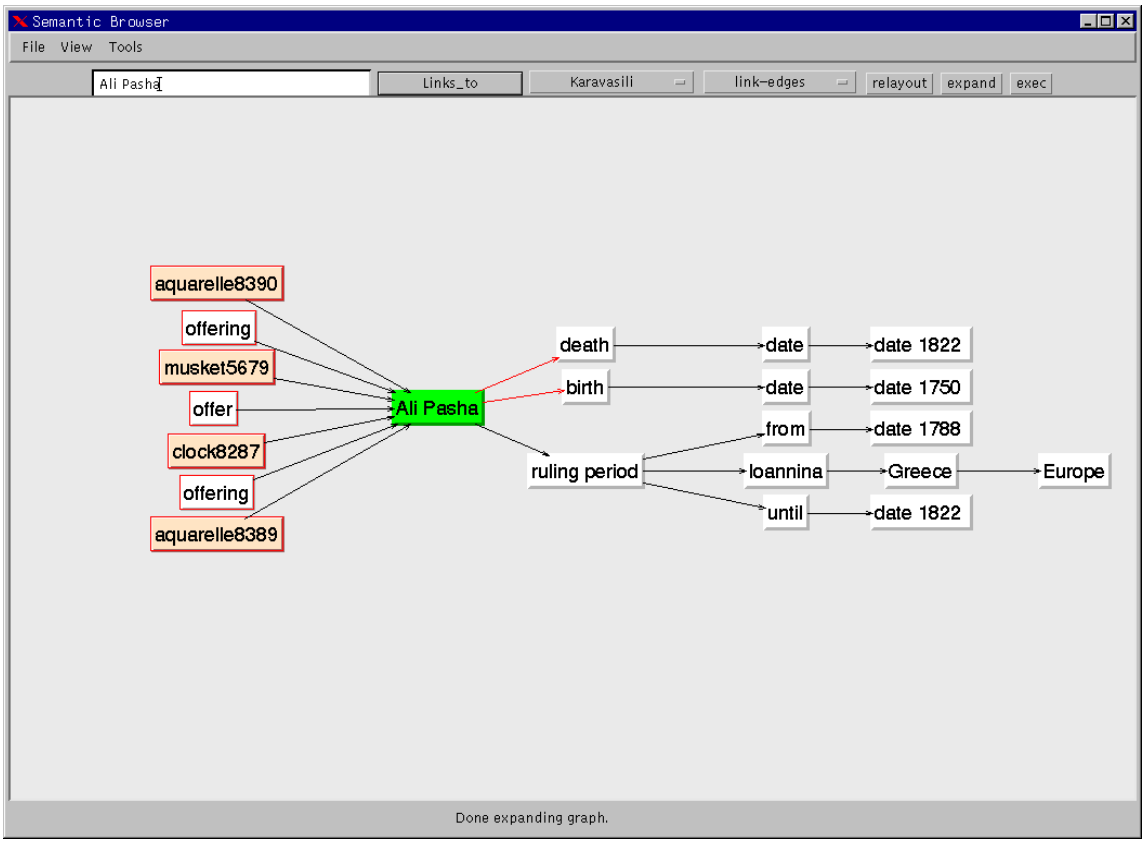
Όλες οι παραπάνω δυνατότητες παρέχονται από το σύστημά μας: μια συγκεκριμένη επερώτηση μπορεί να εφαρμοστεί επάνω σε ένα σύνολο από προεπιλεγμένα αντικείμενα, και τα αποτελέσματα να εμφανιστούν στο χρήστη με μιας, σε ένα βήμα.

4.1.1.2 Οπτικοποίηση της επέκτασης

Όπως είδαμε, το αποτέλεσμα μιας επερώτησης παρουσιάζεται στο χρήστη με μορφή γράφου. Έτσι, προτού ο χρήστης θελήσει να επεκτείνει το αποτέλεσμά του, έχει ήδη εξερευνήσει και εξοικειωθεί με έναν γράφο απάντησης, ο οποίος είναι σχεδιασμένος κατά ένα συγκεκριμένο τρόπο. Κατά την επέκταση ερώτησης, παρουσιάζεται το κρίσιμο ζήτημα του πως θα τοποθετηθούν οι κόμβοι και οι ακμές που προκύπτουν. Ο παλιός γράφος θα πρέπει να μείνει κατά το δυνατόν αμετάβλητος, ώστε ο χρήστης να μη χάσει τον ''προσανατολισμό'' που είχε αποκτήσει, ενώ οι νέοι κόμβοι και ακμές πρέπει να τοποθετηθούν κατά τρόπο τέτοιον, ώστε ο τελικός γράφος να είναι ευνόητος και αισθητικά ευχάριστος. Όπως είδαμε στο Κεφ. 3, το ζήτημα μπορεί να λυθεί με χρήση ενός αλγορίθμου αυξητικής σχεδίασης, προσέγγιση που ακολουθείται λ.χ. στο [50]. Όμως τέτοιοι αλγόριθμοι αρχίζουν να γίνονται διαθέσιμοι μόλις τον τελευταίο καιρό, και δεν είναι γενικά διαθέσιμοι. Ακόμα, αρκετοί απ' αυτούς πάσχουν από διάφορα προβλήματα, που προαναφέραμε στην παράγραφο 3.3. Με τους αλγόριθμους σχεδίασης που διαθέτουμε, και οι οποίοι είναι πολύ ευαίσθητοι σε αλλαγές του γράφου, αντιμετωπίζουμε το πρόβλημα

με τον παρακάτω τρόπο: Για κάθε επερώτηση επέκτασης, δημιουργείται ένας ξεχωριστός γράφος με την απάντησή της, και σχεδιάζεται με βάση οποιονδήποτε αλγόριθμο έχει επιλέξει ο χρήστης, σαν να επρόκειτο για απλή επερώτηση. Η συνένωση με τον αρχικό γράφο απάντησης γίνεται κατόπιν με συγχώνευση των δύο γράφων. Κατά τη συγχώνευση οι κοινοί (παλιοί) κόμβοι διατηρούν τη θέση που είχαν στον αρχικό γράφο, ώστε ο χρήστης να μην χάνει τον προσανατολισμό του, ενώ οι νέοι κόμβοι τοποθετούνται σε θέσεις σχετικές με τον κόμβο-οντότητα πάνω στον οποίο εκτελέστηκε η επερώτηση.

Ένα απλό παράδειγμα επέκτασης μπορούμε να δούμε στο σχ. 4.2. Σε αυτό το παράδειγμα, ο χρήστης αφού έχει εκτελέσει μια αρχική επερώτηση για τα ιστορικά στοιχεία του Αλή Πασά, εκτελεί μια επέκταση της επερώτησης, ζητώντας όλες τις οντότητες που αναφέρονται στον Αλή Πασά. Το αποτέλεσμα είναι οι τονισμένες οντότητες στα αριστερά του Αλή Πασα.



Σχήμα 4.2: Επέκταση επερώτησης

Ας σημειώσουμε τέλος ότι υπάρχει και εναλλακτική προσέγγιση για την τοποθέτηση των κόμβων της επέκτασης. Κατ' αυτήν, αντί για σχεδίαση πρώτα και κατόπιν συγχώνευση του γράφου επέκτασης, υπολογίζεται μια ''διαφορά'' του γράφου επέκτασης από τον αρχικό γράφο, οπότε προκύπτει εν γένει ένα σύνολο ακμών και κόμβων μικρότερο από

ολόκληρο το σύνολο της επέκτασης. Αυτό μένει να σχεδιαστεί και να ‘‘συγχωνευθεί’’, τώρα πια με απλή προσθήκη, στον αρχικό γράφο. Το αποτέλεσμα και των δυο μεθόδων περιέχει ακριβώς τους ίδους κόμβους και ακμές, αλλά έχει εν γένει διαφορετική χωροδιάταξη . Προτιμήσαμε τον πρώτο τρόπο, λόγω της παρατήρησης ότι στη τελευταία περίπτωση, ο γράφος που απομένει για σχεδίαση είναι πολύ συχνά ασύνδετος. Αρκετοί αλγόριθμοι σχεδίασης γράφων χειρίζονται μόνο (ή πολύ καλύτερα) συνδεδεμένους γράφους, όπως π.χ. οι αλγόριθμοι ελατηρίου.

Στην επόμενη ενότητα, παρουσιάζουμε την εμπειρία μας από τη μέχρι τώρα χρήση του συστήματος, και την εκτίμηση για την αποδοτικότητα της προσέγγισής μας.

4.1.1.3 Η λειτουργία επέκτασης στην πράξη

Έπειτα από έναν αριθμό επεκτάσεων επερώτησης πάνω σε έναν αρχικό γράφο, το τελικό σχέδιο του γράφου απάντησης δεν είναι εγγυημένα ικανοποιητικό. Σε περίπτωση π.χ. επέκτασης ερώτησης επάνω σε έναν κόμβο που βρίσκεται στο κέντρο του αρχικού γράφου, οι νέοι κόμβοι πιθανότατα θα επικαλύπτονται με παλιούς, και αρκετοί σύνδεσμοι θα διασταυρώνονται.

Με κατάλληλη όμως επιλογή από μέρους του χρήστη αλγορίθμων σχεδίασης[1], τα αποτελέσματα μπορούν να βελτιωθούν δραματικά. Για παράδειγμα, αλγόριθμοι που τείνουν να σχεδιάζουν το γράφο προς συγκεκριμένη κατεύθυνση, μπορούν να δώσουν ικανοποιητικότερα αποτελέσματα όταν ο κόμβος-στόχος βρίσκεται προς την περιφέρεια του γράφου: Για έναν κόμβο π.χ. που βρίσκεται ‘‘ψηλά’’ στο σχέδιο του γράφου, καλό είναι ο γράφος επέκτασης να σχεδιάζεται ‘‘προς τα πάνω’’.

Από την εμπειρία μας με το μηχανισμό επέκτασης στην πράξη, μπορούμε να πούμε ότι είναι αρκετά εύχρηστος, τουλάχιστον για έμπειρους χρήστες.

Συμπερασματικά, θα μπορούσαμε να πούμε ότι με κατάλληλη χρήση, ο μηχανισμός επέκτασης μπορεί να χρησιμοποιηθεί με ικανοποιητικά αποτελέσματα στην πράξη, για έναν μικρό αριθμό διαδοχικών επεκτάσεων (πέντε ή έξι), οι οποίες όμως είναι συνήθως αρκετές.

Σημαντικό τέλος ρόλο για την ευχρηστία του μηχανισμού παίζουν αρκετές λειτουργίες που έχουν προβλεφθεί για να βοηθήσουν το χρήστη κατά τη διαδικασία της επέκτασης ερώτησης: Η καινούργια πληροφορία που προκύπτει, τοποθετείται σε ένα ξεχωριστό *στρώμα (layer)* του γράφου (παρ. 4.3.3), και μπορεί να σκιαστεί ή να αποκρυβεί κατά βούληση ώστε να γίνει εμφανής η σχέση της με τον προϋπάρχον γράφο (παρ. 4.4.1.3).

[1]Οι αλγόριθμοι που προσφέρονται από το σύστημά μας, αναφέρονται στην παρ. 4.2.

Σε περιπτώσεις που προκύπτουν επικαλύψεις κόμβων, μπορεί να χρησιμοποιηθεί μια λειτουργία αυτόματης εξάλειψης των επικαλύψεων, η οποία παρέχεται από το σύστημά μας (παρ. 4.2.4). Επίσης, σε περιπτώσεις που ο γράφος που έχει προκύψει είναι υπερβολικά μπερδεμένος, μια αυτόματη ανασχεδίαση του γράφου μπορεί να γίνει, κατόπιν επιλογής του χρήστη (παρ. 4.2.3). Σε αντίστοιχες ενότητες στη συνέχεια εξετάζουμε με μεγαλύτερη λεπτομέρεια κάθε μια από αυτές τις λειτουργίες.

4.2 Χωροδιάταξη του γράφου απάντησης

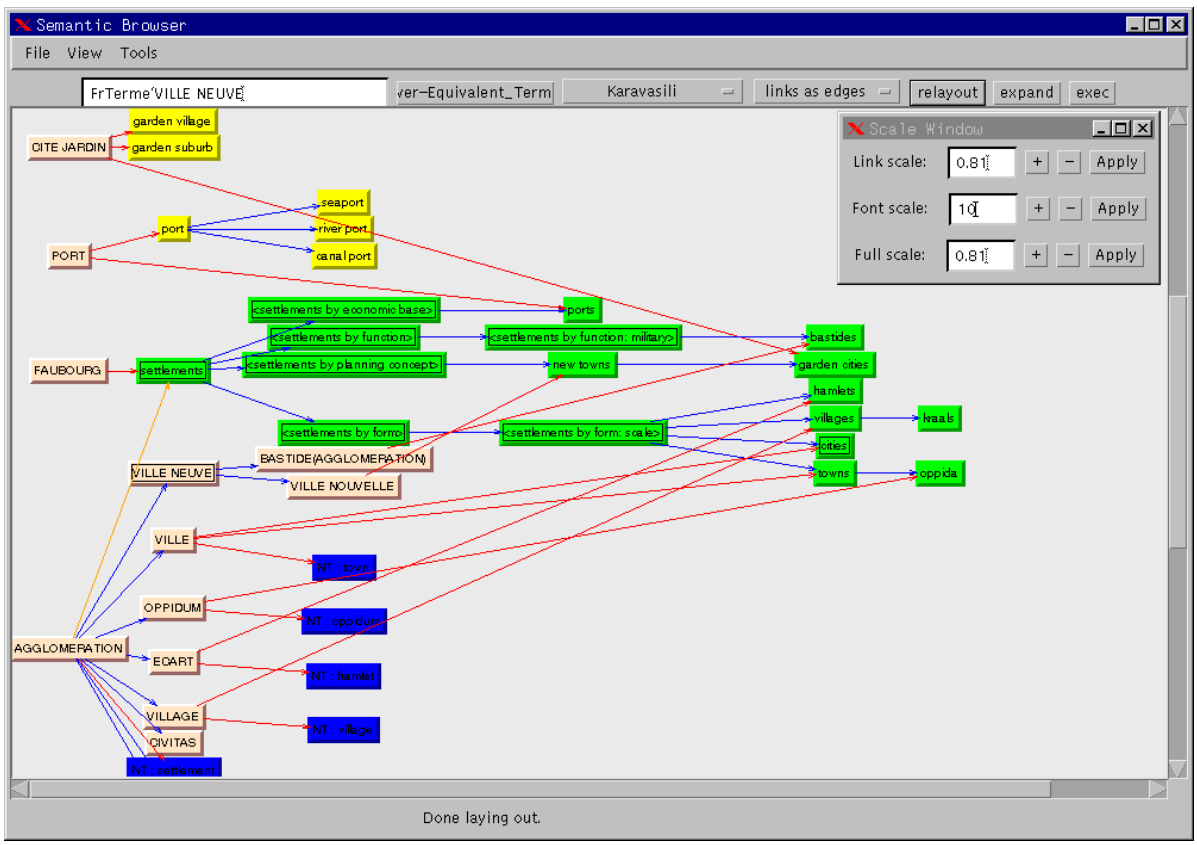
Όπως έχουμε ήδη πει, η χωροδιάταξη ενός γράφου απάντησης είναι κρίσιμη για την εύκολη κατανόησή του. Στην εργασία μας δεν υλοποιήσαμε νέο αλγόριθμο για σχεδίαση γράφων, αλλά χρησιμοποιήσαμε έτοιμους αλγόριθμους. Συγκεκριμένα, χρησιμοποιούμε τον ιεραρχικό αλγόριθμο της Μ. Καραβασίλη [1], τον ιεραρχικό αλγόριθμο dot [33] ο οποίος επιτρέπει προσδιορισμό κατεύθυνσης σχεδίασης, καθώς και τους αλγόριθμους ελατηρίου του Kamada [32], του Tunkelang [49] και του Frick [21], οι οποίοι περιέχονται στο σύστημα σχεδίασης γράφων Graphlet [25]. Ο προεπιλεγμένος αλγόριθμος που χρησιμοποιήται είναι της Καραβασίλη, λόγω της ταχύτητάς του και της ποιότητας του αποτελέσματός του, όσον αφορά την ελαχιστοποίηση των διασταυρώσεων των ακμών και το μέγεθος της περιοχής σχεδίασης. Ο χρήστης μπορεί κατ' επιλογή να χρησιμοποιήσει κάποιον από τους υπόλοιπους αλγόριθμους. Σε μια μελλοντική επέκταση του συστήματος, η επιλογή του κατάλληλου αλγορίθμου θα μπορούσε να γίνεται αυτόματα, βάσει χαρακτηριστικών του προς χωροδιάταξη γράφου. Το σύστημα μπορεί επίσης να επεκταθεί αρκετά εύκολα, ώστε να χρησιμοποιεί νέους αλγόριθμους. Λεπτομέρειες για τον τρόπο υπάρχουν στο Κεφ. 5.

Έως τώρα, υποθέσαμε ότι οι συχετίσεις παριστάνονται με ακμές στο γράφο απάντησης. Παρακάτω, εξετάζουμε τη δυνατότητα *οι συσχετίσεις να παριστάνονται ως κόμβοι*, και την ευχέρεια που μας παρέχει αυτό στην κατανόηση του γράφου. Ένα άλλο ζήτημα που εντοπίσαμε ήδη, είναι *η εξάλειψη τυχόν επικαλύψεων των κόμβων*, σε περίπτωση που επικαλύψεις προκύπτουν κατά τη σχεδίαση ή κατόπιν επεκτάσεων του γράφου. Παρακάτω παρουσιάζουμε αναλυτικότερα πώς χειριζόμαστε αυτά τα ζητήματα, και περιγράφουμε επίσης τις λειτουργίες της *ανασχεδίασης* και της *συγχώνευσης γράφων*, που προκύπτουν ως απαραίτητες στη διαδικασία της διερεύνησης.

4.2.1 Παράσταση των συσχετίσεων ως ακμών ή κόμβων

Ο απλούστερος τρόπος να παραστήσουμε μια συσχέτιση ανάμεσα σε δυο οντότητες είναι με μια ακμή ανάμεσα στους κόμβους των οντοτήτων [24]. Η προσέγγιση αυτή

είναι απλή, και είναι η προσφορότερη όταν είναι απλό για το χρήστη να αντιληφθεί την σημασία του κάθε συνδέσμου, για παράδειγμα από το χρωματισμό του.



Σχήμα 4.3: Συσχετίσεις ανάμεσα σε όρους διαφόρων θησαυρών όρων.

Στο σχ.4.3, έχουμε έναν γράφο με συσχετίσεις ανάμεσα σε όρους θησαυρών[2] [18]. Οι συσχετίσεις που εμφανίζονται είναι (κυρίως) δύο κατηγοριών, συγκεκριμένα *στενότερος όρος* και *συνώνυμος όρος*. Σε μια τέτοια περίπτωση, είναι σχετικά εύκολο για τον χρήστη να θυμάται ότι οι μπλέ σύνδεσμοι αντιστοιχούν σε στενότερους όρους, και οι κόκκινοι σε ισοδύναμους όρους. Παρατηρούμε ότι μια χρωματισμένη (ανώνυμη) ακμή σε τέτοιες περιπτώσεις αρκεί για να μεταφέρει την απαραίτητη πληροφορία.

Σε περιπτώσεις όμως που υπάρχουν πολλά διαφορετικά είδη συσχετίσεων, η χρήση χρώματος δεν επαρκεί για την αντιστοίχιση των συνδέσμων στις κατηγορίες τους: ο χρήστης δεν μπορεί να θυμάται την έννοια πάρα πολλών χρωμάτων και τα πράγματα είναι χειρότερα με τον αρχάριο ή περιστασιακό χρήστη, για τον οποίο δεν είναι πάντα ευχάριστο να ξοδεύει χρόνο για να απομνημονεύει σημασίες χρωμάτων. Σε αυτές τις περιπτώσεις, η σχέση, στην οποία αντιστοιχεί κάθε συνδέσμος, γίνεται σαφής με πιο άμεσο τρόπο, με

[2]Μερικοί κόμβοι στο γράφο εμφανίζονται με διπλό περιθώριο. Οι κόμβοι αυτοί περιέχουν συμπτυγμένη πληροφορία. Το γράφο αυτό θα τον αναφέρουμε και παρακάτω, στην ενότητα 4.3.2.2, όπου θα συζητήσουμε την λειτουργία της σύμπτυξης κόμβων

την παρουσίαση του ονόματος του γνωρίσματος (ή της κατηγορίας στην οποία ανήκει, εαν αυτό είναι ανώνυμο). Το όνομα μπορεί να αναγράφεται κοντά στη γραμμή που παριστάνει την ακμή. Δυστυχώς όμως, μια απλοϊκή αυτόματη τοποθέτηση ετικετών στους συνδέσμους πολύ συχνά οδηγεί σε ένα δυσανάγνωστο σχήμα, με επικαλύψεις ετικετών. Η τοποθέτηση ετικετών στη γενικότητά της είναι δύσκολο πρόβλημα [27], και ελάχιστοι από τους αλγόριθμους για σχεδίαση γράφων που κυκλοφορούν σήμερα την υποστηρίζουν [31].

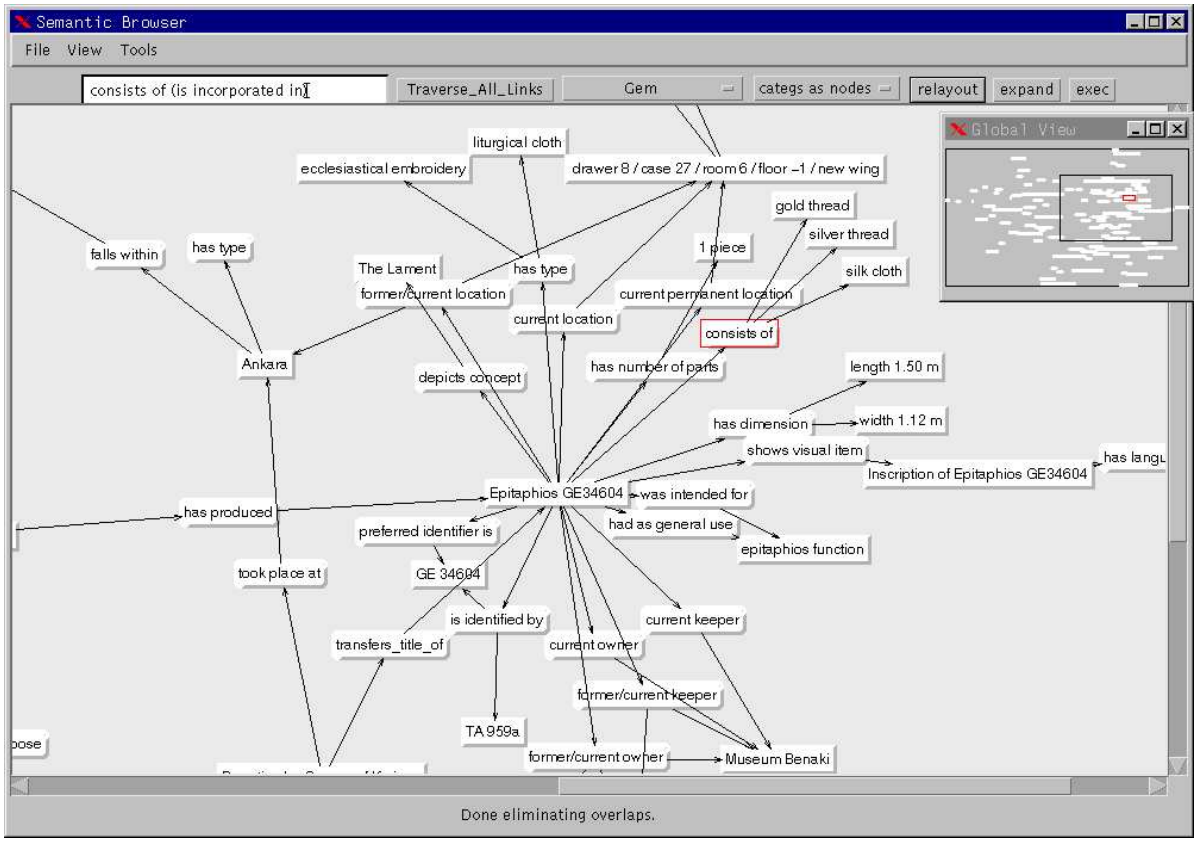
Η εναλλακτική προσέγγιση που ακολουθούμε για να παρακάμψουμε το πρόβλημα, συνίσταται στην *παράσταση και των συσχετίσεων με κόμβους* (σχ. 4.4). Για να ξεχωρίζουν από τους κόμβους των οντοτήτων, οι κόμβοι συσχετίσεων έχουν διαφορετικό σχήμα. Η προσέγγισή μας αυτή μας επιτρέπει να χειριζόμαστε ομοιόμορφα και συσχετίσεις συσχετίσεων: Οι αλγόριθμοι σχεδίασης τους χειρίζονται με τον ίδιο τρόπο όπως όλους τους υπόλοιπους κόμβους.

Τέλος, μια ιδέα που προκύπτει φυσιολογικά για περαιτέρω βελτίωση της σαφήνειας του γράφου, είναι η παράσταση ενός ολόκληρου σύνολου συνδέσμων με έναν κόμβο βάσει την κατηγορία τους: Για κάθε σύνολο συνδέσμων που ξεκινούν από τον ίδιο κόμβο και ανήκουν στην ίδια κατηγορία, δημιουργείται *ένας* μόνο κόμβος, όπως φαίνεται στο σχ. 4.4 (οι κόμβοι με τις στρογγυλεμένες γωνίες), με ετικέτα το όνομα της κατηγορίας.

4.2.2 Συγχώνευση γράφων

Ένας αριθμός από γράφους απαντήσεων μπορούν να συγχωνευθούν ώστε να πάρουμε έναν συνολικό γράφο, που περιλαμβάνει συγκεντρωμένη όλη την πληροφορία που έχουν οι αρχικοί γράφοι. Κάθε αρχικός γράφος μπορεί να είναι σχεδιασμένος με οποιονδήποτε αλγόριθμο. Μετά τη συγχώνευση, η χωροδιάταξη του συνολικού γράφου προέρχεται από τα επιμέρους σχέδια του κάθε υπο-γράφου. Έτσι, για παράδειγμα, μπορούμε να έχουμε προς συγχώνευση ένα ‘‘κεντρικό’’ γράφο στον οποίο οι συμμετρίες έχουν τονιστεί από κάποιον αλγόριθμο, μαζί με μικρότερους γράφους που περιέχουν λιγότερο σημαντική πληροφορία, π.χ. ‘‘ιδιότητες’’ κόμβων του κεντρικού, και οι οποίοι έχουν μορφή δέντρων. Η λειτουργία της συγχώνευσης μπορεί να χρησιμοποιηθεί λ.χ. για σύγκριση ενός γράφου απάντησης *πριν* και *μετά* την ενημέρωση της βάσεως με νέα δεδομένα, ώστε να γίνουν φανερές οι διαφορές. Στο σύστημά μας, χρησιμοποιείται για την επέκτασης ερώτησης, όπως περιγράψαμε στην παρ. 4.1.1.

Κατά την συγχώνευση, χρησιμοποιείται ο μηχανισμός των στρωμάτων: κάθε γράφος που συγχωνεύεται, τοποθετείται σε ένα ξεχωριστό στρώμα του νέου γράφου. Κατ’ αυτόν τον τρόπο, η εποπτεία των αλλαγών στο γράφο γίνεται πολύ εύκολη (βλ. και τις σχετικές



Σχήμα 4.4: Κόμβοι κατηγοριών

τεχνικές για εποπτεία των στρωμάτων στην παρ. 4.4.1.3).

Σημειώνουμε τέλος ότι για να συγχωνευτεί ένα σύνολο γράφων απαντήσεων, θα πρέπει οι συσχετίσεις να παριστάνονται σε όλους τους γράφους με τον ίδιο τρόπο, ως κόμβοι ή ακμές. Διαφορετικά δημιουργούμε γράφους που δεν μπορούμε να χειριστούμε με ομοιόμορφο τρόπο.

4.2.3 Ανασχεδίαση του γράφου

Μετά την παρουσίαση ενός γράφου απάντησης στο χρήστη, αυτός μπορεί να τον επεξεργαστεί κατά διάφορους τρόπους. Μπορεί να τον επεκτείνει βάσει μιας νέας επερώτησης, να συμπτύξει κόμβους του και να σβήσει άλλους, ή να μετακινήσει κομμάτια του γράφου. Δεν είναι ασυνήθιστο να παρουσιάζονται περιπτώσεις όπου ο γράφος καταλήγει να είναι πολύ ‘‘μπερδεμένος’’ στην οθόνη. Σε τέτοιες περιπτώσεις ο χρήστης μπορεί να επιλέξει *ανασχεδίαση* του γράφου, σύμφωνα με κάποιον αλγόριθμο της προτίμησής του. Έτσι, καταλήγει σε μια καθαρότερη εικόνα του γράφου, όπως τον έχει διαμορφώσει σε κάθε φάση.

4.2.4 Εξάλειψη επικαλύψεων

Για την περίπτωση που πολλοί κόμβοι επικαλύπτουν ο ένας τον άλλον στο σχέδιο κάποιου γράφου, παρέχεται μια απλή λειτουργία εξάλειψης των επικαλύψεων, με αυτόματη μετα-τοποθέτηση των κόμβων. Η λειτουργία αυτή προέκυψε ως αναγκαιότητα για τη γρήγορη και εύχρηστη αντιμετώπιση ‘‘εύκολων’’ περιπτώσεων: μπορεί να βοηθήσει για ξεκαθάρισμα του γράφου όταν αυτός είναι ελαφρώς μπερδεμένος, και απλές μετακινήσεις μερικών κόμβων αρκούν. Για πολύπλοκες όμως καταστάσεις δεν επαρκεί, γιατί δεν παίρνει υπ’ όψιν της τους κρυμμένους ή διασταυρούμενους σύνδεσμους που προκύπτουν. Σε τέτοιες περιπτώσεις, χρειάζεται μια ανασχεδίαση του γράφου, όπως περιγράψαμε στην προηγούμενη παράγραφο.

4.3 Σημασιολογική και τοπολογική πληροφορία στο γράφο

Εκτός από τη σημασιολογία του, ένας γράφος σημασιολογικού δικτύου παρέχει πληροφορία και με τη χωροδιάταξή του, καθώς και με την τοπολογία του.

Κάθε σχεδιασμένος γράφος (χωρίς να είναι κατ’ ανάγκη σημασιολογικό δίκτυο) παρέχει πληροφορία με τη *χωροδιάταξή* του. Για παράδειγμα, ένα κατάλληλο σχέδιο του γράφου ενός πλέγματος (διχτυού) ή ενός στατικού δομικού μοντέλου, ας πούμε ενός κτιρίου, μας κάνει να αντιληφθούμε περί τίνος πρόκειται. Επίσης, κάθε γράφος παρέχει πληροφορία με την *τοπολογία* του, μπορεί για παράδειγμα να είναι δενδρικός, οπότε καθορίζει μιαν ιεραρχία των κόμβων, να είναι πλήρως συνδεδεμένος, κλπ. (Παρατηρούμε ότι χωροδιάταξη και τοπολογία είναι διαφορετικές ιδιότητες: ένα δέντρο μπορεί να έχει σχεδιαστεί έτσι ώστε να ‘‘θυμίζει’’ πλέγμα).

Την σημασιολογική και την τοπολογική αυτή πληροφορία, μπορούμε να τη χρησιμοποιήσουμε ώστε να βοηθήσουμε το χρήστη στην εξερεύνηση του σημασιολογικού δικτύου. Μπορούμε να συμπτύξουμε πληροφορία από πολλούς κόμβους σε έναν βάσει είτε της τοπολογίας του γράφου ή της σημασιολογίας των συνδέσμων.

Παρακάτω παρουσιάζουμε αναλυτικότερα αυτούς τους μηχανισμούς. Ξεκινάμε συζητώντας εκτενέστερα τις δυνατότητες που μας δίνονται από την κατηγοριοποίηση των συνδέσμων, στην επόμενη ενότητα.

4.3.1 Κατηγοριοποίηση των συνδέσμων

Οι συσχετίσεις ανάμεσα σε οντότητες της Telos μπορούν να χωριστούν σε διάφορα

είδη, καθένα εκ των οποίο προσδίδει στη σχέση επιπρόσθετη σημασία και θέτει επιπλέον περιορισμούς, όπως επισημαίνει ο Θεοδωράκης [2]. Η κατηγοριοποίησή τους γίνεται βάσει δύο ειδών περιορισμών: με *περιορισμούς πλήθους*, οι οποίοι περιορίζουν τον αριθμό των οντοτήτων με τις οποίες μπορεί να σχετίζεται μία οντότητα (σχέσεις *μονοσήμαντες, αντίστροφα μονοσήμαντες, και αμφιμονοσήμαντες*), και με *περιορισμούς ακεραιότητας*, οι οποίοι περιορίζουν τις εξαρτήσεις των συσχετιζόμενων οντοτήτων μεταξύ τους (σχέσεις *ολικές, επί, ολικές και επί*).

Είναι χρήσιμο να έχουμε διαθέσιμη την πληροφορία για το είδος κάθε κατηγορίας συνδέσμων, γιατί μπορούμε να την χρησιμοποιήσουμε για να απλοποιήσουμε το γράφο που παρουσιάζουμε στο χρήστη.

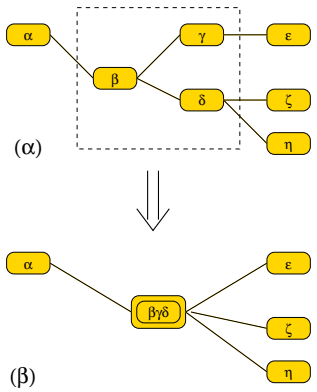
Ας εξετάσουμε συγκεκριμένα το είδος των ολικών σχέσεων, που στη βιβλογραφία των Βάσεων Δεδομένων αναφέρονται ως *dependent (εξαρτώμενες).* Αυτές οι εξαρτώμενες κατηγορίες συνδέσμων στην πράξη χρησιμοποιούνται για να παραστήσουν πληροφορία κατά τρόπο αντίστοιχο των *''σύνθετων αντικειμένων''*, που γνωρίζουμε από τις Βάσεις Δεδομένων: οι οντότητες στις οποίες καταλήγουν οι σύνδεσμοι, δεν έχουν αυτόνομη ύπαρξη (στο βαθμό που έχουν άλλες οντότητες της βάσης), παρά έχουν νόημα μόνο σε σχέση με την οντότητα με την οποία σχετίζονται. Έτσι, αποτελούν στην ουσία ''ιδιότητες'' μιας άλλης, ''κυρίαρχης'' οντότητας. Κατ' αυτή την έννοια, η πληροφορία που παριστάνουν έχει νόημα για το χρήστη μόνο σε σχέση με μια συγκεκριμένη οντότητα της βάσης. Αυτό μπορούμε να το εκμεταλλευτούμε συμπτύσσοντας την πληροφορία που περιλαμβάνουν σε έναν κύριο κόμβο που παριστάνει την κυρίαρχη οντότητα. Η συγκεκριμένη λειτουργία, για σύμπτυξη ειδικά των εξαρτώμενων κατηγοριών, δεν παρέχεται από το σύστημά μας, αλλά αποτελεί μια πιθανότατα χρήσιμη, και εύκολη στην υλοποίηση επέκτασή του.

Το είδος των εξαρτώμενων κατηγοριών που παρουσιάσαμε παραπάνω, είναι ένα μόνο παράδειγμα της χρησιμότητας της σημασιολογικής πληροφορίας των συνδέσμων. Γενικότερα, μπορούμε να χρησιμοποιήσουμε την πληροφορία που παρέχεται από τις κατηγορίες των συνδέσμων για να απλοποιήσουμε το γράφο κατά διάφορους τρόπους, όχι μόνο βάσει μιας προκαθορισμένης σύμβασης όπως στο παραπάνω παράδειγμα, αλλά κάτω από τον δυναμικό έλεγχο του χρήστη. Στο σύστημά μας, η πληροφορία αυτή χρησιμοποιήται για την σύμπτυξη κόμβων που συσχετίζονται μεταξύ τους μέσω σύνδεσμων μιας κατηγορίας, όπως περιγράφεται στην ενότητα 4.3.2.3.

Τέλος, η πληροφορία των κατηγοριών των συνδέσμων χρησιμοποιείται και με πιο άμεσο (αυτόματο) τρόπο, για την ονοματοδοσία των ανώνυμων συνδέσμων ή την αντικατάσταση πολλών συνδέσμων της ίδιας κατηγορίας με έναν, όπως αναφέραμε στην παρ. 4.2.1.

4.3.2 Σύμπτυξη πληροφορίας σε έναν κόμβο

Ένας άμεσος τρόπος να μειώσουμε την πολυπλοκότητα ενός γράφου, είναι να ελαττώσουμε τον αριθμό των κόμβων του. Αυτό μπορεί να γίνει *συμπυκνώνοντας* πληροφορία από πολλούς κόμβους σε έναν (ας τον ονομάσουμε *συνοπτικό κόμβο*).



Σχήμα 4.5: Σύμπτυξη κόμβων

Ένα παράδειγμα αυτής της λειτουργίας φαίνεται στο σχήμα 4.5. Ένας νέος κόμβος δημιουργείται και αντικαθιστά έναν αριθμό από παλιούς κόμβους. Οι εσωτερικές ακμές ανάμεσα στους παλιούς κόμβους εξαφανίζονται στο νέο γράφο που προκύπτει, ενώ οι εξωτερικές ακμές από και προς τους παλιούς κόμβους αντικαθιστώνται από ακμές αντίστοιχα από και προς το νέο κόμβο. Η ετικέτα του νέου κόμβου είναι εν γένει συνάρτηση των ετικετών των ‘‘περιλαμβανομένων’’ κόμβων. Ο χρήστης μπορεί κατ’ επιλογή να βλέπει είτε μια σύντομη ετικέτα στον συνοπτικό κόμβο, είτε την πλήρη ετικέτα που περιλαμβάνει τις ετικέτες όλων των συμπτυγμένων κόμβων.

Αυτή η απλή λειτουργία μπορεί να χρησιμοποιηθεί με πλήθος τρόπων, ώστε να μας παρέχει έναν αρκετά ισχυρό και εύχρηστο μηχανισμό για την διερεύνηση. Στο σύστημά μας τη χρησιμοποιούμε για να παρέχουμε λειτουργίες ομαδοποίησης των κόμβων και τοπολογικής και σημασιολογικής σύμπτυξης.

4.3.2.1 Ομαδοποίηση

Ένα αυθαίρετο σύνολο κόμβων επιλεγμένων από το χρήστη μπορεί να *ομαδοποιηθεί*, συμπτύσσοντάς το σε έναν συνοπτικό κόμβο, όπως στο σχ. 4.5.

Με αυτή τη λειτουργία ο χρήστης στην πράξη θεωρεί (σημασιολογικά) ότι το συνόλο των κόμβων-οντοτήτων που επιλέγει προσδιορίζει μιαν έννοια που ως μια γενικότερη ‘‘οντότητα’’ περιλαμβάνει τους κόμβους αυτούς. Η σύμπτυξη των κόμβων σε έναν του επιτρέπει να έχει την εποπτεία των συσχετίσεων των ‘‘εξωτερικών’’ οντοτήτων με το

σύνολο των κόμβων, ως ‘‘οντότητα’’.

Δεν μπορούμε να πούμε με σιγουριά αν αυτή η λειτουργία, ως έχει, θα φανεί χρήσιμη και πόσο στην πράξη. Η ευχρηστία της όμως, και η διαισθητική της απλότητα, υποδεικνύει ότι πιθανόν να φανεί αρκετά χρήσιμη, ιδιαίτερα στους άπειρους χρήστες.

4.3.2.2 Τοπολογική Σύμπτυξη

Συχνά η επερώτηση του χρήστη σε σχέση με την οργάνωση και το περιεχόμενο της βάσης είναι τέτοια, ώστε η ίδια η τοπολογία του γράφου απάντησης περιέχει πολύτιμη ‘‘σημασιολογική’’ πληροφορία.

Ειδικότερα, μια χρήσιμη παρατήρηση είναι ότι το σύνολο των *κόμβων-φύλλων* (τοπολογική ιδιότητα) που συνδέονται με κάποιον συγκεκριμένο (ας τον πούμε *‘‘ριζικό’’*) κόμβο, συχνά μπορεί να θεωρηθεί ως ένα σύνολο *‘‘ιδιοτήτων’’* του κόμβου.

Επεκτείνοντας αυτή τη θεώρηση, παρατηρούμε ότι πολλές φορές μπορούμε να χαρακτηρίζουμε ολόκληρα *δέντρα*, που εμφανίζονται ως υπο-γράφοι γενικότερων γράφων απάντησης, ως ‘‘ιδιότητες’’ ενός κόμβου: Οι οντότητες-κόμβοι του δένδρου, συνδέονται με τον υπόλοιπο γράφο μόνο δια μέσου της οντότητας-ρίζας του. Συνεπώς, θα ήταν ιδιαίτερα πρακτική μια σύμπτυξη των δέντρων αυτών μέσα[3] στο ριζικό τους κόμβο, υπό την έννοια ότι απλοποιεί δραματικά το γράφο, χωρίς να μας στερεί από κρίσιμη σημασιολογική πληροφορία. Μια ειδικότερη περίπτωση που παρουσιάζει ενδιαφέρον, είναι η σύμπτυξη ενός ‘‘άχρηστου’’ *κλάδου*, αντί ολόκληρου του δέντρου. Μπορούμε να θέσουμε περιορισμούς βάθους κατά την διάσχιση του δέντρου, ώστε να συμπτύσσονται κόμβοι σε βάθος π.χ. ένα ή δύο, και όχι ολόκληρο το δέντρο.

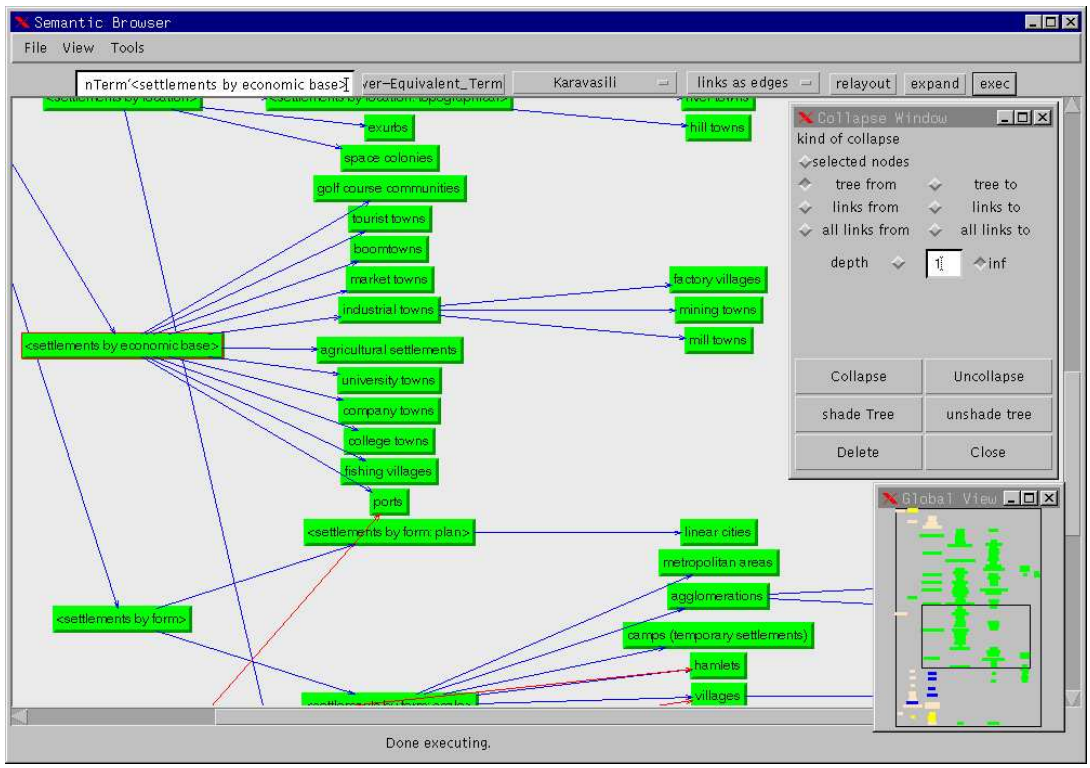
Τη μέθοδο αυτή μπορούμε να τη χρησιμοποιήσουμε και κατά την αντίθετη φορά των ακμών, συμπτύσσοντας *δέντρα γόνεων*, κόμβους δηλαδή που αναφέρονται *στον* ριζικό, αντί να αναφέρονται *από* αυτόν.

Στο σύστημά μας, ο νέος κόμβος που προκύπτει από τη σύμπτυξη, περιλαμβάνει άμεσα την πληροφορία των κόμβων που έχουν συμπτυχθεί: Η πλήρης ετικέτα του περιλαμβάνει τις ετικέτες όλων των συμπτυγμένων κόμβων, παραθεμένες με τρόπο ώστε να γίνεται οπτικά εμφανής η ιεραρχική δομή του δέντρου.

Ένα παράδειγμα αυτής της λειτουργίας φαίνεται στα σχ.4.6 και 4.7. Πρόκειται για ολόκληρο το γράφο που είχαμε δεί για πρώτη φορά συμπτυγμένο στο σχ.4.3, και περιλαμβάνει συγκεκριμένα όλους τους στενότερους όρους του όρου *settlements*, καθώς

[3]Παρακάτω, όταν γράφουμε ότι ένας κόμβος Χ (ή σύνολο κόμβων) συμπτύσσεται ‘‘μέσα’’ σε έναν άλλο κόμβο Κ, εννοούμε ότι ένας νέος κόμβος δημιουργείται στη θέση του Κ, ο οποίος περιλαμβάνει και τον Κ και τον Χ.

και τους ισοδύναμούς τους.

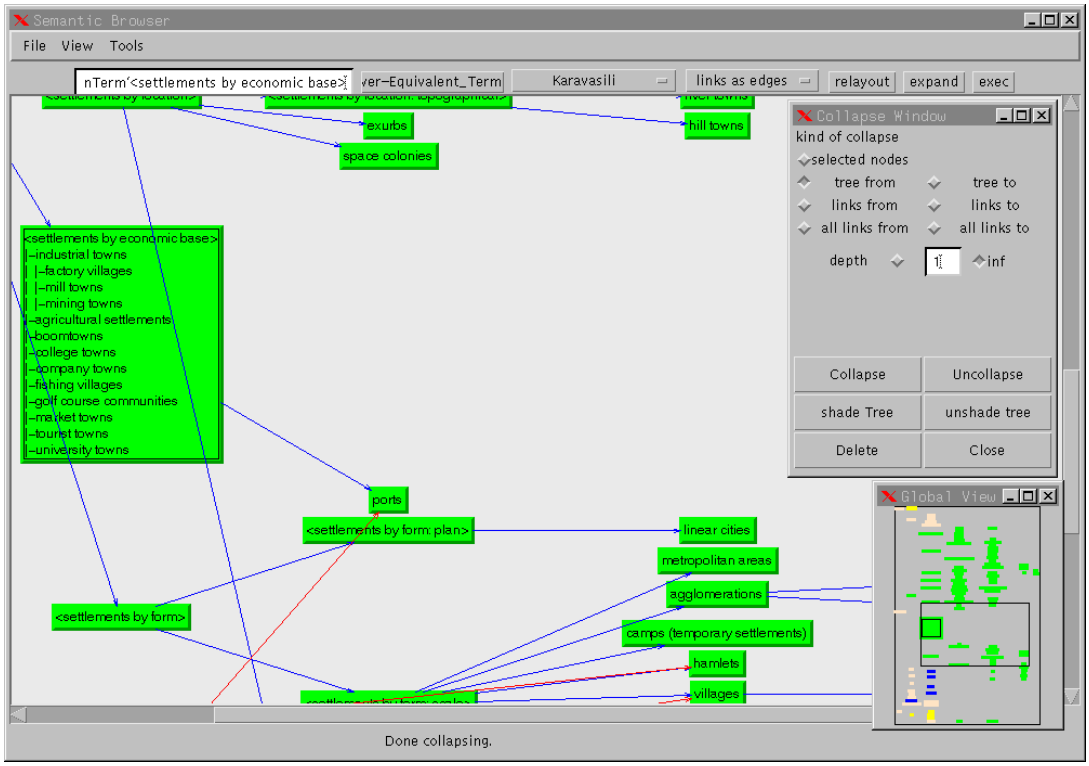


Σχήμα 4.6: Πριν την τοπολογική σύμπτυξη στον κόμβο <settlements by economic base>.

Σε πρώτη φάση, στο σχ. 4.6, έχουμε εντοπίσει τον όρο <*settlements by economic base*>, και σκοπεύουμε να συμπτύξουμε το υποδέντρο του γράφου που τον έχει ως ρίζα.

Μετά την εκτέλεση της λειτουργίας, στο σχ.4.7, παρατηρούμε ότι η οντότητα *ports*, η οποία έχει και δεύτερο γονέα (και άρα δεν ανήκει στο υποδέντρο), μένει έξω από τον συνοπτικό κόμβο. Είναι έτσι πολύ ευκολότερο για το χρήστη να εντοπίσει τους κόμβους που έχουν ‘‘εξωτερικές’’ συσχετίσεις. Με μια σειρά από τέτοιες συμπτύξεις, ο χρήστης καταλήγει σε έναν ελάχιστο γράφο, τον οποίο μάλιστα μπορεί να ανασχεδιάσει, όπως έχουμε κάνει στην εικόνα του σχήματος 4.3, Στο συγκεκριμένο παράδειγμα που παραθέτουμε, αυτό δίνει μια καλύτερη εποπτεία των συσχετίσεων που εκφράζουν τις *αντιστοιχίσεις* ανάμεσα σε όρους δύο διαφορετικών θησαυρών όρων. Κάτι τέτοιο δεν ήταν εύκολο να εντοπιστεί σε ολόκληρο το γράφο.

Η χρησιμότητα μιας λειτουργίας σαν την παραπάνω, έχει ήδη εντοπιστεί από χρήστες του GAIN, και αναμένεται να διευκολύνει σε μεγάλο βαθμό την κατανόηση ιδιαίτερα μεγάλων και περίπλοκων γράφων.



Σχήμα 4.7: Μετά την τοπολογική σύμπτυξη στον κόμβο <settlements by economic base>.

4.3.2.3 Σημασιολογική Σύμπτυξη

Όπως αναφέραμε πρωτύτερα, την πληροφορία για τις κατηγορίες των συνδέσμων μπορούμε να την αξιοποιήσουμε για *σύμπτυξη ανά κατηγορία:* Μπορούμε να συμπτύξουμε μια κατηγορία συνδέσμων που ξεκινούν από τον ίδιο κόμβο, ή όλους ανεξαιρέτως τους συνδέσμους αυτής της κατηγορίας.

Η *σύμπτυξη ενός συνδέσμου* μπορεί να γίνει με δύο τρόπους: Είτε ο κόμβος-προορισμός του συνδέσμου συμπτύσσεται μέσα στον κόμβο αφετηρία, ή το αντίστροφο.

Σύμπτυξη ανά κατηγορία σε έναν κόμβο

Ο χρήστης μπορεί να επιλέξει να συμπτυχθούν όλοι οι σύνδεσμοι μιας κατηγορίας *που ξεκινούν ή καταλήγουν σε έναν συγκεκριμένο κόμβο.*

Για να επιλέξει ο χρήστης την κατηγορία που τον ενδιαφέρει, επιλέγει έναν σύνδεσμο αυτής της κατηγορίας, προτού εκτελέσει τη λειτουργία. Οι σύνδεσμοι που δεν ανήκουν σε αυτή την κατηγορία δεν συμπύσσονται, και είναι έτσι ευκολότερο για το χρήστη να τους επιθεωρήσει.

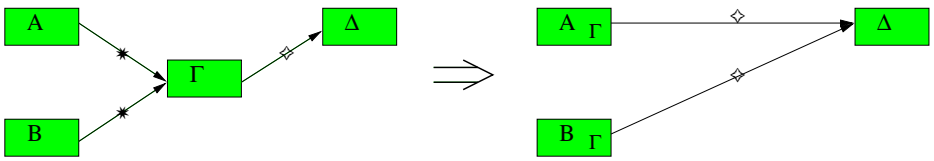
Γενίκευσή αυτής της λειτουργίας αποτελεί η παρακάτω.

Σύμπτυξη κατηγορίας σε όλο το γράφο

Η προηγούμενη λειτουργία, μπορεί να εφαρμοστεί εναλλακτικά σε ολόκληρο το γράφο, αντί για ένα συγκεκριμένο κόμβο. Ο χρήστης μπορεί να συμπτύξει όλους ανεξαιρέτως τους συνδέσμους μιας κατηγορίας.

Για παράδειγμα, σε έναν γράφο ενός θησαυρού όρων όπου παρουσιάζονται σχέσεις ‘‘γενικότερος όρος’’ και ‘‘συνώνυμο’’, ο χρήστης είναι πιθανόν να θέλει να επικεντρωθεί στην ιεραρχία των όρων, που αποτελείται μόνο από τις σχέσεις ‘‘γενικότερος όρος’’, συμπτύσσοντας κάθε σύνολο συνώνυμων όρων σε ένα.

Κατ’ αυτή τη διαδικασία, είναι πιθανό να προκύψουν διάφορες αμφισημίες. Για παράδειγμα, η περίπτωση του σχ.4.8, όπου παρουσιάζεται το πρόβλημα σε ποιόν κόμβο, τον Α ή τον Β, συμπτύσσεται ο κόμβος Γ. Σε αυτή την περίπτωση, η πληροφορία του Γ πρέπει να περιληφθεί και στους δύο κόμβους.



Σχήμα 4.8: Αμφισημία κατά τη σύμπτυξη και επίλυσή της

Στην εργασία μας, υποθέτουμε ότι η σύμπτυξη κόμβων χρησιμοποιείται κατά κύριο λόγο για την μείωση της πολυπλοκότητας του γράφου: ο χρήστης ενδιαφέρεται κυρίως για τον γράφο που απομένει έπειτα από μία σύμπτυξη. Ως εκ τούτου, δεν έχουμε ασχοληθεί εκτενώς με την επίλυση των αμφισημιών: στο σύστημά μας, ένας από τους πιθανούς συνοπτικούς κόμβους επιλέγεται (ο πρώτος τυχόν), και μέσα σε αυτόν γίνεται η σύμπτυξη.

Αν προκύψει ότι η αντιμετώπιση των αμφισημιών είναι απαραίτητη, μπορεί να υλοποιηθεί σε μια μελλοντική επέκταση του συστήματος.

Σύμπτυξη ανα μετακατηγορία

Στην Telos, οι κατηγορίες συνδέσμων μπορούν παραπέρα να οργανώνονται σε μετακατηγορίες. Η σύμπτυξη ολόκληρης μετακατηγορίας, με σύμπτυξη κάθε κατηγορίας που ανήκει σε αυτή, είναι πιθανόν μια χρήσιμη λειτουργία. Ιδιαίτερο ενδιαφέρον παρουσιάζει, όπως ήδη επισημάναμε (παρ. 4.3.1), η σύμπτυξη της ω-κατηγορίας dependent, λειτουργία που αντιστοιχεί στην ερμηνεία του υπογράφου, ο οποίος συμπτύσεται, ως μια οντότητα

που αντιστοιχεί σε ‘‘σύνθετο αντικείμενο’’.

Στην εργασία μας αυτή η λειτουργία δεν παρέχεται, υπάρχουν όμως όλοι οι απαραίτητοι μηχανισμοί ώστε να μπορεί να υλοποιηθεί πολύ εύκολα: για κάποια μετακατηγορία, βρίσκουμε τις κατηγορίες της και τις συμπτύσσουμε όπως παρουσιάζουμε στην προηγούμενη παράγραφο.

4.3.3 Στρώματα του γράφου

Ο χωρισμός του γράφου σε *στρώματα (layers)*, αποτελεί ένα σημασιολογικό μηχανισμό για τον εύκολο χειρισμό υπογράφων του.

Ένα επίπεδο περιλαμβάνει ένα υποσύνολο των κόμβων και τον ακμών του γράφου, ενώ ένας κόμβος ή ακμή μπορεί να ανήκει μόνο σε ένα επίπεδο (Σε μια γενικότερη υλοποίηση θα μπορούσαν να ανήκουν σε περισσότερα από ένα). Ακμές μπορούν να υπάρχουν και μεταξύ κόμβων που ανήκουν σε διαφορετικά επίπεδα. Ένας γράφος μπορεί να αποτελείται από ένα περισσότερα επίπεδα.

Η έννοια του επιπέδου είναι γνωστή κυρίως από σχεδιαστικά προγράμματα (π.χ. Photo Shop). Αποτελεί ένα διαισθητικά απλό και εύχρηστο μηχανισμό μείωσης της πολυπλοκότητας ενός σχεδίου, στην περίπτωσή μας ενός γράφου: ο χρήστης μπορεί να συγκεντρωθεί στη μελέτη ενός ή μερικών μόνο επιπέδων αντί ολόκληρου του γράφου, ή να μελετήσει τις συσχετίσεις που εμφανίζονται ανάμεσα σε δύο ή περισσότερα διαφορετικά επίπεδα.

Στην εργασία μας χρησιμοποιούμε τα επίπεδα κατά τη διαδικασία της επέκτασης ερώτησης (παρ. 4.1.1). Έχουμε υλοποιήσει *απόκρυψη* (hidding), *σκίαση* (shading) και *σβήσιμο* με βάση τα επίπεδα, κατά την βούληση του χρήστη. Περισσότερες πληροφορίες για τη χρήση τους θα δούμε στην παράγραφο 4.4.1.3, που ασχολήται με την οπτικοποίηση του γράφου.

Σημειώνουμε τελος, ότι ο μηχανισμός αυτός μπορεί να χρησιμοποιηθεί σε μελλοντική επέκταση του συστήματος ώστε να επιτρέπει εισαγωγή πληροφορίας στη βάση με επέμβαση του χρήστη στο γράφο: Όλη η προς εισαγωγή πληροφορία μπορεί να παρέχεται σε ένα νέο επίπεδο.

4.4 Λειτουργίες οπτικοποίησης και διεπαφής χρήσης

Για να είναι ένα σύστημα διερεύνησης χρησιμοποιήσιμο στην πράξη, δεν αρκεί να παρέχει ορισμένες βασικές λειτουργίες. Είναι απαραίτητο να προσφέρει και διευκολύνσεις

στο χρήστη ώστε να κάνει πιο άνετα και αποδοτικά τη δουλειά του. Οι δυνατότητες του συστήματος θα πρέπει να προσφέρονται μέσα από μια καλά μελετημένη διεπιφάνεια χρήσης, που θα είναι οργανωμένη με σκοπό την ευχρηστία του.

Παρόλο που δεν είχαμε ως στόχο την παραγωγή μιας πλήρους εφαρμογής (με βάση τις απαιτήσεις από εμπορικές εφαρμογές), στο σύστημά μας παρέχονται πολλές από τις απαραίτητες συμπληρωματικές λειτουργίες που σχετίζονται με την οπτικοποίηση του γράφου απάντησης και τη διεπιφάνεια χρήσης. Ο χρήστης μπορεί να ενεργοποιήσει αυτές τις λειτουργίες μέσα από κατάλληλους πίνακες ελέγχου.

Ο χρήστης μπορεί να δει το γράφο σε επιλεγμένη κλίμακα μεγέθους, καθώς και να έχει μια συνολική του άποψη. Μπορεί να σκιάσει ή να κρύψει τμήματα του γράφου με βάση τα επίπεδά του, όπως επίσης και με βάση σημασιολογικά ή τοπολογικά κριτήρια. Υποστηρίζονται ακόμα πολλαπλά παράθυρα γράφων. Τέλος, παρέχεται δυνατότητα για αποθήκευση και φόρτωμα γράφων από το δίσκο.

Παρακάτω παρουσιάζουμε αναλυτικότερα αυτές τις λειτουργίες.

4.4.1 Οπτικοποίηση του γράφου

4.4.1.1 Οπτική κλιμάκωση του γράφου

Οπτική κλιμάκωση (scaling) του γράφου μπορεί να γίνει στο σύστημά μας με τρεις τρόπους: κλιμάκωση του μήκους των ακμών, κλιμάκωση του μεγέθους των κόμβων (με αντίστοιχη κλιμάκωση της γραμματοσειράς που χρησιμοποιείται για την ετικέτα τους), και τέλος, με συνδυασμό αυτών των δύο, συνολική κλιμάκωση του μεγέθους του γράφου.

Η κλιμάκωση των ακμών μπορεί να είναι ένας εναλλακτικός χρήσιμος τρόπος για περιορισμό των επικαλύψεων μεταξύ κόμβων σε ορισμένες ειδικές περιπτώσεις. Για παράδειγμα, είναι πολύ χρήσιμη όταν κάποιος αλγόριθμος σχεδίασης δεν λαμβάνει υπ' όψιν το μέγεθος των κόμβων, και συνεπώς δεν υπολογίζει κατάλληλες αποστάσεις ανάμεσά τους κατά την τοποθέτηση, με αποτέλεσμα να προκύπτουν επικαλύψεις.

4.4.1.2 Ολική Όψη του γράφου

Η *Ολική Όψη* (Global View ή Birdeye View) είναι ένας γνωστός μηχανισμός που χρησιμοποιείται σε διάφορα συστήματα για συνολική εποπτεία ενός μεγάλου γράφου.

Το παράθυρο Ολικής Όψης που παρέχουμε στο χρήστη είναι κλιμακούμενου μεγέθους, και περιέχει μόνο τους κόμβους του γράφου (χωρίς τις ετικέτες τους), και όχι τις ακμές του. Ένα *''παράθυρο όψης''* που τοποθετείται με το ποντίκι στο σημείο που θέλει ο χρήστης,

του επιτρέπει να κάνει ορατό στο κύριο παράθυρο το ζητούμενο τμήμα του γράφου.

4.4.1.3 Απόκρυψη και Σκίαση τμημάτων του γράφου

Όπως έχουμε ήδη αναφέρει, η απόκρυψη τμήματος του γράφου μπορεί να βοηθήσει στην καλύτερη κατανόηση της δομής του. Παρόμοιο αποτέλεσμα μπορούμε να επιτύχουμε με *σκίαση* τμήματος του γράφου, που επιτρέπει στο χρήστη να διαμερίσει οπτικά το γράφο σε υπο-γράφους και να συγκεντρωθεί στη μελέτη καθενός ξεχωριστά, ή στη μελέτη των συσχετίσεων ανάμεσά τους.

Απόκρυψη και σκίαση τμημάτων του γράφου μπορεί να γίνει με βάση τα στρώματα του, την τοπολογία του , ή σημασιολογικά κριτήρια.

Η απόκρυψη και η σκίαση με βάση τα στρώματα, όταν αυτά χρησιμοποιούνται για να φιλοξενήσουν τις επεκτάσεις ενός γράφου, φαίνεται ότι είναι μια πολύ χρήσιμη λειτουργία. Οι επεκτάσεις που έχουν γίνει σε κάθε βήμα, μπορούν κατά βούληση να σκιάζονται, να αποκρύπτονται, και να επανεμφανίζονται, ώστε οι διαδοχικές αλλαγές να γίνονται εύκολα αντιληπτές.

Τα τοπολογικά κριτήρια που χρησιμοποιούνται για σύμπτυξη πληροφορίας σε έναν κόμβο (παρ. 4.3.2) μπορούν να χρησιμοποιηθούν εναλλακτικά για απόκρυψη και σκίαση της πληροφορίας αυτής. Το ίδιο μπορεί να γίνει και με τα σημασιολογικά κριτήρια.

Οι παραπάνω λειτουργίες πρέπει να αξιολογηθούν στην πράξη, και να μελετηθεί περισσότερο η διεπαφή χρήσης που τις κάνει διαθέσιμες στο χρήστη, ώστε να μπορέσουμε να εγγυηθούμε την ευχρηστία του συστήματος.

4.4.2 Διεπαφή χρήσης

4.4.2.1 Πολλαπλά παράθυρα γράφων

Ο χρήστης μπορεί να έχει πολλά παράθυρα γράφων παράλληλα στην οθόνη του. Σε κάθε παράθυρο είναι δυνατόν να εμφανίζεται οποιοσδήποτε γράφος. Έτσι για παράδειγμα ο ίδιος γράφος απάντησης μπορεί να εμφανίζεται σε διαφορετικά παράθυρα, με διαφορετική χωροδιάταξη ή και διαφορετικό τρόπο παράστασης των συσχετίσεων (με κόμβους ή ακμές). Άλλη ενδιαφέρουσα περίπτωση είναι διαφορετικές επερωτήσεις επάνω στον ίδιο στόχο μπορούν να εμφανίζονται ταυτόχρονα σε διαφορετικά παράθυρα: ένα παράθυρο λ.χ. να περιέχει την ιεραρχία κλάσεων στην οποία ανήκει μια κλάση, και ένα άλλο τα γνωρίσματά της.

Κάθε παράθυρο μπορεί να έχει δική του Ολική Όψη και παλέτες ελέγχου, καθώς και

δικά του χαρακτηριστικά εμφάνισης (γραμματοσειρά, κλιμάκωση γράφου κλπ).

Ένα σημαντικό πλεονέκτημα είναι ότι ο χρήστης μπορεί να εργάζεται σε ένα παράθυρο όταν κάποιο άλλο περιμένει αποτελέσματα, πράγμα που βελτιώνει την ευχρηστία του συστήματος.

Μπορούμε να φανταστούμε και να προτείνουμε σαν επέκταση σενάρια κατά τα οποία ο χρήστης θα μπορεί να μεταφέρει πληροφορία, π.χ. έναν ολόκληρο υπο-γράφο, από το ένα παράθυρο στο άλλο. Πολύ χρήσιμη μπορεί να είναι η οπτική αντιστοίχιση πληροφορίας που είναι κοινή ή σχετική ανάμεσα σε δύο παράθυρα, με διατονισμό (highlighting). Για παράδειγμα, θα μπορούσαμε να έχουμε ένα παράθυρο με την ιεραρχία των κυριότερων κλάσεων μιας βάσης, και για όποιο αντικείμενο επιλέγει ο χρήστης σε κάποιο άλλο παράθυρο, οι κλάσεις στις οποίες ανήκει να διατονίζονται στο πρώτο παράθυρο. Άλλη ιδέα που έχει χρησιμοποιηθεί σε διάφορα συστήματα, είναι το ένα παράθυρο να παρουσιάζει περιληπτικά όλη την πληροφορία ενός ομαδοποιημένου γράφου, και με επιλογή τμημάτων του οι σχετικές λεπτομέρειες να εμφανίζονται σε κάποιο άλλο παράθυρο, όπως π.χ. στο [34]. Οι δυνατότητες που ανοίγονται προς αυτή την κατεύθυνση είναι πολλές και αξιόλογες.

4.4.2.2 Αποθήκευση και Φόρτωμα του γράφου

Αφού ο χρήστης καταλήξει σε κάποιον γράφο που τον ενδιαφέρει, μπορεί να τον αποθηκεύσει για κατοπινή χρήση ή επεξεργασία από άλλα προγράμματα. Η αποθήκευση γίνεται στη γλώσσα GML (Graph Markup Language) [26], μια γλώσσα που είναι πιο κοντά από όλες τις σχετικές στο να γίνει de facto πρότυπο για παράσταση γράφων.

Συστήματα που υποστηρίζουν αυτή τη γλώσσα μπορούν να χρησιμοποιηθούν off-line από το χρήστη, μετά την αποθήκευση του γράφου. Για παράδειγμα, μπορεί να φορτωθεί ο γράφος από το Graphlet [25], και να μετατραπεί σε postscript μορφή για εκτύπωση, ή να χρησιμοποιηθούν οι αλγόριθμοι του Graph Drawing Server [42], για ένα νέο είδος σχεδίασης.

4.5 Σύνοψη λειτουργιών

Συνοπτικά, οι λειτουργίες που παρέχονται στο σύστημά μας, ομαδοποιημένες ανά επίπεδο λειτουργικότητας, είναι οι εξής:

1. Επερωτήσεις:
    - εκτέλεση επερώτησης

- επέκταση επερώτησης

2. Χωροδιάταξη του γράφου:
   - αυτόματη χωροδιάταξη του γράφου
   - παράσταση των συσχετίσεων ως ακμές ή κόμβους, ή ολόκληρων κατηγοριών συσχετίσεων ως κόμβους
   - ανασχεδίαση του γράφου
   - εξάλειψη επικαλύψεων
3. Χρήση σημασιολογικής και τοπολογικής πληροφορίας:
   - σύμπτυξη πληροφορίας με βάση τοπολογικά ή σημασιολογικά κριτήρια
   - χωρισμός του γράφου σε στρώματα
4. Λειτουργίες οπτικοποίησης και διεπαφής χρήσης:
   - Οπτικοποίηση του γράφου
     -- οπτική κλιμάκωση του γράφου
     -- Ολική Όψη του γράφου
     -- απόκρυψη και σκίαση τμημάτων του γράφου
   - Διεπαφή χρήσης
     -- πολλαπλά παράθυρα γράφων
     -- αποθήκευση και φόρτωμα του γράφου

Κεφάλαιο 5

Αρχιτεκτονική του συστήματος

Στο κεφάλαιο αυτό παρουσιάζουμε την αρχιτεκτονική του Συστήματος Σημασιολογικής Διερεύνησης. Ορισμένες από τις επιλογές της αρχιτεκτονικής καθορίστηκαν από την απαίτηση για απομακρυσμένη πρόσβαση των χρηστών στο σύστημα, ενώ άλλες προέκυψαν ως προσφορότερες με τη χρήση των δυνατοτήτων της γλώσσας υλοποίησης, της Java. Παρουσιάζουμε ως εκ τούτου και τα κυριότερα σημεία που έχουν να κάνουν με την υλοποίηση του συστήματος.

5.1 Καθοριστικές απαιτήσεις

Μια βασική απαίτηση για το σύστημά μας ήταν να είναι δυνατή η απομακρυσμένη χρήση του συστήματος μέσω του Παγκόσμιου Ιστού (WWW). Θα θέλαμε επίσης να μπορεί να χρησιμοποιηθεί χωρίς μετατροπές και για τοπική χρήση. Για το λόγω αυτό ακολουθήσαμε την αρχιτεκτονική client-server.

Ο server είναι μια διεργασία που έχει πρόσβαση στη βάση του ΣΣΕ. Ονομάζεται, για ευνόητους λόγους, *Εργοστάσιο Γράφων (Ε.Γ., Graph Factory).* Υποθέτουμε ότι τρέχει σε ένα ισχυρό μηχάνημα, ενώ ο client μπορεί να τρέχει σε οποιοδήποτε μηχάνημα στο δίκτυο. Για το λόγο αυτό (και άλλους που θα δούμε παρακάτω), η υπολογιστικά βαρειά λειτουργία της χωροδιάταξης των γράφων εκτελείται στο Ε.Γ.

Ο client ονομάζεται *Σημασιολογικός Διερευνητής (Σ.Δ., Semantic Browser)* και αποτελείται κυρίως από το τμήμα του συστήματος που έχει να κάνει με τη διεπαφή χρήσης. Επιπλέον, ορισμένες λειτουργίες επάνω στο γράφο απάντησης γίνονται κατ' ευθείαν στον Σ.Δ., ενώ για άλλες απαιτείται επικοινωνία με το Ε.Γ.

Ως γλώσσα υλοποίησης επιλέξαμε όπως προαναφέραμε τη Java. Η γλώσσα αυτή

είναι σχεδιασμένη ώστε να παρέχει πολλές ευκολίες για επικοινωνία προγραμμάτων μέσω δικτύου, και επιπλέον είναι ανεξάρτητη πλατφόρμας.

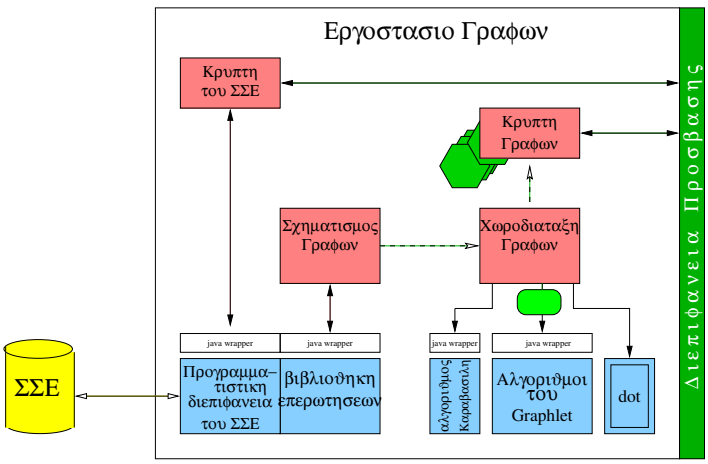
5.2 Αρχιτεκτονική

Ο γενικός τρόπος λειτουργίας του συστήματος, είναι ως εξής: Οι εντολές που δίνει ο χρήστης μέσω της διεπιφάνειας χρήσης, ενεργοποιούν μια αντίστοιχη διαδικασία του Σημασιολογικού Διερευνητή. Για την περάτωσή της, ο Σ.Δ. καλεί τις κατάλληλες μεθόδους του Ε.Γ. Αυτό με τη σειρά του, εκτελεί όποιες λειτουργίες είναι απαραίτητες, και επιστρέφει τα αποτελέσματα στον Σ.Δ., προς επίδειξη στο χρήστη.

Παρακάτω, εξετάζουμε αναλυτικότερα το κάθε μέρος του συστήματος.

5.2.1 Το Εργοστάσιο Γράφων

Το Ε.Γ. του συστήματος είναι υπεύθυνο για τη δημιουργία και τη σχεδίαση γράφων απαντήσεων, και επιπλέον παρέχει στον Σ.Δ. μια *κρύπτη* (cache) με ιδιότητες των διαφόρων αντικειμένων της βάσης. Συνδέεται με έναν εξυπηρέτη του ΣΣΕ μέσω της προγραμματιστικής διεπιφάνειας *pqi* [15].



Σχήμα 5.1: Αρχιτεκτονική του Ε.Γ.

Το γενικό σχεδιάγραμμα της αρχιτεκτονικής του φαίνεται στο σχ. 5.1.

Ορισμένα από τα τμήματα του είναι γραμμένα σε Java, και άλλα σε C/C++. Για τα τμήματα που δεν είναι σε κώδικα της Java, παρέχονται Java wrappers, ώστε να είναι δυνατή η χρήση τους.

1. **Βιβλιοθήκη επερωτήσεων :**

Η βιβλιοθήκη αυτή περιέχει τις κλάσεις σε C++ QueryMacro και GetFunctions, που έχουν αποσπαστεί από τον κώδικα του GAIN, και χρησιμοποιεί η ίδια την προγραμματιστική διεπιφάνεια του ΣΣΕ. Η κλάση QueryMacro είναι υπεύθυνη για την ανάκληση (από την βάση όπου είναι αποθηκευμένες) των προκαθορισμένων επερωτήσεων, και την εκτέλεσή τους. Η κλάση GetFunctions προσφέρει έναν ομοιόμορφο τρόπο πρόσβασης στα αποτελέσματα των επερωτήσεων, ώστε να είναι εύκολη η μετατροπή τους σε μορφή γράφου.

Για τη χρήση αυτής της βιβλιοθήκης μέσα από κώδικα σε Java έχει δημιουργηθεί ένας Java wrapper. Οι λεπτομέρειες της κλάσης QueryMacro, όπως αρχικοποίηση, πέρασμα της οντότητας-στόχου ως παράμετρο, κλπ, κρύβονται εντελώς από το wrapper. Με αυτό τον τρόπο, τελικά ο προγραμματιστής έχει να κάνει μόνο με την κλάση GetFunctions, στην οποία παρέχεται πρόσβαση μέσω μιας απλής προγραμματιστικής διεπιφάνειας σε Java.

2. **Κρύπτη του ΣΣΕ :**
   Για την εκτέλεση ορισμένων λειτουργιών του ΣΣΔ, είναι απαραίτητη σημασιολογική πληροφορία που υπάρχει στην βάση του ΣΣΕ, όπως είδαμε στην παρ. 4.3.

   Στην περίπτωση που το σύστημα χρησιμοποιείται μέσω WWW, οι περιορισμοί ασφάλειας που επιβάλλονται από τις σχετικές συμβάσεις, επιτρέπουν στον απομακρυσμένο client να έχει σύνδεση με ένα και μόνο μηχάνημα, αυτό από το οποίο έχει μεταφερθεί ο κώδικάς του, στο οποίο και πρέπει να τρέχει το Ε.Γ. Ως συνέπεια, ο client δεν μπορεί να συνδεθεί άμεσα με τον εξυπηρέτη του ΣΣΕ, που τρέχει εν γένει σε άλλο μηχάνημα, για να πάρει τις απαιτούμενες πληροφορίες.

   Η κρύπτη του ΣΣΕ εξυπηρετεί διπλό σκοπό: Αφ' ενός αποτελεί μια ενδιάμεση διεπιφάνεια πρόσβασης (interface) στον εξυπηρέτη του ΣΣΕ, την οποία παρέχει το Ε.Γ. στον Σ.Δ., αφ' ετέρου χρησιμεύει ως κρύπτη ιδιοτήτων των αντικειμένων. Η προεπιλεγμένη χωρητικότητά της είναι 200 αντικείμενα, και μπορεί να αλλάξει με επιλογή του προγραμματιστή: Αν κριθεί απαραίτητο, με μια απλή επέκταση του συστήματος μπορεί να προσδιορίζεται από το χρήστη κατά την εκκίνηση του Ε.Γ. Εαν μηδενίσουμε τη χωρητικότητά της κρύπτης, αυτή λειτουργεί ως μια απλή διεπιφάνεια πρόσβασης στον εξυπηρέτη του ΣΣΕ, και παρέχει τις προγραμματισμένες λειτουργίες της.

   Σε περίπτωση που η επικοινωνία μεταξύ client-server δεν γίνεται μέσω του WWW και δεν έχουμε τους περιορισμούς που προαναφέραμε, ο Σ.Δ. μπορεί να διαθέτει δική του κρύπτη, η οποία συνδέεται άμεσα με τον εξυπηρέτη του ΣΣΕ, διαφορετικά χρησιμοποιεί την κρύπτη του Ε.Γ.

3. **Σχηματισμός Γράφων:**
   Αποτελεί το κομμάτι του Ε.Γ. που είναι υπεύθυνο για το σχηματισμό γράφων από τα αποτελέσματα επερωτήσεων. Χρησιμοποιεί την *βιβλιοθήκη επερωτήσεων*. Το τμήμα αυτό φροντίζει για όλα τα χαρακτηριστικά που θα έχει ο γράφος που θα παραχθεί. Φροντίζει για το χρώμα και το σχήμα των κόμβων, καθώς και για την παράσταση των συνδέσμων με κόμβους ή ακμές, ανάλογα με την απαίτηση του χρήστη. Σε αυτό το τμήμα επίσης γίνεται η οποιαδήποτε προεπεξεργασία του γράφου: Στη συγκεκριμένη υλοποίηση, παρέχεται μόνο η προεπεξεργασία που αντικαθιστά ένα σύνολο ακμών μιας κατηγορίας με ένα κόμβο (βλ. παρ. 4.2.1).

   Ο γράφος που προκύπτει από το τμήμα αυτό χρειάζεται πια μόνο χωροδιάταξη, για να μπορεί να παρουσιαστεί στο χρήστη.

4. **Χωροδιάταξη :**
   Το τμήμα αυτό του Ε.Γ. παρέχει τη σύζευξη με τα διάφορα πακέτα και βιβλιοθήκες αλγορίθμων σχεδίασης. Είναι υπεύθυνο για τη μετατροπή του γράφου από τη μορφή παράστασης που χρησιμοποιείται εσωτερικά στο Ε.Γ. στη μορφή που απαιτείται από κάθε βιβλιοθήκη αλγορίθμων, και αντίστροφα. Η διαδικασία αυτή δεν είναι απαραίτητο να περιλαμβάνει πλήρη μετατροπή του γράφου: Ορισμένες ιδιότητες, όπως π.χ. το χρώμα των κόμβων, μπορεί να μην είναι σημαντικές κατά τη χωροδιάταξη. Αντίστοιχα, μετά τη χωροδιάταξη από έναν αλγόριθμο, μόνο η πληροφορία για την τοποθεσία κάθε κόμβου (και τα λυγίσματα των ακμών) είναι απαραίτητη. Αν είμαστε σε θέση να αντιστοιχίσουμε τους κόμβους και τις ακμές ανάμεσα στις δύο παραστάσεις, όπως συμβαίνει σχεδόν για όλα τα σχετικά πακέτα, δεν είναι καν ανάγκη να κάνουμε πλήρη μετατροπή ‘‘προς τα πίσω’’ του γράφου: αρκεί να αποσπάσουμε την πληροφορία που μας ενδιαφέρει.

   Στη συγκεκριμένη υλοποίηση παρέχεται πρόσβαση στα εξής πακέτα και βιβλιοθήκες αλγορίθμων, όπως προαναφέραμε στην παρ.4.2: Στον αλγόριθμο σχεδίασης της Καραβασίλη, στο πακέτο αλγορίθμων Graphlet (για τους αλγόριθμους Kamada, Frick και Tunkelang), και στο πρόγραμμα ιεραρχικής σχεδίασης γράφων dot.

   - Σύνδεση με τον αλγόριθμο της Καραβασίλη:
     Η σύνδεση με τον αλγόριθμο της Καραβασίλη γίνεται προγραμματιστικά. Αρχικά ενημερώνεται η δομή δεδομένων του αλγορίθμου με το γράφο, κατόπιν καλείται ο αλγόριθμος, και τέλικα τα αποτελέσματα παραλαμβάνονται με κατάλληλες ρουτίνες. Όλες οι παραπάνω διαδικασίες καλούνται μέσω ενός Java wrapper.
   - Σύνδεση με τη βιβλιοθήκη αλγορίθμων του Graphlet:

Για τη βιβλιοθήκη του Graphlet, ‘‘χαλαρή’’ μόνο σύνδεση παρέχεται. Ο γράφος μετατρέπεται σε ενδιάμεση μορφή κειμένου με χρήση της γλώσσας GML. Κατόπιν μεταφράζεται από ρουτίνες του Graphlet στην εσωτερική του μορφή, γίνεται η χωροδιάταξη, και το αποτέλεσμα επιστρέφεται αφού μετατραπεί ξανά σε GML. Τελικά, γίνεται συντακτική ανάλυση (parsing) του αποτελέσματος για να εξαχθεί η πληροφορία τοποθέτησης των κόμβων. Η δημιουργία και η μετάφραση των κειμένων GML γίνονται σε γραμμικό χρόνο. Παρ’ όλα αυτά, σε μια μελλοντική έκδοση του συστήματος θα ήταν σαφώς γρηγορότερο να εξαλειφόταν αυτό το βήμα, και να γινόταν απευθείας ενημέρωση των δομών του Graphlet.

- Σύνδεση με το dot:
  Το dot είναι ένα πρόγραμμα που υλοποιεί έναν ιεραρχικό αλγόριθμο σχεδίασης γράφων. Η επικοινωνία με το dot γίνεται μέσω περιγραφών κειμένου.

  Το dot διαθέτει μια δική του γλώσσα για την περιγραφή των γράφων. Στην απλούστερή της περίπτωση, που επαρκεί για τις ανάγκες μας, αρκεί η παράθεση των ακμών του γράφου στη μορφή ‘‘κόμβος πηγή -> κόμβος προορισμός ;’’ ανά γραμμή εισόδου. Η γλώσσα αυτή χρησιμοποιείται μόνο για είσοδο στο dot των γράφων. Για την έξοδο των αποτελεσμάτων του, παρέχει μια εξίσου απλή παράσταση κειμένου με πληροφορία για έναν κόμβο (ή ακμή) ανά γραμμή εξόδου, την οποία και χρησιμοποιούμε.

5. **Κρύπτη Γράφων:**
   Συχνά οι χρήστες επιστρέφουν για να θυμηθούν κάποιον γράφο που έβλεπαν προηγουμένως. Για το λόγο αυτό, ένας αριθμός γράφων, με το σχέδιό τους, κρατιέται σε μια κλάση κρύπτης. Η χρήση της κρύπτης γίνεται αυτόματα από το σύστημα, όταν η ερώτηση του χρήστη ανακαλυφθεί ότι έχει ήδη απαντηθεί και τα αποτελέσματα είναι αποθηκευμένα. Η προεπιλεγμένη χωρητικότητα της κρύπτης είναι 12 γράφοι και μπορεί να αλλάξει με τον τρόπο που περιγράφουμε για την κρύπτη του ΣΣΕ παραπάνω. Τέλος, μπορεί εύκολα, σε μια μελλοντική επέκταση, να υλοποιηθεί μια λίστα ιστορικού, από την οποία ο χρήστης θα επιλέγει προηγούμενες ερωτήσεις και η οποία θα χρησιμοποιεί επίσης την κρύπτη.

6. **Διεπιφάνεια πρόσβασης στο Ε.Γ.:**
   Για την πρόσβαση ενός Σ.Δ. στο Ε.Γ. δεν επινοήθηκε κάποιο πρωτόκολλο επικοινωνίας: γίνεται με τη μέθοδο της Remote Method Invocation (RMI), που παρέχει η Java. Για όλες τις λειτουργίες που χρειάζεται ο Σ.Δ. από το Ε.Γ., παρέχεται μια διεπιφάνεια πρόσβασης (Java Interface). Επιπλέον, όλα τα αντικείμενα που χρειάζεται να μεταφερθούν από το Ε.Γ. στον Σ.Δ., όπως είναι ο γράφος συμπεριλαμβανομένων

των κόμβων και των ακμών του, έχουν σχεδιαστεί ώστε να είναι κατάλληλα για αυτόματη μεταφορά, χρησιμοποιώντας το data marshaling/unmarshaling που παρέχεται από το περιβάλλον της γλώσσας. Με αυτό τον τρόπο, η επικοινωνία απλοποιείται σε μεγάλο βαθμό, και αλλαγές στον κώδικα μπορούν να γίνουν αρκετά εύκολα.

Η διεπιφάνεια πρόσβασης του Ε.Γ., επιτρέπει στον Σ.Δ.:

- να έχει πρόσβαση στην κρύπτη του ΣΣΕ του Ε.Γ.
- να εκτελέσει μια επερώτηση και να παραλάβει το γράφο-αποτέλεσμα, σχεδιασμένο κατά έναν αλγόριθμο της επιλογής του χρήστη
- να ζητήσει ανασχεδίαση ενός γράφου
- να ζητήσει τη σχεδίαση ενός οποιουδήποτε γράφου, τον οποίο παρέχει ο Σ.Δ. στο Ε.Γ.

5.2.2 Ο Σημασιολογικός Διερευνητής

Ο Σ.Δ. παρέχει τη διεπιφάνεια χρήσης, η οποία παρουσιάζει στο χρήστη μια ενιαία εικόνα του συστήματος. Ο χρήστης δεν απασχολείται με το πού τρέχει κάθε λειτουργία που επιλέγει, αν τρέχει δηλαδή τοπικά στο σύστημά του ή απομακρυσμένα στο Ε.Γ.

Όπως είδαμε στην αρχιτεκτονική του Ε.Γ., επιλέξαμε η δημιουργία της δομής του γράφου να γίνεται στο Ε.Γ, και κατόπιν αυτός να μεταφέρεται έτοιμος στον Σ.Δ. Αυτό αφ' ενός γιατί το Ε.Γ. έχει άμεση σύνδεση με τον εξυπηρέτη του ΣΣΕ, απ' όπου παίρνει τις απαραίτητες πληροφορίες, κυριότερα όμως επειδή μόνο αυτό μπορεί να διαθέτει τους αλγόριθμους για αυτόματη σχεδίαση: Οι αλγόριθμοι αυτοί είναι συνήθως γραμμένοι σε μια συμβατική (μή ανεξάρτητη πλατφόρμας) γλώσσα, και οι βιβλιοθήκες με τον κώδικά τους δεν μπορούν να μεταφερθούν μέσω δικτύου για εκτέλεση. Συνεπώς, ο Σ.Δ. όταν τρέχει σε ένα απομακρυσμένο μηχάνημα δεν μπορεί να έχει πρόσβαση σε αυτές τις βιβλιοθήκες[1].

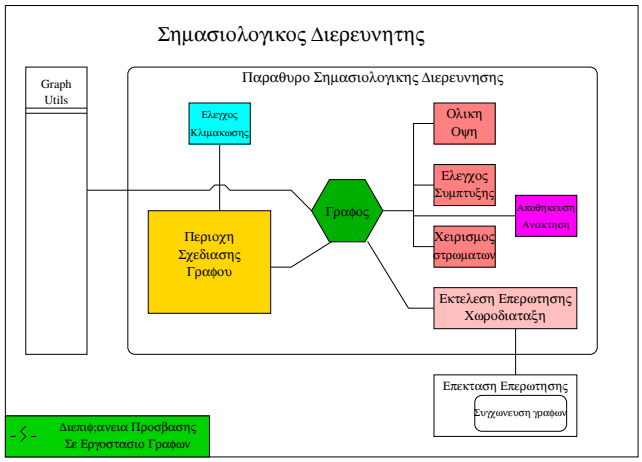
Ο Σ.Δ. θα μπορούσε να έχει πρόσβαση στη δομή γράφου του Ε.Γ., και να μην διατηρεί δικιά του δομή. Αυτή η σκέψη όμως αποδείχτηκε μή πρακτική, καθότι η ταχύτητα του δικτύου καθιστά κάτι τέτοιο αδύνατο: Για παράδειγμα, πρόσβαση σε κάθε κόμβο και ακμή είναι απαραίτητη κάθε φορά που γίνεται μια απλή ανασχεδίαση του γράφου, π.χ. για κύλιση της οθόνης. Για αυτούς τους λόγους, ένα αντίγραφο του γράφου μεταφέρεται στον Σ.Δ., και πάνω του γίνονται οι κατοπινές λειτουργίες.

Στην πράξη, αποδείχτηκε ότι και αυτή η επιλογή είναι αρκετά αργή: για την αντιγραφή του γράφου μέσω δικτύου, απαιτείται κωδικοποίηση και αποκωδικοποίηση όλων των

[1] Εξαίρεση σε αυτό θα ήταν αν είχαμε αλγόριθμους σχεδίασης γραμμένους απ' ευθείας σε Java. Αυτό είναι μεν εφικτό, αλλά η Java είναι ακόμα πολύ αργή σε σύγκριση με τις άλλες γλώσσες, κατά περίπου 20 φορές.

αντικειμένων (ακμών και κόμβων) του γράφου (marshaling και unmarshaling), πράγμα το οποίο είναι αρκετά χρονοβόρο (τουλάχιστον όμως σε αυτή την περίπτωση γίνεται μόνο μια φορά). Το πλεονέκτημα είναι ότι δεν χρειάζεται ειδικό πρωτόκολλο επικοινωνίας μεταξύ Σ.Δ. και Ε.Γ. για τη μεταφορά του γράφου. Το μειονέκτημα είναι η απόδοση. Για καλύτερη όμως απόδοση, ίσως θα πρέπει μελλοντικά να φτιαχτεί ένα τέτοιο πρωτόκολλο, το οποίο θα εξαλείφει την ανάγκη για πλήρη μετατροπή και μεταφορά της πληροφορίας όλων των δομών του γράφου μέσω του δικτύου.

Η αρχιτεκτονική του Σ.Δ. φαίνεται στο σχ. 5.2.



Σχήμα 5.2: Αρχιτεκτονική του Σ.Δ.

Αποτελείται από τα εξής λειτουργικά μέρη:

1. **Περιοχή Σχεδίασης του Γράφου:**
   Μια κλάση για τη σχεδίαση και παρουσίαση ενός γράφου, και το χειρισμό του. Περιλαμβάνει παραμέτρους για την κλιμάκωση του γράφου, το μέγεθος και το είδος της γραμματοσειράς για τις ετικέτες, το χρώμα του φόντου σχεδίασης και ό,τι άλλο είναι απαραίτητο.

   Παρέχει τη λειτουργικότητα σε επίπεδο εντολών του ποντικιού, π.χ. για την επιλογή ενός κόμβου ή ακμής και τη μετακίνηση κόμβων, καθώς και την κύλιση της περιοχής σχεδίασης, όταν αυτή είναι πολύ μεγάλη για να χωρέσει σε μια οθόνη.

2. **Παράθυρο Σημασιολογικής Διερεύνησης:**
   Αποτελεί την κλάση που παρέχει τη διεπιφάνεια χρήσης για την εξερεύνηση του γράφου. Μέσω αυτής παρέχεται πρόσβαση σε όλες τις λειτουργίες του συστήματος.

   Κάθε ΠΣΔ μπορεί να παρουσιάζει *ένα* γράφο κάθε χρονική στιγμή. Για το χειρισμό και τη σχεδίαση του γράφου περιλαμβάνει μία Περιοχή Σχεδίασης Γράφου. Επίσης παρέχει πίνακα ελέγχου για τις διάφορες παραμέτρους του συστήματος, για εκτέλεση

επερώτησης και χωροδιάταξη του γράφου, για σύμπτυξη των κόμβων, και επιλογή για αποθήκευση/φόρτωμα του γράφου.

Για τις λειτουργίες του συστήματος, απαραίτητες είναι επιπλέον ορισμένες ρουτίνες, όπως για τη σκίαση των κόμβων, την εύρεση των υποδέντρων του γράφου κλπ, καθώς και διαδικασίες για την επέκταση ερώτησης. Αυτές περιλαμβάνονται αντίστοιχα στις κλάσεις GraphUtils και Graph Functions, και μπορούν εναλλακτικά να χρησιμοποιηθούν και από το Ε.Γ., σε περίπτωση π.χ. που επιθυμούμε διαφορετική κατανομή των βαρών της επεξεργασίας ανάμεσα στα μέρη του συστήματος.

3. **Έλεγχος Κλιμάκωσης:**
   Ένας πίνακας ελέγχου *(Scale Window)* για τα διάφορα είδη κλιμάκωσης (βλ. παρ. 4.4.1.1).

4. **Ολική Όψη:**
   Παράθυρο *(Global View)* για τη συνοπτική παρουσίαση ολόκληρου του γράφου (βλ. παρ. 4.4.1.2).

5. **Χειρισμός Στρωμάτων:**
   Ένας πίνακας ελέγχου *(Layers Window)* για αποκρυβή, σκίαση ή σβήσιμο στρωμάτων του γράφου (βλ. παρ. 4.3.3).

6. **Έλεγχος σύμπτυξης:**
   Ένας πίνακας ελέγχου *(Collapse Window)* για την εφαρμογή των σημασιολογικών και τοπολογικών κριτηρίων σύμπτυξης στο γράφο (βλ. παρ. 4.3).

7. **Εκτέλεση Επερώτησης/Χωροδιάταξη :**
   Αυτός ο πίνακας παρέχεται μόνιμα στο χρήστη.

   Παρέχει έλεγχο για:

   - εκτέλεση επερώτησης:
     Ο χρήστης μπορεί να επιλέξει το είδος της επερώτησης, την οντότητα-στόχο, τον τρόπο με τον οποίο θα παριστάνονται οι σύνδεσμοι στο γράφο αποτελέσματος (ως ακμές, κόμβοι ή κόμβοι ανά κατηγορία), και τον αλγόριθμο για τη σχεδίαση του γράφου.
   - επέκταση επερώτησης:
     Επιλέγεται ο αλγόριθμος με τον οποίο θα γίνει η χωροδιάταξη στο αποτέλεσμα της επέκτασης. Ως οντότητες-στόχοι πάνω στις οποίες γίνεται η επέκταση λαμβάνονται οι οντότητες που έχουν επιλεγεί από το χρήστη στην Περιοχή Σχεδίασης του γράφου.

- ανασχεδίαση του γράφου:
  Επιλέγεται ο αλγόριθμος για την ανασχεδίαση του τρέχοντος γράφου.

8. **Βοηθητικές κλάσεις:**
   Γενικές κλάσεις περιέχουν τις ρουτίνες για το χειρισμό του γράφου: Παίρνοντας ως είσοδο έναν συγκεκριμένο γράφο και κατάλληλες παραμέτρους, παρέχουν τοπολογική και σημασιολογική σύμπτυξη, σκίαση κόμβων, εξάλειψη επικαλύψεων, υπολογισμό των ετικετών των συνοπτικών κόμβων κ.α. Επίσης, παρέχουν τις απαραίτητες λειτουργίες για την επέκταση επερώτησης.

Κεφάλαιο 6

Συμπεράσματα - Μελλοντικές Κατευθύνσεις

6.1 Συμπεράσματα

Είναι αναμφισβήτητη η χρησιμότητα των σημασιολογικών δικτύων για παράσταση πολύπλοκης γνώσης. Η γνώση αυτή μπορεί εύκολα να χρησιμοποιηθεί από αυτόματα προγράμματα, όμως όταν τίθεται το ζήτημα κατά πόσον υπάρχει η δυνατότητα για αποδοτική μετάδοσή της στο χρήστη, οι απαντήσεις δεν είναι πάντα καταφατικές. Αυτό έχει οδηγήσει πολλές φορές στην υιοθέτηση πιο κλασικών και συντηρητικών λύσεων σε διάφορα προβλήματα.

Στην εργασία μας, εισάγαμε μια δυναμική θεώρηση της διαδικασίας αναζήτησης: Η αναζήτηση αποτελείται από ημι-αυτόματες διαδικασίες, για το συντονισμό των οποίων η ανάμειξη του χρήστη παίζει κεντρικό ρόλο. Έτσι, το βάρος στην ανάπτυξη συστημάτων διερεύνησης πληροφορίας μετακινείται από την ανάπτυξη λειτουργιών που κάνουν πολλή και ολοκληρωμένη δουλειά αυτόματα, σε ημι-αυτόματες λειτουργίες που αποτελούν συστατικά της συνολικότερης διαδικασίας.

Η εργασία μας είναι ένα πρώτο βήμα προς την κατανόηση αυτής της διαδικασίας. Διακρίνουμε τις ανεξάρτητες μεταξύ τους παραμέτρους της, που αντιστοιχούν σε ανεξάρτητα επίπεδα λειτουργικότητας, και τα οργανώνουμε κατά έναν πειθαρχημένο τρόπο.

Τέλος, το Σύστημα Σημασιολογικής Διερεύνησης το οποίο αναπτύξαμε, μπορεί να χρησιμοποιηθεί ως πλατφόρμα πάνω στην οποία θα δοκιμαστεί η πραγματική αποτελεσματικότητα και η συνολική χρησιμότητα της προσέγγισής μας.

6.2 Σύγκριση με το GAIN

Σημείο εκκίνησης για την εργασία μας ήταν το μέρος του GAIN που ασχολήται με τη γραφική διερεύνηση της βάσης γνώσεων. Είναι σκόπιμο συνεπώς να κάνουμε μια σύγκριση των δύο συστημάτων.

Από άποψη πληρότητας, το GAIN υπερτερεί. Είναι ώριμο προϊόν, πράγμα που σημαίνει πλήρη έλεγχο λαθών, κατατοπιστικά μυνήματα στο χρήστη, καθώς και παραμετροποίηση του προγράμματος, που ελέγχεται μέσω της διεπαφής χρήσης. Ακόμα, δυνατότητες για εκτυπώσιμη έξοδο ενός γράφου, και μηχανισμό για σύνδεση με εξωτερικά εργαλεία, όπως π.χ. διορθωτές κειμένου ή προγράμματα παρουσίασης γραφικών.

Από άποψη λειτουργικότητας, το GAIN παρέχει επιπλέον προκατασκευασμένες επερωτήσεις με την απάντηση σε μορφή λίστας κειμένου και πολυκριτηριακές ερωτήσεις με απάντηση σε λίστα κειμένου. Ακόμα, διαθέτει προγραμματισμένους (hard-wired) μηχανισμούς για την πλήρη παρουσίαση μιας οντότητας με όλα της τα χαρακτηριστικά σε μορφή κειμένου (*Object Card*) ή σε μορφή γράφου, με έναν απλό αλγόριθμο (*Star View*).

Το ΣΣΔ υπερτερεί λειτουργικά κυρίως στη δυνατότητά του για δυναμικό χειρισμό και επεξεργασία του γραφικού αποτελέσματος, με μετακινήσεις και συμπτύξεις κόμβων, καθώς και στη δυνατότητα για δυναμική επέκταση των επερωτήσεων. Οι αλγόριθμοι σχεδίασης που διαθέτει μπορούν να φανούν ιδιαίτερα χρήσιμοι σε συνδιασμό με τις παραπάνω δυνατότητες. Ακόμα, έχει τη δυνατότητα να εμφανίζει τους συνδέσμους με διάφορες μορφές, ώς ακμές, κόμβους ή κόμβους κατηγοριών, κάτι που είναι πολύ χρήσιμο, ιδιαίτερα για χρήστες που δεν έχουν εξοικειοποιηθεί με το μοντέλο μιας βάσης.

Από άποψη απόδοσης, το GAIN είναι σαφώς ταχύτερο: Είναι προγραμματισμένο σε C++, που είναι πολύ γρηγορότερη από την Java. Επιπλέον, κατά την ανάπτυξη του ΣΣΔ δώσαμε ιδιαίτερη σημασία στην απλότητα της σχεδίασής του και στην ευκολία για μελλοντικές μετατροπές από τρίτους, θυσιάζοντας αρκετές φορές την απόδοση και αποφεύγοντας τις βελτιστοποιήσεις.

Τέλος, σημειώνουμε τη σημαντική δυνατότητα του ΣΣΔ να τρέχει μέσω δικτύου, πράγμα που δίνει για παράδειγμα τη δυνατότητα σε κάποιον που ενδιαφέρεται να γνωρίσει το ΣΣΕ να έχει μια άμεση επαφή με ένα παράδειγμα βάσης γνώσεων, και της μεθόδου εξερεύνησής της.

Συμπερασματικά μπορούμε να πούμε ότι το ΣΣΔ είναι ένα περισσότερο πιλοτικό (prototype) σύστημα, ακατάλληλο ίσως για ‘‘βιομηχανική’’ χρήση στη θέση του GAIN, κυρίως λόγω της ταχύτητάς του. Από την άλλη, είναι χρήσιμο για σκοπούς επίδειξης ή

εκπαιδευτικής χρήσης, ενώ προσφέρει μια πλατφόρμα για την αξιολόγηση νέων δυνατοτήτων στη διερεύνηση. Με μια βελτιστοποίησή του, μπορεί να χρησιμοποιηθεί για κανονική χρήση παρέχοντας χρήσιμες δυνατότητες, πολλές από τις οποίες έχουν ήδη εντοπιστεί από χρήστες του GAIN.

6.3 Μελλοντικές Κατευθύνσεις

Στο Κεφ. 4 αναφέραμε αρκετές από τις πιθανές μελλοντικές επεκτάσεις του συστήματός μας, για να έχουμε μια συνολικότερη εικόνα της προσέγγισής μας.

Εντοπίσαμε ότι ιδιαίτερα οι απαιτήσεις μας για την αυτόματη σχεδίαση γράφων αγγίζουν ή υπερβαίνουν τα όρια της σύγχρονης τεχνολογίας και έρευνας σε αυτόν τον τομέα, πράγμα που σημαίνει ότι στο προσεχές μέλλον μπορούμε να έχουμε δραματικες βελτιώσεις σε αυτή την κατεύθυνση.

Σε αυτό το κεφάλαιο, παρουσιάζουμε αναλυτικότερα τις σημαντικότερες κατευθύνσεις οι οποίες ανοίγονται σε κάθε τομέα λειτουργικότητας.

6.3.1 Σχεδίαση του γράφου

Σχεδόν απεριόριστες δυνατότητες υπάρχουν ακόμα για αποτελεσματικότερη πληροφόρηση του χρήστη με κατάλληλη σχεδίαση των γράφων απάντησης. Όπως επισημάναμε στην παρ. 4.1.1.2, αυξητικοί τρόποι σχεδίασης μπορούν να χρησιμοποιηθούν, σε συνδιασμό με την επέκταση των επερωτήσεων. Επιπλέον, μπορεί να γίνει σχεδίαση βάσει *δηλωτικών μεθόδων*, σχεδίαση *βάσει παραδείγματος*, ή με τεχνικές για ομαδοποίηση του γράφου. Επιπλέον, μπορούμε να εξετάσουμε *εναλλακτικές μορφές παράστασης των γράφων*, πέραν της παράστασής τους με κόμβους και ακμές. Στη συνέχεια, αναφερόμαστε σε καθεμία από τις δυνατότητες αυτές χωριστά.

6.3.1.1 Δηλωτική σχεδίαση

Η δηλωτική σχεδίαση των γράφων ταιριάζει ιδιαίτερα σε γράφους σημασιολογικών δικτύων. Είναι ιδανική για το μοντέλο δεδομένων του ΣΣΕ, καθώς κανόνες του τύπου ‘‘σχεδιάζουμε την σχέση isA από κάτω προς τα πάνω’’ ή ‘‘σχεδιάζουμε τα γνωρίσματα προς τα πλάγια’’ χρησιμοποιούνται ως συνηθισμένη πρακτική στη σχεδίαση των βάσεων με το χέρι.

Τα περισσότερα από τα σημερινά συστήματα δηλωτικής σχεδίασης έχουν προκύψει με

επεκτάσεις γνωστών μεθοδολογιών για χωροδιατάξεις γενικών σχεδίων, και χειρίζονται *γεωμετρικούς* ή *χωρικούς* περιορισμούς, όπως π.χ. ''η μέγιστη απόσταση μεταξύ αυτών των κόμβων να είναι d'', ή ''κράτα αυτούς τους κόμβους στο ίδιο οριζόντιο επίπεδο'', ή ''αυτός ο κόμβος να είναι στα αριστερά εκείνου''. Οι περιορισμοί που απαιτούν τα σημασιολογικά δίκτυα, π.χ. ο περιορισμός για την isA σχέση, αν εκφραστεί σε τέτοιου είδους περιορισμούς, οδηγεί αφ' ενός σε ένα μεγάλο σύνολο κανόνων, αφ' ετέρου σε αμφισημίες (π.χ. ένας κύκλος από κόμβους καθένας εκ των οποίων είναι στα αριστερά του άλλου). Τα ζητήματα αυτά (πολλοί κανόνες και αμφισημίες) δεν αντιμετωπίζονται ικανοποιητικά ακόμα από τα υπάρχοντα συστήματα.

Είναι πάντως οπωσδήποτε ενδιαφέρον να εξεταστεί αναλυτικότερα σε ποιόν βαθμό οι δυνατότητες για περιορισμούς που παρέχουν μπορούν να φανούν χρήσιμες, και να εξεταστεί πιο συγκεκριμένα το ζήτημα του χρόνου εκτέλεσης που απαιτούν.

6.3.1.2 Σχεδίαση βάσει παραδείγματος

Η σχεδίαση βάσει παραδείγματος είναι άλλη μια χαρακτηριστική τεχνική, που χρησιμοποιείται κατά κόρο στο μοντέλο δεδομένων του ΣΣΕ: Μια ''περίπτωση'' ενός γενικότερου ''σχήματος'' σχεδιάζεται με βάση το σχέδιο του ''σχήματος'', το οποίο παρέχει συνήθως ο σχεδιαστής της βάσης. Η μέθοδος αυτή αποτελεί ιδανικό τρόπο κατανόησης των ''δεδομένων'' δοθέντος του ''σχήματος'', ή κατανόησης του ''σχήματος'' με τη βοήθεια μιας περίπτωσής του. Το DOODLE [14], είναι ένα σύγχρονο σύστημα για σχεδίαση γράφων, η τεχνική του οποίου βασίζεται σε αυτήν την ιδέα. Χειρίζεται όμως ακόμα μόνο ειδικές κατηγορίες γράφων, συγκεκριμένα δέντρα και ακυκλικούς επίπεδους γράφους.

6.3.1.3 Σχεδίαση με ομαδοποίηση

Ίσως η πιο άμεσα αποδοτική λύση στο πρόβλημα της σχεδίασης πολύπλοκων γράφων με την υπάρχουσα τεχνολογία, είναι αυτή της ομαδοποίησης των γράφων και της ξεχωριστής σχεδίασης κάθε συμπλέγματος (βλ. παρ. 3.5). Στο [20] υπάρχει ένα εντυπωσιακό παράδειγμα αυτής της τεχνικής, όπου ''ένας γράφος που θα ήταν σχεδόν αδύνατο να τον εξερευνήσει και να τον κατανοήσει κανείς με ένα μοναδικό σχέδιο, έγινε αρκετά ευανάγνωστος με χρήση καλά σχεδιασμένων δυναμικών σχεδίων, που χρησιμοποιούν μια ειδική μεταφορά τύπου 'επέκταση/σύμπτυξη' για την εξερεύνηση του γράφου''.

Στο δικό μας σύστημα, παρέχεται ήδη μια περιορισμένη μορφή αυτής της λειτουργίας: μια ομαδοποίηση των κόμβων προκύπτει με την σύμπτυξη, με τοπολογικά ή σημασιολογικά κριτήρια, ενός υπογράφου. Ο χρήστης μπορεί να ζητήσει κατόπιν την ανασχεδίαση του

γράφου που προκύπτει.

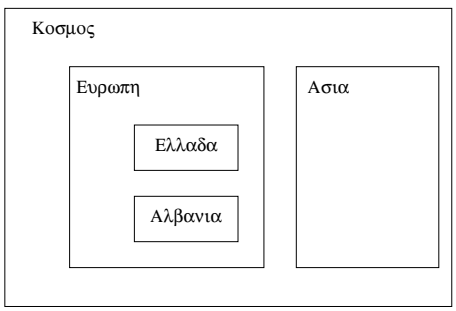
Με ελάχιστες προσθήκες στο επίπεδο της υλοποίησης, η διαδικασία αυτή μπορεί εναλλακτικά να εφαρμοστεί *πρίν* την αρχική σχεδίαση του γράφου και την παρουσίασή του στο χρήστη, ώστε να έχουμε άμεσα το επιθυμητό αποτέλεσμα. Το ζήτημα που τίθεται και είναι πιο ενδιαφέρον να εξεταστεί, είναι με ποιο τρόπο προσδιορίζονται εκ των προτέρων οι επιθυμητές συμπτύξεις (πιθανώς με ένα σύνολο κανόνων που μπορεί ο χρήστης να τροποποιεί).

Επιπλέον, ειδικοί αλγόριθμοι για ομαδοποίηση, όπως π.χ. στα [43] [41], μπορούν να ενσωματωθούν, για καλύτερα αποτελέσματα σε μεγάλους γράφους.

Σημειώνουμε εδώ ότι ένα από τα σημαντικά ζητήματα που παρουσιάζονται με χρήση αυτής της προεπεξεργασίας, είναι ότι χρειάζεται μελέτη για τον τρόπο ολοκλήρωσής της με το μηχανισμό επέκτασης επερώτησης. Ερωτήματα που πρέπει να απαντηθούν είναι τί γίνεται σε περίπτωση επέκτασης του γράφου επάνω σε κόμβους που έχουν περιληφθεί σε ένα σύμπλεγμα. Η επέκταση γίνεται μέσα στο σύμπλεγμα; Οι κανόνες που έχουν δημιουργήσει το σύμπλεγμα συνεχίζουν να ισχύουν αν εφαρμοστούν στον νέο γράφο που προκύπτει; Προκύπτουν νέα συμπλέγματα; Όλα αυτά τα ζητήματα είναι μάλλον μή τετριμμένα.

6.3.1.4 Εναλλακτικές μορφές παράστασης γράφων

Σε όλη μας τη συζήτηση, θεωρήσαμε ότι οι γράφοι παριστάνονται με κόμβους και ακμές. Υπάρχουν όμως κι άλλοι τρόποι παράστασης των γράφων. Ένας τρόπος που είναι ιδιαίτερα πρόσφορος για χρήση σε σημασιολογικά δίκτυα, είναι με χρήση *σχεδίων εγκλεισμού* (inclusion drawings), όπως στο σχ. 6.1.



Σχήμα 6.1: Σχέδιο εγκλεισμού

Ο τρόπος αυτός παριστάνει πολύ παραστατικά την ειδική σχέση ‘‘εγκλεισμού’’. Η χρήση του παρουσιάζει ιδιαίτερο ενδιαφέρον, εφόσον μάλιστα υπάρχουν αρκετά συστήματα για την αυτόματη παραγωγή τέτοιων σχεδίων [36], [14].

6.3.1.5 Πολλαπλές όψεις

Μια ακόμα τεχνική είναι η χρήση πολλαπλών όψεων για την εξερεύνηση ενός μοναδικού γράφου [37]. Η κάθε όψη μπορεί να προσδιορίζεται με βάση ένα σημασιολογικό κριτήριο, για παράδειγμα μια κατηγορία συνδέσμων.

Στη δική μας προσέγγιση, η χρήση γενικών προκατασκευασμένων επερωτήσεων, ουσιαστικά υποκαθιστά μια τέτοια μέθοδο: Για κάθε όψη που μπορεί να είναι χρήσιμη, ο σχεδιαστής του συστήματος αρκεί να παρέχει μια κατάλληλη προκατασκευασμένη επερώτηση.

6.3.1.6 Αυτόματη επιλογή αλγορίθμου σχεδίασης

Όπως έχουμε ήδη αναφέρει, πολλοί από τους αλγορίθμους σχεδίασης χειρίζονται μόνο ειδικές μορφές γράφων, π.χ. δένδρα ή επίπεδους γράφους. Η αναγνώριση της μορφής ενός γράφου μπορεί να γίνει σε γραμμικό χρόνο. Συνεπώς, θα ήταν σκόπιμη μια λειτουργία που επιλέγει αυτόματα τον κατάλληλο αλγόριθμο για τη σχεδίαση κάθε δεδομένου γράφου.

6.3.2 Άλλες επεκτάσεις

Οι λειτουργίες που παρέχονται στο σύστημά μας και έχουν να κάνουν με τοπολογικά και σημασιολογικά κριτήρια, ασχολούνται με σύμπτυξη, σκίαση και απόκρυψη υπογράφων. Νέες λειτουργίες μπορούν να προστίθενται αν φανεί χρήσιμο, όπως π.χ. αλγόριθμοι για εύρεση του συντομότερου μονοπατιού ανάμεσα σε δύο κόμβους.

Η διεπαφή χρήσης μπορεί να γίνει περισσότερο διαλειτουργική, όπως επισημάναμε στην παρ. 4.4.2.1, και περισσότερος ‘‘άμεσος χειρισμός’’ (direct manipulation) των κόμβων οντοτήτων μπορεί να ενσωματωθεί, για μεγαλύτερη ευχρηστία του συστήματος (π.χ. η επιλογή της επέκτασης επερώτησης να γίνεται ‘‘πάνω’’ στον κόμβο- στόχο).

Για την παρουσίαση των γράφων, νέες τεχνικές, όπως η σχεδίασή τους σε υπερβολικές επιφάνειες, μπορεί να εξεταστούν, αν και μέχρι σήμερα δεν είναι ξεκάθαρο αν αυτή η μέθοδος είναι πραγματικά βοηθητική [35].

Σημαντικό είναι ακόμα, το σύστημα να παρακολουθήσει τις εξελίξεις που συμβαίνουν στην ερευνητική κοινότητα που ασχολείται με την επεξεργασία και σχεδίαση γράφων, για να υιοθετήσει λ.χ. μια τυποποιημένη μορφή εσωτερικής παράστασης του γράφου, αν προκύψει μια τέτοια, πράγμα που θα βοηθήσει στην καλύτερη διαλειτουργικότητά του με άλλα συστήματα.

Τέλος, μπορούμε να προτείνουμε την σύνδεση του συστήματός μας με ένα σύστημα παρουσίασης πληροφορίας σε μορφή κειμένου, όπως αυτό της Κ. Τόλιου [5], για την τελική παροχή ενός πληρέστερου συστήματος εξερεύνησης σημασιολογικών βάσεων.

Βιβλιογραφία

[1] Μαρία Καραβασίλη. Ένας αλγόριθμος σχεδίασης κατευθυνόμενων γράφων. Μεταπτυχιακή Εργασία, Τμ. Επιστήμης Υπολογιστών, Πανεπιστήμιο Κρήτης, Φεβρουάριος 1994.

[2] Μάνος Θεοδωράκης. Εμβέλεια Ονόματος σε Σημασιολογικά Μοντέλα Δεδομένων. Μεταπτυχιακή Εργασία, Τμ. Επιστήμης Υπολογιστών, Πανεπιστήμιο Κρήτης, Οκτώβριος 1995.

[3] Ειρήνη Φουντουλάκη. Σύστημα Υποστήριξης Διαγνωστικών Διαδικασιών σε Τεχνικές Εφαρμογές. Μεταπτυχιακή Εργασία, Τμ. Επιστήμης Υπολογιστών, Πανεπιστήμιο Κρήτης, Οκτώβριος 1996.

[4] Ελένη Πατάκη. Σχεδίαση και Υλοποίηση ενός Εργαλείου Επιλογής Λογισμικού για Αναχρησιμοποίηση. Μεταπτυχιακή Εργασία, Τμ. Επιστήμης Υπολογιστών, Πανεπιστήμιο Κρήτης, 1993.

[5] Κατερίνα Τόλιου. Πύλη διασύνδεσης του Παγκόσμιου Ιστού με το Σήστημα Σημασιολογικού Ευρετηριασμού. Μεταπτυχιακή Εργασία, Τμ. Επιστήμης Υπολογιστών, Πανεπιστήμιο Κρήτης, Οκτώβριος 1998.

[6] Γιάννης Βελεγράκης. VRW: Ενα Σύστημα Κατασκευής και Μορφοποίησης Αναφορών. Technical Report FORTH-ICS-TR-197, Ινστιτούτο Πληροφορικής, 'Ιδρυμα Τεχνολογίας και 'Ερευνας, May 1997.

[7] Giuseppe Di Battista, Peter Eades, Roberto Tamassia, and Ioannis G. Tollis. Algorithms for drawing graphs: an annotated bibliography. *Comput. Geom. Theory Appl.*, 4:235--282, 1994. Available via FTP at ftp://wilma.cs.brown.edu/pub/papers/compgeo/gdbiblio.tex.Z.

[8] F. J. Brandenburg. Designing graph drawings by layout graph grammars. In R. Tamassia and I. G. Tollis, editors, *Graph Drawing (Proc. GD '94)*, volume 894 of *Lecture Notes in Computer Science*, pages 416--427. Springer-Verlag, 1995.

[9] Maria Christoforaki, Panos Constantopoulos, and Martin Dörr. CLIO: An object-oriented model of cultural data. Technical Report ICS-FORTH.MUIS.92.1, Institute of Computer Science Foundation for Research and Technology Hellas, November 1992.

[10] R.F. Cohen, G. Di Battista, R. Tamassia, and I.G. Tollis. Dynamic Graph Drawings: Trees, Series-Parallel Digraphs, and Planar ST-Digraphs. *SIAM J. Comput.*, 24(no. 5):970--1001, 1995.

[11] R.F. Cohen, P. Eades, T. Lin, and F. Ruskey. Three-Dimensional Graph Drawing. In R. Tamassia and I.G. Tollis, editors, *Graph Drawing (Proc. GD '94)*, volume 894 of *LNCS*, pages 1--11. Springer-Verlag, 1995.

[12] Panos Constantopoulos. Cultural Documentation: The CLIO System. Technical Report 115, Institute of Computer Science Foundation for Research and Technology Hellas, January 1994.

[13] Panos Constantopoulos and Martin Dörr. The Semantic Index System: A brief presentation. Institute of Computer Science, Foundation of Research and Technology - Hellas, May 1994. Available in http://www.ics.forth.gr/proj/isst/Systems/sis.

[14] I. F. Cruz. Doodle: A visual language for object-oriented databases. *Proc. ACM SIGMOD Conf.*, page 71, June 1992.

[15] Costas Dadouris, Martin Dörr, Antonis Kladogenis, and Christos Georgis. Sis - application programmatic interface reference manual. manuscript, Institute of Computer Science, Foundation of Research and Technology - Hellas, 1997.

[16] Giuseppe DiBattista, editor. *Graph Drawing, 5th International Symposium, GD '97, Rome, Italy, September 1997*, volume 1353 of *LNCS*. Springer-Verlag, 1997.

[17] Ugur Dogrusoez, Brendan Madden, and Patrick Madden. Circular Layout in the Graph Layout Toolkit. In Stephen North, editor, *Graph Drawing (Proc. GD '96)*, volume 1190 of *Lecture Notes in Computer Science*, pages 92--100. Springer-Verlag, 1996.

[18] M. Dörr. Effective Terminology Support for Distributed Digital Collections. In *6th DELOS Workshop, Preservation of Digital Information*, Tomar, Portugal 17-19 June, 1998.

[19] P. Eades. A Heuristic for Graph Drawing. *Congressus Numerantuim*, 42:149--160, 1984.

[20] Peter Eades, Joe Marks, and Stephen North. Graph-Drawing Contest Report. In Stephen North, editor, *Graph Drawing (Proc. GD '96)*, volume 1190 of *Lecture Notes in Computer Science*, pages 129--138. Springer-Verlag, 1996.

[21] A. Frick, A. Ludwig, and H. Mehldau. A fast adaptive layout algorithm for undirected graphs. In R. Tamassia and I. G. Tollis, editors, *Graph Drawing (Proc. GD '94)*, volume 894 of *Lecture Notes in Computer Science*. Springer-Verlag, 1995.

[22] Sis: Graphical analysis interface user manual. manuscript, Institute of Computer Science, Foundation of Research and Technology - Hellas, 1997.

[23] Cristian Ghezzi. Three dimensional visualization of structured data. Master's thesis, Politecnico di Milano, Italy, December 1996.

[24] Gene Golovchinsky, Klaus Reichenberger, and Thomas Kamps. Subverting Structure: Data-Driven Diagram Generation. In *Proceedings of the 6th IEEE Visualization '95*, pages 217--223, Atlanta, Georgia, October 1995. Available in http://anarch.ie.utoronto.ca/people/gene.html/publications/vis95/.

[25] M. Himsolt. The Graphlet System. In Stephen North, editor, *Graph Drawing (Proc. GD '96)*, volume 1190 of *Lecture Notes in Computer Science*, pages 233--240. Springer-Verlag, 1996.

[26] Michael Himsolt. Gml: Graph modeling language, December 1996. Available through http://www.uni-passau.de/Graphlet/GML/index.html.

[27] Claudia Iturriaga and Anna Lubiw. Elastic Labels: The Two-Axis Case. In Giuseppe DiBattista, editor, *Graph Drawing, 5th International Symposium, GD '97*, volume 1353 of *Lecture Notes in Computer Science*, pages 181--192. Springer-Verlag, 1997.

[28] Conceptbase v5.0 user manual. RWTH Aachen, Germany, 1998. Available in http://www-i5.informatik.rwth-aachen.de/CBdoc/userManual-V50.

[29] S. Tagawa K. Sugiyama and M. Toda. Methods for visual understanding of hierarchical system structures. *IEEE Trans. Syst. Man Cybern.*, SMC-11(2):109--125, February 1981.

[30] Yael Kahn and Terry Baylis. A survey of graph drawing algorithms based on graph grammars. presented at the Fifth International Workshop on Graph Grammars and Their Application to Computer Science, Williamsburg, Virginia, November 1994. Available in http://ltc.law.warwick.ac.uk/yael/gragra/1.htm.

[31] Konstantinos G. Kakoulis and I. G. Tollis. An Algorithm for Labeling Edges of Hierarchical Drawings. In Giuseppe DiBattista, editor, *Graph Drawing, 5th International Symposium, GD '97*, volume 1353 of *Lecture Notes in Computer Science*, pages 169--180. Springer-Verlag, 1997.

[32] T. Kamada and S. Kawai. An algorithm for drawing general undirected graphs. *Information Processing Letters*, 31:7--15, 1989.

[33] E. Koutsofios and S. North. Drawing graphs with dot: User's manual. Bell Laboratories, 1993. Available through http://www.research.att.com/sw/tools/graphviz/.

[34] V. Lalioti and P. Loucopoulos. Visualization for Validation. In C. Rolland, F. Bodart, and C. Cauvet, editors, *Advanced Information Systems Engineering, CAiSE '93, Paris, France*, volume 685 of *LNCS*, pages 143--164, June 1993.

[35] John Lamping, Ramana Rao, and Peter Pirolli. A Focus+Context Technique Based on Hyperbolic Geometry for Visualizing Large Hierarchies. In *Proc. of ACM CHI '95 Conference: Human Factors in Computing Systems*, New York, 1995. ACM.

[36] K. Misue, K. Nitta, K. Sugiyama, T. Koshiba, and R.Inder. Enhancing D-ABDUCTOR Towards a Diagrammatic User Interface Platform. In *Proceedings of the 2nd International Conference on Knowledge-Based Intelligent Electronic Systems (KES '98)*. IEEE Press, 1998.

[37] Mukherjea, Sougata, Foley, James D., and Scott Hudson. Visualizing complex hypermedia networks through multiple hierarchical views. In *Proc. of ACM CHI '95 Conference: Human Factors in Computing Systems*, pages 331--337, New York, 1995. ACM.

[38] John Mylopoulos, Alexander Borgida, Matthias Jarke, and Manolis Koubarakis. Telos : Representing Knowledge about Information Systems. *ACM Transactions on Information Systems*, 8(4), October 1990. University of Toronto.

[39] S. C. North. Incremental Layout in DynaDAG. In *Graph Drawing (Proc. GD '95)*, number 1027 in LNCS. Springer-Verlag, 1995.

[40] Stephen North, editor. *Graph Drawing (Proc. GD '96)*, volume 1190 of *LNCS*. Springer-Verlag, Berkeley, California, USA, September 1996.

[41] T. Roxborought and A. Sen. Graph Clustering Using Multiway Ratio Cut. In Giuseppe DiBattista, editor, *Graph Drawing, 5th International Symposium, GD '97*, volume 1353 of *Lecture Notes in Computer Science*, pages 291--296. Springer-Verlag, 1997.

[42] S. Bridgeman and A. Garg and R. Tamassia. A Graph Drawing and Translation Service on the WWW. In Stephen North, editor, *Graph Drawing (Proc. GD '96)*, volume 1190 of *Lecture Notes in Computer Science*, pages 45--52. Springer-Verlag, 1996.

[43] Reinhard Sablowski and Arne Frick. Automatic Graph Clustering (System Demonstration). In Stephen North, editor, *Graph Drawing (Proc. GD '96)*, volume 1190 of *Lecture Notes in Computer Science*, pages 395--400. Springer-Verlag, 1996.

[44] G. Sander. Graph layout throught the vcg tool. In R. Tamassia and I. G. Tollis, editors, *Graph Drawing (Proc. GD '94)*, volume 894 of *Lecture Notes in Computer Science*, pages 194--205. Springer-Verlag, 1995.

[45] Frisco Schutte. Correlating documents with common knowledge by using attribute links. Institute of Computer Science, Foundation of Research and Technology - Hellas, June 1996. Available via http://www.ics.forth.gr/proj/isst/Publications/TechnicalReports.html.

[46] Shin Takahashi, Satoshi Matsuoka, Akinori Yonezawa, and Tomihisa Kamada. A General Framework for Bi-Directional Translation between Abstract and Pictorial Data. In *Proceedings of the ACM SIGGRAPH and SIGCHI Symposium on User Interface Software and Technology (UIST'91)*, pages 165--174. ACM Press, November 1991.

[47] R. Tamassia, Di Battista G., and C. Batini. Automatic Graph Drawing and Readability of Diagrams. *IEEE Trans. Syst. Man Cybern.*, SMC-18(no. 1):61--79, 1988.

[48] R. Tamassia and I. G. Tollis, editors. *Graph Drawing (Proc. GD '94)*, volume 894 of *LNCS*. Springer-Verlag, 1995.

[49] D. Tunkelang. A layout algorithm for undirected graphs. In *Proc. ALCOM Intl. Workshop on Graph Drawing and Topological Graph Algorithms*, September 1993.

[50] D. Tunkelang, R. J. Byrd, and Cooper J. W. Lexical Navigation: Using Incremental Graph Drawing for Query Refinement. In Giuseppe DiBattista, editor, *Graph Drawing, 5th International Symposium, GD '97*, volume 1353 of *Lecture Notes in Computer Science*, pages 316--321. Springer-Verlag, 1997.

[51] Gaby Zinssmeister and C. McCreary. Drawing graphs with attribute graph grammars. In *Graph Grammar Workshop*, pages 355--369. Williamsburg, 1994.