



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ

**Σημασιολογικός Εμπλουτισμός της Πληροφορίας σε Σχεσιακές
Βάσεις Δεδομένων και Συστήματα με Επίγνωση Περιβάλλοντος**

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

του

Νικόλαου Η. Κωνσταντίνου

Αθήνα, Δεκέμβριος 2009



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ
ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ
ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

Σημασιολογικός Εμπλουτισμός της Πληροφορίας σε Σχεσιακές Βάσεις Δεδομένων και Συστήματα με Επίγνωση Περιβάλλοντος

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

του

Νικόλαου Η. Κωνσταντίνου

Συμβουλευτική Επιτροπή: Νικόλαος Μήτρου
Γεώργιος Στασινόπουλος
Ελευθέριος Καγιάφας

Εγκρίθηκε από την επταμελή εξεταστική επιτροπή την 7η Δεκεμβρίου 2009.

.....
Νικόλαος Μήτρου
Καθηγητής Ε.Μ.Π.

.....
Γεώργιος Στασινόπουλος
Καθηγητής Ε.Μ.Π.

.....
Ελευθέριος Καγιάφας
Καθηγητής Ε.Μ.Π.

.....
Ευστάθιος Συκάς
Καθηγητής Ε.Μ.Π.

.....
Ιωάννης Βασιλείου
Καθηγητής Ε.Μ.Π.

.....
Ιωάννα Ρουσάκη
Λέκτορας Ε.Μ.Π.

.....
Παναγιώτης Δεμέστιχας
Αν. Καθηγητής Πανεπιστημίου
Πειραιώς

Αθήνα, Δεκέμβριος 2009.

.....
Νικόλαος Η. Κωνσταντίνου
Διδάκτωρ Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

© (2009) Εθνικό Μετσόβιο Πολυτεχνείο. All rights reserved.

Περιεχόμενα

1	Εισαγωγή	1
1.1	Δεδομένα και πληροφορία	1
1.2	Διαλειτουργικότητα και ολοκλήρωση κατανεμημένων πηγών πληροφορίας	2
1.3	Σημασιολογική επισημείωση	6
1.4	Η ανάγκη για προσθήκη σημασιολογίας	7
1.5	Αναπαράσταση γνώσεως με RDF γράφο	10
1.5.1	Ορισμός RDF γράφων	12
1.6	Περιγραφική λογική	15
1.6.1	Εκφραστικότητα περιγραφικής λογικής	16
1.7	Βάση Γνώσεως	19
1.8	Οντολογίες	21
1.9	Δομή της διατριβής	23
2	Γλώσσες σημασιολογικής περιγραφής και αναζήτησης - Επισκόπηση	25
2.1	Το σχήμα RDFS	25
2.2	Η γλώσσα OIL	26
2.3	Η γλώσσα DAML+OIL	28
2.4	Η γλώσσα OWL	29
2.5	Συνήθη λεξικά και μικροπρότυπα	36
2.6	Σύνταξη κανόνων	41
2.7	Γλώσσες ερωτημάτων	44
2.7.1	Διάσχιση γράφων: Οικογένεια RQL	46
2.7.2	Πρότυπα τριάδων: Οικογένεια SPARQL	48
2.7.3	Σημασιολογικά χαρακτηριστικά της SPARQL	50
2.8	Συμπεράσματα	52
3	Διαχείριση, επεξεργασία και αξιοποίηση σημασιολογικής πληροφορίας	54
3.1	Εισαγωγή	54
3.2	Σύνταξη οντολογιών	55
3.2.1	Protégé	56
3.2.2	SWOOP και SMORE	57
3.3	Σημασιολογική επισημείωση	58
3.4	Προγραμματιστικές διεπαφές	60
3.4.1	Jena	60
3.5	Η οντολογία Cyc	61
3.6	Εξυπηρετητές υπηρεσιών συλλογιστικής	63
3.7	Τεχνολογίες Διαδικτύου	68
3.7.1	Πράκτορες λογισμικού	68
3.7.2	Συστήματα με επίγνωση περιβάλλοντος	70
3.7.3	Σημασιολογικές Υπηρεσίες Διαδικτύου	73
3.7.4	Μηχανές αναζήτησης	75
3.7.5	Πολυμεσικά δεδομένα	78

3.8	Συμπεράσματα	80
4	Συνεργασία οντολογιών με Βάσεις Δεδομένων	84
4.1	Εισαγωγή	84
4.1.1	Αντιστοίχιση και ολοκλήρωση δεδομένων	85
4.1.2	Προτυποποίηση συστημάτων με τη χρήση Τεχνικών Χώρων	86
4.1.3	Σχεσιακό μοντέλο	87
4.1.4	Αντιστοίχιση σχεσιακών Βάσεων Δεδομένων με οντολογίες	88
4.1.5	Θεώρηση ανοιχτού και κλειστού κόσμου	89
4.1.6	Θεωρητικοί περιορισμοί	90
4.2	Ανασκόπηση βιβλιογραφίας	90
4.2.1	Ανάλυση των προσεγγίσεων	96
4.2.2	Μελέτη εφικτότητας (Proof of concept)	101
4.3	Σχεδιασμός και υλοποίηση ενός πρωτότυπου συστήματος (Vis-A-Vis)	104
4.3.1	Κεντρική ιδέα του συστήματος	105
4.3.2	Έλεγχοι συνέπειας	106
4.3.3	Εκτέλεση σημασιολογικού ερωτήματος	111
4.4	Σχολιασμός	112
4.5	Πληρότητα, εγκυρότητα και απόδοση	113
4.6	Συμπεράσματα, παρατηρήσεις και προτάσεις	114
5	Προσθήκη σημασιολογίας σε πραγματικό χρόνο και καταναμημένο περιβάλλον	118
5.1	Εισαγωγή	118
5.1.1	Δεδομένα και Πληροφορία	119
5.1.2	Μεσισμικό	120
5.1.3	Επεξεργασία πραγματικού χρόνου	121
5.1.4	Σύγχρονη και ασύγχρονη επεξεργασία πληροφορίας	122
5.2	Εξαγωγή σημασιολογικής πληροφορίας από πολυμεσική πληροφορία	123
5.2.1	Σχετικές Εργασίες	124
5.2.2	Αρχιτεκτονική του μεσισμικού	125
5.2.3	Κανόνες αντιστοίχισης και Σημασιολογικοί κανόνες	129
5.2.4	Ερωτήματα σημασιολογικού επιπέδου	131
5.2.5	Χρήστες συστήματος και παρεχόμενη λειτουργικότητα	133
5.3	Παράδειγμα επεξεργασίας πραγματικού χρόνου	135
5.3.1	Παρακολούθηση χώρων	135
5.3.2	Εφαρμογή υποστήριξης συνεδριάσεων	139
5.3.3	Διαχωρισμός και Σύντηξη Μηνυμάτων	141
5.4	Μετρήσεις απόδοσης στην επεξεργασία πραγματικού χρόνου	143
5.4.1	Πειραματικές μετρήσεις	145
5.5	Αντιμετώπιση Σφαλμάτων και Ασυνεπειών	154
5.6	Συμπεράσματα	156
5.7	Μελλοντική εργασία - προκλήσεις	159
5.7.1	Ασφάλεια	161
5.7.2	Δίκτυα αισθητήρων	162

6 Συμπεράσματα - Μελλοντική έρευνα	163
6.1 Σύνοψη της διατριβής και συμπεράσματα	163
6.2 Ανακεφαλαίωση των καινοτομικών στοιχείων της διατριβής	164
6.3 Θέματα για περαιτέρω διερεύνηση	165
Βιβλιογραφία	167
Πίνακας ακρωνυμίων	181
Παράρτημα	183
Κατάλογος δημοσιεύσεων του συγγραφέα	195

Ευρετήριο Εικόνων

1	Παράδειγμα χρήσης ανώνυμου πόρου σε RDF γράφο	14
2	Αρχιτεκτονική συστήματος αναπαράστασης γνώσεως βασισμένου σε Περιγραφική Λογική	21
3	Στρώματα Σημασιολογικού Ιστού	26
4	Σχέση κλάσης υποκλάσης μεταξύ της OWL και του RDFS	30
5	Κατηγοριοποίηση των προσεγγίσεων σημασιολογικής επισημείωσης	59
6	Περιγραφή Υπηρεσιών Διαδικτύου με OWL-S	74
7	Ιστορία του Διαδικτύου	82
8	Αποτέλεσμα σημασιολογικά εμπλουτισμένο με RDFa σε μηχανή αναζήτησης	82
9	Συνεργασία οντολογιών με Βάσεις Δεδομένων	84
10	Αντιστοίχιση σχεσιακών Βάσεων Δεδομένων με οντολογίες	89
11	Αρχιτεκτονική του D2RQ	93
12	Απλή ταξινόμηση των προσεγγίσεων για την αντιστοίχιση Βάσεων Δεδομένων με οντολογίες	100
13	Visavis	105
14	Μη αποδεκτή αντιστοίχιση σε περίπτωση ξένων μεταξύ τους κλάσεων	108
15	Αποδεκτή αντιστοίχιση σε περίπτωση σχέσης υπερκλάσης-υποκλάσης	109
16	Αποδεκτή αντιστοίχιση σε περίπτωση ξένου κλειδιού	110
17	Γενική αρχιτεκτονική συστήματος αυτόματης εξαγωγής σημασιολογίας	126
18	Επεξεργασία εισερχόμενων μηνυμάτων	128
19	Σύνταξη Κανόνων αντιστοίχισης	130
20	Σύνταξη Σημασιολογικών κανόνων	131
21	Στιγμιότυπα της εφαρμογής παρακολούθησης χώρων	137
22	Διάγραμμα καταστάσεων εφαρμογής υποστήριξης συνεδριάσεων	140
23	Στιγμιότυπα πειραματικών δεδομένων	146

24	Στην (a) φαίνεται η συμπεριφορά του μεσισμικού με υποστήριξη από υπηρεσίες συλλογιστικής. Στην (b) φαίνεται πως χωρίς τη χρήση υπηρεσιών συλλογιστικής ο χρόνος απόκρισης είναι αμελητέος και δεν εξαρτάται από το μέγεθος της Βάσης Γνώσεως	147
25	Στις (a) και (b) παρουσιάζεται η απόδοση του μεσισμικού σε σχέση με τις υπηρεσίες συλλογιστικής και το σύνολο κανόνων, αντίστοιχα. Στις (c) και (d) παρουσιάζεται η απόδοση του μεσισμικού κατά τις εφαρμογές παρακολούθησης χώρων και υποστήριξης συνεδριάσεων, αντίστοιχα. Σε κάθε περίπτωση το σύστημα υποστηρίζεται από υπηρεσίες συλλογιστικής ενώ δε λαμβάνει χώρα κάποια διαχείριση του χρόνου επεξεργασίας των μηνυμάτων	149
26	Στις (a) και (b) παρουσιάζεται ο χρόνος απόκρισης της εφαρμογής παρακολούθησης χώρων, ενώ στις (c) και (d) ο χρόνος απόκρισης της εφαρμογής υποστήριξης συνεδριάσεων. Σε κάθε περίπτωση οι εφαρμογές υποστηρίζονται από υπηρεσίες συλλογιστικής και ο χρόνος απόκρισης διατηρείται σε συγκεκριμένο κατώφλι	151
27	Στην (a) παρατηρείται ότι μεγαλύτερα κατώφλια οδηγούν σε πιο συχνές διαδικασίες συντήρησης. Στην (b) φαίνεται το γεγονός ότι ο χρόνος επεξεργασίας του κάθε μηνύματος είναι ευθέως ανάλογος του αριθμού των τριάδων της προσωρινής Βάσης Γνώσεως. Στην (c) φαίνεται ότι ο χρόνος επεξεργασίας ανά τριάδα της προσωρινής Βάσης Γνώσεως είναι υψηλός για σχετικά μικρές οντολογίες, σταθεροποιείται όμως για μεγαλύτερες οντολογίες	152
28	Ένας πελάτης ο οποίος επεξεργάζεται offline δεδομένα. Παρατηρείται ότι η απόδοσή του είναι παρόμοια του εξυπηρετητή. Η απόδοση αυτή επηρεάζεται κυρίως από τον αριθμό των τριάδων στην προσωρινή Βάση Γνώσεως, ενώ η καθυστέρηση που οφείλεται σε άλλους παράγοντες μπορεί να θεωρηθεί αμελητέα	153
29	Τρεις πελάτες σε παράλληλη επεξεργασία offline πληροφορίας	154
30	Εννέα πελάτες σε παράλληλη επεξεργασία offline πληροφορίας	155

Ευρετήριο Πινάκων

1	Συνήθη namespaces στο Σημασιολογικό Ιστό	11
2	Ένας RDF γράφος ορίζεται από πέντε σύνολα	13
3	Το Λεξικό του RDF, <i>rdfV</i> είναι ένα σύνολο από URI αναφορές στο namespace RDF	15
4	Περιγραφή εννοιών στην \mathcal{AL}	16
5	Εκφραστικότητα στην Περιγραφική Λογική	18
6	Επεξήγηση της γλώσσας <i>SHOIN(D)</i>	19
7	RDFS λεξικό	27
8	OWL λεξικό	33
9	Τα tags της Dublin Core για περιγραφή ψηφιακού περιεχομένου	37

10	Δηλώσεις της SPARQL	50
11	Κατηγοριοποίηση προσεγγίσεων (πίνακας 1 από 3)	97
12	Κατηγοριοποίηση προσεγγίσεων (πίνακας 2 από 3)	98
13	Κατηγοριοποίηση προσεγγίσεων (πίνακας 3 από 3)	99
14	Αντιστοίχιση οντολογίας και Βάσης Δεδομένων με τη χρήση του R ₂ O . . .	104
15	Σύνοψη των προγραμματιστικών διεπαφών του μεσισμικού	134

Περίληψη

Το σημαντικότερο πρόβλημα στη διάθεση της πληροφορίας είναι ότι η μορφή στην οποία διατίθεται δεν είναι συνήθως ομοιογενής τόσο συντακτικά όσο και σημασιολογικά. Τεράστιος όγκος πληροφορίας υπάρχει, τόσο στο Διαδίκτυο όσο και σε λιγότερο διασυνδεδεμένες πηγές. Η ύπαρξη μεταδεδομένων τα οποία θα προσέθεταν έναν ανώτερο βαθμό σαφήνειας δε θεωρείται δεδομένη. Το ζήτημα που προκύπτει είναι το πώς θα ολοκληρωθεί ο όγκος αυτός πληροφορίας ώστε να μετατραπεί σε μια Βάση Γνώσεως από την οποία θα μπορούν να εξάγονται άμεσα αλλά και έμμεσα συμπεράσματα.

Οι υπάρχουσες τεχνολογίες διαχείρισης, επεξεργασίας και αξιοποίησης της πληροφορίας είναι συνήθως βασισμένες σε συντακτική επισημείωση. Οι σημερινές μορφές επισημείωσης υστερούν στο ότι στην πλειοψηφία τους αποτελούν μονολιθικές και εξειδικευμένες προσεγγίσεις –λύσεις κατά περίπτωση– και προσφέρουν περιορισμένες δυνατότητες εξαγωγής αποτελεσμάτων και συμπερασμάτων. Εξυπηρετούν το στόχο για τον οποίο παρήχθησαν μεν, δεν επιτρέπουν την περαιτέρω αξιοποίηση σε σημασιολογικό επίπεδο, δε. Αντιπροσωπευτικό παράδειγμα είναι το Βαθύ Διαδίκτυο [1], το οποίο συγκεντρώνει τεράστια ποσότητα πληροφορίας η οποία δύσκολα καθίσταται αξιοποιήσιμη και εκμεταλλεύσιμη από τον τελικό χρήστη του Διαδικτύου [2]. Αντίστοιχα, η φύση του προβλήματος είναι παρόμοια και στα συστήματα με επίγνωση περιβάλλοντος στα οποία ο όγκος της πληροφορίας αυξάνεται με μεγάλο ρυθμό, δυσχεραίνοντας τη διαχείρισή του, την ολοκλήρωση με άλλες πηγές πληροφορίας και κατ' επέκταση την ευφυή εκμετάλλευσή του.

Η παρούσα διατριβή εστιάζει στην έρευνα στο ευρύτερο πλαίσιο του σημασιολογικού εμπλουτισμού της πληροφορίας. Ειδικότερα, επικεντρώνεται στην έρευνα των αλλαγών και προσθηκών που χρειάζεται να γίνουν σε τεχνικό και θεωρητικό επίπεδο, ώστε η επόμενη γενιά πληροφοριακών συστημάτων να ενσωματώνει και να εκμεταλλεύεται σημασιολογική πληροφορία. Ως λύση προς την κατεύθυνση της ολοκλήρωσης της πληροφορίας μέσω του σημασιολογικού εμπλουτισμού της υιοθετείται ο Σημασιολογικός Ιστός, ο οποίος επιτρέπει την προσθήκη μεταδεδομένων τα οποία υπακούν σε κοινά πρότυπα και επιτρέπουν την εξαγωγή γνώσεως από την ήδη υπάρχουσα αλλά και το συνδυασμό της υπάρχουσας πληροφορίας με στόχο την εξαγωγή συμπερασμάτων. Η παρούσα εργασία αναλύει το πρόβλημα του σημασιολογικού εμπλουτισμού της πληροφορίας παρουσιάζοντας τις συνιστώσες που το συνθέτουν και που προσθέτουν στη δυσκολία επίλυσής του. Ειδικότερα, εξετάζονται πιο αναλυτικά δυο περιπτώσεις του ίδιου προβλήματος.

Η πρώτη περίπτωση αφορά την ολοκλήρωση Βάσεων Δεδομένων με στόχο τη

δημιουργία μιας κατανεμημένης Βάσης Γνώσεως η οποία θα ενσωματώνει ευφυή χαρακτηριστικά. Το πρόβλημα δεν είναι νέο, εξετάζεται όμως υπό το πρίσμα των τελευταίων τεχνολογικών εξελίξεων και συγκεκριμένα στο χώρο του αναπτυσσόμενου Σημαιολογικού Ιστού. Η συνεισφορά της διατριβής εδώ εντοπίζεται στην ανάλυση του χώρου της συνεργασίας των Βάσεων Δεδομένων με τις οντολογίες, τη λεπτομερή καταγραφή και κατηγοριοποίηση των προσεγγίσεων που υπάρχουν στη βιβλιογραφία και μαζί συνθέτουν την αιχμή των εξελίξεων καθώς και την αναλυτική αποτύπωση των συμπερασμάτων που προκύπτουν από την ενασχόληση με το χώρο αυτό. Σημειώνεται ότι λόγω της δυναμικής των εξελίξεων στο χώρο, η καταγραφή των προσεγγίσεων που υπάρχουν στη βιβλιογραφία συμβάλλει στην έρευνα στο χώρο καθώς παρατηρείται απουσία από αντίστοιχες μελέτες. Επιπλέον, υλοποιείται και παρουσιάζεται μια πρωτότυπη προσέγγιση στην επίλυση του προβλήματος και αναλύονται οι περιορισμοί και οι δυνατότητες των συστημάτων που αντιμετωπίζουν το πρόβλημα της συνεργασίας Βάσεων Δεδομένων και οντολογιών.

Η δεύτερη περίπτωση αφορά τα συστήματα με επίγνωση περιβάλλοντος. Η ουσιαστική διαφορά εδώ σε σχέση με την ολοκλήρωση Βάσεων Δεδομένων έγκειται στον παράγοντα του χρόνου. Τυπικά, τα συστήματα αυτά συγκεντρώνουν τεράστιο όγκο πληροφορίας ο οποίος προέρχεται από δεδομένα που συλλέγονται από αισθητήρες. Ενώ είναι εύκολη μια χαμηλού επιπέδου επεξεργασία, η οποία θα μπορεί ενδεχομένως να εξυπηρετεί έναν περιορισμένο αριθμό απαιτήσεων, το πρόβλημα εντοπίζεται επίσης στον τρόπο με τον οποίο η πληροφορία αυτή θα είναι δυνατό να ολοκληρωθεί και να αξιοποιηθεί σε σημασιολογικό επίπεδο. Για την μελέτη του προβλήματος αναπτύσσεται μια πιλοτική εφαρμογή μεσισμικού η οποία στοχεύει στην επεξεργασία και το σημασιολογικό εμπλουτισμό της πληροφορίας. Τα δεδομένα τα οποία χρησιμοποιούνται πειραματικά είναι μεταδεδωμένα πολυμεσικής πληροφορίας σε ημιδομημένη μορφή τα οποία προέρχονται από ετερογενείς κατανεμημένες πηγές. Με τη χρήση μεσισμικού τα δεδομένα αυτά εισρέουν στο σύστημα σχηματίζοντας μια σημασιολογική Βάση Γνώσεως η οποία είναι ικανή να δίνει απαντήσεις σε ερωτήματα υψηλότερου σημασιολογικού επιπέδου. Ιδιαίτερη αναφορά γίνεται στην επεξεργασία των αποτελεσμάτων σε πραγματικό χρόνο αλλά και στις εκ των υστέρων διαδικασίες που πραγματοποιούνται με στόχο τη βελτίωση της κλιμακωσιμότητας του συστήματος. Αναλύονται και τεκμηριώνονται τα πλεονεκτήματα αλλά και οι περιορισμοί που προκύπτουν από την παρουσιαζόμενη προσέγγιση στη δημιουργία μιας κατανεμημένης Βάσης Γνώσεως.

Λέξεις-κλειδιά: Σημαιολογικός Ιστός, Οντολογία, Βάση Δεδομένων, Βάση Γνώσεως, Κατανεμημένα Συστήματα, Επίγνωση Περιβάλλοντος, Μεσισμικό

Abstract

The major problem of information available is that usually, the form in which it is published lacks both syntactical and semantic homogenisation. There is a huge amount of information, both online and in less interlinked sources. The existence of metadata that would add a higher degree of clarity is not taken for granted. The question that arises is how to integrate this volume information in order to produce a Knowledge Base from which assumptions could be derived, both explicit and implicit.

The existing information management, processing and exploitation technologies are usually based on syntactic annotation. The current forms of annotation in their majority are monolithic and specialized approaches –ad hoc solutions– and offer limited potential of exporting results and conclusions. On one hand, they serve the purpose for which they were produced, but on the other hand they do not allow further exploitation in semantic level. An example is the Deep Web [1], which aggregates an enormous amount of information that is difficult to become useful and usable by the end user of the Web [2]. Accordingly, the nature of the problem is similar in concept to the context-aware systems in which the volume of information is growing at a great pace, hindering its management, integration with other sources of information and hence the intelligent exploitation.

This thesis focuses on the research in the broader context of semantic information enrichment. In particular, the research focuses on the changes and additions that need to be made at the technical and theoretical level, so that the next generation of information systems will integrate and exploit semantic information. As a solution towards the integration of information through semantic enrichment, the Semantic Web is adopted, which allows the addition of metadata which are subject to common standards and permit knowledge extraction from the existing one, and the combination of existing information in order to infer implicit knowledge, as well. The current work analyzes the problem of semantic information enrichment by presenting the components that compose it and add to the difficulty of solving it. In particular, two cases of the same problem are discussed in more detail.

The first case concerns Database integration aiming at creating a distributed Knowledge Base that will incorporate intelligent features. The problem is not new, but it is considered in the light of recent technological developments and particularly in the expanding Semantic Web. The contribution of the thesis here lies in the analysis of the domain of Database and ontology collaboration, the detailed recording and categorizing of the approaches that exist in the bibliography and together they form the state of the art of the developments, and the analytical assessing of the conclusions drawn by close-examining

this domain. It is noted that due to the dynamic developments in the domain, the recording of the approaches that exist in the bibliography contributes to the research in the area as a lack of relevant studies can be observed. In addition, an original approach to solving the problem is implemented and presented, and the limitations and possibilities of the systems facing the problem of Database and ontology collaboration are analyzed.

The second case concerns context-aware systems. The substantial difference here in comparison with the Database integration lies in the time factor. Typically, these systems collect huge amounts of information that originates from data collected by sensors. While a low-level processing which may serve a limited number of requirements is easy, the problem also lies in how this information will be integrated and used in semantic level. For the study of the problem, a pilot middleware application is developed that is designed to process and semantically enrich information. The data which is used for the experiments conducted are semistructured multimedia metadata that come from heterogeneous distributed sources. By using the middleware, these data flow in the system forming a semantic Knowledge Base that is able to provide answers to queries at a higher semantic level. Specific reference is made to the real-time processing of the results, but also to the post procedures carried out in order to improve the system's scalability. Also, the advantages and limitations resulting from the ensuing approach to creating a distributed Knowledge Base are analyzed and documented.

Keywords: Semantic Web, Ontology, Database, Knowledge Base, Distributed Systems, Context-aware, Middleware

Ευχαριστίες

Θα ήθελα καταρχήν να ευχαριστήσω τον επιβλέποντά μου καθ. Ν. Μήτρου για την ευκαιρία που μου έδωσε επιλέγοντάς με ως υποψήφιο διδάκτορα αλλά και για την καθοδήγηση και συμπαράσταση κατά όλη τη διάρκεια εκπόνησης της διατριβής μου. Ευχαριστώ ιδιαίτερα και τα υπόλοιπα μέλη της τριμελούς συμβουλευτικής μου επιτροπής, καθ. Γ. Στασινόπουλο και καθ. Ε. Καγιάφα για το χρόνο, τη διάθεσή τους αλλά και τις παρατηρήσεις τους επί της παρούσας έρευνας και του κειμένου της διατριβής. Ευχαριστώ ιδιαίτερα τους καθ. Ν. Μήτρου, καθ. Γ. Στασινόπουλο, καθ. Ε. Καγιάφα, καθ. Ε. Συκά, καθ. Ι. Βασιλείου, λέκτ. Ι. Ρουσσάκη και αν. καθ. Π. Δεμέστιχα που μου έκαναν την τιμή να συμμετέχουν στην επταμελή επιτροπή εξέτασής μου.

Θερμές ευχαριστίες στον Τάσο Ζαφειρόπουλο, το Δημήτρη Σπανό, το Γιάννη Βαλαβάνη, το Γιάννη Παπαϊωάννου, τον Παναγιώτη Μπούρο, το Νάσο Κατσαμάνη, και το Νίκο Ζαρμπούτη για τις καρποφόρες συζητήσεις που είχαμε κατά την κοινή μας πορεία στο χώρο της έρευνας.

Επίσης ένα μεγάλο ευχαριστώ στον Παναγιώτη Σταθόπουλο, τη Βούλα Ζώη και τους συνεργάτες από το ΑΙΤ για τις υποδείξεις και τα σχόλιά τους κατά τη διάρκεια της συνεργασίας μας στα πλαίσια των ερευνητικών προγραμμάτων που μαζί φέραμε εις πέρας.

Θα ήθελα επίσης να ευχαριστήσω τους ανώνυμους κριτές οι οποίοι με τις εύστοχες παρατηρήσεις και τα σχόλιά τους βοήθησαν ουσιαστικά στη βελτίωση των αποτελεσμάτων της παρούσας έρευνας. Ιδιαίτερα, ευχαριστώ τον Flavius Fräsincar για την έγκριση και επιδοκιμασία σημαντικού τμήματος της παρούσας ερευνητικής προσπάθειας.

Να εκφράσω επίσης την εκτίμησή μου για την οικονομική στήριξη από τα ερευνητικά προγράμματα της Γενικής Γραμματείας Έρευνας και Τεχνολογίας. Τέλος, ένα μεγάλο ευχαριστώ στην οικογένεια που με στήριξε ποικιλοτρόπως, αλλά και τους ανθρώπους του στενού μου κοινωνικού κύκλου για την κατανόηση και τη συμπαράστασή τους σε όλα αυτά τα χρόνια.

1 Εισαγωγή

Η παρούσα διατριβή πραγματεύεται το πρόβλημα της διαχείρισης και ολοκλήρωσης της πληροφορίας ενώ παράλληλα διατηρείται και εμπλουτίζεται η σημασιολογία των κατανεμημένων πηγών δεδομένων. Αναλύεται και εν τέλει υιοθετείται ως βάση για την παρούσα προσέγγιση ο Σημασιολογικός Ιστός, οι τεχνολογίες που παρέχει στην κατεύθυνση διαχείρισης της σημασιολογίας καθώς και το θεωρητικό υπόβαθρο στο οποίο στηρίζεται η εξέλιξη των τεχνολογιών αυτών. Αναλύονται οι δυνατότητες των σύγχρονων τεχνολογιών και παρέχεται το υπόβαθρο για την πλήρη κατανόησή τους. Εξετάζονται σημαντικά έργα, τα αποτελέσματα των οποίων συντέλεσαν στη διαμόρφωση του τοπίου με τη σημερινή του μορφή. Ερευνάται σε βάθος η βιβλιογραφία, παρουσιάζονται τα ερευνητικά αποτελέσματα μέχρι σήμερα, διερευνάται ο αντίκτυπος του Σημασιολογικού Ιστού στις τεχνολογίες Διαδικτύου και αναλύεται η συνεισφορά αλλά και οι αδυναμίες του Σημασιολογικού Ιστού στη διαχείριση και αξιοποίηση της πληροφορίας. Καταρχήν, οριοθετείται το πεδίο στο οποίο τοποθετείται η συνεισφορά της διατριβής, παρουσιάζεται το κίνητρο, το σκεπτικό και η μεθοδολογία που ακολουθήθηκε κατά τη μελέτη του προβλήματος και, τέλος, αναλύεται η συνεισφορά της διατριβής, τα αποτελέσματα και τα συμπεράσματα που προκύπτουν.

1.1 Δεδομένα και πληροφορία

Ο όρος Δεδομένα όπως και ο όρος Πληροφορία είναι έννοιες οι οποίες συχνά χρησιμοποιούνται χωρίς διάκριση. Χρειάζεται όμως να αποσαφηνιστεί η χρήση των όρων καθώς αποτελούν βασικούς δομικούς λίθους για την παρούσα διατριβή. Γενικότερα, με τον όρο πληροφορία γίνεται συνήθως αναφορά σε ειδήσεις, γεγονότα και έννοιες που αποκτώνται από την καθημερινή επικοινωνία και θεωρούνται ως αποκτηθείσα γνώση. Τα δεδομένα, αντίθετα, μπορούν να είναι μη κατάλληλα επεξεργασμένα και μη ταξινομημένα σύνολα πληροφοριών. Ένας πιο αυστηρός ορισμός για το τι είναι δεδομένα και τι είναι πληροφορία, σύμφωνα με το Oxford English Dictionary:

Δεδομένα (data) είναι μια παράσταση, όπως γράμματα, αριθμοί, σύμβολα, εικόνες κ.ά. στα οποία μπορεί να αποδοθεί κάποια σημασία (έννοια) και μπορούν να υποστούν επεξεργασία και ανάλυση. Ο όρος αναφέρεται σε μια συλλογή από περιγραφές φαινομένων τα οποία μπορούν να περιλαμβάνουν πειραματικά αποτελέσματα, περιγραφές εμπειριών ή, γενικότερα, μια βάση για περαιτέρω συλλογισμό.

Πληροφορία (information) είναι η σημασία που αποδίδεται σε ένα σύνολο από δεδομένα, τα οποία έχουν υποβληθεί σε επεξεργασία βάσει προκαθορισμένων κανόνων και κάποια χρήσιμα συμπεράσματα έχουν εξαχθεί. Με τις πληροφορίες περιορίζεται

η αβεβαιότητα για διάφορα πράγματα και βοηθείται έτσι κανείς στο να λάβει σωστές αποφάσεις. Πληροφορία μπορεί να σημαίνει μια συλλογή από σχετικά δεδομένα, γνώση σχετικά με ένα θέμα, απάντηση σε ερώτηση ή, γενικότερα, δεδομένα τα οποία αναπαριστώνται με συγκεκριμένο σχήμα και διάταξη ούτως ώστε να είναι κατανοητά από κοινό στο οποίο απευθύνονται.

Το ελάχιστο τμήμα πληροφορίας είναι το bit, το οποίο απαντά με ναι ή όχι. Αντίστοιχα, όμως, το bit μπορεί να αποτελεί δεδομένο για παράδειγμα επιτυχούς ή εσφαλμένης κατάληξης μιας πειραματικής μέτρησης. Επομένως, παρατηρείται ότι στο χώρο της επιστήμης των υπολογιστών, τα περιεχόμενα μιας Βάσης Δεδομένων μπορεί να αποτελούν δεδομένα για κάποιο χρήστη της Βάσης, πληροφορία για κάποιον άλλον, πάντα ανάλογα με την οπτική γωνία και το ενδιαφέρον του χρήστη. Για παράδειγμα, η τιμή του πεδίου “θερμοκρασία” σε μια Βάση Δεδομένων που συλλέγει μετρήσεις αισθητήρων μπορεί να αποτελεί την πληροφορία που παράγεται από τους αισθητήρες θερμοκρασίας ή μέρος των δεδομένων πάνω στα οποία θα βασιστεί μια στατιστική ανάλυση για της μεταβολές της θερμοκρασίας· η διάκριση επομένως είναι υποκειμενική.

Για το λόγο αυτό χρειάζεται να οριστεί ότι στην παρούσα εργασία ο όρος πληροφορία αναφέρεται στα περιεχόμενα των Βάσεων Δεδομένων αλλά και στα περιεχόμενα αρχείων – τυπικά σε ημιδομημένη σύνταξη – τα οποία φέρουν πληροφορία καθώς ήδη εμπεριέχουν αποθηκευμένα τα αποτελέσματα μιας πρώτης επεξεργασίας δεδομένων.

Η παρούσα εργασία πραγματεύεται τον τρόπο με τον οποίο η πληροφορία αυτή μπορεί να επεξεργαστεί επιπλέον ώστε να εμπλουτιστεί με σημασιολογία. Γίνεται λόγος για “σημασιολογικό εμπλουτισμό της πληροφορίας”, ο οποίος δεν έγκειται στην αντικατάσταση της υπάρχουσας πληροφορίας αλλά στην προσθήκη σημασιολογίας σε αυτή ούτως ώστε να καθίσταται σαφές και ξεκάθαρο το περιεχόμενό της τόσο σε συντακτικό όσο και σε εννοιολογικό επίπεδο.

1.2 Διαλειτουργικότητα και ολοκλήρωση κατανεμημένων πηγών πληροφορίας

Η παρούσα διατριβή μελετά το πρόβλημα της διαχείρισης της πληροφορίας όταν η πληροφορία βρίσκεται αποθηκευμένη σε κατανεμημένες πηγές οι οποίες είναι συνδεδεμένες μεν, διαφορετικές δε. Επομένως, το πρόβλημα της διαχείρισης της πληροφορίας στο Διαδίκτυο εστιάζεται στο ότι η ετερογένεια των πηγών αυτών πληροφορίας σε συντακτικό και σημασιολογικό επίπεδο καθιστά πρακτικά αδύνατο ένα εγχείρημα ομογενοποίησής τους. Είναι επομένως επιτακτική η ανάγκη για την διαλειτουργικότητα μεταξύ πηγών πληροφορίας σε επίπεδο υλοποίησης.

Ο ιδανικός στόχος της διαλειτουργικότητας είναι να δύνανται δυο διαφορετικοί πράκτορες (foreign agents) ανταλλάσουν επιτυχώς πληροφορία [3]. Οι υπάρχουσες προσεγγίσεις που επιτυγχάνουν τη διαλειτουργικότητα μεταξύ κατανεμημένων τμημάτων λογισμικού μπορούν να κατηγοριοποιηθούν ως εξής [4]:

- *Αντιστοίχιση* μεταξύ εννοιών της κάθε πηγής. Η κατασκευή ενός συνολικού σχήματος περιγραφής της πληροφορίας αναλαμβάνει την αντιστοίχιση μεταξύ εννοιολογικά συσχετιζόμενων πηγών πληροφορίας.
- *Μεσολάβηση για μετάφραση* των ερωτημάτων. Κατά την προσέγγιση αυτή, παρεμβάλλεται ένα ενδιάμεσο στάδιο μετάφρασης μεταξύ των ερωτημάτων από τη μια πηγή στην άλλη, το οποίο μπορεί να αποτελείται από ένα τμήμα λογισμικού, ένα σύνολο κανόνων, μια οντολογία, ή έναν πράκτορα λογισμικού.
- *Προσέγγιση βασισμένη στα ερωτήματα*, κατά την οποία ο χρήστης είναι δυνατόν να αποστέλλει ερωτήματα τα οποία θα αποτιμηθούν σε όλες τις επιμέρους πηγές πληροφορίας.

Σημειώνεται ότι οι προσεγγίσεις αυτές δεν είναι μεταξύ τους αμοιβαία αποκλειόμενες. Ένα σύστημα μπορεί να διαθέτει μεσολαβητές για την επικοινωνία μεταξύ ορισμένων πηγών αλλά ταυτόχρονα να υπακούει και σε ένα καθολικό σχήμα περιγραφής της πληροφορίας.

Η πρώτη προσέγγιση στην αντιμετώπιση του προβλήματος της διαλειτουργικότητας μεταξύ κατανεμημένων πηγών πληροφορίας συνίσταται στη χρήση προτύπων. Προσπάθειες όπως η SOAP και η WSDL αλλά και πλήθος προτύπων και μικρο-προτύπων (ενότητα 2.5) από το W3C, το OMG και άλλες κοινοπραξίες (consortia) διευκολύνουν τη διαλειτουργικότητα, την επικοινωνία και κατ' επέκταση τη διαχείριση της πληροφορίας καθώς, σύμφωνα με αυτή τη λογική, όλοι αποδέχονται και συμμορφώνονται σύμφωνα με ένα πρότυπο και ουσιαστικά είναι σαν τα δεδομένα να ανήκουν σε μια ενιαία βάση πληροφορίας.

Αξίζει να σημειωθεί ότι τα πρότυπα αυτά αναφέρονται ως *συστάσεις* του W3C (W3C recommendations) και αποτελούν στον ανοιχτό χώρο του Διαδικτύου ό,τι πιο συναφές υπάρχει στην έννοια *πρότυπο* και τον τρόπο με τον οποίο ο όρος χρησιμοποιείται σε άλλους τομείς.

Αντίθετα, η ολοκλήρωση των πηγών πληροφορίας αποτελεί προσέγγιση διαφορετικής φιλοσοφίας από τη διαλειτουργικότητα. Προκειμένου να γίνεται λόγος για ολοκλήρωση πληροφορίας, δεν αρκεί η απλή συμμόρφωση στη χρήση προτύπων για την επικοινωνία μεταξύ κατανεμημένων πηγών. Ο όρος *ολοκλήρωση* στο πεδίο της επιστήμης

υπολογιστών μπορεί να αναφέρεται σε ένα από τα εξής¹:

- *Ολοκλήρωση Συστημάτων*, η διαδικασία κατά την οποία μικρότερα τμήματα υλικού ή λογισμικού συνδυάζονται και συνεργάζονται σχηματίζοντας ένα μεγαλύτερο σύστημα το οποίο επιτυγχάνει τη λύση ενός προβλήματος.
- *Ολοκλήρωση πληροφορίας*, η διαδικασία κατά την οποία η πληροφορία που προέρχεται από διάφορες διατάξεις ή από τμήματα λογισμικού συνενώνεται ώστε να είναι δυνατή η συνολική επεξεργασία της.
- *Ολοκλήρωση επιχειρηματικών εφαρμογών* ή *ολοκλήρωση επιχειρηματικών συστημάτων* είναι η ολοκλήρωση εφαρμογών ούτως ώστε να διατίθεται ένα σύνολο από συναφείς επιχειρηματικές λύσεις.

Η παρούσα διατριβή πραγματεύεται το πρόβλημα της ολοκλήρωσης της πληροφορίας, το οποίο πηγάζει από την ύπαρξη ετερογενών πηγών αποθήκευσης και επεξεργασίας δεδομένων. Η ετερογένεια αυτή οφείλεται στην ποικιλία τρόπων με του οποίους τα δεδομένα μπορεί να οργανωθούν. Ειδικότερα, το πρόβλημα εντοπίζεται στο ότι μπορεί η πληροφορία που χρειάζεται να ολοκληρωθεί χαρακτηρίζεται από:

- Διαφορετικά μοντέλα και σχήματα. Για παράδειγμα, η πληροφορία μπορεί να είναι δομημένη σε σχεσιακό μοντέλο, σε XML ή σε Βάση Γνώσεως.
- Διαφορές λεξιλογίου. Για παράδειγμα η ιδιότητα “χρόνος” σε ένα σχήμα να εμφανίζεται ως “διάρκεια” σε ένα άλλο.
- Συντακτικές διαφορές. Μπορεί η ίδια πληροφορία να εμφανίζεται στη διαδρομή `/appearance/description` σε κάποιο XML αρχείο και στην διαδρομή `/appearance/@description` ή ακόμη και `/description/appearance` σε κάποιο άλλο.
- Σημασιολογικές διαφορές. Για παράδειγμα στο RDF μοντέλο υπάρχει η έννοια κλάσης/υποκλάσης ενώ στο σχεσιακό μοντέλο δεν υπάρχει.

Γενικά, οι υπάρχουσες προσεγγίσεις μπορούν να κατηγοριοποιηθούν σε δύο είδη. Κατά την πρώτη προσέγγιση (GAV, Global-As-View) κάθε στοιχείο του καθολικού σχήματος εκφράζεται ως ερώτηση/όψη (query/view) πάνω στα σχήματα των πηγών. Κατά τη δεύτερη (LAV, Local-As-View) κάθε στοιχείο του σχήματος μιας πηγής εκφράζεται ως ερώτηση/όψη πάνω στο καθολικό σχήμα. Γενικότερα, η GAV προσέγγιση είναι προτιμητέα όταν τα σχήματα των πηγών δεν υπόκεινται σε συχνές αλλαγές.

Στο [5] παρουσιάζεται επίσης και η GLAV (Global-Local-As-View) προσέγγιση κατά την οποία οι σχέσεις μεταξύ των σχημάτων και των πηγών ορίζονται με χρήση τόσο

¹en.wikipedia.org/wiki/Integration

LAV όσο και GAV δηλώσεων. Επίσης, μια προσέγγιση κατά την οποία υπάρχουν αντιστοιχίσεις μεταξύ των πηγών αλλά απουσιάζει ένα καθολικό σχήμα ονομάζεται P2P.

Ιδανικά, η ζητούμενη αρχιτεκτονική ενός συστήματος ολοκλήρωσης πληροφορίας έχει ως εξής:

- Πηγή Π_1 με σχήμα Σ_1
- Πηγή Π_2 με σχήμα Σ_2
- ...
- Πηγή Π_ν με σχήμα Σ_ν
- Καθολικό σχήμα Σ : το σχήμα στο οποίο υποβάλλονται οι ερωτήσεις υψηλότερου επιπέδου.

Ο στόχος της ολοκλήρωσης είναι η παροχή δυνατότητας στο χρήστη να υποβάλλει ερωτήσεις στο σχήμα Σ και να δέχεται απαντήσεις από τα $\Sigma_1, \Sigma_2, \dots, \Sigma_\nu$ χωρίς να τον ενδιαφέρει η ετερογένεια μεταξύ των σχημάτων. Το όφελος που αποκομίζεται αφορά το ότι το καθολικό σχήμα Σ είναι σε θέση να απαντά υψηλότερου επιπέδου ερωτήσεις, σε αντίθεση με τα σχήματα των πηγών.

Σε κάθε περίπτωση είναι επιθυμητή η ενιαία πρόσβαση στην πληροφορία, ανεξάρτητα από το πώς είναι αποθηκευμένη, πώς και ποιος τη διαχειρίζεται. Ειδικότερα, σε επόμενο στάδιο είναι επιθυμητή η *σημασιολογική* ολοκλήρωση της πληροφορίας. Στο κεφάλαιο 4 γίνεται αναλυτική αναφορά στο πρόβλημα της ολοκλήρωσης της πληροφορίας στην περίπτωση οντολογιών και Βάσεων Δεδομένων, ενώ στο κεφάλαιο 5 παρουσιάζεται ένα σύστημα για την ολοκλήρωση της πληροφορίας σε πραγματικό χρόνο από κατανεμημένες πηγές. Η έννοια της αντιστοίχισης η οποία συνδέεται στενά με την έννοια της ολοκλήρωσης παρουσιάζεται διεξοδικότερα στην ενότητα 4.1.1.

Παράδειγμα ολοκλήρωσης πηγών με χρήση RDF/S δίνεται στο [6] όπου παρουσιάζεται ολοκλήρωση XML πηγών σε RDF. Επίσης, στο [7] παρουσιάζεται προσέγγιση ολοκλήρωσης όπου το σχήμα των πηγών είναι σε RDF, το συνολικό σχήμα είναι επίσης σε RDFS και η γλώσσα ερωτημάτων είναι η RDQL. Επίσης, στο σύστημα Agora [8] τα σχήματα των πηγών είναι σχεσιακά ενώ το καθολικό σχήμα είναι XML, οι ερωτήσεις χρειάζεται να έχουν συγκεκριμένη μορφή ενώ η προσέγγιση που ακολουθείται είναι η LAV. Στο σύστημα Xyleme [9] ερευνάται η ολοκλήρωση κατά την οποία τα σχήματα πηγών είναι DTD ενώ επίσης, το καθολικό σχήμα είναι DTD και η προσέγγιση που ακολουθείται είναι η LAV. Το ανοιχτό ζήτημα όμως παραμένει ότι οι κανόνες αφορούν απλά συντακτική αντιστοίχιση, χωρίς να λαμβάνεται υπόψη κάποιας μορφής σημασιολογία.

Το ζήτημα ενσωμάτωσης της σημασιολογίας στην ολοκλήρωση αυτή της πληροφορίας ονομάζεται *σημασιολογική ολοκλήρωση* και ουσιαστικά το πρόβλημα

επικεντρώνεται στο πώς μπορεί να παραχθεί ένα κοινό σχήμα και επίσης οι μηχανισμοί μεταφοράς από την υπάρχουσα πληροφορία σε σημασιολογικά εμπλουτισμένη.

Χρειάζεται να οριστεί ότι ο όρος *ολοκλήρωση* (integration) της πληροφορίας διαφοροποιείται από τον όρο *ενοποίηση* (merging) της πληροφορίας καθώς η ενοποίηση, σε αντίθεση με την ολοκλήρωση υπονοεί την συνένωση της πληροφορίας σε επίπεδο υλοποίησης σε μια πηγή.

1.3 Σημασιολογική επισημείωση

Ο όρος *επισημείωση* (annotation) ουσιαστικά σημαίνει την προσθήκη μεταδεδομένων στην πληροφορία. Η πληροφορία μπορεί να είναι μορφοποιημένη με οποιοδήποτε πρότυπο, πιο σημαντική όμως είναι η συνεισφορά της επισημείωσης σε πληροφορία η οποία στην πρωτογενή της μορφή δεν είναι κατανοητή από τον άνθρωπο όπως για παράδειγμα η πολυμεσική πληροφορία (ενότητα 3.7.5). Ο προσδιορισμός “σημασιολογική” εκφράζει την προσθήκη συγκεκριμένης σημασιολογίας (semantics), ούτως ώστε τα μεταδεδομένα που προστίθενται και αντιστοιχίζονται στα πραγματικά δεδομένα να μπορούν να επεξεργαστούν από τον υπολογιστή με στόχο την εξαγωγή περαιτέρω γνώσεως. Η διαφοροποίηση της σημασιολογικής επισημείωσης από την καθιερωμένη επισημείωση έγκειται στο ότι η πρώτη περιγράφει την πληροφορία με τρόπο βασισμένο σε κοινά αποδεκτή σημασιολογία ενώ η δεύτερη με λέξεις κλειδιά ή με αναφερόμενη σε ειδικό σκοπό μορφοποίηση η οποία περιορίζει την επιπλέον επεξεργασία της.

Δεδομένης της σημασίας της επισημείωσης, κάποιος θα μπορούσε να αναρωτηθεί γιατί δεν υπάρχει πάντα. Η απάντηση είναι ένας συνδυασμός πολλών παραγόντων. Κατ’ αρχήν, είναι μια χρονοβόρα διαδικασία. Οι χρήστες δεν έχουν αρκετό χρόνο ή δεν τη θεωρούν τόσο σημαντική ώστε να αφιερώσουν χρόνο για να επισημειώσουν το ήδη υπάρχον περιεχόμενο. Από την άλλη, οι εταιρείες πιστεύουν ότι η επισημείωση είναι σπατάλη πόρων (χρόνου και χρήματος). Ο χρήστης που θα πραγματοποιήσει την επισημείωση χρειάζεται να έχει αρκετή εμπειρία στο χώρο και να τον κατανοεί πλήρως, ειδάλλως η επισημείωση ενδέχεται μάλλον να επιβαρύνει ένα σύστημα χωρίς ουσιαστικό όφελος. Η επαναχρησιμοποίηση της πληροφορίας επισημείωσης είναι συχνά δύσκολη, καθώς μπορεί να είναι πλεονάζουσα, μερική, ή αποθηκευμένη σε διάφορα μορφότυπα (formats). Το αποτέλεσμα είναι συνήθως στατικό και δεν ενσωματώνει αυτόματα αλλαγές στην πληροφορία. Επιπλέον, η πληροφορία επισημείωσης μπορεί να γίνει ξεπερασμένη πολύ σύντομα. Ειδικά το γεγονός αυτό, σύμφωνα με τους Uren et al. [10], θέτει σε κίνδυνο την εμπορική εξάπλωση του Σημασιολογικού Ιστού. Η *αυτοματοποίηση* της διαδικασίας

επισημείωσης θα αποτελέσει ένα βήμα προς την ευρύτερη εφαρμογή της.

Οι χρήστες των συστημάτων αυτόματης επισημείωσης χρειάζεται να γνωρίζουν τους περιορισμούς τους, όπως χαμένες επισημειώσεις (που αναφέρονται ως χαμηλή ανάμνηση, low recall) και ανακριβείς επισημειώσεις (που αναφέρονται ως χαμηλή ακρίβεια, low precision). Βέβαια, οργανισμοί με μεγάλες συλλογές περιεχομένου, πολλές φορές προτιμούν μια ατελή επισημείωση από καθόλου επισημείωση.

Σε κάθε περίπτωση ωστόσο, τα μεγέθη που χρειάζεται να αντισταθμιστούν όσον αφορά την πραγματοποίηση αναζητήσεων είναι η *ακρίβεια* και η *ανάμνηση*. Η ακρίβεια προσμετρά τον αριθμό της σωστής πληροφορίας ως ποσοστό της συνολικής πληροφορίας. Όσο μεγαλύτερη είναι η ακρίβεια ενός συστήματος τόσο καλύτερα μπορεί να διαβεβαιώσει το σύστημα ότι η πληροφορία που παρέχει είναι ορθή. Η ανάμνηση υπολογίζει την ορθά αναγνωρισμένη πληροφορία συγκριτικά με το σύνολο της ορθής πληροφορίας. Για παράδειγμα, σε μια μηχανή αναζήτησης στο Διαδίκτυο, υψηλή ανάμνηση σημαίνει ότι από τα αποτελέσματα δεν απουσιάζει καμία σχετική ιστοσελίδα, χωρίς να ενδιαφέρει πόσες μη σχετικές επιστράφηκαν ως αποτέλεσμα. Γενικότερα, όσο υψηλότερος είναι ο ρυθμός ανάμνησης τόσο καλύτερο είναι ένα σύστημα στο να μη χάνει τα σωστά αποτελέσματα. Η ακρίβεια και η ανάμνηση είναι αντιστρόφως ανάλογες μεταξύ τους.

1.4 Η ανάγκη για προσθήκη σημασιολογίας

Η αναζήτηση συγκεκριμένης πληροφορίας με μεθόδους που δεν ενσωματώνουν και δεν υποστηρίζουν σημασιολογία δεν είναι εξασφαλισμένο ότι αποδίδει πάντα τα αναμενόμενα. Οι σημερινές μηχανές αναζήτησης εφαρμόζουν αναζήτηση πάνω σε κατανοητή μόνο από τον άνθρωπο (human-understandable) πληροφορία και κατά κανόνα βασισμένη σε λέξεις-κλειδιά. Επιπλέον, η γλώσσα HTML παρέχει κυρίως πληροφορία για τη γραφική απεικόνιση κειμένων προοριζόμενη για την κατανόηση από ανθρώπινη λογική. Ως αποτέλεσμα, οι μηχανές αναζήτησης δεν επιστρέφουν σχεδόν ποτέ τα επιθυμητά αποτελέσματα. Αντίθετα, επιστρέφουν συνήθως τεράστιες ποσότητες πληροφορίας αμφίβολης σχέσης με το αντικείμενο αναζήτησης. Πολλά από τα δεδομένα μπορεί να είναι άχρηστα όπως επίσης και μερικά σημαντικά να μην προβληθούν.

Ο Σημασιολογικός Ιστός στοχεύει στην επίλυση του προβλήματος αυτού μέσω της περιγραφής των πόρων του Διαδικτύου. Στην πράξη, ο σκοπός είναι ο εμπλουτισμός της υπάρχουσας πληροφορίας με την προσθήκη σημασιολογικών μεταδεδομένων τα οποία συγκεκριμενοποιούν τη φύση των πόρων κατά τρόπο αντιληπτό τόσο από τον άνθρωπο όσο και από τον υπολογιστή.

Επομένως, ο στόχος του Σημασιολογικού Ιστού δεν είναι η αντικατάσταση του υπάρχοντος Ιστού αλλά η επέκτασή του με την προσθήκη σημασιολογίας. Ο στόχος αυτός υλοποιείται με την ενοποίηση των νέων τεχνολογιών με τις υπάρχουσες. Σύμφωνα με το όραμα του Σημασιολογικού Ιστού, τα δεδομένα θα είναι πιο εύκολο να τα αναζητήσει και να τα ταξινομήσει κανείς σε σύγκριση με τις σημερινές πρακτικές.

Η χρήση των οντολογιών για την περιγραφή της γνώσεως που υπάρχει στο Διαδίκτυο κρίνεται απαραίτητη λόγω του ότι η διαρκής αύξηση της ψηφιακής πληροφορίας καθιστά την εύρεση συγκεκριμένης πληροφορίας αρκετά δύσκολη. Η δυσκολία αυτή εντείνεται από το γεγονός ότι η υπάρχουσα πληροφορία σπάνια συνοδεύεται από επεξηγηματική σημασιολογία. Η προσθήκη σημασιολογικής περιγραφής στα περιεχόμενα του Διαδικτύου θα διευκολύνει την ολοκλήρωσή τους ορίζοντας σημασιολογικές σχέσεις μεταξύ των διαφόρων πόρων.

Η περιγραφή των πόρων του Διαδικτύου με οντολογίες στοχεύει ανάμεσα σε άλλα στο να τους καταστήσει κατανοητούς από μηχανές (machine-understandable) [11]. Η σημασία του να είναι μια πληροφορία κατανοητή από μηχανή είναι να μπορεί ένα αυτοματοποιημένο εργαλείο όπως μια μηχανή αναζήτησης ή ένας πράκτορας λογισμικού να εξάγει συμπεράσματα και να κατανοεί το περιεχόμενό της. Για την επίτευξη του στόχου αυτού κρίνεται απαραίτητη η αντιστοίχιση περιγραφής της σημασιολογίας με την κάθε πληροφορία (σημασιολογική επισημείωση). Μια σημείωση στην κάθε σελίδα του Διαδικτύου με περιγραφή του περιεχομένου της, ένα είδος μεταδεδομένων με τη διαφορά ότι η γλώσσα περιγραφής θα είναι ένα κοινώς αποδεκτό πρότυπο.

Συνήθως, τα μεταδεδομένα παρουσιάζονται σε ημιδομημένη σύνταξη, η οποία αφορά δεδομένα τα οποία αποθηκεύονται μαζί με τη δομή τους συνήθως σε ξεχωριστά αρχεία, σε αντίθεση με τα *δομημένα* δεδομένα στα οποία η δομή αποθηκεύεται ξεχωριστά από τα δεδομένα (πλειάδες) όπως για παράδειγμα σε ένα σχήμα Βάσης Δεδομένων. Δημοφιλής γλώσσα ημιδομημένων αρχείων αποτελεί η XML, η οποία εξυπηρετεί αρκετά ικανοποιητικά τις ανάγκες περιγραφής μεταδεδομένων. Παρέχει αυστηρά καθορισμένο συντακτικό και semantics και μπορεί εύκολα να προσαρμοστεί στις ανάγκες κάθε εφαρμογής. Τεχνολογίες έχουν αναπτυχθεί σε όλες τις γλώσσες προγραμματισμού που καθιστούν πλέον τη διαχείριση εγγράφων XML μια απλή, αυτοματοποιημένη διαδικασία.

Αξιοποιώντας τις δυνατότητες της XML, έχουν αναπτυχθεί πακέτα λογισμικού που καλύπτουν – σχεδόν – κάθε ανάγκη που αφορά έγγραφα του Διαδικτύου. Ενδεικτικά, XML αρχεία συναντάμε σε αρχεία ρυθμίσεων, σε πληροφορία που αποστέλλεται ως Υπηρεσία Διαδικτύου (Web Service) υπό τη μορφή SOAP φακέλων από πολλούς ιστοτόπους, σε περιεχόμενο που διατίθεται σε RSS μορφή για κοινοπραξία μεταξύ ιστοτόπων, ακόμη και ως βάση για την παραγωγή διαφορετικής μορφής εγγράφων, με

τη χρήση της XSL.

Στα ισχυρά χαρακτηριστικά της XML είναι ότι τα έγγραφά της είναι εύκολα στην κατανόηση τόσο από τον άνθρωπο όσο και από τον υπολογιστή. Η πρακτικά απεριόριστα επεκτάσιμη ιεραρχική της δομή καλύπτει τις περισσότερες από τις ανάγκες περιγραφής εγγράφων. Η μορφή αποθήκευσής της σε απλά αρχεία κειμένου επιτρέπει τη συνεργασία της με τα περισσότερα πρωτόκολλα επικοινωνίας και καθιστά ακόμη πιο εύκολη τη διαχείρισή της. Είναι ανεξάρτητη από το περιβάλλον στο οποίο βρίσκεται και δεν επηρεάζεται από τις τεχνολογικές εξελίξεις του.

Ωστόσο, ο τρόπος αυτός περιγραφής μεταδεδομένων παρουσιάζει αρκετά αρνητικά σημεία. Ανάμεσα σε αυτά είναι το ότι το δεντρικό μοντέλο περιγραφής δεν είναι αρκετά εκφραστικό, όπως για παράδειγμα το σχεσιακό μοντέλο που ακολουθείται από τις σχεσιακές Βάσεις Δεδομένων. Στις αδυναμίες του επιπλέον συγκαταλέγεται και το ότι παρέχει περιορισμένη περιγραφή της δομής αρχείων ενώ οι γλώσσες περιγραφής οντολογιών επιτρέπουν πιο πλούσια περιγραφή οποιασδήποτε μορφής πληροφορίας, είτε πρόκειται για το Διαδίκτυο, είτε όχι. Με το Σημασιολογικό Ιστό μπορούν να οριστούν για παράδειγμα σχέσεις μεταξύ εννοιών, όπως κλάσης/υποκλάσης, αμοιβαία αποκλειόμενες ή αντίστροφες κλάσεις και άλλες οι οποίες αναλύονται διεξοδικότερα στη συνέχεια (ενότητα 2.4). Ο ρόλος πλέον των ημιδομημένων μεταδεδομένων περιορίζεται στην παροχή απλά ενός ομοιόμορφου τρόπου σύνταξης των αρχείων αυτών [12]. Είναι βέβαιο ότι όταν ο όγκος των δεδομένων ή άλλοι παράγοντες απαιτούν ενσωματωμένη σημασιολογία, τα υπάρχοντα μεταδεδομένα χρειάζεται να αντικατασταθούν από ένα άλλο είδος αρχείου με πλουσιότερες περιγραφικές δυνατότητες και πιο συγκεκριμένη τερμινολογία.

Συγκεκριμενοποιώντας τώρα σε έγγραφα του Διαδικτύου και δεδομένου του όγκου του Διαδικτύου, έχει προκύψει η ανάγκη για την ανάπτυξη ενός πρότυπου κοινά αποδεκτού, πιο περιγραφικού από την XML. Η XML αδυνατεί να καλύψει τις ανάγκες που υπάρχουν για μια κοινή επισήμανση των εγγράφων, κάτι που θα οδηγούσε στην εύκολη αναζήτηση συγκεκριμένων πληροφοριών στο Διαδίκτυο. Γι αυτό το λόγο σχεδιάστηκε και προτάθηκε το Resource Definition Framework (RDF) το οποίο, σαν εξέλιξη της XML προσφέρει δύναμη στην κατεύθυνση αυτή. Το RDF έχει ως κύριο στόχο – αλλά δεν περιορίζεται σε αυτό – την περιγραφή πόρων του Διαδικτύου. Η συνήθης σύνταξη ενός RDF αρχείου είναι σε XML μορφή, αλλά τα σημασιολογικά χαρακτηριστικά του είναι αρκετά διαφοροποιημένα (ενότητα 1.5.1).

Έχοντας ως στόχο την περιεκτικότερη, ακριβέστερη και συνεπέστερη περιγραφή πόρων του Διαδικτύου έχει αναπτυχθεί και διαδοθεί επίσης η γλώσσα Web Ontology Language (OWL) (ενότητα 2.4). Χρησιμοποιώντας έννοιες και πρακτικές που προέρχονται

κατά κύριο λόγο από το χώρο της τεχνητής νοημοσύνης, με την OWL μπορούν να περιγραφούν με περισσότερη ακρίβεια οι διαθέσιμες πηγές πληροφορίας. Ακριβέστερα, μπορούν να δημιουργηθούν δομημένες Βάσεις Γνώσεως μέσα από τις οποίες να εξάγονται συμπεράσματα τα οποία να προκύπτουν από την αρχική πληροφορία.

Το αρχικό άρθρο του T. Berners-Lee στο *Scientific American* περιέγραφε μια αναμενόμενη εξέλιξη του υπάρχοντος Ιστού σε έναν Σημασιολογικό Ιστό [11]. Ωστόσο, αυτή η εξέλιξη ωστόσο δεν έχει επέλθει ακόμη. Πράγματι, σε ένα πιο πρόσφατο άρθρο [13], ο T. Berners-Lee δηλώνει ότι “... παρ’ όλα αυτά, αυτή η απλή ιδέα παραμένει σε μεγάλο βαθμό μη υλοποιημένη”.

1.5 Αναπαράσταση γνώσεως με RDF γράφο

Το RDF παρέχει τη δυνατότητα περιγραφής γράφων και όχι απλά δεντρικών δομών. Η δημιουργία του RDF πηγάζει από τη φιλοδοξία να δοθεί ένας κοινός παρονομαστής στα περιεχόμενα του Διαδικτύου και στην ουσία είναι η πρώτη προσπάθεια σε αυτήν την κατεύθυνση. Σημειώνεται ότι η αναφορά σε ξεχωριστούς δικτυακούς πόρους εννοεί ξεχωριστά αναγνωριστικά ταυτότητας των πόρων (URIs). Η κύρια ιδέα του RDF βασίζεται στη μοντελοποίηση κάθε δικτυακού πόρου (resource) ως προς τις σχέσεις του (property) με άλλους δικτυακούς πόρους. Στις τριάδες αυτές, συνηθίζεται ο πρώτος πόρος να ονομάζεται υποκείμενο (subject) και ο τρίτος πόρος αντικείμενο (object). Αποτέλεσμα αυτής της διαδικασίας μοντελοποίησης είναι η παραγωγή προτάσεων (statements) που περιέχουν τριάδες (triples) της μορφής:

Πρόταση RDF: (resource, property, resource)
ή (subject, property, object)

Επεκτείνοντας την έννοια του πόρου (resource), στο RDF είναι δυνατή η ύπαρξη και απλών λεκτικών εκτός από URIs ή οτιδήποτε δηλωθεί.

Πριν γίνει αναλυτικότερη η περιγραφή των RDF γράφων χρειάζεται να γίνει ειδική αναφορά στα *namespaces*, μια ιδέα ευρέως χρησιμοποιούμενη στην XML. Τα namespaces σχεδιάστηκαν ώστε να αποφεύγεται η σύγχυση ονομάτων στα XML αρχεία που μπορεί να προέρχονται από διαφορετικές πηγές. Για παράδειγμα, αν ένα XML έγγραφο περιέχει το στοιχείο `element` και χρειάζεται να διαφοροποιείται από το στοιχείο `element` ενός άλλου εγγράφου – γεγονός που συμβαίνει αρκετά συχνά στα XML αρχεία – τότε το κάθε XML χρειάζεται να έχει το δικό του namespace, ώστε το κάθε στοιχείο του να είναι μοναδικό. Το namespace δηλώνεται στην αρχή των εγγράφων ως εξής:

```
xmlns:prefix="location"
```


Πρόθεμα (prefix)	Namespace URI (location)
rdf	http://www.w3.org/1999/02/22-rdf-syntax-ns
rdfs	http://www.w3.org/2000/01/rdf-schema
xsd	http://www.w3.org/2001/XMLSchema
dc	http://purl.org/dc/elements/1.1/
daml	http://www.daml.org/2001/03/daml+oil
owl	http://www.w3.org/2002/07/owl
cc	http://creativecommons.org/ns
rdfa	http://www.w3.org/ns/rdfa
foaf	http://xmlns.com/foaf/0.1
swrl	http://www.w3.org/2003/11/swrl

Πίνακας 1: Συνήθη namespaces στο Σημασιολογικό Ιστό

Στο Σημασιολογικό Ιστό συναντούνται συχνά συγκεκριμένα namespaces, μερικά από τα οποία συγκεντρώνονται στον πίνακα 1. Τα συγκεκριμένα αυτά namespaces συνθέτουν το “λεξιλόγιο” που χρησιμοποιείται στο Σημασιολογικό Ιστό για την απόδοση μοναδικής σημασίας σε πόρους του Διαδικτύου και αναφέρονται συνήθως ως μικροπρότυπα. Εκτενέστερη αναφορά για τα μικροπρότυπα γίνεται στην ενότητα 2.5.

Στο παρόν έγγραφο, χρησιμοποιείται το namespace `ex` σε παραδείγματα, ως εξής:

```
xmlns:ex="http://www.example.org"
```

Επομένως, ένας πόρος του namespace `ex` θα μπορούσε να είναι το `ex:phd student`. Οι πόροι (resources) που γράφονται σε αυτή τη μορφή ονομάζονται *συντομευμένα URIs* ή συνήθως *Qnames*.

Πρακτικά, η δομή RDF μπορεί να υλοποιηθεί σε πολλές διατάξεις, η πιο συνηθισμένη είναι αυτή της XML, η οποία συναντάται και ως RDF/XML. Εκτός αυτής όμως συναντάται η Notation 3 (N3), η N-Triples και η Turtle η οποία αποτελεί επέκταση της N-Triples και υιοθετώντας τις πιο χρήσιμες ιδιότητες της N3 (www.dajobe.org/2004/01/turtle/). Επίσης συνηθισμένο είναι οι RDF τριάδες να διατηρούνται αποθηκευμένες σε *triplestores*, Βάσεις Δεδομένων ειδικά σχεδιασμένες για την αποθήκευση και ανάκτηση πληροφορίας που περιγράφεται με τριάδες RDF. Ο τύπος MIME του RDF, `application/rdf+xml`, καθορίζεται στο RFC 3870 (tools.ietf.org/html/rfc3870).

Παρατίθεται στη συνέχεια ένα απλό παράδειγμα που φανερώνει τις δυνατότητες του RDF σε μορφοποίηση RDF/XML, N3 και αποθήκευσης σε *triplestore*:

- Μορφοποίηση XML/RDF

```
<ex:grad_student rdf:about="2393">
  <ex:name>John</ex:name>
</ex:grad_student>
```

- Μορφοποίηση N3

```
@prefix default: <http://www.example.org#> .
default:John a default:grad_student .
```

- Αποθήκευση σε Βάση Δεδομένων χρησιμοποιώντας τη διεπαφή Jena. Ο πίνακας Jena_g1t1_stmt(Subj, Prop, Obj) αποθηκεύει μια τριάδα ως εξής

```
Subj Uv::http://www.example.org#John:
Prop Uv::http://www.w3.org/1999/02/22-rdf-syntax-ns#type:
Obj Uv::http://example.org#Postgraduate:
```

Ωστόσο, στα αρνητικά στοιχεία του RDF συγκαταλέγεται ότι η απλή περιγραφή γράφων δε μπορεί να αναπαραστήσει πολύπλοκα μοντέλα πληροφορίας. Για πιο σύνθετη περιγραφή χρειάζεται η χρήση πιο εκφραστικών γλωσσών, οι οποίες προέρχονται από την Περιγραφική Λογική.

1.5.1 Ορισμός RDF γράφων

Ένας RDF γράφος είναι παρόμοιος με έναν κατευθυνόμενο γράφο (Directed Labeled Graph). Διαφέρει όμως στο ότι το RDF επιτρέπει περισσότερες από μια ακμές μεταξύ δυο κόμβων². Ο γράφος του RDF δεν είναι ανάγκη να είναι συνδεδεμένος και επιτρέπεται η ύπαρξη κύκλων. Οι κόμβοι του RDF μπορεί να περιέχουν είτε πόρους (resources) ή λεκτικά (literals). Τα λεκτικά διαφέρουν από τους πόρους στο ότι δεν τυγχάνουν επεξεργασίας κατά την επεξεργασία του RDF. Αντίθετα, οι κόμβοι που περιέχουν πόρους, κατατάσσονται ανάλογα με το αν περιέχουν URI αναφορές ή *ανώνυμους πόρους* (blank nodes ή bNodes). Κάθε ανώνυμος πόρος θεωρείται ότι είναι μοναδικός στο γράφο. Οι υπόλοιποι πόροι ονομάζονται αντίστοιχα με μια URI αναφορά ή ένα λεκτικό. Ένας *RDF γράφος* είναι ένα σύνολο από RDF τριάδες (triples). Στον πίνακα 2 ορίζονται τα σύνολα U , B , L , P και R .

Στα σύνολα αυτά ισχύουν τα εξής:

²Ορισμός του RDF μοντέλου και της σύνταξής του. Πρόταση του W3C από τις 22 Φεβ. 1999 www.w3.org/TR/1999/REC-rdf-syntax-19990222

Σύνολο	Περιεχόμενα Συνόλου
U	URI αναφορές
B	Ανώνυμοι πόροι (Blank nodes)
L	RDF λεκτικά
P	Ιδιότητες (Properties)
R	Πόροι (Resources)

Πίνακας 2: Ένας RDF γράφος ορίζεται από πέντε σύνολα

1. $P \subset R$. Οι ιδιότητες είναι υποσύνολο των πόρων. Επομένως, είναι πόροι και οι ίδιες οι ιδιότητες.
2. $R = U \cup B$. Το σύνολο των πόρων αποτελείται από URI αναφορές και ανώνυμους πόρους.
3. $\text{rdf:type} \in P$. Η δήλωση rdf:type ανήκει στο σύνολο των ιδιοτήτων. Πρακτικά αυτό σημαίνει ότι εμφανίζεται στη μέση των τριάδων.
4. $\text{rdf:Property} \in U$. Η δήλωση rdf:Property είναι μια αναφορά URI.
5. Τα σύνολα U , B και L είναι μεταξύ τους ανά δυο αμοιβαία αποκλειόμενα.

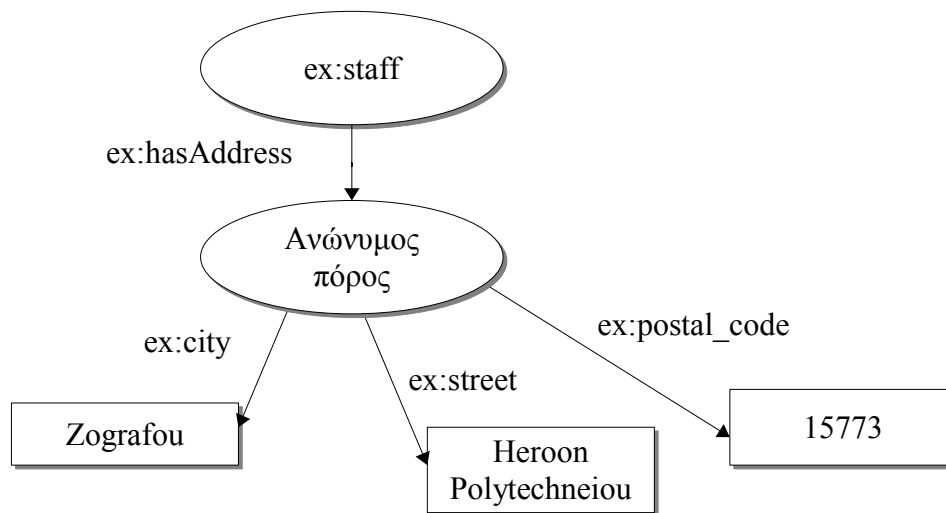
Επίσημα τώρα, κάθε τριάδα της μορφής $(v_1, v_2, v_3) \in (U \cup B) \times U \times (U \cup B \cup L)$ ονομάζεται *τριάδα RDF* (RDF triple). Ένας *RDF γράφος* είναι ένα πεπερασμένο σύνολο από τριάδες.

Οι ανώνυμοι πόροι δηλώνουν στην ουσία την ύπαρξη ενός κόμβου στον RDF γράφο χωρίς να δίνουν περισσότερη πληροφορία γι' αυτόν. Γραφικά, ένας RDF γράφος αναπαρίσταται ως εξής: Για κάθε τριάδα (a, b, c) προστίθενται στο γράφο οι κόμβοι a , c και η ακμή b ως εξής: $a \xrightarrow{b} c$. Στην εικόνα 1 έχει σχεδιαστεί ένα τμήμα ενός RDF γράφου ο οποίος περιέχει έναν ανώνυμο πόρο.

Ο πίνακας 3 συγκεντρώνει το σύνολο των δηλώσεων που θα μπορεί να περιέχει ένα RDF έγγραφο. Ως RDF έγγραφο ορίζεται ένα έγγραφο το οποίο περιέχει αναφορά στο namespace του RDF.

Οι αρχικές τριάδες της RDF, οι *αξιοματικές τριάδες* (axiomatic triples) οι οποίες είναι έγκυρες για κάθε γράφο RDF είναι οι εξής:

```
(rdf:type,      rdf:type, rdf:Property)
(rdf:subject,  rdf:type, rdf:Property)
(rdf:predicate, rdf:type, rdf:Property)
(rdf:object,   rdf:type, rdf:Property)
(rdf:first,    rdf:type, rdf:Property)
(rdf:rest,     rdf:type, rdf:Property)
```



Εικόνα 1: Παράδειγμα χρήσης ανώνυμου πόρου σε RDF γράφο

```

(rdf:value,      rdf:type, rdf:Property)
(rdf:_1,        rdf:type, rdf:Property)
(rdf:_2,        rdf:type, rdf:Property)
...
(rdf:nil,       rdf:type, rdf:List)

```

Σημειώνεται ότι η `rdf:nil` είναι μια κενή λίστα. Επιπλέον, στο RDF υπάρχει κληρονομικότητα τόσο στους πόρους όσο και στις ιδιότητές τους. Έτσι, μπορεί να οριστεί μια ιδιότητα `Class` και μια ιδιότητα `subClassOf` που να είναι *κατώτερη* ιεραρχικά από την `Class`. Αντίστοιχα, μπορεί να οριστεί για παράδειγμα ένας πόρος `Application` και ένας πόρος `Web_Application` σε πιο κάτω επίπεδο της ιεραρχίας.

Ο τελεστής `^^` χρησιμοποιείται για να δηλώσει τον τύπο ενός λεκτικού. Για παράδειγμα στην τριάδα

```
ex:John ex:hasAge "21"^^xsd:integer
```

δηλώνεται ότι ο `John` έχει ηλικία 21. Αναλυτικότερα, δηλώνεται ότι ο πόρος `ex:John` έχει την ιδιότητα `ex:hasAge` η οποία έχει τιμή "21". Η τιμή αυτή είναι μια τιμή του τύπου που δίνεται στο `xsd:integer`. Αν θεωρηθεί ότι το namespace `ex` έχει τη μορφή που δηλώθηκε προηγουμένως, τότε η πλήρης μορφή της τριάδας είναι:

```
<http://www.example.org#John>
```

Λεξικό RDF	
rdf:type	rdf:object
rdf:Property	rdf:first
rdf:XMLLiteral	rdf:rest
rdf:nil	rdf:Seq
rdf:List	rdf:Bag
rdf:Statement	rdf:Alt
rdf:subject	rdf:_1 rdf:_2 ...rdf:value
rdf:predicate	

Πίνακας 3: Το Λεξικό του RDF, $rd\mathcal{V}$ είναι ένα σύνολο από URI αναφορές στο namespace RDF

```
<http://www.example.org#hasAge>
"21"^^<http://www.w3.org/2001/XMLSchema#integer>
```

1.6 Περιγραφική λογική

Η Περιγραφική Λογική (Description Logic) προσφέρει τη γλώσσα με την οποία γίνεται δυνατή η διαχείριση ανεξάρτητων αντικειμένων, ρόλων και εννοιών [14]. Τα σημεία που χρειάζεται να τονιστούν ιδιαίτερα κατά περίπτωση είναι η γλώσσα που χρησιμοποιείται και η σημασιολογία (τα semantics) της λογικής. Οι γλώσσες χρησιμοποιούνται για την περιγραφή του κόσμου με τη χρήση φορμουλών. Οι φόρμουλες κατασκευάζονται χρησιμοποιώντας στοιχεία όπως predicate symbols, συνήθως P, S, R ή συμβολοσειρές, για παράδειγμα “Greater”, “Adequate”, μεταβλητές όπως x, y, z οι οποίες δηλώνουν αυθαίρετα αντικείμενα του υπό περιγραφή κόσμου, σταθερές όπως a, b, c οι οποίες δηλώνουν σταθερά αντικείμενα, functional symbols όπως f, g και constructors οι πιο συνήθεις από τους οποίους είναι η τομή \wedge , η ένωση \vee , η επαγωγή \rightarrow , υπάρχει \exists και για κάθε \forall . Για παράδειγμα, μια φόρμουλα Περιγραφικής Λογικής θα μπορούσε να δηλώνει ότι “υπάρχει ένας αριθμός x μεγαλύτερος του 0”: $\exists x.greaterThan(x, 0)$.

Η βασική γλώσσα της Περιγραφικής Λογικής είναι η \mathcal{AL} [15]. Στη συνέχεια περιγράφεται το πώς η \mathcal{AL} επεκτείνεται ώστε να μπορέσει να περιγράψει πιο σύνθετες έννοιες. Στην \mathcal{AL} , οι περιγραφές των εννοιών γίνονται σύμφωνα με τον κανόνα σύνταξης που παρουσιάζεται στον πίνακα 4.

Αξίζει να σημειωθεί ότι η άρνηση επιτρέπεται μόνο σε ατομικές έννοιες. Χάρην παραδείγματος σχετικά με το τι μπορεί να περιγραφεί με τη γλώσσα \mathcal{AL} , έστω μια έννοια $Person$, μια έννοια $Female$ και ένας ρόλος $hasChild$. Τότε, η έννοια όλων των οποίων όλα τα παιδιά είναι $Female$ γράφεται $Person \sqcap \forall hasChild.Female$.

Σύμβολο	Επεξήγηση
$C, D \rightarrow A$	ατομική έννοια (atomic concept)
\top	καθολική έννοια (universal concept)
\perp	κενή έννοια (bottom concept)
$\neg A$	άρνηση μόνο σε ατομικούς ρόλους (atomic negation)
$C \cap D$	τομή (intersection)
$\forall R.C$	περιορισμός τιμής (value restriction)
$\exists R.\top$	περιορισμένος υπαρξιακός περιορισμός (limited existential qualification)

Πίνακας 4: Περιγραφή εννοιών στην \mathcal{AL}

Οι έννοιες απλά αναπαριστούν τη δομή του κόσμου η οποία μπορεί και να μην έχει ύπαρξη. Προκειμένου οι έννοιες να αποκτήσουν κάποιο νόημα, χρειάζεται μια *διερμηνεία* (interpretation) I η οποία και θα τους προσδώσει νόημα. Μια διερμηνεία I είναι η δομή (Δ^I, \cdot^I) όπου Δ^I είναι το πεδίο τιμών (domain) της διερμηνείας, για παράδειγμα οι φυσικοί αριθμοί \mathbb{N} και \cdot^I μια συνάρτηση διερμηνείας η οποία αντιστοιχίζει σε κάθε ατομική έννοια A ένα σύνολο $A^I \subseteq \Delta^I$ και με κάθε ατομικό ρόλο R με μια δυαδική σχέση $R^I \subseteq \Delta^I \times \Delta^I$. Το δεύτερο argument σε ένα ζεύγος (a, b) το οποίο ανήκει σε κάποια σχέση R ονομάζεται συνήθως *κατασκευαστής ρόλου* (role-filler ή R-filler).

1.6.1 Εκφραστικότητα περιγραφικής λογικής

Προσθέτοντας περισσότερους constructors είναι δυνατή η κατασκευή πιο εκφραστικών γλωσσών. Έτσι, η εκφραστικότητα στην περιγραφή ενός μοντέλου κόσμου με την Περιγραφική Λογική ποικίλλει ανάλογα με το υποσύνολο της γλώσσας που χρησιμοποιείται. Η διαφοροποίηση αυτή προκύπτει από το γεγονός ότι δεν έχουν όλες οι γλώσσες τις ίδιες δυνατότητες περιγραφής και κατ' επέκταση τις ίδιες επιδόσεις σε αλγόριθμους που τις επεξεργάζονται. Όταν χρησιμοποιείται ένα πολύ εκφραστικό υποσύνολο της Περιγραφικής Λογικής, δεν υπάρχει εγγύηση ότι οι αλγόριθμοι τερματίζουν. Αντίστοιχα, αν το υποσύνολο που επιλέγεται είναι αρκετά περιορισμένο, οι αλγόριθμοι θα έχουν καλύτερες επιδόσεις αλλά η περιγραφή του μοντέλου θα στερείται εκφραστικότητας. Στη συνέχεια περιγράφεται το πώς μπορεί να επεκταθεί η εκφραστικότητα, ξεκινώντας από τη βασική γλώσσα \mathcal{AL} [16].

Η ένωση (conjunction) εννοιών γράφεται $C \cup D$ και μεταφράζεται ως εξής: $(C \cup D)^I = C^I \cup D^I$. Η γλώσσα που προκύπτει είναι η \mathcal{ALU} . Με την \mathcal{ALU} , μπορεί για παράδειγμα να οριστεί η έννοια ενός *Parent* ως εξής: $Parent = Father \cup Mother$.

Μπορεί επίσης να οριστεί ο πλήρης υπαρξιακός περιορισμός (full existential quantifi-

cation) γράφοντας $\exists R.C$ και μεταφράζοντας

$$(\exists R.C)^I = \{\alpha \in \Delta^I \mid \exists b. (\alpha, b) \in R^I \wedge b \in C^I\}.$$

Έτσι προκύπτει η γλώσσα $\mathcal{AL}\mathcal{E}$. Με την $\mathcal{AL}\mathcal{E}$ μπορεί να οριστεί η έννοια των ανθρώπων που έχουν ένα ανήλικο παιδί: $Person \cap \exists hasChild.Juvenile$.

Στη συνέχεια, μπορεί να οριστεί η πλήρης άρνηση (άρνηση σε αυθαίρετο αριθμό εννοιών) γράφοντας $\neg C$ και μεταφράζοντας $(\neg C)^I = \Delta^I \setminus C^I$. Έτσι προκύπτει η γλώσσα περιγραφής $\mathcal{AL}\mathcal{C}$.

Τέλος, μπορεί να περιοριστεί ο αριθμός των κατασκευαστών ρόλων (R-filler) τους οποίους μπορεί να έχει ένα άτομο (individual) σε μια σχέση R χρησιμοποιώντας αριθμητικούς περιορισμούς. Οι περιορισμοί αυτοί έχουν τη μορφή $\geq nR$ ή $\leq nR$ όπου n είναι ένας μη αρνητικός ακέραιος. Η μετάφρασή τους είναι

$$(nR)^I = \{\alpha \in \Delta^I \mid \#\{b \mid (\alpha, b) \in R^I\} \geq n\} \text{ και}$$

$$(\leq nR)^I = \{\alpha \in \Delta^I \mid \#\{b \mid (\alpha, b) \in R^I\} \leq n\} \text{ αντίστοιχα,}$$

όπου με $\#$ συμβολίζεται ο αριθμός των ατόμων (cardinality) κάθε συνόλου. Η γλώσσα που προκύπτει ονομάζεται $\mathcal{AL}\mathcal{N}$. Με την $\mathcal{AL}\mathcal{N}$ μπορεί να οριστεί για παράδειγμα η έννοια ενός πατέρα ο οποίος έχει περισσότερα από τρία παιδιά ως εξής $ParentWithManyChildren \equiv Parent \cap \geq 3hasChild$.

Συνήθως, οι αριθμητικοί αυτοί περιορισμοί αναφέρονται ως unqualified ώστε να διαχωρίζονται από τους qualified οι οποίοι υποδηλώνονται από το γράμμα \mathcal{Q} και έχουν τη σύνταξη $\geq R.C$ ή $\leq R.C$.

Χρησιμοποιώντας οποιοδήποτε συνδυασμό των παραπάνω constructors μπορούν να οριστούν πιο σύνθετες γλώσσες οι οποίες δηλώνονται με ένα όνομα της μορφής $\mathcal{AL}[\mathcal{M}][\mathcal{E}][\mathcal{C}][\mathcal{N}]$.

Από σημασιολογικής άποψης, οι γλώσσες αυτές δεν είναι διακριτές μεταξύ τους. Για παράδειγμα, χρησιμοποιώντας τους κανόνες De Morgan μπορεί κανείς να ισχυριστεί ότι $C \cup D \equiv \neg(\neg C \cap \neg D)$, γεγονός που συνεπάγεται ότι η $\mathcal{AL}\mathcal{C}$ μπορεί επίσης να έχει τον \mathcal{U} constructor. Αντίστοιχα, διαπιστώνεται ότι $\exists R.C \equiv \neg \forall R. \neg C$. Αυτό σημαίνει ότι στην $\mathcal{AL}\mathcal{C}$ μπορεί να υπάρξει full existential qualification (\mathcal{E}). Καταλήγοντας, οι $\mathcal{AL}\mathcal{C}$ και $\mathcal{AL}\mathcal{C}\mathcal{N}$ είναι οι πιο βασικές γλώσσες Περιγραφικής Λογικής. Στον πίνακα 5 περιέχεται το σύνολο των γλωσσών που μπορούν να προκύψουν.

Η σύνθεση γλωσσών παράγει υποσύνολα της Περιγραφικής Λογικής. Για παράδειγμα, ο πίνακας 6 περιγράφει τις επιτρεπόμενες δηλώσεις στο υποσύνολο $\mathcal{SHOIN}(\mathcal{D})$ της Περιγραφικής Λογικής.

Η γλώσσα OWL συνδέεται άμεσα με την Περιγραφική Λογική καθώς τα τρία υποσύνολα που χρησιμοποιούνται αντιστοιχούν σε τρεις κλιμακούμενα εκφραστικές γλώσσες.

Γλώσσα	Περιγραφή
\mathcal{AL}	Η γλώσσα αυτή αποτελεί τη βάση για άλλες γλώσσες και επιτρέπει: <ol style="list-style-type: none"> 1. Άρνηση (atomic negation). Επιτρέπει στην ουσία την προσθήκη του τελεστή άρνησης \neg σε έννοιες οι οποίες δεν εμφανίζονται στο αριστερό τμήμα των δηλώσεων 2. Τομή εννοιών (Τελεστής \sqcap) 3. Περιορισμούς 4. Περιορισμένη υπαρξιακή ποσοτικοποίηση (existential quantification)
FL^-	Ένα υποσύνολο της \mathcal{AL} το οποίο προκύπτει αφαιρώντας τη δυνατότητα atomic negation. (Τα αρχικά \mathcal{FL} προκύπτουν από την Frame Language)
FL_o	Υποσύνολο της FL^- που προκύπτει αφαιρώντας την limited existential quantification.
\mathcal{C}	Επιτρέπει την σύνθετη άρνηση εννοιών (complex concept negation). Στην ουσία, ελευθερώνεται η χρήση του τελεστή \neg .
\mathcal{S}	Είναι συντομογραφία για την \mathcal{ALC} με την προσθήκη μεταβατικών ιδιοτήτων. Μεταβατικές ιδιότητες είναι αυτές που κληρονομούνται από τις κλάσεις στις υποκλάσεις τους.
\mathcal{H}	Ιεραρχία ρόλων. Είναι στην ουσία η δήλωση <code>subPropertyOf</code> της RDF.
\mathcal{R}	Limited complex role inclusion axioms; reflexivity and irreflexivity; role disjointness
\mathcal{O}	Επιτρέπει την προσθήκη ονομάτων στις έννοιες (Nominals)
\mathcal{I}	Επιτρέπει τη δήλωση αντίστροφων ιδιοτήτων (Inverse properties)
\mathcal{N}	Προσθήκη περιορισμών στα στοιχεία ενός συνόλου (cardinality restrictions). Περιορισμοί με τους οποίους δηλώνεται ότι μια κλάση μπορεί να έχει ακριβώς έναν αριθμό υποκλάσεων ή να τίθενται όρια για τον αριθμό τους, αντίστοιχα με τις δηλώσεις <code>owl:Cardinality</code> και <code>owl:MaxCardinality</code> της OWL
\mathcal{Q}	Qualified cardinality restrictions
\mathcal{F}	Functional properties
\mathcal{E}	Full existential qualification (Existential restrictions που έχουν fillers διάφορους από το <code>owl:Thing</code>)
\mathcal{U}	Ένωση εννοιών (Concept Union)
(\mathcal{D})	Χρήση datatype properties, data values ή data types

Πίνακας 5: Εκφραστικότητα στην Περιγραφική Λογική

Υποσύνολο	Περιγραφή
\mathcal{S}	η \mathcal{ALC} με μεταβατικές ιδιότητες
\mathcal{H}	Ιεραρχία
\mathcal{O}	Ονοματισμένες έννοιες
\mathcal{I}	Αντίστροφες ιδιότητες
\mathcal{N}	Περιορισμοί στον αριθμό στοιχείων συνόλου
(\mathcal{D})	Χρήση Datatype properties, data values ή data types

Πίνακας 6: Επεξήγηση της γλώσσας $\mathcal{SHOIN}(\mathcal{D})$

1.7 Βάση Γνώσεως

Η ενότητα αυτή ορίζει την έννοια της Βάσης Γνώσεως σύμφωνα με την Περιγραφική Λογική. Οι constructors που έχουν αναφερθεί ως τώρα μπορούν να χρησιμοποιηθούν για τον ορισμό πολύπλοκων εννοιών, όχι όμως και τις σχέσεις μεταξύ των εννοιών αυτών. Οι σχέσεις αυτές αποκαλούνται τερμινολογικά αξιώματα (terminological axioms) και μπορούν να πάρουν τη μια από τις ακόλουθες δυο μορφές:

$$C \subseteq D (R \subseteq S) \text{ και}$$

$$C \equiv D (R \equiv S)$$

όπου C και D είναι έννοιες (concepts) και R και S είναι ρόλοι (roles) [17]. Τα αξιώματα του πρώτου είδους ονομάζονται *inclusions* ενώ τα αξιώματα του δεύτερου είδους ονομάζονται *equalities*. Από τη σκοπιά των semantics μια διερμηνεία I (interpretation) ικανοποιεί μια inclusion $C \subseteq D$ αν $C^I \subseteq D^I$ και ικανοποιεί μια ισότητα $C \equiv D$ αν $C^I \equiv D^I$. Τα ίδια semantics εφαρμόζονται και στους ρόλους. Αν μια I ικανοποιεί ένα αξίωμα, τότε λέμε ότι είναι ένα μοντέλο του.

Τα inclusion axioms χρησιμοποιούνται για τη δημιουργία ιεραρχιών εννοιών ή ρόλων. Επομένως, αν $C \subseteq D$ τότε η D είναι η πιο γενική έννοια και η C η πιο συγκεκριμένη. Η ιεραρχία ρόλων δηλώνεται με το γράμμα \mathcal{H} και επιφέρει μεγάλη πολυπλοκότητα στις διαδικασίες reasoning. Με την ιεράρχηση εννοιών όμως μπορεί κανείς να εκφράσει ακόμη πιο πολύπλοκες έννοιες όπως για παράδειγμα ότι το ποδόσφαιρο είναι υποκλάση των αθλημάτων $Football \subseteq Sport$ ή ότι το να έχει κάποιος ένα γιο είναι ειδικότερη περίπτωση του να έχει παιδί $hasSon \subseteq hasChild$.

Το σύνολο των inclusion και equality axioms ονομάζεται $TBox(T)$. Αν μια έννοια του $TBox$ δεν εμφανίζεται στο αριστερό μέρος των δηλώσεων, τότε η έννοια αυτή ονομάζεται

primitive. Αν μια έννοια του δεξιού μέρους των δηλώσεων περιέχει μια έννοια η οποία ορίζεται σε άλλο μέρος των δηλώσεων τότε μπορεί να αντικατασταθεί με τη δήλωσή της. Αν εφαρμοστεί η διαδικασία αυτή σε όλες της δηλώσεις ενός TBox T , τότε προκύπτει το TBox T' το οποίο θα περιέχει μόνο primitive έννοιες. Η διαδικασία αυτή ονομάζεται *επέκταση* (expansion) του T .

Τα TBoxes μπορούν να κατηγοριοποιηθούν ως *κυκλικά* ή *ακυκλικά*. Ένα TBox περιέχει κύκλο αν μια έννοια χρησιμοποιεί τον εαυτό της στον ορισμό της είτε άμεσα είτε έμμεσα. Έμμεσα, σημαίνει ότι η έννοια αυτή χρησιμοποιεί στον ορισμό της μια δεύτερη έννοια η οποία χρησιμοποιεί στον ορισμό της άμεσα ή έμμεσα την πρώτη έννοια. Ένα παράδειγμα κυκλικού ορισμού είναι ο ορισμός της δομής του δυαδικού δένδρου.

$$BinaryTree \equiv Tree \cap \leq 2hasBranch \cap hasBranch.BinaryTree$$

Λόγω ιδιαίτερων προβλημάτων που προκύπτουν στα κυκλικά TBoxes, στη μελέτη που γίνεται σήμερα στην Περιγραφική λογική για το Σημασιολογικό ιστό, τα TBoxes θεωρούνται ότι είναι ακυκλικά εκτός κι αν δηλωθεί διαφορετικά.

Έχοντας παράξει τις έννοιες που περιγράφουν τον κόσμο μέσα σε ένα TBox, μπορεί κανείς να εφαρμόσει τις έννοιες αυτές σε συγκεκριμένα αντικείμενα και να περιγράψει σχέσεις που ισχύουν για υπαρκτά αντικείμενα στον κόσμο. Οι περιγραφές αυτές περιέχονται σε μια ειδική οντότητα η οποία ονομάζεται *ABox* και συμβολίζεται με το γράμμα A . Μέσα στο ABox μπορούν να εισαχθούν individuals, στα οποία να αποδοθούν ονόματα και ιδιότητες.

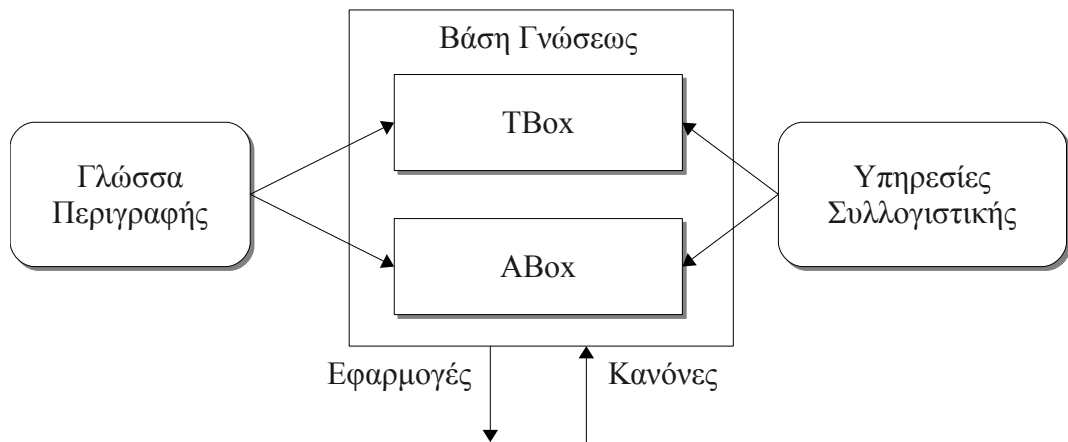
Έστω C μια έννοια, R ένας ρόλος και a, b, c τρία ονόματα individuals. Οι ισχυρισμοί που μπορούν να γίνουν μέσα στο ABox είναι δυο ειδών³, $C(a)$ (concept assertion) και $R(b, c)$ (role assertion). Ο πρώτος ισχυρισμός δηλώνει ότι το individual a ανήκει στην έννοια C . Για παράδειγμα, ο ισχυρισμός $Parent(JOHN)$ δηλώνει ότι ο John είναι πατέρας. Ο δεύτερος ισχυρισμός δηλώνει ότι το c είναι ο filler του ρόλου R του b . Για παράδειγμα, ο ισχυρισμός $hasDaughter(JOHN, JANE)$ δηλώνει ότι ο John έχει κόρη την Jane.

Η έννοια της διερμηνείας επεκτείνεται ώστε να συμπεριλαμβάνονται και τα individuals. Μια διερμηνεία ενός ατόμου αντιστοιχίζει ένα άτομο a σε ένα στοιχείο $a^I \in \Delta^I$.

Στην Περιγραφική Λογική, μια διερμηνεία υπακούει στη λεγόμενη Unique Name Assumption, προϋποθέτοντας ότι δυο διαφορετικά ονόματα από individuals αντιστοιχίζονται σε διαφορετικά στοιχεία του συνόλου Δ^I .

Ένα TBox T και ένα ABox A , ονομάζονται μαζί Βάση Γνώσεως $\mathcal{K} = \langle T, A \rangle$ [20]. Η Βάση Γνώσεως είναι στην ουσία ο πυρήνας συστημάτων αναπαράστασης της γνώσεως, η αρχιτεκτονική των οποίων παρουσιάζεται στην εικόνα 2.

³Ένα ABox αποτελείται από μια πεπερασμένη σειρά δηλώσεων αυτών των δυο τύπων [18, 19]



Εικόνα 2: Αρχιτεκτονική συστήματος αναπαράστασης γνώσεως βασισμένου σε Περιγραφική Λογική

Κατά την εξέλιξη της Περιγραφικής Λογικής έχουν προκύψει πολλά συστήματα με κύριο αντικείμενο την αποθήκευση, διαχείριση και αξιοποίηση γνώσεως. Ανάμεσα σε αυτά είναι τα Loom [21] το 1987, Back το 1988, Kris το 1991, Classic το 1991, FaCT [22] το 1998, Racer [23] το 2001, Cel το 2005, KAON2 [24] το 2004 και Pellet[25] το 2005.

1.8 Οντολογίες

Για την ύπαρξη μιας Βάσης Γνώσεως είναι απαραίτητη η συνύπαρξη μιας οντολογίας. Μια οντολογία είναι υπεύθυνη για την περιγραφή κάθε συγκεκριμένου πεδίου ενδιαφέροντος, περιγράφοντας τις έννοιες και την αλληλεπίδρασή τους. Η χρήση της μπορεί να οδηγήσει στην αυτόματη αύξηση του πεδίου γνώσεως με τη χρήση κανόνων λογικής ανάλυσης που μπορεί να εφαρμόζονται στην υπάρχουσα γνώση με σκοπό την εξαγωγή βέβαιων συμπερασμάτων [26].

Ο ορισμός έχει δοθεί από τον Gruber [27] ως “προσδιορισμός της κατανόησης κάποιου τομέα (specification of a conceptualization)”. Ο ορισμός αυτός έχει δανειστεί από την Φιλοσοφία, σύμφωνα με την οποία η “Οντολογία”⁴ είναι η συστηματική καταγραφή της “Υπαρξης”. Για ένα σύστημα λογισμικού, αυτό που “υπάρχει” είναι αυτό που μπορεί να αναπαρασταθεί και, επομένως, το σύνολο των αντικειμένων τα οποία μπορούν να

⁴Είναι γενικότερα αποδεκτό ο όρος “Οντολογία” με “Ο” κεφαλαίο να αναφέρεται στον κλάδο της Φιλοσοφίας ενώ ο όρος “οντολογία” με “ο” μικρό να αναφέρεται στις οντολογίες του Σημαιολογικού Ιστού

αναπαρασταθούν αποτελεί το πεδίο το οποίο η οντολογία πραγματεύεται (universe of discourse).

Η οντολογία είναι μια επιπλέον περιγραφή σε ένα ασαφές μοντέλο με στόχο να αποσαφηνιστεί ακόμα περισσότερο. Οι οντολογίες παρέχουν εννοιολογικά μοντέλα περιγραφής τα οποία είναι κατανοητά τόσο από τον άνθρωπο όσο και από τον υπολογιστή, υπό τη μορφή διαμοιρασμένης περιγραφής εννοιών. Με τη χρήση των οντολογιών, το περιεχόμενο μπορεί να γίνει καταλληλότερο για επεξεργασία από τον υπολογιστή σε αντίθεση με τις υπάρχουσες πρακτικές κατά τις οποίες το περιεχόμενο μορφοποιείται ώστε να είναι κατανοητό από τον άνθρωπο και να επιτευχθεί εν τέλει η ενοποίηση πολλαπλών ετερογενών πηγών δεδομένων σε σημασιολογικό επίπεδο.

Μαθηματική θεμελίωση της κατανόησης ενός τομέα (conceptualization) έχει δοθεί στο [28] όπου ορίζεται ως μια δομή $\langle D, \mathbf{R} \rangle$ όπου D είναι ένα πεδίο ορισμού (domain) και \mathbf{R} ένα σύνολο από σχετικές σχέσεις (relevant relations) στο D . Το σύνολο αυτό των σχετικών σχέσεων αποτελείται από τις *intensional* και *extensional* σχέσεις οι οποίες αναφέρονται και ως *conceptual* και *ordinary* σχέσεις αντίστοιχα. Το *domain space* αναφέρεται ως μια δομή $\langle D, W \rangle$ όπου W είναι ένα σύνολο από maximal states of affairs of such domain (επίσης ονομάζεται και *possible worlds*). Μια *conceptualization* τώρα ονομάζεται ένα σύνολο διατεταγμένων τριάδων $C = \langle D, W, R \rangle$ όπου R είναι ένα σύνολο conceptual relations στο domain space $\langle D, W \rangle$.

Με άλλα λόγια, μπορεί να οριστεί ως οντολογία ένα σύνολο από ορισμούς οι οποίοι αντιστοιχίζουν τα ονόματα των οντοτήτων (entities) στο universe of discourse με αναγνώσιμο από τον άνθρωπο κείμενο και ένα σύνολο από formal axioms τα οποία καθορίζουν τη διερμηνεία (interpretation) και την καλή χρήση των όρων (terms) αυτών [29].

Πιο γενικά, μια οντολογία μπορεί να οριστεί ως ένα μοντέλο μιας Βάσης Γνώσεως. Πρακτικά, οι γλώσσες που χρησιμοποιούνται για να σχηματίσουν οντολογίες είναι πιο κοντά στις γλώσσες Περιγραφικής Λογικής Πρώτου Βαθμού παρά στις γλώσσες που χρησιμοποιούνται για τις Βάσεις Δεδομένων. Για το λόγο αυτό, οι γλώσσες οντολογιών λέγεται ότι είναι σε *σημασιολογικό* επίπεδο ενώ οι γλώσσες των Βάσεων Δεδομένων σε *λογικό* ή *φυσικό* επίπεδο. Ο όρος οντολογία στο Σημασιολογικό Ιστό αναφέρεται τυπικά σε δυο ξεχωριστούς τρόπους μοντελοποίησης της γνώσεως ενός συστήματος (RDF/RDFS) και OWL [30].

Ανάλογα με το επίπεδο γενίκευσης διακρίνονται έξι τύποι οντολογιών:

- Η οντολογία “υψηλού” επιπέδου (top-level ontology) ορίζει πολύ γενικές έννοιες όπως χώρος, χρόνος, αντικείμενο, ιδιότητα, γεγονός, κλπ. οι οποίες ισχύουν ανεξάρτητα από οποιοδήποτε γνωστικό πεδίο.

- Η γενική οντολογία (general ontology) ορίζει ένα πλήθος εννοιών που σχετίζονται με βασική ανθρώπινη γνώση.
- Η οντολογία πεδίου (domain ontology) ορίζει έννοιες που σχετίζονται με ένα συγκεκριμένο γνωστικό πεδίο.
- Η οντολογία έργου (task ontology) ορίζει έννοιες που σχετίζονται με την εκτέλεση ενός έργου ή μιας δραστηριότητας.
- Η οντολογία εφαρμογής (application ontology) ορίζει τις απαραίτητες έννοιες για το σχεδιασμό μιας συγκεκριμένης εφαρμογής.
- Η μετα-οντολογία (meta-ontology) ή γενικής χρήσης (generic ontology) ορίζει έννοιες που είναι κοινές σε διάφορα γνωστικά πεδία και οι οποίες μπορούν να εξειδικευτούν περαιτέρω ανάλογα με τις ανάγκες του γνωστικού πεδίου.
- Επιπλέον, στην ολοκλήρωση οντολογιών οι αρχικές οντολογίες συχνά ονομάζονται οντολογίες πηγής (source ontologies) ενώ, όταν χρησιμοποιείται μια κοινή οντολογία για τη σύνδεση των αρχικών, ονομάζεται οντολογία στόχος (target ontology).

Ένα αρνητικό σημείο των οντολογιών είναι αυτό της επεκτασιμότητας. Θεωρητικά, δεν υπάρχει περιορισμός στο επίπεδο προσδιορισμού των δεδομένων. Πρακτικά όμως, η διατήρηση όγκου δεδομένων σε απλά αρχεία είναι μια πρακτική με χαμηλή αποτελεσματικότητα. Αυτό ισχύει γιατί οι οντολογίες προορίζονται κυρίως για να περιγράφουν πληροφορία, όχι να την εμπεριέχουν. Επομένως, τα υπάρχοντα δεδομένα που βρίσκονται αποθηκευμένα για παράδειγμα σε Βάσεις Δεδομένων θα πρέπει να παραμείνουν εκεί καθώς η οντολογική περιγραφή τους στοχεύει μόνο στη βελτίωση της χρηστικότητάς τους.

1.9 Δομή της διατριβής

Η παρούσα διατριβή είναι δομημένη ως εξής:

Το κεφάλαιο 1 αποτελεί μια θεωρητική εισαγωγή στο πρόβλημα που αντιμετωπίζεται με την παρούσα εργασία. Οριοθετείται το πρόβλημα της ολοκλήρωσης της πληροφορίας. Τεκμηριώνεται η βασισμένη στο Σημασιολογικό Ιστό προσέγγιση που ακολουθείται στη συνέχεια και παρέχεται ο ορισμός των εννοιών που απασχολούν την παρούσα εργασία και που συνθέτουν το “οικοσύστημα” του Σημασιολογικού Ιστού.

Στο κεφάλαιο 2 δίνεται μια πιο πρακτική αποτύπωση του περιβλήματος της παρούσας εργασίας. Περιγράφονται οι γλώσσες που έχουν αναπτυχθεί για την περιγραφή, τη διαχείριση και αξιοποίηση σημασιολογίας. Παρουσιάζονται τα λεξικά που εξυπηρετούν τους στόχους αυτούς μέσα από την περιγραφή πόρων, τον ορισμό κανόνων και την υποβολή ερωτημάτων με στόχο την ευφυή εκμετάλλευση της σημασιολογίας.

Το κεφάλαιο 3 συγκεντρώνει την ανασκόπηση του ευρύτερου πεδίου του Σημαιολογικού Ιστού. Παρουσιάζεται τμήμα του λογισμικού που έχει παραχθεί βασισμένο σε τεχνολογίες του Σημαιολογικού Ιστού, όπως περιβάλλοντα συγγραφής οντολογιών, εξυπηρετητές υπηρεσιών συλλογιστικής και έργα με σημαντικά αποτελέσματα. Επιπλέον, γίνεται αναφορά στον αντίκτυπο των τεχνολογιών του Σημαιολογικού Ιστού σε συναφείς τομείς έρευνας όπως οι Υπηρεσίες Διαδικτύου, τα συστήματα πολλαπλών πρακτόρων και τα συστήματα με επίγνωση περιβάλλοντος. Το κεφάλαιο καταλήγει με τα συμπεράσματα σχετικά με τη μελλοντική στην πορεία της έρευνας στο χώρο του Σημαιολογικού Ιστού.

Στο κεφάλαιο 4 εξετάζεται ενδελεχώς το πρόβλημα ενοποίησης του Σημαιολογικού Ιστού με τις Βάσεις Δεδομένων, οι υπάρχουσες προσεγγίσεις, τα πλεονεκτήματα και μειονεκτήματα που παρουσιάζουν, η πρακτική χρησιμότητα και αναγκαιότητα της συνεργασίας αυτής αλλά και οι περιορισμοί που τίθενται στη συνεργασία αυτή. Ουσιαστικά, το κεφάλαιο αυτό μαζί με το τελευταίο αποτελεί τη συνεισφορά της διατριβής.

Στο κεφάλαιο 5 περιγράφεται ο σχεδιασμός ενός πρωτότυπου συστήματος μεσισμικού με στόχο την ολοκλήρωση και το σημαιολογικό εμπλουτισμό της πληροφορίας που συγκεντρώνεται από ετερογενείς πηγές δεδομένων και δίκτυα αισθητήρων. Παρατίθενται οι μετρήσεις απόδοσης και εφικτότητας της προτεινόμενης προσέγγισης και το κεφάλαιο καταλήγει με τη συγκέντρωση των πορισμάτων της εργασίας αλλά και με τις κατευθύνσεις για περαιτέρω ανάπτυξη του μεσισμικού.

Τέλος, στο κεφάλαιο 6 γίνεται μια σύντομη σύνοψη της διατριβής, δίνονται τα σημαντικότερα συμπεράσματα που προκύπτουν από αυτήν, επισημαίνονται τα κυριότερα καινοτομικά της στοιχεία και δίνονται, εν κατακλείδι, κατευθύνσεις προς τις οποίες θα μπορούσε να κινηθεί η μελλοντική έρευνα στο χώρο.

2 Γλώσσες σημασιολογικής περιγραφής και αναζήτησης - Επισκόπηση

Το κεφάλαιο αυτό παρουσιάζει τις γλώσσες σημασιολογικής περιγραφής που χρησιμοποιούνται κυρίως στο Σημασιολογικό Ιστό. Παρουσιάζεται το RDFS με το οποίο υλοποιούνται οι έννοιες του RDF, η OWL, όπως επίσης και η εκφραστικότητα που παρέχεται προκειμένου να περιγραφούν οι πόροι του Διαδικτύου και να παρέχεται η δυνατότητα διαχείρισής τους και αναζήτηση σε αυτούς σε σημασιολογικό επίπεδο. Παρουσιάζονται επίσης σε συντομία οι OIL και DAML+OIL για λόγους πληρότητας. Πρόκειται για γλώσσες οι οποίες δεν αναπτύσσονται σήμερα αλλά η εξέλιξή τους οδήγησε στην σημερινή OWL. Επιπλέον, εξετάζονται οι γλώσσες ερωτημάτων μέσω των οποίων μπορούν να υποβληθούν ερωτήματα στις οντολογίες, οι διαθέσιμες γλώσσες για τη δήλωση κανόνων όπως επίσης και συγκεκριμένα “λεξικά”, ευρέως διαδεδομένα για την αναπαράσταση και ανταλλαγή σημασιολογίας στα πλαίσια περιορισμένων πεδίων ενδιαφέροντος.

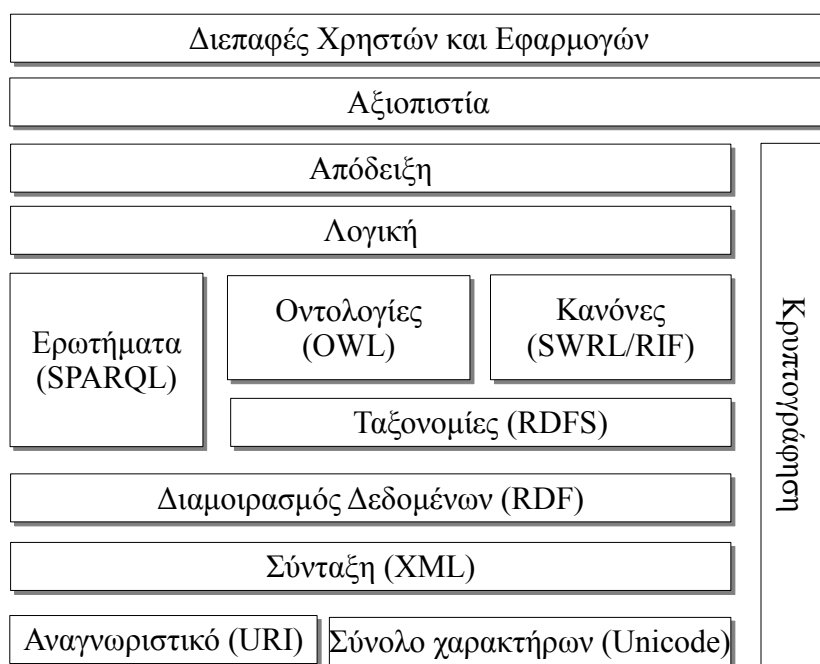
Η ιδέα του Σημασιολογικού Ιστού σε σχέση με τις τεχνολογίες που τον συνθέτουν απεικονίζεται σχηματικά στην εικόνα 3⁵. Στο κεφάλαιο αυτό εξετάζεται το κάθε τμήμα του οικοδομήματος του Σημασιολογικού Ιστού. Στο σύνολό τους οι τεχνολογίες αυτές συνθέτουν το ‘οικοσύστημα’ του Σημασιολογικού Ιστού στο οποίο μπορεί να βασιστεί μια εφαρμογή για να παράξει αξιόπιστη πληροφορία.

2.1 Το σχήμα RDFS

Ένα παράδειγμα σημασιολογικής επέκτασης του RDF είναι το RDF Schema ή RDFS. Το RDFS δεν προϋποθέτει επιπλέον συντακτικούς περιορισμούς. Είναι το σύστημα τύπων του RDF και όπως και το RDF, το RDFS βασίζεται σε κλάσεις και ιδιότητες. Αναφέρθηκε προηγουμένως (στην ενότητα 1.5) ότι το RDF είναι απλά ένα πλαίσιο περιγραφής πόρων. Ο χρήστης μπορεί να δημιουργήσει το δικό του λεξικό βασισμένο στο RDF και να περιγράψει τα δεδομένα που θέλει κατά την κρίση του. Το RDFS καλύπτει τα προβλήματα που θα προέκυπταν από μια τέτοια ελευθερία.

Το RDFS αποτελεί μια πρώτη απόπειρα σημασιολογικής επισημείωσης περιεχομένου του Διαδικτύου. Θα μπορούσαμε να πούμε ότι βάζει τάξη στη σημασιολογία του RDF ορίζοντας για παράδειγμα την κλάση του “is subclass of”. Δεν εξαρτάται πλέον από την εφαρμογή η μετάφραση μιας σχέσης αλλά αυτό γίνεται από το RDFS, δίνοντας τη δυνατότητα με το λεξικό RDFS, οι εφαρμογές να μπορούν να συνεργαστούν σε

⁵Πηγή: en.wikipedia.org/wiki/Semantic_Web_Stack



Εικόνα 3: Στρώματα Σημασιολογικού Ιστού

κοινή βάση. Στην ουσία, το RDF είναι απλά ένα πλαίσιο αναπαράστασης της γνώσεως και RDFS είναι το πρότυπο που περιγράφει πώς μπορεί να χρησιμοποιηθεί το RDF προκειμένου να περιγραφεί περιεχόμενο του Διαδικτύου. Τα πάντα στο RDFS είναι URIs, μια αντίληψη που διατηρήθηκε σε όλη την πορεία ανάπτυξης του Σημασιολογικού Ιστού.

Το λεξικό που χρησιμοποιείται για την περιγραφή πόρων του Διαδικτύου με το RDFS συγκεντρώνεται στον πίνακα 7. Ο καθορισμός της ορολογίας του RDFS βασίζεται πάνω στο RDF. Γι' αυτό και συνήθως αναφέρεται ως RDF(S).

2.2 Η γλώσσα OIL

Ακρωνύμιο που σημαίνει “Ontology Inference Layer”. Τα semantics της OIL [31–33] βασίζονται στην Περιγραφική Λογική αλλά η σύνταξή της στο RDF. Κατά το σχεδιασμό της OIL ένας από τους βασικούς στόχους ήταν η ενοποίηση με τις εφαρμογές RDF. Επομένως, τα περισσότερα RDF σχήματα είναι έγκυρες OIL οντολογίες και οι περισσότερες OIL οντολογίες μπορούν να αναγνωστούν από επεξεργαστές RDF. Σε αντίθεση με το RDF, η OIL έχει καθορισμένη σημασιολογία.

Δήλωση	Επεξήγηση
<code>rdfs:domain</code>	Δηλώνει το <code>domain</code> μιας ιδιότητας P , δηλαδή την κλάση όλων των πόρων που μπορεί να εμφανιστούν ως S σε μια τριάδα (S, P, O)
<code>rdfs:range</code>	Δηλώνει το <code>range</code> μιας ιδιότητας P , δηλαδή την κλάση όλων των πόρων που μπορεί να εμφανιστούν ως O σε μια τριάδα (S, P, O)
<code>rdfs:Resource</code>	Η κλάση όλων των πόρων
<code>rdfs:Literal</code>	Η κλάση για όλα τα λεκτικά (strings)
<code>rdfs:Datatype</code>	Η κλάση των τύπων δεδομένων (Datatypes). Το <code>rdfs:Datatype</code> είναι ταυτόχρονα και <code>instance</code> αλλά και υποκλάση του <code>rdfs:Class</code> . Κάθε <code>instance</code> του <code>rdfs:Datatype</code> είναι και υποκλάση του <code>rdfs:Literal</code>
<code>rdfs:Class</code>	Η κλάση όλων των κλάσεων
<code>rdfs:subClassOf</code>	Αντιστοιχίζει μια κλάση σε μια από τις υπερκλάσεις της. Σημειώνεται ότι μια κλάση μπορεί να έχει περισσότερες από μια υπερκλάσεις
<code>rdfs:subPropertyOf</code>	Αντιστοιχίζει μια ιδιότητα σε μια από τις υπεριδιότητές της. Μια ιδιότητα μπορεί επίσης να έχει περισσότερες από μια υπεριδιότητες
<code>rdfs:member</code>	Είναι υπεριδιότητα σε όλες τις ιδιότητες που δηλώνουν ότι ένας πόρος ανήκει σε μια κλάση που μπορεί να περιέχει αντικείμενα (container)
<code>rdfs:Container</code>	Υπερκλάση όλων των κλάσεων που μπορούν να περιέχουν αντικείμενα, όπως και οι <code>rdf:Bag</code> , <code>rdf:Seq</code> και <code>rdf:Alt</code>
<code>rdfs:Container- MembershipProperty</code>	Είναι υποκλάση του <code>rdf:Property</code> . Είναι ιδιότητα που χρησιμεύει στη δήλωση του ότι ένας πόρος είναι μέλος ενός container. Κάθε <code>instance</code> είναι υποιδιότητα του <code>rdfs:member</code>

Πίνακας 7: RDFS λεξικό

Η OIL αποτελείται από επίπεδα, καθένα από τα οποία προσθέτει επιπλέον λειτουργικότητα στο προηγούμενο. Στη βάση της OIL είναι το RDFS το οποίο θεωρείται ως Core OIL. Η OIL διαχωρίστηκε σε τρία υποσύνολα της γλώσσας ανάλογα με την επιθυμητή εκφραστικότητα του κάθε υποσυνόλου. Η Standard OIL απλά προσθέτει μερικά στοιχεία στην Core OIL. Η Instance OIL προσθέτει τη δυνατότητα στη δημιουργία στιγμιοτύπων του μοντέλου. Στην πράξη επιτρέπει τη χρήση της RDF για περιγραφή instances. Τρίτο και τελευταίο υποσύνολο αποτελεί η Heavy OIL η οποία συμπεριλαμβάνει τις πλήρεις δυνατότητες της γλώσσας. Τα θετικά στοιχεία της OIL πηγάζουν από τη βάση της στην Περιγραφική Λογική. Για παράδειγμα, αν δυο οντολογίες χρησιμοποιούν την ίδια τερμινολογία στους ορισμούς τους, τότε μπορεί να εξαχθεί η ιεραρχία κλάσεων-υποκλάσεων του συνδυασμού των οντολογιών. Υπάρχουν όμως αρνητικά σημεία καθώς είναι πιθανό να προκύψουν λογικές ασυμβατότητες μεταξύ των εννοιών για την αντιμετώπιση των οποίων δε δίνονται οδηγίες. Για παράδειγμα, έστω ότι σε μια κλάση *Student* ορίζεται περιορισμός `oil:maxCardinality` σε μια ιδιότητα – οι ιδιότητες στην ορολογία της OIL ονομάζονται *slots* – *isStudentOf* με την τιμή ένα. Στην περίπτωση αυτή, μπορεί να προκύψουν ασυμβατότητες από διαφορετικές οντολογίες προέρχεται η πληροφορία ότι ο ίδιος φοιτητής είναι φοιτητής σε διαφορετικές σχολές. Για τους λόγους αυτούς, η OIL χρησιμοποιήθηκε ως η πρώτη απόπειρα προσθήκης σημασιολογίας σε RDF γράφους αλλά εξελίχθηκε στην DAML+OIL η οποία παρουσιάζει καλύτερα χαρακτηριστικά όπως διαπιστώνεται στην ενότητα 2.3.

2.3 Η γλώσσα DAML+OIL

Η US Defence Advanced Research Project Agency (DARPA) σε συνεργασία με την EU committee on agent markup languages δημιούργησαν την DAML+OIL [34]. Η DAML+OIL είναι επέκταση του RDF Schema. Οι πρώτες εκδόσεις της DAML+OIL ονομαζόταν DAML-ONT αλλά το όνομά της άλλαξε σε μια προσπάθεια να προσελκύσει τον κόσμο που εργαζόταν σε OIL. Όπως και η OIL, η DAML+OIL βασίζεται σε Περιγραφική Λογική.

Οι περιγραφικές δυνατότητες της DAML+OIL είναι κατά πολύ παρόμοιες με αυτές της OIL. Η DAML+OIL, όπως και η OIL επιτρέπει την τομή, ένωση και άρνηση μεταξύ εκφράσεων κλάσεων. Οι ιδιότητες `daml:intersectionOf`, `daml:unionOf` και `daml:complementOf` είναι αντίστοιχες με τις ιδιότητες `oil:AND`, `oil:OR` και `oil:NOT` της OIL. Όπως και η OIL, η DAML+OIL έχει τη δυνατότητα να θέτει περιορισμούς σε αριθμούς συνόλων (cardinality restrictions). Συγκεκριμένα, επιτρέπει τις δηλώσεις `daml:minCardinality` και `daml:maxCardinality`. Η ιδιότητα

`daml:hasClass` είναι αντίστοιχη με αυτή της `oil:hasClass` και δηλώνει ότι τουλάχιστο μια τιμή του slot πρέπει να αναφέρεται στην κλάση που δηλώνεται, είναι ο τελεστής \exists της λογικής. Επιπλέον, η δήλωση `daml:toClass` σημαίνει ότι όλες οι τιμές του slot πρέπει να αναφέρονται στην κλάση που δηλώνεται και είναι αντίστοιχη στην `oil:ValueType` – τελεστής \forall . Μια ακόμη ομοιότητα είναι στις ιδιότητες `daml:inverseOf` και `daml:TransitiveProperty` οι οποίες είναι αντίστοιχες με αυτές της OIL.

Η DAML+OIL έχει μερικές διαφορές από την OIL. Η DAML περιέχει την ιδιότητα `daml:hasValue` η οποία δεν έχει αντίστοιχη στην OIL. Η ιδιότητα αυτή χρησιμοποιείται για να δηλώσει ότι τουλάχιστο μια από τις τιμές του slot πρέπει να ισούται με τη δηλωθείσα τιμή. Η `daml:UniqueProperty` δηλώνει ότι μια ιδιότητα μπορεί να έχει μόνο μια τιμή ανά instance. Η `daml:UnambiguousProperty` υπονοεί ότι μια τιμή μπορεί να ανήκει μόνο σε ένα instance. Τέλος, επιτρέπει και τη χρήση της `daml:Imports` για την εισαγωγή οντολογιών. Η δήλωση αυτή εισάγει όλες τις δηλώσεις μιας άλλης οντολογίας στην παρούσα. Οι τύποι της XSD όπως `integers`, `strings` κοκ. υποστηρίζονται για πρώτη φορά στην DAML+OIL.

2.4 Η γλώσσα OWL

Η OWL έχει σχεδιαστεί για τις εφαρμογές που πρέπει να επεξεργαστούν το περιεχόμενο των πληροφοριών αντί απλά να παρουσιάσουν τις πληροφορίες στους ανθρώπους. Σύμφωνα με το W3C, ο σκοπός της OWL είναι να παρασχεθεί ένα τυποποιημένο σχήμα που είναι συμβατό με την αρχιτεκτονική του World Wide Web και του Semantic Web. Η τυποποίηση των οντολογιών σε γλώσσα OWL θα κάνει τα δεδομένα στο Web περισσότερο επεξεργάσιμα από μηχανές και επαναχρησιμοποιήσιμα στις εφαρμογές [35].

Η δημιουργία της OWL [36, 37] ουσιαστικά ξεκινά με την επίσημη έναρξη του έργου DAML. Η DAML σε συνδυασμό με την OIL παράγαγε την DAML+OIL η οποία είναι μια επέκταση του RDFS, και η επέκταση της DAML+OIL είναι η OWL η οποία σήμερα φαίνεται ως η πιο συμπαγής λύση. Θα ήταν ιδανικό αν η OWL ήταν συμβατή με το RDF, αν ήταν η OWL μια απλή επέκταση του RDFS, δυστυχώς όμως τα πράγματα περιπλέκονται [38] καθώς στην OWL ο κόσμος θεωρείται ως ένα σύνολο κλάσεων, ιδιοτήτων και ατόμων.

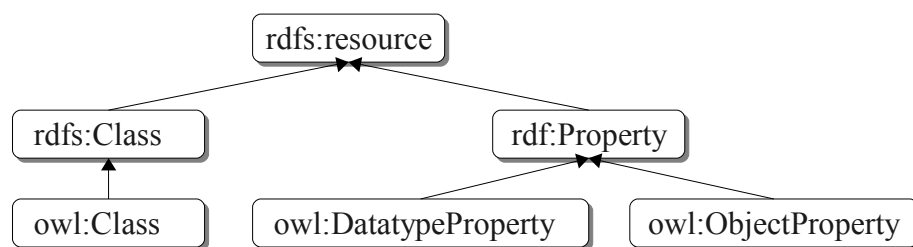
Σύμφωνα με τη θεωρητική σημασιολογία του μοντέλου (model-theoretic semantics) [39] της OWL, ένα λεξικό V της OWL μπορεί να οριστεί ως ένα σύνολο λεκτικών V_L και επτά σύνολα από αναφορές URI: $V_C, V_D, V_I, V_{DP}, V_{IP}, V_{AP}$ και V_O τα οποία δηλώνουν

αντίστοιχα τα ονόματα των κλάσεων, των ιδιοτήτων τύπου δεδομένων (datatype), των ατόμων, των data-valued ιδιοτήτων, των individual-valued ιδιοτήτων, των ιδιοτήτων επισημείωσης (annotation) και των ονομάτων των οντολογιών.

Σημαντικό είναι να αναφερθεί ότι στην OWL ισχύει η θεώρηση ανοιχτού κόσμου. Δηλαδή, οι περιγραφές των πόρων δεν είναι περιορισμένες σε ένα ενιαίο αρχείο ή πεδίο. Ενώ η κλάση C_1 μπορεί να καθοριστεί αρχικά στην οντολογία O_1 , μπορεί να επεκταθεί σε άλλες οντολογίες⁶. Οι συνέπειες αυτών των πρόσθετων προτάσεων για τη C_1 είναι μονοτονικές. Οι νέες πληροφορίες δεν μπορούν να αποσύρουν τις προηγούμενες πληροφορίες. Οι νέες πληροφορίες μπορούν να είναι αντιφατικές, αλλά τα γεγονότα και οι συνεπαγωγές μπορούν μόνο να προστεθούν, ποτέ να διαγραφούν. Η δυνατότητα τέτοιων αντιφάσεων είναι κάτι που ο σχεδιαστής μιας οντολογίας πρέπει να λάβει υπόψη.

Στη γλώσσα OWL συναντώνται τρεις εκδοχές: Lite, DL και Full οι οποίες είναι στην ουσία και οι τρεις πολύ περιγραφικές γλώσσες. Η OWL βασίζεται στην Περιγραφική Λογική η οποία με τη σειρά της βασίζεται στον Κατηγορικό Λογισμό και αποτελεί τη σημερινή πρόταση (recommendation) του W3C η οποία έχει διαδεχθεί την DAML+OIL.

Σημειώνεται ότι σε μια οντολογία OWL χρησιμοποιούνται επίσης και δηλώσεις που προέρχονται από το RDF(S), όπως για παράδειγμα οι `rdfs:subClassOf`, `rdfs:range` και `rdf:resource`. Η OWL χρησιμοποιεί το RDFS και βασίζεται σε αυτό προκειμένου να μοντελοποιήσει τις δομικές έννοιές της. Στην εικόνα 4 περιγράφεται η σχέση κλάσης-υποκλάσης μεταξύ του RDF και της OWL. Η εικόνα δείχνει τον τρόπο με τον οποίο η OWL βασίζεται στις προγενέστερες γλώσσες προκειμένου να ορίσει το λεξικό της.



Εικόνα 4: Σχέση κλάσης υποκλάσης μεταξύ της OWL και του RDFS

Αναλυτικά τώρα, το λεξικό της OWL συμπεριλαμβάνει τα εξής στοιχεία, όπως απεικονίζεται στον πίνακα 8.

⁶Περισσότερα για τη θεώρηση ανοιχτού κόσμου στην ενότητα 4.1.5.

Δηλώσεις επικεφαλίδας

<code>owl:Ontology</code>	Συνήθως επικεφαλίδα των OWL εγγράφων και περιέχει δηλώσεις για τη διαχείριση της οντολογίας, όπως σχόλια, αριθμός έκδοσης και αναφορές σε άλλες οντολογίες
<code>owl:PriorVersion</code>	Μια αναφορά στην προηγούμενη έκδοση της οντολογίας
<code>owl:versionInfo</code>	Περιέχει ένα string με πληροφορίες για την παρούσα έκδοση
<code>owl:backwardCompatibleWith</code>	Περιέχει μια αναφορά στην προηγούμενη έκδοση της οντολογίας η οποία είναι συμβατή με την παρούσα
<code>owl:incompatibleWith</code>	Το αντίθετο της <code>backwardCompatibleWith</code> . Περιέχει μια αναφορά σε μια οντολογία η οποία δεν είναι συμβατή με την παρούσα. Χρησιμεύει στους προγραμματιστές ώστε να γνωρίζουν ότι δε μπορούν για παράδειγμα να αναβαθμίσουν μια αναπτυσσόμενη οντολογία χωρίς να ελέγξουν τι μετατροπές χρειάζονται
<code>owl:Imports</code>	Δίνει λίστα με άλλες οντολογίες των οποίων τα περιεχόμενα θεωρούνται ότι είναι μέρος της παρούσας οντολογίας. Η <code>Imports</code> είναι μεταβατική ιδιότητα, αν δηλαδή η οντολογία <i>A</i> εσωκλείει τη <i>B</i> και η <i>B</i> την <i>C</i> , τότε και η <i>A</i> εσωκλείει την <i>C</i> . Η δήλωση αυτή συμπεριλήφθηκε στην OWL με στόχο η τοποθεσία της κάθε οντολογίας να είναι και το διακριτικό της χαρακτηριστικό

Κλάσεις

<code>owl:Class</code>	Οι κλάσεις στην OWL δηλώνονται με τη χρήση της <code>owl:Class</code>
<code>owl:disjointWith</code>	Δήλωση ότι μια κλάση δεν έχει κοινά στοιχεία με μια άλλη
<code>owl:equivalentClass</code>	Δήλωση ισοδυναμίας μεταξύ δυο κλάσεων

Ιδιότητες

2 Γλώσσες σημασιολογικής περιγραφής και αναζήτησης - Επισκόπηση

<code>owl:ObjectProperty</code>	Ορίζει μια ιδιότητα αντικείμενου η οποία αντιστοιχίζει ένα αντικείμενο σε ένα άλλο
<code>owl:DatatypeProperty</code>	Ορίζει μια ιδιότητα τύπου δεδομένων η οποία αντιστοιχίζει ένα αντικείμενο σε μια τιμή δεδομένων
<code>owl:inverseOf</code>	Ορισμός αντίστροφων ιδιοτήτων, όπως για παράδειγμα οι “isTaughtBy” και “teaches”
<hr/>	
Περιορισμοί Ιδιοτήτων	
<code>owl:Restriction</code>	Περιέχει τους περιορισμούς των ιδιοτήτων. Αναλυτικότερα, περιέχει ένα αντικείμενο <code>onProperty</code> και μία ή περισσότερες δηλώσεις περιορισμών
<code>owl:onProperty</code>	Δηλώνει την ιδιότητα στην οποία εφαρμόζονται οι περιορισμοί
<code>owl:allValuesFrom</code>	Δηλώνει την κλάση των δυνατών τιμών που μπορεί να πάρει η ιδιότητα που δηλώνεται στην <code>onProperty</code> . Με άλλα λόγια, όλες οι τιμές της ιδιότητας που δηλώνεται στην <code>onProperty</code> πρέπει να είναι από την κλάση που δηλώνεται στην <code>allValuesFrom</code> . Σε περιγραφική λογική, αυτή είναι η δήλωση <i>universal quantification</i> (\forall)
<code>owl:someValuesFrom</code>	Δηλώνει ότι υπάρχει ένα instance της κλάσης που δηλώνεται το οποίο να ικανοποιεί την ιδιότητα. Σε όρους περιγραφικής λογικής, δηλώνεται η <i>existential quantification</i> (\exists)
<code>owl:hasValue</code>	Δηλώνει μια συγκεκριμένη τιμή που πρέπει να έχει η ιδιότητα που δηλώνεται με την <code>onProperty</code>
<hr/>	
Ειδικές ιδιότητες	
<code>owl:TransitiveProperty</code>	Δηλώνει μια μεταβατική ιδιότητα, όπως “isTallerThan” ή “isOlderThan”
<code>owl:SymmetricProperty</code>	Δηλώνει μια συμμετρική ιδιότητα όπως “isSiblingOf”
<code>owl:FunctionalProperty</code>	Δηλώνει μια ιδιότητα η οποία έχει το πολύ μια τιμή για κάθε αντικείμενο, όπως “height” ή “directSupervisor”

2 Γλώσσες σημασιολογικής περιγραφής και αναζήτησης - Επισκόπηση

owl:InverseFunctionalProperty	Δηλώνεται μια ιδιότητα για την οποία δυο διαφορετικά αντικείμενα δε μπορούν να έχουν την ίδια τιμή, όπως για παράδειγμα η “isTheID-numberFor” η οποία δε μπορεί να έχει την ίδια τιμή για δυο διαφορετικά αντικείμενα καθώς ο αριθμός ταυτότητας είναι μοναδικός για τον καθένα
Boolean συνδυασμοί	
owl:ComplementOf	Δηλώνει ότι δυο κλάσεις δεν έχουν κοινά στοιχεία μεταξύ τους. Σημειώνεται ότι η ίδια δήλωση θα μπορούσαμε να επιτευχθεί και με χρήση της disjointWith
owl:unionOf	Ένωση κλάσεων
owl:intersectionOf	Τομή κλάσεων. Σημειώνεται ότι τόσο στην ένωση όσο και στην τομή, η κλάση που προκύπτει δεν είναι υποκλάση του συνόλου που προκύπτει αλλά ισοδύναμη (equivalent) του αποτελέσματος
Απαρίθμηση	
owl:oneOf	Χρησιμοποιείται για να δηλώσει μια κλάση απαριθμώντας τα στοιχεία που την αποτελούν
Instances⁷	
owl:AllDifferent	Το στοιχείο που περιέχει τα διακριτά instances
owl:distinctMembers	Περιέχει μια λίστα με διακριτά instances
owl:differentFrom	Μπορεί να χρησιμοποιηθεί για να δηλώσει ότι ένα instance είναι διαφορετικό από ένα άλλο
Προκαθορισμένες κλάσεις	
owl:Thing	Τα πάντα στην OWL είναι υποκλάση του Thing
owl:Nothing	Τα πάντα στην OWL είναι υπερκλάση του Nothing

Πίνακας 8: OWL λεξικό

⁷Η OWL περιέχει ένα μηχανισμό με τον οποίο μπορεί να δηλωθεί ότι κάποια instances είναι διαφορετικά το ένα από το άλλο. Ο μηχανισμός αυτός χρειάζεται γιατί η OWL δεν κάνει χρήση του Unique Name Assumption, γεγονός που σημαίνει ότι στην OWL δυο instances μπορούν να έχουν το ίδιο όνομα. Ωστόσο, για τη σωστή αντιμετώπιση από εξυπηρετητές υπηρεσιών συλλογιστικής χρειάζεται να οριστεί ότι πρόκειται για διαφορετικά στοιχεία

Η *OWL Full* περιέχει στην ουσία ολόκληρη τη γλώσσα OWL και χρησιμοποιεί όλα τα δομικά συστατικά της. Επιτρέπει επιπλέον το συνδυασμό των συστατικών αυτών με οποιοδήποτε αυθαίρετο τρόπο με τα στοιχεία της RDF και της RDF Schema. Στις δυνατότητές της περιέχεται επίσης ακόμη και η τροποποίηση των προκαθορισμένων δομικών της συστατικών εφαρμόζοντας το ένα στο άλλο. Για παράδειγμα, στην OWL Full μπορεί να οριστεί περιορισμός στον αριθμό στοιχείων του συνόλου των κλάσεων στην κλάση που περιέχει όλες τις κλάσεις (`owl:Class`). Ουσιαστικά με αυτόν τον τρόπο μπορεί να περιοριστεί ο αριθμός των κλάσεων που εμφανίζονται σε κάθε οντολογία.

Στα θετικά στοιχεία της OWL Full περιλαμβάνεται το ότι είναι πλήρως αναδρομικά συμβατή με τις RDF, RDFS, OWL Lite και OWL DL. Οποιοδήποτε έγκυρο έγγραφο στις γλώσσες αυτές είναι επίσης και έγκυρο στην OWL Full και κάθε πόρισμα που προκύπτει από αυτά είναι επίσης έγκυρο και στην OWL Full. Το βασικό μειονέκτημα της OWL Full είναι ότι σαν γλώσσα είναι τόσο ισχυρή έτσι ώστε να μην είναι δυνατή η εξασφάλιση υποστήριξης από υπηρεσίες συλλογιστικής (reasoning). Η ελευθερία στην περιγραφή της OWL Full δεν εξασφαλίζει ότι οι διαδικασίες αυτές μπορεί να λήγουν σε πεπερασμένο – πόσο μάλλον σε πολυωνυμικό – χρόνο.

Η *OWL DL* είναι παρόμοια με τη γλώσσα περιγραφής γνώσεως *SHOIN(D+)*. Η διαφορά είναι ότι η OWL DL παρέχει και ιδιότητες επισημείωσης (annotation properties) ενώ η *SHOIN(D+)* όχι. Η OWL DL έχει σχεδιαστεί ούτως ώστε να είναι περιγραφική σαν γλώσσα στο σημείο που να υπάρχουν εξυπηρετητές υπηρεσιών συλλογιστικής (reasoners) των οποίων οι αλγόριθμοι να τερματίζουν σε πολυωνυμικό χρόνο. Τα χαρακτηριστικά της OWL DL από τη σκοπιά της επιστήμης των υπολογιστών είναι τόσο εκφραστικά ώστε να βελτιστοποιείται η υπολογισσιμότητα των εννοιών της διατηρώντας βέλτιστη απόδοση σε θέματα πολυπλοκότητας. Η DL παρέχει τη μέγιστη εκφραστικότητα ενώ η διαδικασίες συλλογιστικής είναι πάντα πεπερασμένες και πάντα τερματίζουν σε πολυωνυμικό χρόνο.

Όσον αφορά το λεξικό που χρησιμοποιείται στην OWL DL, είναι προφανώς περιορισμένο σε σχέση με την πλήρη εκφραστικότητα της OWL Full. Έχουν τεθεί οι εξής περιορισμοί:

1. Ένας πόρος μπορεί να είναι μόνο κλάση, τύπος δεδομένων, ιδιότητα τύπου δεδομένων, ιδιότητα αντικειμένου, άτομο, τιμή δεδομένων ή τμήμα του λεξικού, τίποτε άλλο. Αυτό σημαίνει ότι, για παράδειγμα μια κλάση δε μπορεί να είναι ταυτόχρονα και άτομο ή ότι μια ιδιότητα δε μπορεί να έχει ταυτόχρονα μερικές

τιμές από έναν τύπο δεδομένων και μερικές από μια κλάση (αυτό θα την καθιστούσε ταυτόχρονα ιδιότητα τύπου δεδομένων και ιδιότητα αντικειμένου).

Πρέπει να δηλώνεται ρητά ο τύπος του κάθε πόρου (resource). Για παράδειγμα, έστω το τμήμα OWL:

```
<owl:Class rdf:ID="Class1">
  <rdfs:subClassOf rdf:about=" Class2" />
</owl:Class>
```

Ακόμη και στην περίπτωση που δε δηλώνεται ότι η `Class1` είναι κλάση, δινόταν μόνο `<rdf:ID="Class1">` εξυπακούεται ότι θα επρόκειτο για resource τύπου `Class`, αφού είναι υποκλάση κάποιας άλλης. Στην OWL DL όμως, χρειάζεται να δηλωθεί ρητά.

2. Οι ιδιότητες τύπου δεδομένων δεν έχουν κοινό σημείο με τις ιδιότητες αντικειμένου, πρόκειται για αμοιβαία αποκλειόμενα σύνολα. Επομένως, οι δηλώσεις `owl:inverseOf`, `owl:FunctionalProperty`, `owl:InverseFunctionalProperty` και `owl:SymmetricProperty` δε μπορούν να χρησιμοποιηθούν για ιδιότητες τύπου δεδομένων.
3. Οι περιορισμοί στα στοιχεία των συνόλων (cardinality restrictions) δεν είναι πλέον μεταβατικές και επίσης δε μπορούν να τεθούν περιορισμοί στα στοιχεία συνόλων από μεταβατικές ιδιότητες.
4. Δε γίνεται χρήση από ανώνυμες κλάσεις παρά μόνο ως domain και range των `owl:equivalentClass` και `owl:disjointWith` και range (όχι domain) της `rdfs:subClassOf`.

Στα αρνητικά σημεία της, ωστόσο, περιλαμβάνεται το ότι έχει απολεστεί η συμβατότητα με το RDF. Ένα έγγραφο RDF θα πρέπει να τροποποιηθεί και να επεκταθεί σε αρκετά σημεία ώστε να καταστεί συμβατό με την OWL DL. Αντίθετα, όμως, κάθε έγκυρο OWL DL έγγραφο είναι επίσης και έγκυρο RDF έγγραφο.

Η *OWL Lite* είναι σχεδόν ισοδύναμη με τη γλώσσα *SHIF(D+)*. Σαν γλώσσα, περιέχει μόνο ένα μικρό υποσύνολο του λεξικού της OWL. Συγκεκριμένα, έχει αφαιρεθεί η δήλωση για αριθμημένες (enumerated) κλάσεις, η δήλωση για αμοιβαία αποκλειόμενες (disjoint) κλάσεις και ο αυθαίρετος περιορισμός στον αριθμό των στοιχείων των κλάσεων (cardinality).

Η OWL Lite επιτρέπει τις βασικές έννοιες της Περιγραφικής Λογικής, έχει μεταβατικές ιδιότητες, αντίστροφες έννοιες, functional properties και datatype properties. Μια οντολογία σε OWL Lite πρέπει να ικανοποιεί τους περιορισμούς που τίθενται στην OWL DL και επιπλέον

1. Να μην κάνει χρήση των constructors `owl:oneOf`, `owl:disjointWith`, `owl:unionOf`, `owl:complementOf` και `owl:hasValue`.
2. Οι περιορισμοί σε cardinality μπορούν να έχουν μόνο τις τιμές 0 και 1 – όχι αυθαίρετους μη αρνητικούς ακέραιους όπως στις άλλες υπογλώσσες.
3. Οι δηλώσεις `owl:equivalentClass` δε μπορούν να γίνουν μεταξύ ανώνυμων κλάσεων παρά μόνο μεταξύ class identifiers.

Η OWL Lite έχει το προτέρημα της διευκόλυνσης λόγω της απλότητάς της στην κατανόηση. Είναι πιο απλή και προορίζεται για περιπτώσεις όπου χρειάζεται περισσότερο η ταχύτητα και όχι η εκφραστικότητα. Είναι πολύ πιο εύκολη η συγγραφή εργαλείων και προγραμμάτων και είναι εξασφαλισμένες οι καλύτερες επιδόσεις τους σε σχέση με τις DL και Full παραλλαγές.

Αναφέρεται, τέλος, ότι η τρέχουσα έκδοση της γλώσσας είναι η OWL 2, επίσημη πρόταση του W3C από τις 27 Οκτωβρίου 2009⁸. Η νέα έκδοση περιέχει συντακτικές διευκολύνσεις για την ευκολότερη δήλωση συχνά χρησιμοποιούμενων εννοιών, νέους τύπους δηλώσεων (constructs) Περιγραφικής Λογικής και πιο εκτεταμένη εκφραστικότητα.

2.5 Συνήθη λεξικά και μικροπρότυπα

Τα *μικροπρότυπα* σύμφωνα με το `microformats.org` είναι “μικρά τμήματα HTML τα οποία αναπαριστούν αντικείμενα όπως άνθρωποι και γεγονότα σε σελίδες του Διαδικτύου”. Παρέχουν περισσότερη αξιοπιστία στην πληροφορία που δημοσιεύεται ενώ παράλληλα παρέχουν διευκόλυνση στην διασύνδεση της πληροφορίας με άλλες πηγές.

Ο Σημασιολογικός Ιστός σήμερα βρίσκει ευρεία αποδοχή σε επιμέρους πεδία ενδιαφέροντος τα οποία βασίζονται κυρίως σε RDF σύνταξη προκειμένου να περιγράψουν το γνωστικό τους πεδίο. Ένα από λεξικά που συναντώνται συχνά σε έγγραφα του Σημασιολογικού Ιστού και χρειάζεται ειδική αναφορά είναι αυτό της Dublin Core. Η Dublin Core⁹ είναι ένας οργανισμός με στόχο την παραγωγή προτύπων για μεταδεδομένα. Το συγκεκριμένο πρότυπο που παρουσιάζεται στον πίνακα 9 είναι αυτό της περιγραφής ψηφιακών αντικειμένων. Αποτελείται από 16 προαιρετικά tags, τα οποία παρατίθενται με τυχαία σειρά.

Συνήθως, η χρήση των tags της Dublin Core παρατηρείται στην αρχή των δηλώσεων του εγγράφου μέσω του namespace `dc` (πίνακας 1) και χρησιμοποιείται, για παράδειγμα για τη δήλωση του δημιουργού ενός αντικειμένου με τη χρήση έστω του tag `creator`:

⁸OWL 2 Web Ontology Language: www.w3.org/TR/owl2-overview/

⁹The Dublin Core Metadata Initiative (DCMI): dublincore.org

A/A	Tag	A/A	Tag
1	Title	2	Format
3	Creator	4	Identifier
5	Subject	6	Source
7	Description	8	Language
9	Publisher	10	Relation
11	Contributor	12	Coverage
13	Date	14	Rights
15	Type	16	Audience

Πίνακας 9: Τα tags της Dublin Core για περιγραφή ψηφιακού περιεχομένου

```
<dc:creator>John</dc:creator>
```

Η κοινότητα της DBpedia¹⁰ [40] πραγματοποιεί προσπάθεια για την εξαγωγή δομημένης πληροφορίας από την Wikipedia ώστε να μπορεί όλη αυτή η δομημένη πληροφορία να είναι διαθέσιμη στο Διαδίκτυο ενώ παράλληλα να είναι διασυνδεδεμένη με άλλες πηγές πληροφορίας του Διαδικτύου. Η Wikipedia, σύμφωνα με το www.alexa.com είναι ο έκτος πιο δημοφιλής δικτυακός τόπος (Δεκέμβριος 2009), με εκδόσεις σε περισσότερες από 250 γλώσσες και περίπου δύο εκατομμύρια άρθρα στην αγγλική γλώσσα.

Όπως όμως και σε πολλές άλλες πηγές πληροφορίας, το πρόβλημα της Wikipedia είναι ότι η αναζήτηση περιορίζεται σε απλή αναζήτηση σε λέξεις-κλειδιά καθιστώντας πολύ περιορισμένη την πρόσβαση σε αυτόν τον πλούτο πληροφορίας. Επιπλέον, λόγω της απήχησής της και του ανοιχτού της χαρακτήρα, η Wikipedia πάσχει από το πρόβλημα που παρουσιάζεται σε όλα τα έργα που επιτρέπουν συνεργατική επεξεργασία της πληροφορίας. Παρουσιάζονται ασυνέπειες, μη συγκροτημένες κατηγοριοποιήσεις εννοιών, ακόμα και φαινόμενα spam. Το έργο DBpedia αντιμετωπίζει τα προβλήματα αυτά επιχειρώντας να αποδώσει δομή στην πληροφορία και να επιτρέπει την υποβολή πιο ευφώνων ερωτημάτων ενώ ταυτόχρονα θα διασυνδέει την πληροφορία της Wikipedia με άλλες πηγές πληροφορίας. Η DBpedia δημιουργεί ένα πλαίσιο εξαγωγής των δεδομένων της Wikipedia σε RDF υπό μια κοινή οντολογία-λεξικό. Το σύνολο που προκύπτει παρέχεται ως βάση για την αντιστοίχιση πληροφορίας του Διαδικτύου και τη δημιουργία εφαρμογών Σημασιολογικού Ιστού. Το έργο DBpedia αριθμεί πλέον περισσότερες από 100 εκατομμύρια τριάδες.

¹⁰DBpedia: dbpedia.org

FOAF σημαίνει Friend Of A Friend¹¹. Πρόκειται για ένα έργο που φιλοδοξεί να καταστήσει κατανοητή από τον υπολογιστή την πληροφορία που υπάρχει σε προσωπικές ιστοσελίδες ή επίσης ιστοσελίδες ομάδων, εταιρειών, οργανισμών κοκ. Για το στόχο αυτό έχει αναπτυχθεί η FOAF οντολογία/λεξικό. Η βασική ιδέα του έργου είναι ότι η πληροφορία των σελίδων που αφορούν ανθρώπους είναι συνήθως της ίδιας σημασίας. Αν επομένως, πολλοί άνθρωποι δημιουργούν τις ιστοσελίδες τους σε FOAF, θα μπορεί να δημιουργηθεί μια Βάση Γνώσεως με αυτούς τους ανθρώπους. Η φιλοδοξία της FOAF είναι ότι αν πολύς κόσμος ακολουθήσει την γλώσσα αυτή, θα είναι δυνατή η εξαγωγή πληροφορίας μέσω προγραμμάτων των οποίων θα μπορούν να διασχίζουν τις σελίδες αυτές και να διατηρούν λίστες σε άλλες σελίδες. Θα μπορέσουν να δημιουργηθούν υπηρεσίες ο να βασίζονται στη γνώση αυτή. Το πάντα επίκαιρο θέμα της ασφάλειας προσωπικών δεδομένων, αντιμετωπίζεται στην περίπτωση της FOAF με τη χρήση τεχνικών κρυπτογραφίας. Για παράδειγμα:

```
<foaf:Person>
  <foaf:name>
    John
  </foaf:name>
  <foaf:mbox sha1sum>
    241021fb0e6289f92815fc210f9e9137262c252e
  </foaf:mbox sha1sum>
  <foaf:homepage
    rdf:resource="http://www.example.org/~john/" />
  <foaf:img
    rdf:resource="http://www.example.org/~john/john.jpg" />
</foaf:Person>
```

Εδώ δηλώνεται ότι ο John έχει μια προσωπική ιστοσελίδα, μια φωτογραφία στην οποία – πιθανότατα – θα απεικονίζεται ο ίδιος και μια διεύθυνση email την οποία δεν επιθυμεί να την κοινοποιήσει, πιθανότατα για την αποφυγή spam emails. Η διεύθυνση email είναι το αποτέλεσμα του αλγόριθμου SHA1 στο URI του email. Το DOAC (Description of a Career) συμπληρώνει το FOAF κατά τη δημιουργία πληροφορία βιογραφικών σημειωμάτων.

Αντίστοιχη προσπάθεια είναι και η DOAP (usefulinc.com/doap) η οποία σημαίνει Description Of A Project και στοχεύει στη δημιουργία ενός λεξικού το οποίο να περιγράφει έργα ανοιχτού κώδικα. Το σκεπτικό πίσω από την DOAP είναι ακριβώς ανάλογο της FOAF.

¹¹FOAF: xmlns.com/foaf/spec/

Το σύνολο των RSS γλωσσών χρησιμοποιείται προκειμένου να δημοσιεύεται πληροφορία για τις ανανεώσεις στις οποίες υπόκειται μια ιστοσελίδα. Μία από τις μορφές του RSS, το RDF Site Summary χρησιμοποιεί λεξικό το οποίο υπακούει σε συγκεκριμένο πρότυπο, είναι βασισμένο σε RDF και συμμορφώνεται πλήρως με τις υποδείξεις του W3C (web.resource.org/rss/1.0/spec).

Ο Creative Commons είναι ένας οργανισμός που παράγει άδειες για ψηφιακό περιεχόμενο. Χρησιμοποιεί το RDF για την ενσωμάτωση πληροφορίας σχετικής με την άδεια χρήσης σε ιστοσελίδες ή αρχεία πολυμέσων. Οι έννοιες που περιέχει το λεξικό Creative Commons σχετίζονται κυρίως με δικαιώματα όπως η αναπαραγωγή, η διανομή και ο διαμοιρασμός του περιεχομένου.

Το έργο SIOC¹² [41] έχει στόχο την περιγραφή κοινοτήτων στο Διαδίκτυο. Μέσω της οντολογίας που παρέχει περιγράφει έννοιες όπως Group, Forum, Post και επιτρέπει τη διασύνδεση μεταξύ τους.

Το GRDDL (προφέρεται “griddle”) αποτελεί επίσης επίσημη σύσταση του W3C από τις 17 Σεπ. 2007. Το GRDDL παρέχει έναν μηχανισμό ο οποίος επιτρέπει την αυτόματη μετάφραση πληροφορίας – η οποία μπορεί να περιέχεται σε μικροπρότυπα – σε RDF.

Η έννοια του χρόνου υπάρχει σχεδόν σε κάθε εφαρμογή που καλύπτει ανάγκες πραγματικού κόσμου. Για παράδειγμα, όσον αφορά τις Υπηρεσίες Διαδικτύου, ένα χρονικό στιγμιότυπο της αγοράς, πώλησης, κράτησης ή συναλλαγής γενικότερα είναι αναγκαίο. Με αυτό το σκεπτικό, το Semantic Web Best Practices and Deployment Working Group¹³ ανέπτυξε την DAML-Time οντολογία η οποία σήμερα έχει εξελιχθεί στην OWL-Time¹⁴. Έτσι, οι έννοιες που περιγράφονται στην οντολογία αυτή μπορούν να δώσουν περιγραφή της χρονικής διάστασης Υπηρεσιών Διαδικτύου αλλά και ιστοσελίδων. Η οντολογία μπορεί να σταθεί και σαν περιγραφή από μόνη της αλλά η ομάδα ανάπτυξης έχει εξασφαλίσει σαν κύρια προτεραιότητα τη συνεργασία με την OWL-S, όπως αντίστοιχα και η DAML-Time συνεργάζεται άριστα με την DAML-S¹⁵.

Στην οντολογία αυτή περιγράφονται έννοιες όπως Instant για χρονικές στιγμές, Interval για χρονικά διαστήματα και μπορούν να έχουν ιδιότητες όπως starts, ends, intoOverlaps, TimeZone, unitType και γενικά αποτελεί προσέγγιση που θα καλύπτει τις περισσότερες απαιτήσεις [42].

Επίσης προσπάθεια για την περιγραφή εννοιών του χρόνου γίνεται από το έργο T-owl¹⁶ (IST-2004-2.4.7). Το έργο αυτό στοχεύει στην εισαγωγή της διάστασης του χρόνου

¹²Semantically-Interlinked Online Communities: sioc-project.org/

¹³www.w3.org/2001/sw/BestPractices/

¹⁴Time Ontology in OWL: www.w3.org/TR/owl-time/

¹⁵Περισσότερα για τις DAML-S και OWL-S στην ενότητα 3.7.3.

¹⁶TOWL project: www.towl.org

στις υπάρχουσες γλώσσες του Σημασιολογικού Ιστού. Παράλληλα, με τη δημιουργία *reasoners* οι οποίοι να λαμβάνουν υπόψη τη διάσταση του χρόνου, στα αποτελέσματα του έργου αυτού θα είναι η διαχείριση του χρόνου στα πληροφοριακά συστήματα βασισμένα στο Σημασιολογικό Ιστό. Ανάμεσα στα μέχρι τώρα επιτεύγματα του έργου είναι η δημιουργία της TOWL [43], μιας γλώσσας περιγραφής του χρόνου σε OWL.

Το hCalendar είναι ακόμη ένα πρότυπο για την περιγραφή χρονικών στιγμών και γεγονότων. Βασίζεται στο iCalendar standard (RFC2445 (www.ietf.org/rfc/rfc2445.txt)) και είναι κατάλληλο για την ενσωμάτωση σε αρχεία HTML, XHTML, Atom, RSS, ακόμη και αυθαίρετα XML.

eRDF σημαίνει embedded RDF και αποτελεί μια σύνταξη με τη χρήση της οποίας μπορεί κανείς να γράψει HTML έγγραφα με τέτοιο τρόπο ούτως ώστε το αποτέλεσμα να μπορεί να μετατραπεί σε RDF, είτε μέσω eRDF parser ή από XSLT. Είναι δυνατό να εξαχθούν RDF γράφοι από eRDF έγγραφα, μέσα από την επεξεργασία τους από ένα σύνολο από εργαλεία που το υποστηρίζουν, όπως:

- eRDF scraper simile.mit.edu/wiki/Erdf_scraper
- ARC PHP arc.web-semantics.org/
- Raptor RDF parser toolkit librdf.org/raptor
- an XSLT stylesheet to transform eRDF in XHTML to RDF/XML research.talis.com/2005/erdf/extract-rdf.xsl
- eRDF Greasemonkey userscripts.org/scripts/show/8260
- Web Service: ARC eRDF extract arc.web-semantics.org/demos/erdf-extract
- Web Service: Talis eRDF extract purl.org/NET/erdf/extract
- Web Service: triplr eRDF extract triplr.org

Δεν αποτελεί όμως πρότυπο του W3C. Αντίθετα, το RDFa (RDF attributes) αποτελεί ένα σύνολο από επεκτάσεις της XHTML που από τις 14 Οκτ. 2008 αποτελεί επίσημη πρόταση του W3C¹⁷. Το RDFa ήδη υποστηρίζεται από το Drupal, ένα από τα δημοφιλέστερα συστήματα διαχείρισης περιεχομένου (CMS) και το Yahoo! SearchMonkey (εικόνα 8). Το RDFa δεν είναι μικροπρότυπο, είναι γλώσσα η οποία στην ουσία αποτελεί ένα σύνολο επεκτάσεων στην XHTML. Με την RDFa, μπορεί ο συντάκτης μιας ιστοσελίδας να προσθέσει σημασιολογική επισημείωση σε τμήματα ενός XHTML εγγράφου χρησιμοποιώντας στην πράξη τα στοιχεία *meta* και *link*. Με αυτόν τον τρόπο μπορεί να ενσωματωθεί σημασιολογία σε μια ιστοσελίδα όπως στο παρακάτω παράδειγμα το οποίο χρησιμοποιεί το λεξικό της Dublin Core:

¹⁷RDFa in XHTML: Syntax and Processing, www.w3.org/TR/2008/REC-rdfa-syntax-20081014/

```
<div xmlns:dc="http://purl.org/dc/elements/1.1/"
      about="http://www.example.com/example">
  <span property="dc:title">The RDFa Primer</span>
  <span property="dc:creator">John</span>
  <span property="dc:date">2008-10-01</span>
</div>
```

Το παράδειγμα δεν προσφέρει σημασιολογία ως έχει, μπορεί όμως να οδηγήσει στη δημοσίευση διασυνδεδεμένων δεδομένων [44] ως εξής¹⁸:

```
<p>
  My
  <span rel="foaf:interest" resource="urn:ISBN:0752820907">
    favorite book
  </span>
  is the inspiring
  <span about="urn:ISBN:0752820907">
    <cite property="dc:title">
      Weaving the Web
    </cite>
    by
    <span property="dc:creator">
      Tim Berners-Lee
    </span>
  </span>.
</p>
```

Επομένως, η έννοια της *σημασιολογικής* επισημείωσης σε ένα έγγραφο του Διαδικτύου υφίσταται όταν τα επισημειωμένα τμήματα του εγγράφου διασυνδέονται με άλλα τμήματα του Διαδικτύου. Με αυτόν τον τρόπο τούς αποδίδεται μονοσήμαντη έννοια και ενισχύεται η *αξιοπιστία* τους, σύμφωνα με την ιδέα του Σημασιολογικού Ιστού (εικόνα 3).

2.6 Σύνταξη κανόνων

Οι συμβατικές προσεγγίσεις στη συγγραφή κανόνων συμπεριλαμβάνουν εργαλεία όπως η Jess, το Drools ή το ενσωματωμένο σύστημα κανόνων της Jena. Η Jess είναι μια μηχανή δήλωσης κανόνων για τη γλώσσα Java. Η ιστορία της Jess ξεκινά το 1995 και

¹⁸Πηγή: en.wikipedia.org/wiki/RDFa

η ιδέα πίσω από τη Jess είναι το να μπορεί κανείς να προγραμματίζει όχι με συναρτήσεις αλλά με το να ορίζει ένα σύνολο κανόνων το οποίο θα εφαρμόζεται συνεχώς σε ένα σύνολο δεδομένων. Η μηχανή εκτέλεσης κανόνων της Jess βασίζεται στον αλγόριθμο Rete [45]. Η Jess δε διατίθεται ελεύθερα. Διαθέσιμη είναι μόνο μια μειωμένη έκδοση της Jess, η CLIPS¹⁹. Επίσης χρησιμοποιούμενο και διαδεδομένο είναι το Drools, ένα σύστημα για τη διαχείριση και παραγωγή κανόνων, πάνω στο οποίο βασίζεται και το Jboss rules. Και τα δυο βασίζονται επίσης στον αλγόριθμο Rete, όπως επίσης και η μηχανή επεξεργασίας κανόνων της Jena.

Οι προσεγγίσεις που ακολουθούνται περισσότερο για την εξυπηρέτηση αναγκών του Σημασιολογικού Ιστού περιλαμβάνουν την RuleML και την SWRL. Η RuleML²⁰ είναι μια γλώσσα εφαρμογής κανόνων που χρησιμοποιούνται για την αφαίρεση, την επανασυγγραφή και τις περαιτέρω διεργασίες μετασχηματισμών. Γενικά, είναι μια επεκτάσιμη προδιαγραφή για την άμεση ανταλλαγή κανόνων. Οι κανόνες στην RuleML δηλώνονται σε συνδυασμό φυσικής γλώσσας και σημειώσεων. Οι περαιτέρω σκοποί αυτής της επισημείωσης περιλαμβάνουν την παροχή ενός σχήματος ανταλλαγής κανόνων μεταξύ των διαφορετικών εργαλείων/συστημάτων, το χαρακτηρισμό των κανόνων περιεχομένου στα επιχειρησιακά έγγραφα και την παροχή μιας υψηλού επιπέδου γλώσσας προδιαγραφών για το ενεργό περιεχόμενο στο Διαδίκτυο. Ωστόσο, πολύ λίγες υλοποιήσεις, όπως οι Mandarax και OO jDrew είναι διαθέσιμες.

Η SWRL [46] (Semantic Web Rule Language) είναι μια γλώσσα που συνδυάζει την OWL με την RuleML [47, 48]. Στην πράξη, η SWRL συνδυάζει ένα υποσύνολο της OWL DL με τις Unary/Binary Datalog RuleML υπο-γλώσσες της RuleML. Η βασική ιδέα της SWRL είναι η δήλωση κανόνων με στόχο την επέκταση της OWL DL. Στο παρακάτω παράδειγμα περιγράφεται ένας απλός κανόνας που δηλώνει ότι αν ο $x1$ έχει γονέα τον $x2$ και ο $x2$ έχει αδερφό τον $x3$, τότε ο $x3$ είναι θείος του $x1$. Σε πρόταση λογικής πρώτου βαθμού ο κανόνας γράφεται:

$$hasParent(?x1, ?x2) \wedge hasBrother(?x2, ?x3) \rightarrow hasUncle(?x1, x3)$$

Ενώ σε SWRL παίρνει τη μορφή:

```
<ruleml:imp>
<ruleml: rlab ruleml:href=" example1"/>
<ruleml: body>
  <swrlx:individualPropertyAtom swrlx:property="hasParent">
    <ruleml:var>x1</ruleml:var>
    <ruleml:var>x2</ruleml:var>
```

¹⁹CLIPS, A Tool for Building Expert Systems: clipsrules.sourceforge.net

²⁰The Rule Markup Initiative: www.ruleml.org


```
</swrlx:individualPropertyAtom>
<swrlx:individualPropertyAtom swrlx:property="hasBrother">
  <ruleml:var>x2</ruleml:var>
  <ruleml:var>x3</ruleml:var>
</swrlx:individualPropertyAtom>
</ruleml: body>
<ruleml: head>
  <swrlx:individualPropertyAtom swrlx:property="hasUncle">
    <ruleml:var>x1</ruleml:var>
    <ruleml:var>x3</ruleml:var>
  </swrlx:individualPropertyAtom>
</ruleml: head>
</ruleml:imp>
```

Με τη χρήση της SWRL είναι δυνατή η μετάφραση προγραμμάτων άλλου τύπου σε OWL, όπως για παράδειγμα Prolog ή SQL – όπου οι όψεις και τα ερωτήματα θα μπορούσαν να είναι κανόνες της SWRL. Η SWRL έχει υλοποιηθεί ως επέκταση του Proτέγέ ως SWRLTab [49] και επιτρέπει την έκφραση και εκτέλεση προτάσεων. Επίσης το έργο SweetRules(projects.semwebcentral.org/projects/sweetrules) ασχολήθηκε με την υποστήριξη σε Java της RuleML και της SWRL, δεν εμφανίζει όμως κινητικότητα. Σε κάθε περίπτωση απαιτείται η ενσωμάτωση (`owl:import`) της SWRL οντολογίας στην υπό ανάπτυξη οντολογία. Επιπλέον, οι εξυπηρετητές υπηρεσιών συλλογιστικής Pellet και KAON2 υποστηρίζουν την SWRL, αλλά όχι στην πλήρη εκφραστικότητά της λόγω των πολλών ασυμβατοτήτων μεταξύ OWL DL και έκφραση με κανόνες Horn προτάσεων της λογικής πρώτου βαθμού.

Το σημαντικότερο πρόβλημα είναι η ελευθερία στην προσφερόμενη εκφραστικότητα από τους κανόνες, που την καθιστά συχνά αδύνατη. Με άλλα λόγια, το υποσύνολο μιας γλώσσας κανόνων που να μπορεί να χρησιμοποιηθεί για να επιβεβαιώσει τις πεπερασμένες διαδικασίες συλλογιστικής είναι περιορισμένο. Οι δημιουργοί του Pellet πρότειναν ένα υποσύνολο ασφαλών SWRL κανόνων που βεβαιώνουν τη σωστή απόφαση [50], αλλά η υποστήριξη SWRL ή RuleML στις εφαρμογές είναι ακόμα μια ρέουσα διαδικασία.

Για την μελέτη του προβλήματος της ανταλλαγής κανόνων στα πλαίσια του Σημασιολογικού Ιστού έχει συσταθεί το Rule Interchange Format Working Group (www.w3.org/2005/rules/wiki/RIF_Working_Group) αλλά δεν έχει προκύψει ακόμη κάποιο πρότυπο.

2.7 Γλώσσες ερωτημάτων

Οι γλώσσες ερωτημάτων για τον Σημασιολογικό Ιστό είναι ακόμη στη νεότητά τους καθώς πρόκειται όμως για χώρο ο οποίος ακόμη προσφέρει μεγάλο πεδίο για έρευνα και ανάπτυξη. Πιο εκτεταμένες έρευνες για γλώσσες ερωτημάτων του Σημασιολογικού Ιστού μπορούν να βρεθούν στο [51] και στο [52].

Ένα αφελές ερώτημα θα ήταν το εξής: Γιατί χρειάζονται ειδικές γλώσσες ερωτημάτων για οντολογίες, εφόσον η σύνταξή τους είναι τις περισσότερες φορές σε XML και γιατί δε θα ήταν αρκετή η ανάκτηση των κόμβων που χρειάζονται με μια γλώσσα ερωτημάτων για XML όπως για παράδειγμα η XPath ή η XQuery; Ως απάντηση, έστω το ακόλουθο παράδειγμα στο οποίο σε RDF περιγράφεται το γεγονός ότι ο John είναι υποψήφιος διδάκτωρ.

```
<rdf:description rdf:about="1306">
  <rdf:type rdf:resource="&ex;phd_student" />
  <ex:name>John</ex:name>
  <ex:title>PhD Student</ex:title>
</rdf:description>
```

Σε μια τέτοια περιγραφή, ένα κατάλληλο XPath ερώτημα θα ήταν:

```
/rdf:description[rdf="http://www.example.org#phd_student"]
  /ntua:name
```

Η περιγραφή θα μπορούσε όμως να δίνεται και με κάποιον άλλο τρόπο, για παράδειγμα

```
<ex:phd_student rdf:about="1306">
  <ex:name>John</ex:name>
  <ex:title>PhD Student</ex:title>
</ex:phd_student>
```

Στην περίπτωση αυτή, το προηγούμενο XPath ερώτημα δε λειτουργεί και χρειάζεται να ξαναγραφεί ως εξής:

```
//ex:phd_student/ex:name
```

Μια τρίτη πιθανή περιγραφή θα μπορούσε να είναι ως εξής:

```
<ex:phd_student
  rdf:about="1306"
  ex:name="John"
  ex:title="PhD Student" />
```

Γι' αυτή τη συντακτική παραλλαγή το ερώτημα XPath θα έπρεπε να είναι της μορφής:

```
//ex:phd_student/@ex:name
```

Διαπιστώνεται επομένως η ανεπάρκεια της XPath αλλά και των υπόλοιπων γλωσσών ερωτημάτων για XML στο να ακολουθήσουν την ελευθερία εκφραστικότητας του RDF. Με τη χρήση του RDF, ένα ερώτημα για τους μεταπτυχιακούς φοιτητές όπως για παράδειγμα σε γλώσσα SPARQL (ενότητα 2.7.2)

```
PREFIX ex:http://www.example.org
SELECT ?x, ?name
FROM ex WHERE (?x rdf:type ex:grad_student),
              (?x ex:name ?name)
```

θα επιστρέψει τον John ανεξάρτητα από τη σύνταξη της δήλωσης. Επιπρόσθετα, οι γλώσσες ερωτημάτων για XML αδυνατούν να υποστηρίξουν τις διάφορες μορφές στις οποίες μπορεί να εκφράζεται ένας RDF γράφος όπως η N-Triples ή η Turtle notation. Είναι απαραίτητη η δυνατότητα δημιουργίας ερωτημάτων στο επίπεδο της RDF. Μια γλώσσα ερωτημάτων θα πρέπει να λαμβάνει υπόψη όχι μόνο τη σύνταξη αλλά και το μοντέλο δεδομένων της RDF προκειμένου να επιστρέφει τα σωστά αποτελέσματα. Σημειώνεται ότι δεν έχει αναπτυχθεί ακόμη γλώσσα αρκετά ικανή για την υποβολή ερωτημάτων σε OWL οντολογίες – εκτός από την OWL-QL (www.ksl.stanford.edu/projects/owl-ql) η οποία παρουσιάζεται στο [53].

Στη συνέχεια, παρουσιάζοντας τις υπάρχουσες γλώσσες για RDF, ακολουθείται ο διαχωρισμός σε δυο υποκατηγορίες, δυο οικογένειες γλωσσών, την SPARQL – προφέρεται “σπάρκλ” – και την RQL. Εκτός όμως από αυτές τις επικρατέστερες γλώσσες, έχουν αναπτυχθεί ακόμη μερικές οι οποίες τυγχάνουν μικρότερης αποδοχής. Η N3 είναι γλώσσα ερωτημάτων η οποία και παρέχει πρόσβαση σε RDF βασισμένη σε κείμενο. Υποστηρίζεται από τα συστήματα Euler and CWM [54]. Η γλώσσα Versa υποστηρίζεται από την πλατφόρμα 4Suite (www.4suite.org) και το όνομά της προέρχεται από τη δυνατότητα forward και backward διάσχιση RDF γράφων. Η Triple (triple.semanticweb.org) παρουσιάζεται στο [55] και είναι μια γλώσσα για ερωτήματα σε RDF γράφους η οποία μπορεί επιπλέον να εξάγει συμπεράσματα. Βασίζεται σε F-Λογική, γεγονός που σημαίνει ότι ο RDF τριάδες μετατρέπονται σε εκφράσεις F-Λογικής. ενικά, μια τριάδα (S, P, O) μετατρέπεται στην έκφραση $S[P \rightarrow O]$. Από τα σημαντικότερα επιτεύγματα της υλοποίησης αυτής είναι ότι οι εκφράσεις αυτές μπορούν να είναι εμφωλευμένες. Για παράδειγμα, η έκφραση

$$S[P1 \rightarrow O1, P2 \rightarrow O2][P3 \rightarrow O3]$$

αντιστοιχεί στις τριάδες RDF:

$(S, P1, O1)$, $(S, P2, O2)$ και $(O2, P3, O3)$

Η Triple μπορεί να διαχειριστεί ταυτόχρονα πολλά μοντέλα RDF τα οποία διαχωρίζονται με το πρόθεμα @model. Επίσης, υπάρχει ακόμη μια ποικιλία από γλώσσες όπως η RXPath και η RPath, η RDFQL, η RxSLT και η RxUpdate οι οποίες είναι γλώσσες με σύνταξη βασισμένη στην XPath και η παρουσίασή τους ξεφεύγει από τα όρια της παρούσας έρευνας. Στη συνέχεια παρουσιάζονται οι δυο κύριοι κλάδοι γλωσσών ερωτημάτων στο Σημασιολογικό Ιστό.

2.7.1 Διάσχιση γράφων: Οικογένεια RQL

Η οικογένεια RQL αποτελείται από τις RQL, SeRQL και eRQL. Το κοινό χαρακτηριστικό μεταξύ αυτών των γλωσσών είναι ότι υποστηρίζουν το συνδυασμό ερωτημάτων στο σχήμα και τα δεδομένα της RDF. Το μοντέλο των δεδομένων όμως διαφέρει από αυτό της RDF(S). Στις βασικές διαφορές περιλαμβάνονται το ότι δεν επιτρέπονται κύκλοι στην ιεραρχία και ότι για κάθε ιδιότητα χρειάζεται να ορίζεται το domain και το range της.

Η RQL είναι μια από τις γλώσσες ερωτημάτων σε RDF αρχεία. Έχει υλοποιηθεί στο RDF Suite του ICS-FORTH και ένα τμήμα της είναι διαθέσιμο στο σύστημα Sesame [56]. Είναι πολύ εκφραστική γλώσσα ερωτημάτων αλλά τα semantics της δεν είναι σε πλήρη συμφωνία με αυτά της RDF. Η σύνταξή της είναι παρόμοια με αυτή της OQL. Για την ανάκτηση όλων των instances μιας κλάσης, για παράδειγμα της phd_student, γράφεται phd_student.

Το ερώτημα αυτό θα επιστρέψει επίσης όλα τα instances που κληρονομούνται από την κλάση αυτή στις υποκλάσεις της. Αν δεν επιθυμούμε την ανάκτηση των instances των υποκλάσεων γράφεται ^phd_student.

Για τη σύνθεση πιο πολύπλοκων ερωτημάτων, η RQL προσφέρει σύνταξη SELECT FROM WHERE παρόμοια με αυτή της SQL. Για παράδειγμα, για την ανάκτηση των ονομάτων όλων όσων βρίσκονται στο RDF αρχείο γράφεται SELECT X, Y FROM {X}name{Y}. Αντίστοιχα, για την ανάκτηση των ονομάτων των υποψηφίων διδασκτόρων μπορεί να γραφεί SELECT X, Y FROM {X}name{Y} WHERE {X}title{"PhD Student"}.

Η RQL έχει σε μεγάλο βαθμό ομοιότητα με την SQL καθώς επιτρέπει ερωτήματα τύπου SELECT DISTINCT, ερωτήματα SELECT FROM WHERE LIKE για αναζήτηση λεκτικών και συνθήκες JOIN μεταξύ αποτελεσμάτων. Επιτρέπει τη διάσχιση του γράφου χρησιμοποιώντας την τελεία, για παράδειγμα, το ακόλουθο ερώτημα επιστρέφει όλα τα μαθήματα που μπορεί να διδάσκονται:

```
SELECT X
WHERE StaffMember.teaches.Course.title{X}
```

Το πρόθεμα \$ για κλάσεις ή το πρόθεμα @ για ιδιότητες μπορεί να χρησιμοποιηθεί για την υποβολή ερωτημάτων πάνω στο σχήμα. Επίσης μπορούν να υποβληθούν ερωτήσεις σχετικά με το domain ή το range ιδιοτήτων. Για παράδειγμα, το ερώτημα

```
SELECT $X, @P, RANGE(@P)
FROM {$X}.@P
```

θα επιστρέψει όλες τις κλάσεις που έχουν ιδιότητες μαζί με το range των ιδιοτήτων αυτών. Οι δυνατότητες της RQL δε σταματάνε εδώ καθώς πρόκειται για μια αρκετά πλήρη γλώσσα. Περισσότερα για τη σύνταξη της RQL προσφέρονται στο online εγχειρίδιο²¹.

SeRQL σημαίνει Sesame Rdf Query Language [57] και προφέρεται όπως η λέξη circle. Είναι η γλώσσα ερωτημάτων του Sesame [56]. Η SeRQL – όπως και η eRQL – έχει σαν βάση της την RQL. Στις βασικές της διαφορές από την RQL συμπεριλαμβάνονται τα εξής:

1. Η SeRQL δεν υποστηρίζει RDF και RDFS types εκτός από literal types.
2. Τροποποιεί και επεκτείνει τις εκφράσεις μονοπατιού σε RDF γράφο που χρησιμοποιεί η RQL. Περιέχει τη δήλωση { } για blank nodes.
3. Παρέχει συντακτικές διευκολύνσεις. Μπορούν να εκφραστούν κάποιες δηλώσεις με περισσότερη συντομία, κάτι που την καθιστά πιο περιεκτική, για παράδειγμα, σε SeRQL, αντί για τη δήλωση

```
...FROM {X} <ex:title> {Y}, {X} <ex:title> {Z}
```

μπορεί να δηλωθεί

```
...FROM {X} <ex:title> {Y, Z}
```

Επιπλέον, υποστηρίζονται CONSTRUCT FROM WHERE εντολές οι οποίες επιτρέπουν τη σύνθεση γράφου που να προκύπτει από τα αποτελέσματα των ερωτημάτων. Ωστόσο, δε μπορούν να εκφραστούν aggregated ερωτήματα (minimum, maximum, average, count) και δεν υποστηρίζονται εμφωλευμένα ερωτήματα. Η eRQL είναι μια ακόμα πιο απλουστευμένη έκδοση της RQL. Η φιλοδοξία των σχεδιαστών ήταν η δημιουργία μιας γλώσσας στην οποία τα απλούστερα ερωτήματα να είναι τόσο απλά όσο του Google²². Τα ερωτήματα στην eRQL είναι τριών ειδών.

²¹RQL v2.1 User manual: <http://139.91.183.30:9090/RDF/RQL/Manual.html>

²²eRQL: www.dbis.informatik.uni-frankfurt.de/~tolle/RDF/eRQL/

1. Ερωτήματα απλής λέξης. Μια απλή λέξη είναι έγκυρο eRQL ερώτημα και επιστρέφει όλες τις δηλώσεις στις οποίες εμφανίζεται με οποιοδήποτε τρόπο η ζητούμενη λέξη.
2. Ερωτήματα γειννίας (Neighborhood queries). Για παράδειγμα, το ερώτημα `{{phd_student}}` θα επιστρέψει όλες τις δηλώσεις όπου εμφανίζεται σαν λέξη και επίσης τους γειτονικούς κόμβους του γράφου. Κατ' επέκταση το ερώτημα `{{{phd_student}}}` επεκτείνει τη "neighborhood" κατά ένα επίπεδο ακόμη.
3. Τελεστές AND και OR. Με τη χρήση των τελεστών αυτών μπορεί να δημιουργηθεί οποιοσδήποτε συνδυασμός των παραπάνω. Δεν επιτρέπεται όμως άρνηση.

2.7.2 Πρότυπα τριάδων: Οικογένεια SPARQL

Η οικογένεια SPARQL αποτελείται από τις SPARQL, SquishQL, RDQL και TriQL. Τα κοινά χαρακτηριστικά μεταξύ αυτών των γλωσσών είναι ότι αντιλαμβάνονται τα δεδομένα RDF ως απλές τριάδες χωρίς κάποιο σχήμα ή άλλη οντολογική πληροφορία εκτός κι αν αυτή έχει ρητά οριστεί στο RDF έγγραφο [52].

Οι SquishQL και RDQL προσφέρουν απλά μια πρόσβαση σε RDF γράφους χωρίς περαιτέρω δυνατότητες. Βασικό στόχο, όπως εξάλλου και στην RDQL αποτελεί η ευκολία στη χρήση και η ομοιότητα στην SQL. Η SquishQL – όπως και η RDQL – δε μπορεί να μεταχειριστεί σύνθετα ερωτήματα. Άρνηση μπορεί να χρησιμοποιηθεί στα φίλτρα ή στις δηλώσεις AND αλλά όχι στις δηλώσεις WHERE, που σημαίνει ότι μπορεί να γίνει έλεγχος για τριάδες που όντως υπάρχουν. Δεν μπορούν να εκφραστούν προαιρετικές συνθήκες, όπως για παράδειγμα με τη χρήση του OPTIONAL της SPARQL.

Η RDQL[58] έχει σύνταξη παρόμοια με αυτή της SQL. Είναι μια εξέλιξη της SquishQL επηρεασμένη από την rdfDB. Η σύνταξή της είναι παρόμοια με αυτή της SQL. Μια υλοποίηση της RDQL, η RDQLPlus παρέχει μια επέκταση στην RDQL χρησιμοποιώντας την inference engine της Jena. Η γλώσσα αυτή που προκύπτει ονομάζεται RIDIQL²³. Η RDQL υποστηρίζεται από την Jena, το RAP [59], το οποίο είναι ένα RDF API για τη γλώσσα PHP, και έχει υποβληθεί στο W3C για προτυποποίηση. Όπως και στην SquishQL, και εδώ τα ερωτήματα είναι ουσιαστικά ένα σύνολο μεταβλητών που υποβάλλονται ως pattern στο σύνολο των τριάδων. Από τις γλώσσες αυτές γίνεται αντιληπτό το επίπεδο τριάδων των RDF, δε λαμβάνεται υπόψη όμως η δομή του γράφου όπως για παράδειγμα στην RQL. Η TriQL επεκτείνει την RDQL υποστηρίζοντας επιπλέον την υποβολή ερωτημάτων σε named γράφους²⁴. Οι named γράφοι επιτρέπουν το

²³RDQLPlus: rdqlplus.sourceforge.net/doc/ridiql.html

²⁴Οι named γράφοι είναι η ιδέα ότι μπορούμε σε ένα έγγραφο να διατηρούμε πολλούς RDF γράφους, δίνοντάς τους ένα URI για όνομα, σε συμφωνία με τις προδιαγραφές του RDF

φιλτράρισμα δηλώσεων RDF ανάλογα με τον συγγραφέα τους. Για παράδειγμα, το ερώτημα “Επέστρεψε το τμήμα του γράφου το οποίο έχει υποβληθεί από τον John” εκφράζεται σε TriQL ως εξής:

```
SELECT ?graph
WHERE (?graph swp:assertedBy ?warrant)
(?warrant swp:authority <http://people.net/John>)
USING ex FOR http://www.example.org#,
swp FOR <http://www.w3.org/2004/03/trix/swp-1/>
```

Η SPARQL [60] είναι η εξέλιξη της BrQL και αποτελεί την επίσημη πρόταση του W3C²⁵. Η Jena όπως αναφέρεται και στην ενότητα 3.4.1 υποστηρίζει την ARQ και πρόκειται για μια από τις αρκετές υλοποιήσεις της SPARQL²⁶. Η SPARQL αποτελεί επέκταση της RDQL σε πολλά σημεία και στις δυνατότητές της συμπεριλαμβάνονται οι εξής:

1. Εξαγωγή RDF υπογράφων.
2. Δημιουργία νέων γράφων από αυτούς στους οποίους τίθενται τα ερωτήματα με τη χρήση της εντολής CONSTRUCT. Όπως και στα RDQL ερωτήματα, ο νέος γράφος μπορεί να καθοριστεί με τη χρήση patterns σε γράφο ή τριάδες.
3. Εξαγωγή περιγραφής των πόρων με τη χρήση της DESCRIBE. Η ιδιότητα αυτή δεν έχει καθοριστεί πλήρως, προβλέπεται όμως μελλοντική χρήση.
4. Καθορισμός του τελεστή OPTIONAL στα patterns ερωτημάτων με την έννοια ότι αν υπάρχουν δεδομένα στα ερωτηθέντα, τότε αυτά επιστρέφονται αλλιώς το ερώτημα δεν επηρεάζεται.
5. Εξέταση ύπαρξης ή μη τριάδων.
6. Υποστήριξη ταξινόμησης των αποτελεσμάτων.

Τα SPARQL ερωτήματα έχουν τη μορφή που δηλώνεται στον πίνακα 10. Ένα παράδειγμα SPARQL ερωτήματος είναι το εξής:

```
PREFIX ex: http://www.example.org#
SELECT ?name, ?title
FROM http://www.example.org
WHERE (?name ex:title ?title)
```

²⁵www.w3.org/TR/rdf-sparql-protocol

²⁶SPARQL Implementations: esw.w3.org/topic/SparqlImplementations

Δήλωση	Επεξήγηση
PREFIX	Καθορισμός ονόματος για ένα URI, αντίστοιχο με το USING της RDQL
SELECT	Επιστρέφει όλες ή κάποιες από τις μεταβλητές της WHERE
CONSTRUCT	Επιστρέφει νέο RDF γράφο με όλες ή κάποιες από τις αντιστοιχίσεις μεταβλητών
DESCRIBE	Επιστρέφει μια περιγραφή των πόρων που βρέθηκαν
ASK	Υποβάλλει ερώτηση αν το pattern ταιριάζει ή όχι
WHERE	Μια conjunctive λίστα με patterns σε τριάδες ή γράφο
OPTIONAL	Μια conjunctive λίστα με προαιρετικά patterns σε λίστα ή γράφο
AND	Λογική έκφραση που μπορεί να εφαρμοστεί στο αποτέλεσμα

Πίνακας 10: Δηλώσεις της SPARQL

Στα πολύ δυνατά σημεία της SPARQL είναι η δημιουργία γράφων με την εντολή CONSTRUCT, κάτι που απουσιάζει από τις RQL γλώσσες ερωτημάτων. Επίσης δυνατό σημείο της SPARQL είναι η δυνατότητα επιστροφής των αποτελεσμάτων σε XML, ή σε γράφο, με τη χρήση της GRAPH.

2.7.3 Σημασιολογικά χαρακτηριστικά της SPARQL

Το βασικό δομικό στοιχείο του Σημασιολογικού Ιστού είναι η τριάδα (triple). Ένα σύνολο από τριάδες αποτελεί ένα γράφο, όπως αναφέρθηκε στην ενότητα 1.5.1. Η τριάδα είναι επίσης το βασικό δομικό στοιχείο στα σημασιολογικά χαρακτηριστικά (semantics) των ερωτημάτων. Ένα παράδειγμα τριάδας, έστω t είναι το εξής:

$$t := (?X, name, ?Name).$$

Η τριάδα αποτελείται από τμήμα τριάδων του RDF, όπως το *name* στην τριάδα του παραδείγματος σε συνδυασμό με μεταβλητές, όπως οι $?X$ και $?Name$ του παραδείγματος.

Η βάση για την άλγεβρα της γλώσσας ερωτημάτων είναι το *βασικό πρότυπο γράφου* (Basic Graph Pattern, BGP) το οποίο είναι ένα σύνολο που αποτελείται από πρότυπα τριάδων (triple patterns):

$$\{t_1, t_2, \dots, t_n\}.$$

Παράδειγμα ενός βασικού πρότυπου γράφου είναι το εξής:

$$\{(?X, name, ?Name), (?X, email, ?Email)\}.$$

Θεωρούνται επίσης 3 βασικοί τελεστές (operators): *AND*, *UNION*, και *OPT*.

Επίσημα, ένα ερώτημα σε ένα γράφο αποστέλλεται με τη χρήση *προτύπων γράφων* (graph patterns) τα οποία σχηματίζονται με το συνδυασμό βασικών προτύπων γράφων και τελεστές. Ένα παράδειγμα προτύπου γράφου ή ερωτήματος σε SPARQL είναι το

εξής:

$$(((\{t_1, t_2\}ANDt_3)OPT\{t_4, t_5\})AND(t_6UNION\{t_7, t_8\})).$$

Για την εκτέλεση των ερωτημάτων, το βασικό δομικό στοιχείο είναι η *αντιστοίχιση*. Η αντιστοίχιση είναι μια συνάρτηση η οποία αντιστοιχίζει τις μεταβλητές του πρότυπου γράφου σε όρους του RDF. Επομένως, αν δίνεται μια αντιστοίχιση μ και ένα πρότυπο γράφου P , ορίζεται $dom(\mu)$ ως το domain του μ και $\mu(P)$ το σύνολο των τριάδων που ανακτώνται από το P αντικαθιστώντας τις μεταβλητές του μ .

Η *αποτίμηση* ενός προτύπου γράφου P σε έναν γράφο G , γράφεται $[[P]]_G$ και αποτελείται από το σύνολο των αντιστοιχίσεων που συναντώνται στο P με $\mu(P) \subseteq G$.

Έστω για παράδειγμα το τμήμα ενός γράφου G :

$(R_1, name, john)$
 $(R_1, email, J@ed.ex)$
 $(R_2, name, paul)$

και ένα ερώτημα P , έστω $\{(?X, name, ?Y)\}$. Η αποτίμηση του ερωτήματος P στον γράφο G δηλώνεται $[[\{(?X, name, ?Y)\}]]_G$

Οι αντιστοιχίσεις που προκύπτουν είναι οι

$$\mu_1 = \{?X \rightarrow R_1, ?Y \rightarrow john\}$$

$$\mu_2 = \{?X \rightarrow R_2, ?Y \rightarrow paul\}$$

επομένως το σύνολο των αποτελεσμάτων θα είναι

$?X ?Y$
 $(\mu_1) \quad R_1 \quad john$
 $(\mu_2) \quad R_2 \quad paul$

Στην πράξη, το τμήμα του RDF που περιγράφει τον γράφο G θα μπορούσε να είναι ως εξής, σε N3 notation:

```
@prefix default: <http://www.example.org/example.rdf#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

default:R1 default:name "john"^^xsd:string .
default:R1 default:email "J@ed.ex"^^xsd:string .
default:R2 default:name "paul"^^xsd:string .
```

Επομένως, το ερώτημα του παραδείγματος γράφεται:

```
PREFIX default: http://www.example.org/example.rdf#  
SELECT ?X, ?Y  
WHERE (?X default:name ?Y)
```

Η αποτίμηση του ερωτήματος αυτού θα επιστρέψει, όπως και στο παράδειγμα θεωρητικής εκτέλεσης του ερωτήματος, τις δυο στήλες *?X* και *?Y*.

2.8 Συμπεράσματα

Για μεγάλο χρονικό διάστημα, απουσίαζε το πρότυπο για μια γλώσσα ερωτημάτων για έγγραφα RDF. Πολλές προσεγγίσεις προτάθηκαν, όπως παρουσιάστηκε στο κεφάλαιο 2.7, με αρκετά ανεπτυγμένες γλώσσες. Η προτυποποίηση μιας γλώσσας ερωτημάτων για την RDF υπήρξε μια από τις άμεσες προτεραιότητες της έρευνας στον τομέα αυτό. Όντως, από τις 15 Ιαν. 2008 η SPARQL είναι η προτεινόμενη από το W3C γλώσσα για ερωτήματα στο Σημασιολογικό Ιστό. Δίχως την καθιέρωση προτύπου, το πρότυπο RDF δε θα μπορούσε να οδηγηθεί στην ευρεία αποδοχή και διάδοσή του [30], όπως για παράδειγμα η καθιέρωση της SQL οδήγησε στην καθιέρωση των σχεσιακών Βάσεων Δεδομένων.

Η συνολική εικόνα των υπόλοιπων γλωσσών ερωτημάτων εμφανίζεται ελλιπής. Παρότι είναι αρκετά ανεπτυγμένες, δεν είναι τόσο ολοκληρωμένες όσο χρειάζεται. Η βασικότερη έλλειψη που διαπιστώνεται είναι η ταξινόμηση των αποτελεσμάτων, κάτι αντίστοιχο με την `ORDER BY` της SQL – με εξαίρεση την *Versa* και την SPARQL. Απουσιάζει επίσης η υποστήριξη `grouping` των αποτελεσμάτων, όπως το `GROUP BY` της SQL. Δεν υποστηρίζονται `aggregated` ερωτήματα, όπως για παράδειγμα συναρτήσεις `MIN`, `MAX` ή `SUM`. Ωστόσο, υποστηρίζονται σε ικανοποιητικό βαθμό τα προαιρετικά αποτελέσματα. Οι περισσότερες γλώσσες προσφέρουν λειτουργικότητα αντίστοιχη με την `OPTIONAL` της SPARQL με την οποία επιστρέφονται αποτελέσματα στην περίπτωση που υπάρχουν τιμές ενώ δεν επηρεάζονται όταν δε βρεθούν τιμές.

Όσον αφορά τα επόμενα βήματα και τις προκλήσεις που χρειάζεται να αντιμετωπιστούν για την ευρύτερη χρησιμοποίηση γλωσσών ερωτημάτων, σημαντικό κρίνεται το ζήτημα της ενσωμάτωσης με τις υπάρχουσες τεχνολογίες. Για εφαρμογές πραγματικού κόσμου, η ενσωμάτωση τεχνολογιών Σημασιολογικού Ιστού δεν θα πρέπει να αντικαθιστά τις υπάρχουσες πρακτικές. Στην κατεύθυνση αυτή χρειάζεται να κινηθεί η ανάπτυξη των γλωσσών, ούτως ώστε να ενσωματώνουν λειτουργικότητα για αξιοποίηση Βάσεων Δεδομένων ή άλλων προτύπων οργάνωσης πληροφορίας. Για

παράδειγμα, υλοποιήσεις όπως η LARQ²⁷ η οποία δίνει τη δυνατότητα ερωτημάτων βασισμένων σε λέξεις κλειδιά παράλληλα με τη δυνατότητα υποβολής σημασιολογικών ερωτημάτων, είναι μια κατεύθυνση στην οποία χρειάζεται να κινηθεί η έρευνα. Πιο τεχνικά, προκειμένου να αξιοποιηθεί η SPARQL σε πραγματικές εφαρμογές θα ήταν επιθυμητή η εμφώλευση ερωτημάτων και σε άλλα πρότυπα, όπως για παράδειγμα XQuery. Σημειώνεται ότι η SQL – και συγκεκριμένα η TSQL, η υλοποίηση του προτύπου SQL που προσφέρει ο Microsoft SQL Server 2005 – υποστηρίζει ερωτήματα της μορφής:

```
SELECT MyXml.query(
    'declare namespace ex="http://www.example.org";
     /ex:root/ex:product[@ex:id="3024"]/ex:name' )
FROM ExampleTable
```

²⁷Η LARQ είναι ο συνδυασμός μιας υλοποίησης της SPARQL (ARQ) με την Lucene, την μηχανή για αναζήτηση σε κείμενο του Apache Group (jena.sourceforge.net/ARQ/lucene-arq.html)

3 Διαχείριση, επεξεργασία και αξιοποίηση σημασιολογικής πληροφορίας

3.1 Εισαγωγή

Με τον όρο “διαχείριση συστήματος” ορίζεται κάθε ενέργεια που στόχο έχει τις δραστηριότητες που σχετίζονται με θέματα:

- παρακολούθησης των ενεργειών του συστήματος,
- καταγραφής των ενεργειών των χρηστών και του συστήματος,
- χειρισμού των επιδόσεων του συστήματος με στόχο τη βελτίωσή τους,
- εγκατάστασης κι απεγκατάστασης λογισμικού,
- ρύθμισης πολιτικών ασφαλείας στη χρήση,
- εξασφάλισης διαθεσιμότητας.

Ο όρος “επεξεργασία σημασιολογικής πληροφορίας” αναφέρεται στις ενέργειες που στόχο έχουν:

- την προσθήκη και αφαίρεση σημασιολογικής πληροφορίας από το σύστημα,
- την αντιστοίχιση σημασιολογικών μεταδεδομένων με πραγματικά δεδομένα,
- την εξαγωγή συμπερασμάτων από την υπάρχουσα γνώση με τη χρήση υπηρεσιών συλλογιστικής.

Ως “αξιοποίηση σημασιολογικής πληροφορίας” ορίζεται:

- η υποβολή και αποτίμηση σημασιολογικών ερωτημάτων υψηλού επιπέδου,
- η δημιουργία Βάσης Γνώσεως η οποία αναπαριστά πληροφορία με σαφώς καθορισμένη σημασιολογία,
- η δυνατότητα εξαγωγής της πληροφορίας αυτής σε μορφή κατανοητή τόσο από τη μηχανή όσο και από τον άνθρωπο.

Για τη διαχείριση, επεξεργασία και αξιοποίηση της σημασιολογικής πληροφορίας έχει αναπτυχθεί μεγάλη ποικιλία από προγράμματα και πρότυπα τα οποία διευκολύνουν τις διαδικασίες αυτές. Στο κεφάλαιο αυτό παρουσιάζεται μια ανασκόπηση της βιβλιογραφίας σχετικά με τον τρόπο κατά τον οποίο μπορεί κανείς να παράξει σημασιολογία, να την επεξεργαστεί αλλά και να την αξιοποιήσει εκμεταλλευόμενος την επιπλέον γνώση που μπορεί να εξάγει. Παρατίθενται επίσης τα σημαντικότερα εργαλεία και τα ουσιαστικότερα αποτελέσματα ερευνών σε χώρους συναφείς με το Σημασιολογικό Ιστό. Ειδικότερα, παρουσιάζονται τα βασικότερα εργαλεία διαχείρισης σημασιολογικής πληροφορίας και εξετάζονται οι επιπτώσεις που έχει η προσθήκη σημασιολογίας στα συστήματα με επίγνωση περιβάλλοντος, στις μηχανές αναζήτησης, στα συστήματα επεξεργασίας πολυμεσικής πληροφορίας.

3.2 Σύνταξη οντολογιών

Η δημιουργία μιας οντολογίας δεν είναι διαδικασία γραμμική, υπό την έννοια του ότι δε μπορεί κανείς να κατηγοριοποιήσει τα βήματα που χρειάζονται για τη συγγραφή της. Μπορεί κανείς να προσεγγίσει το επιθυμητό αποτέλεσμα από πολλές απόψεις, είτε από πιο γενικά σε πιο αφηρημένα είτε το αντίστροφο. Είναι επίσης επαναληπτική διαδικασία καθώς μια δομή η οποία αποτελεί το σκελετό της οντολογίας μπορεί να εμπλουτιστεί με την προσθήκη πιο εξειδικευμένων περιφερειακών εννοιών προκαλώντας πιο σύνθετες σχέσεις μεταξύ τους. Όσο πιο πολύ προχωράει η ανάπτυξη της οντολογίας τόσο πιο σύνθετη γίνεται η προσθήκη επιπλέον σημασιολογίας.

Η διαδικασία ανάπτυξης μιας οντολογίας μπορεί να εμπεριέχει προβλήματα όπως για παράδειγμα στην πλήρη αποσαφήνιση των εννοιών, την απόκτηση και ανάλυση του πεδίου γνώσεως ενός τομέα, το θεμελιώδη σχεδιασμό και την υπακοή σε πιθανόν υπάρχουσες οντολογίες. Στη συνέχεια ακολουθεί η διαδικασία συγγραφής με την παράλληλη αναθεώρηση του εννοιολογικού περιεχομένου, ο καθορισμός της ορολογίας (terminology) της οντολογίας και πιθανότατα η συμπλήρωσή της με άτομα (individuals). Πρόκειται για μια διαδικασία πιο δύσκολη απ' ό,τι ίσως ακούγεται, μάλλον επίπονη και συνήθως μη γραμμική. Γι' αυτό το λόγο έχει αναπτυχθεί μια πληθώρα εργαλείων που να διευκολύνει τη διαδικασία συγγραφής.

Επιπλέον, κατά την ανάπτυξη του Σημασιολογικού Ιστού με απώτερο στόχο την ενοποίηση της παγκόσμιας πληροφορίας χρειάζεται η προσθήκη περιεχομένου από άτομα κάθε επιπέδου εξοικείωσης. Οι οντολογίες όμως κατά κανόνα δεν είναι κατανοητές στο μη εξοικειωμένο μάτι. Για παράδειγμα, ένα τμήμα οντολογίας γραμμένο σε OWL είναι απαγορευτικά δυσνόητο για τον απλό χρήστη του Διαδικτύου που απλά επιθυμεί να προσθέσει σημασιολογικό περιεχόμενο στη σελίδα του. Δεν είναι εύκολα αναγνώσιμη κατά γενική ομολογία η πληροφορία αυτή, είναι επομένως απαραίτητη η παροχή εργαλείων συγγραφής οντολογιών. Επιπλέον, χρειάζεται έλεγχος συνέπειας για την κάθε οντολογία. Ο δημιουργός της κάθε οντολογίας δεν έχει την απόλυτη ελευθερία να εισάγει περιεχόμενο, έστω και σωστό συντακτικά. Είναι αναγκαίο οι έννοιες που θα περιγράφονται στην οντολογία να μην είναι αλληλοσυγκρουόμενες.

Παρά όμως το γεγονός ότι υπάρχουν περισσότεροι από ενενήντα συντάκτες οντολογιών, στην πράξη ελάχιστοι χρησιμοποιούνται γιατί ελάχιστοι παρέχουν ικανή βοήθεια στην ανάπτυξη. Τα εργαλεία που παρουσιάζονται στη συνέχεια έχουν επιλεγεί λόγω των δυνατοτήτων τους και της συνεχούς αναβάθμισής τους η οποία φανερώνει την κοινότητα που τα αναπτύσσει, συντηρεί και βελτιώνει. Χρειάζεται επίσης να αναφερθεί η ύπαρξη μιας σειράς από εμπορικούς ontology editors όπως ο Altova Semantic Works 2009 και ο TopBraid Composer, γεγονός που φανερώνει την αναγκαιότητα υποστήριξης

στη διαδικασία συγγραφής οντολογιών.

Χρειάζεται επίσης να αναφερθεί ότι υπάρχει μια σειρά από παλιότερα περιβάλλοντα συγγραφής οντολογιών τα οποία έφτασαν σε εκδόσεις με αξιοπρόσεκτα χαρακτηριστικά. Φαίνεται όμως ότι οι ομάδες που τα ανέπτυξαν σταμάτησαν την υποστήριξή τους. Μπορεί να μη χρησιμοποιούνται σήμερα, αλλά τα OntoSaurus, Ontolingua, Ontotrack [61], OilEd²⁸ και WebOnto παρείχαν πριν λίγα χρόνια πεδίο για έρευνα και ανάπτυξη και συμβάλλανε στη διαμόρφωση της σημερινής εικόνας του Σημασιολογικού Ιστού. Επίσης σημαντικό είναι και το OilEd [62], το οποίο μπορεί να διαχειριστεί RDF, DAML+OIL ή OWL αρχεία ενώ συνεργάζεται και με τον FaCT reasoner. Πρόκειται για ένα αρκετά προχωρημένο περιβάλλον συγγραφής οντολογιών του οποίου η αρχική έκδοση υποστήριζε τη γλώσσα OIL. Η ανάπτυξη και υποστήριξή του όμως έχει σταματήσει. Η τελευταία έκδοση (3.5.7) παραμένει η ίδια από το Μάρτιο του 2004. Υπάρχει επίσης ο Web Editor pOWL²⁹ [63] ο οποίος βασίζεται στο RAP [59]. Ο pOWL έχει περιβάλλον που υλοποιείται μέσα στον περιηγητή (browser). Υποστηρίζει RDQL ερωτήματα, όπως επίσης και αναζήτηση κειμένου σε RDF και OWL οντολογίες. Διατίθεται σε δυο μεταφράσεις, αγγλικά και γερμανικά. Το IsaViz³⁰ είναι ένα εργαλείο για τη συγγραφή RDF γράφων. Το IsaViz υποστηρίζει την Fresnel, μια γλώσσα για την οπτική παρουσίαση RDF γράφων που παρουσιάστηκε στο [64].

Σύμφωνα με το [65], από τα πιο δημοφιλή πλέον περιβάλλοντα συγγραφής οντολογιών είναι το Protégé (68,2%), με δεύτερο το SWOOP (13,6%) και τρίτο το OntoEdit (12,2%). Οι περισσότεροι από τους συμμετέχοντες στην έρευνα ανήκουν στον ακαδημαϊκό τομέα.

3.2.1 Protégé

Το Protégé το οποίο είναι βασισμένο σε Java και ανοιχτού κώδικα αποτελεί αυτή τη στιγμή την πιο ολοκληρωμένη λύση για συγγραφή και διαχείριση οντολογιών. Πρόκειται για το πιο παλιό και πιο πλήρες περιβάλλον διαχείρισης οντολογιών. Μετά από αρκετά χρόνια ανάπτυξης έχει ήδη φτάσει στην έκδοση 3.3.1 (Αύγουστος 08) ενώ η κοινότητα εγγεγραμμένων χρηστών ξεπερνά τα 100.000 μέλη³¹. Το Protégé αριθμεί πληθώρα plugins³² για οτιδήποτε σχετικό με RDF(S) και OWL. Υποστηρίζει πλήρως τη γλώσσα OWL και συνεργάζεται με εξωτερικούς εξυπηρετητές υπηρεσιών συλλογιστικής σύμφωνα με

²⁸OilEd: oiled.man.ac.uk/

²⁹OntoWiki / pOWL: powl.sourceforge.net

³⁰IsaViz: www.w3.org/2001/11/IsaViz/

³¹Επίσημη ιστοσελίδα του Protégé: protege.stanford.edu

³²Protégé plugins: protege.stanford.edu/download/plugins.html

τη διαλειτουργικότητα DIG (περισσότερα για το DIG στην ενότητα 3.6). Είναι απόλυτα παραμετροποιήσιμο και επεκτάσιμο σε σημείο να αποτελεί τη βασικότερη επιλογή δωρεάν προσφερόμενων περιβαλλόντων για συγγραφή οντολογιών. Η ευχρηστία είναι βασικό χαρακτηριστικό του Protégé. Υπάρχει βοήθεια για οτιδήποτε στην οθόνη του χρήστη και μπορεί κανείς να ξεκινήσει την ανάπτυξη της οντολογίας χωρίς να έχει ιδιαίτερο θεωρητικό υπόβαθρο.

Αξιοσημείωτο είναι ότι το Protégé υποστηρίζει την αποθήκευση μιας οντολογίας σε Βάση Δεδομένων, περισσότερο όμως αυτό γίνεται για λόγους διατήρησης της οντολογίας. Δεν υπάρχει αντιστοιχία μεταξύ του αναπτυσσόμενου μοντέλου και του σχήματος της Βάσης Δεδομένων κατά τρόπο ώστε να είναι δυνατή η αξιοποίηση της Βάσης από τρίτους. Οι δημιουργοί του Protégé “απολογούνται” για το σχήμα αυτό της Βάσης³³ εξηγώντας ότι προτιμητέο είναι οι developers να αναπτύσσουν τα δικά τους σχήματα Βάσης εφόσον τους ενδιαφέρει.

3.2.2 SWOOP και SMORE

Ο SWOOP αποτελεί έναν διαχειριστή οντολογιών με έμφαση στο ελαφρύ και λειτουργικό περιβάλλον. Υλοποιήθηκε από το εργαστήριο UMIACS³⁴ του πανεπιστημίου του Maryland αλλά πλέον αποτελεί έργο ανοιχτού κώδικα. Ο SWOOP έχει το χαρακτηριστικό ότι μπορεί να δημιουργεί partitions σε οντολογίες μετατρέποντάς τις σε *ε-Connections*³⁵. Πρόκειται για ένα λιτό περιβάλλον, ταυτόχρονα όμως με ευχρηστία, ταχύτητα και σταθερότητα. Είναι υλοποιημένος σε Java, υποστηρίζει αρχεία OWL και παρέχει δυνατότητα αναζήτησης. Δεν παρέχει γραφική αναπαράσταση της οντολογίας και δεν υπάρχει υποστήριξη από plugins.

Η ίδια ομάδα αναπτύσσει και τον SMORE, τα αρχικά του οποίου προέρχονται από το “Semantic Markup, Ontology and RDF Editor” [67]. Ο SMORE είναι σχεδιασμένος ώστε να επιτρέπει τη συγγραφή OWL εγγράφων ταυτόχρονα με τη σχεδίαση ιστοσελίδων καθώς διαθέτει ενσωματωμένο και τον Ekit HTML Editor. Μπορεί ο χρήστης να σχεδιάσει έναν δικτυακό τόπο, να σχεδιάσει παράλληλα και τα semantics (οντολογία) και να δημιουργήσει μια ολοκληρωμένη, σημασιολογικά τεκμηριωμένη σελίδα. Είναι, επομένως, ένα ισχυρό εργαλείο το οποίο αποτελεί στην ουσία επέκταση του SWOOP. Ιδιαίτερη βαρύτητα δίνει ο SMORE στην υποστήριξη της OWL. Ενώ οι υπόλοιποι edi-

³³Why we did it the way we did: protege.stanford.edu/doc/design/jdbc_backend.html

³⁴University of Maryland Institute for Advanced Computer Studies, www.mindswap.org

³⁵Οι *ε-Connections* που παρουσιάζονται στο [66] είναι μια επέκταση της OWL DL και παρέχει εναλλακτική στη δήλωση `owl:imports`

tors υποστηρίζουν την RDF κατά κύριο λόγο, ο SMORE είναι προσανατολισμένος στην επεξεργασία OWL εγγράφων.

Ο SMORE είναι κατά πολύ πιο ελαφρύς editor από το Protégé και είναι όπως και ο SWOOP γρήγορος και εύχρηστος. Η διαφορά του από τον SWOOP είναι ότι ενώ ο SWOOP είναι ένα εργαλείο καθαρά για τη συγγραφή οντολογιών, ο SMORE χρησιμεύει για την προσθήκη οντολογικών επισημειώσεων σε περιεχόμενο ιστοσελίδων. Στις δυνατότητές του περιλαμβάνεται η αυτόματη δημιουργία κλάσεων, ιδιοτήτων και individuals από επιλεγμένο κείμενο που εισάγεται από το χρήστη στο υπό ανάπτυξη site στον HTML Editor.

Ο individual editor του SMORE χρησιμοποιεί τους περιορισμούς στο domain και το range των individuals για την παροχή των πιθανών links στη σχεδίαση των ατόμων (individuals) της οντολογίας. Για τις ιδιότητες τύπου δεδομένων (datatype properties) παρέχεται ο ανάλογος editor. Όσον αφορά τη διαχείριση και αποφυγή λαθών, ο SMORE παρέχει λειτουργίες αποσφαλμάτωσης, αναγνωρίζοντας τις τριάδες στον πίνακα των τριάδων που δεν είναι σύμφωνες με τους περιορισμούς. Μπορεί εύκολα να γίνεται έλεγχος συνέπειας της οντολογίας.

3.3 Σημασιολογική επισημείωση

Η σημασιολογική επισημείωση αφορά την αντιστοίχιση σημασιολογικών μεταδεδομένων με τα πραγματικά δεδομένα. Αποτελεί δραστηριότητα πιο στοχευμένη από την απλή συγγραφή οντολογίας καθώς εδώ ο στόχος δεν είναι η μοντελοποίηση ενός πεδίου ενδιαφέροντος αλλά η μοντελοποίηση υπάρχουσας πληροφορίας.

Η επισημείωση περιεχομένου διαχωρίζεται σε δυο είδη: χειροκίνητη και αυτοματοποιημένη, ανάλογα με το αν απαιτείται ανθρώπινη παρέμβαση, η οποία συνήθως υποβοηθείται από αλγόριθμους ημι-αυτόματης εξαγωγής πληροφορίας, ή όχι. Η πρώτη προσέγγιση περιλαμβάνει εργαλεία όπως το COHSE [68] το οποίο στοχεύει στην επισημείωση περιεχομένου κατά τη διάρκεια του χρόνου ανάκτησης, καθώς οι αναγνώστες διαβάζουν το έγγραφο, ή κατά το χρόνο που οι χρήστες συντάσσουν το έγγραφο. Ο Amaya³⁶, είναι ο editor/browser του W3C που υποστηρίζει επισημείωση και ομοίως ο SMORE ανήκει στην ίδια κατηγορία και αναφέρεται σε περιεχόμενο του Διαδικτύου. Το Lixto [69] είναι εργαλείο το οποίο δίνει τη δυνατότητα στους χρήστες να δημιουργούν δυναμικά προσαρμοστές έχοντας σαν βάση συσχετιζόμενες πληροφορίες. Επιπλέον το Thresher [70] δημιουργεί δυναμικά RDF γράφους καθώς οι χρήστες κάνουν πλοήγηση στο Διαδίκτυο. Επίσης, το SHOE [71] επιτρέπει στους

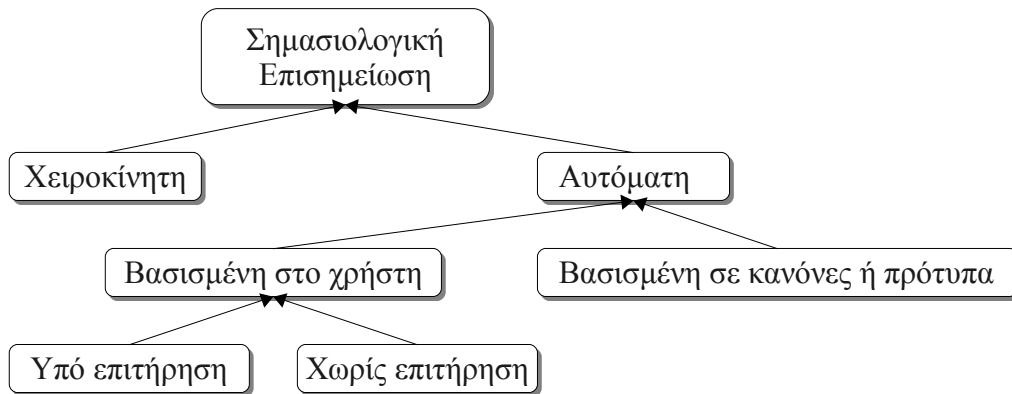
³⁶Amaya: www.w3.org/Amaya

χρήστες να επισημειώνουν HTML σελίδες βάσει αποθηκευμένων τοπικά ή εξωτερικών οντολογιών.

Η αυτόματη επισημείωση μπορεί να διαιρεθεί περαιτέρω σε δύο κατηγορίες:

- Βασισμένη στον χρήστη
- Βασισμένη σε κανόνες

Οι τεχνικές επισημείωσης που είναι βασισμένες στον χρήστη μπορούν να διαιρεθούν επιπλέον σε δύο κατηγορίες, ανάλογα με το αν τελούν υπό επιτήρηση ή όχι. Οι προσεγγίσεις επισημείωσης που δεν χρησιμοποιούν τεχνικές επιτήρησης περιλαμβάνουν το MnM [72] που παρέχει ένα περιβάλλον για χειροκίνητη επισημείωση. Το Melita [73] είναι ένα εργαλείο αυτόματης επισημείωσης το οποίο υποστηρίζεται από την Amilcare, μια μηχανή εξαγωγής πληροφορίας. Ο στόχος του έργου αυτού είναι να αλλάξει σταδιακά το ρόλο του χρήστη στην διαδικασία επισημείωσης. Στις τεχνικές που δεν χρησιμοποιούν επιτήρηση περιλαμβάνονται οι Armadillo [74] και KnowItAll [75] που αυτοματοποιούν την διαδικασία εξαγωγής πληροφορίας με παρόμοιο τρόπο. Στο έργο SmartWeb διερευνάται μια “μη επιτηρούμενη” προσέγγιση βασισμένη σε RDF. Η εικόνα 5 δίνει μια σχηματική αναπαράσταση της κατηγοριοποίησης των εργαλείων που εξετάστηκαν.



Εικόνα 5: Κατηγοριοποίηση των προσεγγίσεων σημασιολογικής επισημείωσης

Συνολικά, όμως παρατηρείται ότι πρόκειται για ένα χώρο στον οποίο ελάχιστα έχουν γίνει και υπάρχει ακόμη πεδίο για έρευνα και ανάπτυξη ώστε να παραχθούν πιο έξυπνες εφαρμογές και να καλυφθούν οι ανάγκες για περισσότερη αυτοματοποίηση. Στο κεφάλαιο 5 παρουσιάζεται η συμβολή της παρούσας διατριβής στην κατεύθυνση ενσωμάτωσης σημασιολογικής επισημείωσης σε πολυμεσική πληροφορία, βασισμένη σε κανόνες.

3.4 Προγραμματιστικές διεπαφές

3.4.1 Jena

Η Jena [76] αποτελεί μια προγραμματιστική διεπαφή (API) σε Java, είναι ανοιχτού κώδικα και κατά κύριο λόγο παρέχει υποστήριξη για τη διαχείριση οντολογιών και προγραμματισμό εφαρμογών Σημασιολογικού Ιστού. Αναπτύσσεται κατά κύριο λόγο από τα Hewlett Packard Labs. Η πρώτη έκδοση της Jena ήταν διαθέσιμη το 2000. Η αρχιτεκτονική της επανασχεδιάστηκε και η Jena2 έγινε διαθέσιμη τον Αύγουστο του 2003, αποτελώντας πλέον την πρακτικότερη λύση για προγραμματιστική διαχείριση οντολογιών. Η Jena παρέχει ουσιαστικά μια συλλογή εργαλείων. Εκτός από το API για διαχείριση οντολογιών, ενσωματώνει επίσης τα εξής:

- Υλοποίηση του πρωτοκόλλου SPARQL (ARQ). Η γλώσσα ερωτημάτων της Jena ήταν η RDQL μέχρι την έκδοση 2.2. Από την 2.3 και μετά η γλώσσα που υιοθετείται είναι η SPARQL ενώ η RDQL εξακολουθεί να υποστηρίζεται για λόγους προς τα πίσω συμβατότητας.
- διαχείριση τριάδων αποθηκευμένων σε Βάση Δεδομένων (SDB).
- εξυπηρετητής εγγράφων RDF μέσω HTTP (Joseki). Ο Joseki μπορεί να διαχειρίζεται μέσω της διεπαφής Jena τριάδες οι οποίες μπορεί να υπάρχουν σε έγγραφο RDF σε οποιαδήποτε μορφή διάταξης (RDF/XML, N3, Turtle) είτε σε μοντέλο αποθηκευμένο σε Βάση Δεδομένων σύμφωνα με τις προδιαγραφές της Jena.
- Η Jena μπορεί να λειτουργήσει επίσης ως εξυπηρετητής υπηρεσιών συλλογιστικής καθώς ενσωματώνει μηχανή εξαγωγής συμπερασμάτων η οποία βασίζεται σε κανόνες Jena. Μπορεί να εξάγει συμπεράσματα από τα μοντέλα που της δίνονται και υποστηρίζει σχεδόν κάθε έκφραση RDFS. Όσον αφορά όμως την OWL DL, η Jena είναι ακόμα ημιτελής. Παρέχει τρεις υλοποιήσεις για OWL reasoners, standard, mini και macro αλλά κανένας από αυτούς δεν καλύπτει την εκφραστικότητα της OWL DL. Υποστηρίζεται ωστόσο η διαλειτουργικότητα DIG για τη σύνδεση των μοντέλων με εξωτερικό reasoner. Οι κανόνες της μηχανής εξαγωγής συμπερασμάτων της Jena έχουν τη μορφή

```
[rdfs5a: (?a rdfs:subPropertyOf ?b), (?b rdfs:subPropertyOf ?c)
-> (?a rdfs:subPropertyOf ?c)]
```

Ο εσωτερικός αυτός reasoner μπορεί να υποστηρίξει DAML, RDFS και OWL (Mini και Micro reasoner) ανάλογα με τις απαιτήσεις της κάθε εφαρμογής ενώ ο

χρήστης μπορεί επίσης να επεκτείνει τη λειτουργία των κανόνων γράφοντας τους δικούς του και δημιουργώντας τις δικές του επαγωγικές διαδικασίες συλλογιστικής.

Στη βάση των εργαλείων της Jena είναι η δομή των γράφων (Graph Layer) η οποία αποτελείται από τριάδες (subject, predicate, object). Η αποθήκευση των γράφων μπορεί να γίνει είτε στην κύρια μνήμη είτε σε Βάση Δεδομένων. Το επίπεδο αφαίρεσης στην αποθήκευση επιτρέπει τη συνεργασία με πρακτικά οποιαδήποτε μορφή αποθήκευσης. Πάνω από το επίπεδο των γράφων βρίσκεται το επίπεδο των ενισχυμένων γράφων (Enh-Graph Layer) το οποίο είναι ουσιαστικά η διαλειτουργικότητα με τα παραπάνω επίπεδα. Το επίπεδο αυτό σχεδιάστηκε ώστε να επιτρέπει δημιουργία όψεων (views) πάνω στο γράφο και παράλληλα είναι σημαντικό ώστε οι δηλώσεις του RDFS να μπορούν να αντιστοιχηθούν σε συναρτήσεις Java. Το επίπεδο του μοντέλου (Model Layer) είναι αυτό με το οποίο ο προγραμματιστής μπορεί να έχει πρόσβαση μέσω ενός πλούσιου συνόλου από μεθόδους και συναρτήσεις τόσο στους κόμβους του γράφου όσο και στην ίδια τη δομή του γράφου. Οι γλώσσες περιγραφής οντολογιών που υποστηρίζονται από τη Jena είναι οι RDF(S), DAML+OIL και OWL.

Σημαντική είναι και η παρουσία του Joseki SPARQL Server (www.joseki.org). Ο Joseki είναι ανοιχτού κώδικα, δημιουργήθηκε και αναπτύσσεται από την Hewlett Packard, από τους συντελεστές που αναπτύσσουν την Jena. Η ιδέα πίσω από τον Joseki είναι η δημοσίευση στο Διαδίκτυο δεδομένων που είναι αποθηκευμένα στο σύστημα αποθήκευσης γράφων της Jena. Με τη χρήση του Joseki, μια Βάση Γνώσεως δημιουργημένη σε Jena αλλά και οποιαδήποτε αποθήκη σημασιολογικών δεδομένων αποκτά υποστήριξη ερωτημάτων σε SPARQL μέσω διαδικτυακής διεπαφής (http GET και POST ερωτημάτων). Πρόκειται για ένα ιδιαίτερα δραστήριο έργο το οποίο έχει φτάσει στην έκδοση 3.2 και υποστηρίζεται από την κοινότητα της Jena.

Τέλος, αξίζει να σημειωθεί ότι η mailing list της κοινότητας όσων προγραμματίζουν χρησιμοποιώντας τις βιβλιοθήκες της Jena (jena-dev@yahoogroups.com), αριθμεί περισσότερα από 2500 μέλη (Σεπτέμβριος 2009), γεγονός που φανερώνει ότι πρόκειται για ιδιαίτερα δραστήριο έργο.

3.5 Η οντολογία Cyc

Ένα χαρακτηριστικό παράδειγμα μέτριας συμπεριφοράς είναι η οντολογία Cyc [77] η οποία συντηρείται από την εταιρεία Cycorp, και η αντίστοιχη ελεύθερη έκδοσή της, η OpenCyc η οποία διατίθεται υπό την ειδική άδεια Apache Licence³⁷. Το έργο ξεκίνησε

³⁷Apache Licence: www.apache.org/licenses/LICENSE-2.0

επίσημα το 1984 με στόχο την καταγραφή της ανθρώπινης γνώσεως και είναι υπό ανάπτυξη ακόμα και σήμερα.

Πρόκειται για μια από τις μεγαλύτερες Βάσεις Γνώσεως που έχουν δημιουργηθεί καθώς περιλαμβάνει εκατοντάδες χιλιάδων όρους και εκατομμύρια ισχυρισμούς που συνδέουν τους όρους μεταξύ τους. Πρόκειται για μια οντολογία που το πεδίο ορισμού της καλύπτει σχεδόν το σύνολο της ανθρώπινης γνώσεως. Ανάμεσα στις έννοιές της βρίσκονται έννοιες οι οποίες έχουν παρθεί από θέματα που αφορούν πολλά επιστημονικά πεδία όπως για παράδειγμα, τα γεωγραφικά δεδομένα ή θέματα που σχετίζονται με το χρόνο. Η φιλοδοξία της δηλώνεται και από το όνομά της καθώς τα γράμματα cyc προέρχονται από τη λέξη enCYClopedia, και αποτελεί την καταγραφή των πιο γενικών εννοιών της ανθρώπινης πραγματικότητας.

Από την έκδοση 1.0 η οποία εκδόθηκε τον Ιούλιο του 2006, η Cyc προσφέρει μαζί με τη Βάση Γνώσεως εργαλεία για την εξαγωγή OWL οντολογιών, απάντηση DAML ερωτημάτων και αλληλεπίδραση με τη γλώσσα Java (Java API). Σαν εργαλείο, η Cyc προσφέρεται για τη γρήγορη ανάπτυξη οντολογιών επεκτείνοντας ή περιορίζοντας την υπάρχουσα γνώση σύμφωνα με τις εκάστοτε ανάγκες. Επιπλέον, μπορεί να χρησιμεύει σε πεδία όπου μια μεγάλη Βάση Γνώσεως είναι απαραίτητη όπως στην αναγνώριση φωνής ή τη δημιουργία λεξικών. Έχει υλοποιηθεί αντιστοίχιση των εννοιών της οντολογίας Cyc στο online λεξικό WordNet [78] το οποίο έχει αναπτυχθεί κατά κύριο λόγο στο Πανεπιστήμιο του Princeton.

Σημαντική μπορεί να είναι η χρήση της Cyc και σε έγγραφα του Διαδικτύου καθώς θα μπορούσαν αυτά να παρέχουν συνδέσμους σε έννοιες της Cyc, κατ' αναλογία με την DBPedia (ενότητα 2.5).

Η OpenCyc 1.0 περιλαμβάνει ολόκληρη την Cyc οντολογία η οποία περιέχει 47.000 έννοιες (concepts), 306.000 δεδομένα (facts) και μπορεί κανείς να περιηγηθεί σε αυτήν μέσα στον ιστότοπό της (www.openencyc.org). Η πρώτη έκδοσή της, το Μάιο του 2001 περιείχε μόνο 6.000 έννοιες και 60.000 δεδομένα. Η OpenCyc έχει δεχθεί κριτική για τα πολλά αρνητικά της σημεία τα οποία προκύπτουν από τη μεγάλου βαθμού πολυπλοκότητά της. Είναι πλέον εξαιρετικά δύσκολο για τον αρχάριο χρήστη να τη χρησιμοποιήσει επιτυχώς, χρειάζεται αρκετή τεκμηρίωση η οποία δυστυχώς δεν προσφέρεται στην ιστοσελίδα. Έχει πλέον σοβαρά προβλήματα επεκτασιμότητας. Επίσης, δεν υπάρχουν εργαλεία με τα οποία θα μπορούσε κανείς να μετρήσει την απόδοσή της ή το κατά πόσο καλύπτει τη γνώση που ισχυρίζεται. Δεν υπάρχει καν κάποιο συναφές έργο με το οποίο θα μπορούσε να συγκριθεί η OpenCyc.

Τον Ιούλιο του 2006, η Cycorp, η εταιρεία που συντηρεί την Cyc κατέστησε διαθέσιμη και την ResearchCyc 1.0, μια ελεύθερη αλλά όχι ανοιχτού κώδικα οντολογία. Η Research-

Cyc προσφέρει περισσότερη σημασιολογικά εμπλουτισμένη πληροφορία σε σχέση με την Cyc, περιέχει ένα μεγάλο λεξικό όρων στην Αγγλική γλώσσα και συνοδεύεται από πλήθος εργαλείων και διεπαφών για τη διαχείριση της γνώσεως που προσφέρει.

Δυστυχώς, η OWL έκδοση της OpenCyc είναι ένα τεράστιο αρχείο, μεγαλύτερο από 700MB το οποίο για να το φορτώσει το Protégé³⁸ χρειάζεται αρκετές ώρες. Χρησιμοποιώντας το Swoogle [79], μπορεί κανείς να βρει μειωμένες εκδόσεις της OpenCyc. Η OpenCyc είναι συνδεδεμένη με το λεξικό WordNet [80], το ανοιχτό λεξικό το οποίο αναπτύσσεται κυρίως από το Πανεπιστήμιο του Princeton. Λόγω του μεγέθους της, η OpenCyc είναι δύστροπη στη διαχείριση. Είναι δύσκολο να συντηρηθεί, να αναπτυχθεί και να διαμοιραστεί μια οντολογία τέτοιου μεγέθους. Θα ήταν ίσως προτιμότερο η οντολογία να διαχωριστεί σε μικρότερα τμήματα, τα οποία να συντηρούνται από ξεχωριστές κοινότητες.

Συνολικά ωστόσο, πρόκειται για μια εξαιρετικά φιλόδοξη απόπειρα καταγραφής της ανθρώπινης γνώσεως με αμφιλεγόμενα προς το παρόν αποτελέσματα. Σίγουρα έχει αρκετό δρόμο ακόμα να διανύσει, δεν παύει όμως να αποτελεί από τη μια μεριά ένα καλό σημείο αφετηρίας για τη συγγραφή οντολογιών, από την άλλη μια τεράστια πηγή δεδομένων διαθέσιμων για επεξεργασία για ερευνητικούς σκοπούς.

3.6 Εξυπηρετητές υπηρεσιών συλλογιστικής

Οι εξυπηρετητές υπηρεσιών συλλογιστικής (reasoners) είναι τα προγράμματα που πρωτίστως αναλαμβάνουν τον εννοιολογικό έλεγχο της οντολογίας εφαρμόζοντας σε αυτή ελέγχους συνέπειας (consistency checks). Επίσης μπορούν να αξιοποιήσουν την υπάρχουσα Βάση Γνώσεως απαντώντας σε ερωτήσεις των οποίων οι απάντηση προκύπτει από τα δεδομένα με την απλή επαγωγική εφαρμογή κανόνων λογικής. Επιπλέον, οι reasoners είναι αυτοί που στην πράξη χρησιμοποιούν την υπάρχουσα γνώση για να την εμπλουτίσουν εξάγοντας – συνήθως επαγωγικά, χρησιμοποιώντας tableaux αλγόριθμους – περισσότερη πληροφορία. Οι reasoners στην ουσία είναι αυτοί που θα εξάγουν την πληροφορία από μια Βάση Γνώσεως. Ανάλογα με τους αλγόριθμους επαγωγής που έχει ο κάθε εξυπηρετητής μπορεί να συμπεράνει την ισχύ γεγονότων που δεν έχουν δηλωθεί ρητά. Δεν είναι δυνατό να εξαχθεί συμπερασματική γνώση (inferred knowledge) χωρίς τη χρήση διαδικασιών συλλογιστικής. Οι βασικές υπηρεσίες συλλογιστικής μπορούν να καταταγούν στις παρακάτω [81].

- Έλεγχος συνέπειας (Consistency checking). Εξασφαλίζει ότι η οντολογία δεν

³⁸ Αναλυτική επισκόπηση των δυνατοτήτων του Protégé στην ενότητα 3.2.1

περιέχει αντικρουόμενες έννοιες. Χρησιμοποιώντας DL τερμινολογία, η υπηρεσία αυτή δηλώνει την εξασφάλιση ότι το ABox είναι συνεπές ως προς το TBox.

- Ικανοποιησιμότητα (Concept satisfiability). Ελέγχει αν είναι δυνατό η κάθε κλάση να έχει άτομα. Αν μια κλάση λόγω του ορισμού της δε μπορεί να έχει άτομα αυτό σημαίνει ότι η οντολογία είναι ασυνεπής στο σύνολό της.
- Κατηγοριοποίηση (Concept Subsumption, Classification). Μπορεί να υπολογίσει το ιεραρχικό δένδρο κλάσεων/υποκλάσεων. Το δένδρο αυτό μπορεί να χρησιμοποιηθεί για παράδειγμα στην αποτίμηση ερωτημάτων πάνω σε μια κλάση και να επιστρέψει επαγωγικά όλες τις υποκλάσεις της ή μόνο τις άμεσες υποκλάσεις.
- Instance Checking (Realization). Έχει τη δυνατότητα να υπολογίσει τις κλάσεις στις οποίες ανήκει ένα άτομο. Με άλλα λόγια μπορεί να υπολογίσει τον τύπο του κάθε ατόμου. Η λειτουργία realization μπορεί να λειτουργήσει μόνο μετά από την κατηγοριοποίηση καθώς η ιεραρχία των κλάσεων είναι απαραίτητη.

Όσον αφορά τώρα, τις ιδιότητες των εξυπηρετητών υπηρεσιών συλλογιστικής σχετικά με τη συμπεριφορά τους, τρεις είναι ιδιαίτερου ενδιαφέροντος όσον αφορά την Περιγραφική Λογική – και τη Λογική γενικότερα.

1. Τερματισμός (Termination). Δηλώνει αν ο αλγόριθμος τερματίζει πάντα ή υπάρχει η περίπτωση να εκτελείται επ' άπειρον.
2. Ορθότητα (Soundness). Μια διαδικασία συλλογιστικής είναι ορθή (sound) αν για κάθε φόρμουλα που αποδεικνύεται ικανοποιήσιμη τότε η φόρμουλα είναι όντως ικανοποιήσιμη.
3. Πληρότητα (Completeness). Μια διαδικασία συλλογιστικής είναι πλήρης όταν, κάθε φορά που εκτελείται σε ικανοποιήσιμες έννοιες πάντα τερματίζει με έγκυρο αποτέλεσμα (είναι clash-free).

Χρειάζεται να αναφερθούμε ειδικά στη διασύνδεση DIG (DL Implementation Group) η οποία βασίζεται σε απλές HTTP get/put μεθόδου και ο ρόλος της είναι διαμεσολαβητικός μεταξύ των εξυπηρετητών υπηρεσιών συλλογιστικής και των περιβαλλόντων διαχείρισης οντολογιών. Αν και δεν έχει προκύψει ακόμα κάποιο πρότυπο, η DIG υποστηρίζεται από τους περισσότερους εξυπηρετητές όπως ο KAON2, ο FaCT++, ο Racer, ο Pellet και η Jena. Παράλληλα, εργαλεία όπως το Protégé, το SWOOP ή το OilEd υποστηρίζουν επίσης τη DIG. Είναι δυνατός οποιοσδήποτε συνδυασμός εξυπηρετητή και περιβάλλοντος συγγραφής από τα παραπάνω.

Η DIG (dig.cs.manchester.ac.uk) δε σχεδιάστηκε να παρέχει τη γενικότερη διαλειτουργικότητα που απαιτείται για την παροχή υπηρεσιών συλλογιστικής, αν και σε

αυτό χρησιμοποιείται ευρέως. Παρέχει έναν (σχετικά) ελαφρύ μηχανισμό για τα διάφορα εργαλεία που χρειάζονται να αλληλεπιδρούν με υπηρεσίες συλλογιστικής Περιγραφικής Λογικής. Θέματα όπως διαχείριση πελατών, συναλλαγών, μόνιμη αποθήκευση σε Βάσεις Δεδομένων δεν καλύπτονται από τις προδιαγραφές της DIG. Η συνήθης χρήση της είναι ως διεπαφή για OWL υπηρεσίες συλλογιστικής. Η γλώσσα που υποστηρίζεται από τη DIG είναι διαφορετική από την OWL και για τη διαλειτουργικότητα απαιτείται μετάφραση η οποία είναι συνήθως εύκολη. Η γλώσσα αυτή αναπτύσσεται από το DIG Working Group και διατηρείται ξεχωριστή ώστε το Group να διατηρεί την αυτονομία του και να μην επηρεάζεται από τις εξελίξεις στις γλώσσες περιγραφής της γνώσεως. Για την επικοινωνία με τον εξυπηρετητή, υλοποιούνται δυο τύποι αιτημάτων, ASK και TELL με τα οποία αποστέλλονται ερωτήματα ή ενημερώνεται ο εξυπηρετητής, αντίστοιχα. Η επικοινωνία γίνεται χωρίς αποκατάσταση συνεδρίας (είναι stateless) και πραγματοποιείται με συγκεκριμένα XML μηνύματα που αποστέλλονται μέσω του πρωτοκόλλου HTTP.

Στη συνέχεια παρουσιάζονται οι σημαντικότεροι εξυπηρετητές υπηρεσιών συλλογιστικής. Έχουν επιλεγεί για ξεχωριστή παρουσίαση αυτοί που είναι ελεύθερου κώδικα, εμφανίζουν καλές επιδόσεις και το κυριότερο, φαίνεται ότι υπάρχει κοινότητα η οποία τους αναπτύσσει και τους υποστηρίζει.

Ο KAON2 (kaon2.semanticweb.org) είναι ο διάδοχος του KAON [82]. Η διαφορά του με τον προκάτοχό του είναι ότι δεν υποστηρίζει μόνο RDFS. Ο KAON2 παρέχει υποδομή με στόχο τη διαχείριση οντολογιών σε OWL DL, SWRL και F-Logic. Παρέχει προγραμματιστικό περιβάλλον ώστε να είναι δυνατή η πρόσβαση με κώδικα στις οντολογίες και υποστηρίζει – σχεδόν ολόκληρη – την SPARQL. Οι διαδικασίες συλλογιστικής που ακολουθεί ο KAON2 δεν είναι βασισμένες σε tableaux αλγόριθμους όπως στους περισσότερους reasoners. Αντ' αυτού, η διαδικασίες συλλογιστικής υλοποιούνται μειώνοντας τις Βάσεις Γνώσεως σε διαζευκτικά προγράμματα datalog. Οι συγγραφείς του ισχυρίζονται στο [24] ότι επέλεξαν αυτή την προσέγγιση γιατί στους υπόλοιπους reasoners η απόδοση πέφτει πολύ όσο αυξάνει το σύνολο ABox. Αντίθετα, στον KAON2, κάθε άτομο δεν αντιμετωπίζεται ξεχωριστά, όπως προβλέπεται στην Περιγραφική Λογική, αλλά μπορεί να ταξινομηθεί σε σύνολο ατόμων με ίδιες ιδιότητες. Η διαδικασία αυτή επωφελείται από το ότι στην datalog έχει γίνει πολλή δουλειά βελτιστοποίησης, στην απάντηση conjunctive queries, όπως τα magic sets [83]. Οι αποδόσεις του KAON2 είναι κατά πολύ καλύτερες σε σχέση με τους υπόλοιπους εξυπηρετητές όπως αποδεικνύεται στο [84]. Τέλος, ο KAON2 μπορεί να παρέχει υπηρεσίες μέσω HTTP και DIG μηνυμάτων. Επιπλέον μπορεί να γίνει και εξυπηρετητής οντολογιών παρέχοντας πρόσβαση μέσω RMI. Η γλώσσα που υποστηρίζεται είναι η *SHIQ(D)*.

Ο Pellet (pellet.owldl.com) είναι ο πρώτος [25] sound and complete reasoner για την OWL DL. Είναι ανοιχτού λογισμικού, γραμμένος σε java και διατίθεται υπό ειδική άδεια των κατασκευαστών του. Οι υπηρεσίες συλλογιστικής που προσφέρει ο Pellet είναι οι υπηρεσίες των reasoners που αναφέρθηκαν παραπάνω και είναι προσβάσιμες μέσω υποβολής ερωτημάτων στον Pellet, για τα οποία χρησιμοποιείται η διαλειτουργικότητα DIG. Ο Pellet είναι σε θέση να ελέγξει OWL Full έγγραφα και να διαπιστώσει αν είναι δυνατό να εκφραστούν χρησιμοποιώντας το DL υποσύνολο της OWL. Αν αυτό ισχύει, τότε μπορεί να το επεξεργαστεί χωρίς πρόβλημα. Αξιοσημείωτο είναι ότι εκτός από το σύννηθες σύνολο υπηρεσιών reasoning που προσφέρει ο Pellet επιπλέον παρέχει και υπηρεσίες για αποσφαλμάτωση και επεξήγηση οντολογιών. Η διαφορά του Pellet όμως από τους υπόλοιπους reasoners είναι ότι από την αρχή σχεδιάστηκε ώστε να συνεργάζεται με την OWL. Σχεδιάστηκε έτσι ώστε ο πυρήνας του κώδικά του να είναι όσο το δυνατόν μικρότερος με αποτέλεσμα να είναι επεκτάσιμος. Ακόμα κι ο πυρήνας είναι επεκτάσιμος, γεγονός που έχει επιτρέψει στους δημιουργούς του να αναπτύξουν επεκτάσεις όπως για παράδειγμα τη διαχείριση των *ℰ-Connections*. Στη βάση του Pellet είναι ο tableau reasoner. Οι OWL οντολογίες φορτώνονται στον reasoner μετά από εκτίμηση του τύπου τους και πιθανές διορθώσεις. Κατά τη διαδικασία αυτή οι δηλώσεις για τις κλάσεις τοποθετούνται στο TBox και οι ισχυρισμοί για τα individuals στο ABox. Το σύστημα προσφέρει προγραμματιστική πρόσβαση μέσω του Service Protocol Interface (SPI) στις συλλογιστικές υπηρεσίες που παρέχει.

Ο FaCT++ (owl.man.ac.uk/factplusplus) [85] είναι ένας OWL DL reasoner που έχει αναπτυχθεί στο πανεπιστήμιο του Manchester και διατίθεται υπό την GNU άδεια ελεύθερου λογισμικού. Έχει φτάσει την έκδοση 1.3.0 (από το Μάιο του 2009) και υποστηρίζει πλήρως την OWL DL. Είναι υλοποιημένος σε C++ και υποστηρίζει τη διαλειτουργικότητα DIG. Η πρώτη έκδοση του reasoner παρουσιάστηκε το 1998 [22] και ήταν υλοποιημένη στη γλώσσα προγραμματισμού LISP. Στο σύστημα είναι υλοποιημένη η υποστήριξη της περιγραφικής γλώσσας *SHOIQ*. Ο FaCT++ προκειμένου να επεξεργαστεί μια Βάση Γνώσεως περνά πρώτα από ένα στάδιο προεπεξεργασίας της. Η Βάση Γνώσεως κανονικοποιείται και μεταφράζεται στην εσωτερική γλώσσα αναπαράστασης γνώσεως του FaCT++. Κατά τη διαδικασία αυτή γίνονται βελτιστοποιήσεις υπό τη μορφή συντακτικών τροποποιήσεων στην αρχική Βάση Γνώσεως. Το δεύτερο στάδιο είναι η κατηγοριοποίηση (classification) όπου υπολογίζεται και αποθηκεύεται στη μνήμη η ταξινόμηση των εννοιών. Η βελτιστοποίηση που γίνεται στο στάδιο αυτό οφείλεται στο ότι ο αλγόριθμος επιλέγει τη σειρά – κυρίως με ευριστικές συναρτήσεις – με την οποία θα επεξεργαστεί τις έννοιες ούτως ώστε να εφαρμόσει τους λιγότερους δυνατούς ελέγχους. Στη συνέχεια πραγματοποιείται ο

έλεγχος ικανοποιησιμότητας της Βάσης Γνώσεως ώστε να εξακριβωθεί ότι δεν υπάρχουν προβλήματα υπαγωγής (subsumption). Αυτό είναι και το κεντρικό δομικό στοιχείο στο οποίο έχει γίνει αρκετή δουλειά βελτιστοποίησης καθώς η ανάπτυξη του reasoner έχει πάνω από δέκα χρόνια ανάπτυξης.

Υπάρχει επιπλέον ένα σύνολο από εξυπηρετητές πιο περιορισμένου ενδιαφέροντος. Ο Racer³⁹ ο οποίος παρουσιάστηκε στο [23] και πλέον διατίθεται εμπορικά υπό το όνομα RacerPro. Ο Racer, σε αντίθεση με τον RacerPro, δεν υποστηρίζει τη διαλειτουργικότητα DIG. Ο μηχανισμός αποδείξεων Euler⁴⁰ είναι ένας reasoner που χρησιμοποιεί την τεχνική backward chaining και είναι ενισχυμένος με Euler path detection. Χρησιμοποιεί τη γλώσσα N3 για την αλληλεπίδραση σύμφωνα με τα πρότυπα του Σημασιολογικού Ιστού. Η τελευταία έκδοσή του είναι η EulerSharp η οποία δεν υποστηρίζει την OWL αλλά μόνο RDF γράφους. Ο Algernon (algernon-j.sourceforge.net) είναι σύστημα εξαγωγής συμπερασμάτων βασισμένο σε java και ανοιχτό κώδικα. Μπορεί να εκτελεστεί και σαν προσθήκη του Protégé εκτός από standalone εφαρμογή. Εφαρμόζει forward chaining και backward chaining επεξεργασία κανόνων. Η ανάπτυξη του algernon ξεκίνησε στις αρχές της δεκαετίας του 90, ο κώδικας επανασχεδιάστηκε στα τέλη της δεκαετίας και το 2002 υλοποιήθηκε σε java. Ο JTP (www.ksl.stanford.edu/software/JTP) υποστηρίζει DAML και OWL, αναπτύχθηκε στο Stanford και παρουσιάστηκε στο [86]. Είναι υλοποιημένος σε Java, υποστηρίζει το KIF και μπορεί να εκφράσει προτάσεις του κατηγορικού λογισμού πρώτης τάξης. Η αρχιτεκτονική του αποτελείται από modules που ονομάζονται reasoners – η χρήση του όρου εδώ δε σχετίζεται με το υπόλοιπο έγγραφο – οι οποίοι χωρίζονται σε δυο κατηγορίες, backward-chaining και forward-chaining. Δυστυχώς, οι Euler, Algernon και JTP δεν υποστηρίζουν τη διαλειτουργικότητα DIG, κάτι που καθιστά δύσκολη την ενσωμάτωσή τους σε εφαρμογές Σημασιολογικού Ιστού.

Αξίζει να αναφερθεί, τέλος και ο OWLJessKB [87], ο απόγονος του DAMLJessKB ο οποίος βασίζεται στην μηχανή εξαγωγής συμπερασμάτων Jess (αλγόριθμος Rete [45]) και στη βιβλιοθήκη της Jena (ενότητα 3.4.1).

³⁹Racer: www.racer-systems.com

⁴⁰Euler: sourceforge.net/projects/eulersharp

3.7 Τεχνολογίες Διαδικτύου

Αναφέρθηκε και προηγουμένως ότι ο Σημασιολογικός Ιστός δε σχεδιάζεται να αντικαταστήσει την υπάρχουσα υποδομή του Διαδικτύου, έρχεται μάλλον για να τη συμπληρώσει, να την επεκτείνει και να διευκολύνει τη διαχείρισή της. Στην κατεύθυνση αυτή, η έρευνα που γίνεται για συνδυασμό με τις υπάρχουσες τεχνολογίες προσανατολίζεται στην ενοποίηση με αυτές και όχι στην αντικατάστασή τους. Στην ενότητα αυτή αποτυπώνεται πώς επιτυγχάνεται η συνεργασία όλων όσων περιγράφηκαν προηγουμένως με αρκετούς τομείς εξελεγχόμενων τεχνολογιών όπως οι πράκτορες λογισμικού, τα συστήματα με επίγνωση περιβάλλοντος, οι Υπηρεσίες Διαδικτύου (Web Services), οι μηχανές αναζήτησης και τα συστήματα διαχείρισης πολυμεσικής πληροφορίας. Η συνεργασία των οντολογιών με τις Βάσεις Δεδομένων αναλύεται διεξοδικά στο κεφάλαιο 4.

3.7.1 Πράκτορες λογισμικού

Η έννοια του πράκτορα λογισμικού εισήχθη το 1977, υπό το πλαίσιο έρευνας για την Κατανεμημένη Τεχνητή Νοημοσύνη (Distributed Artificial Intelligence – DAI). Η έννοια του “actor” εισάγεται ως ένας υπολογιστικός πράκτορας ο οποίος χαρακτηρίζεται από την διεύθυνσή του και την συμπεριφορά του, ενώ έχει μία εσωτερική κατάσταση και μπορεί να απαντά σε μηνύματα άλλων παρόμοιων αντικειμένων [88]. Υπάρχει αρκετή μετέπειτα έρευνα σχετικά με τους συνεργατικούς (collaborative) πράκτορες, με έμφαση στην αλληλεπίδραση και επικοινωνία μεταξύ πρακτόρων με σκοπό τον καταμερισμό και συντονισμό των εργασιών. Επομένως, η ιδέα του πράκτορα λογισμικού αποκτά μία πολύ ευρύτερη έννοια καθιστώντας δύσκολο να δοθεί ένας σαφής ορισμός όσον αφορά το τι είναι πράκτορας σε αντίθεση με ένα απλό πρόγραμμα. Ένας από τους πιο περιγραφικούς ορισμούς δίνεται στο [89]: “Οι πράκτορες λογισμικού (software agents) είναι αυτόνομες οντότητες λογισμικού που αποτελούν μέρος ενός περιβάλλοντος, ανιχνεύουν την κατάστασή του και δρουν πάνω σε αυτό, ούτως ώστε να επιτύχουν κάποιο σκοπό των χρηστών τους. Με τη δράση τους, επηρεάζουν το σύστημα στο οποίο βρίσκονται και την κατάσταση που θα ανιχνεύσουν στο μέλλον”.

Οι πράκτορες λογισμικού μπορούν να διακριθούν από τα απλά προγράμματα λόγω της ιδιότητας των πρώτων να επηρεάζουν το περιβάλλον που στην συνέχεια ανιχνεύουν αλλά και από την χρονική συνέχεια της δράσης τους, σε αντίθεση με πολλά προγράμματα που εκτελούνται για πεπερασμένο χρόνο. Κάθε πράκτορας λογισμικού αποτελεί και ένα πρόγραμμα ενώ αντίθετα ένα πρόγραμμα δεν είναι αναγκαστικά και πράκτορας. Οι πράκτορες συνήθως είναι μικροί σε μέγεθος. Δεν αποτελούν από μόνοι τους μία ολόκληρη

εφαρμογή, αλλά σχηματίζουν μία εφαρμογή σε συνδυασμό με μία πλατφόρμα πρακτόρων και άλλους πράκτορες. Η πλατφόρμα αυτή οφείλει να επιτρέπει σε πολλούς πράκτορες να συνυπάρχουν και να εκτελούνται ταυτόχρονα. Οι πράκτορες μπορούν να αποτελούνται από υποσύνολα που είναι και αυτά πράκτορες λογισμικού τα οποία συνεργάζονται μεταξύ τους ως ένα σύστημα πολλαπλών πρακτόρων.

Μερικά από τα χαρακτηριστικά των πρακτόρων είναι ότι είναι αυτόνομοι, αντιδρούν στις μεταβολές του περιβάλλοντος, δρουν έτσι ώστε να επιτύχουν έναν συγκεκριμένο σκοπό χωρίς πάντα να περιμένουν κάποιο έναυσμα από το περιβάλλον τους και η δράση τους αυτή είναι συνεχής στο χρόνο. Ακόμη επικοινωνούν με άλλους πράκτορες ή χρήστες, είναι ευέλικτοι και προσαρμόζουν τις λειτουργίες τους με βάση την προηγούμενη εμπειρία.

Ένα σύστημα πολλαπλών πρακτόρων (Multi-Agent System) αποτελείται από πολλούς αυτόνομους πράκτορες που αλληλεπιδρούν μεταξύ τους και σύμφωνα με τον R. Flores-Mendez [90], παρουσιάζει τα εξής χαρακτηριστικά: Κάθε πράκτορας δεν μπορεί μόνος του να λύσει κάποιο πρόβλημα, δεν υπάρχει κεντρικό σύστημα ελέγχου, τα δεδομένα είναι αποκεντρωμένα και οι λειτουργίες είναι ασύγχρονες. Η αλληλεπίδραση μπορεί να είναι είτε συνεργατική είτε ιδιοτελής, που σημαίνει ότι οι πράκτορες μπορούν είτε να μοιράζονται έναν κοινό στόχο ή να εξυπηρετεί ο καθένας το ξεχωριστό του συμφέρον. Μπορεί επίσης να έχει τη μορφή μηνυμάτων μεταξύ πρακτόρων ή παραγωγής αλλαγών στο κοινό περιβάλλον των πρακτόρων.

Οι σύγχρονες τάσεις στα online συστήματα παροχής πληροφορίας συχνά απαιτούν τη συνεργασία ικανού αριθμού εταιρών. Τα συστήματα που παρέχουν πρόσβαση σε μεγάλους όγκους δεδομένων όπως για παράδειγμα βιβλιοθήκες, βιολογικά δεδομένα ή γεωγραφικά δεδομένα⁴¹ συχνά αποτελούνται από ένα δίκτυο παρόχων. Σε αυτά τα συστήματα δεν υπάρχει ανάγκη συγκέντρωσης των δεδομένων σε ένα μοναδικό εξυπηρετητή καθώς αυτά είναι διαμοιρασμένα μέσω ενός ομογενοποιημένου μηχανισμού πρόσβασης.

Συστήματα πρακτόρων λογισμικού έχουν αναπτυχθεί με στόχο την παροχή υπηρεσιών όπως η προαναφερθείσα. Για την προτυποποίηση και προώθηση των πρακτόρων λογισμικού, κύριος οργανισμός είναι ο FIPA ο οποίος έγινε αποδεκτός από το IEEE ως η ενδέκατη επιτροπή προτύπων στις 8 Ιουνίου 2005⁴². Παρά το γεγονός ότι η τεχνολογία πρακτόρων λογισμικού δεν είναι ακόμα αρκετά διαδεδομένη σε εμπορικές εφαρμογές [91], σημαντικά βήματα έχουν γίνει στην ανάπτυξή της. Έχει αποδειχθεί ότι

⁴¹Federal Geographic Data Committee (FGDC) Data and Services: www.fgdc.gov/dataandservices

⁴²The Foundation for Intelligent Physical Agents: www.fipa.org

οι σύγχρονες πλατφόρμες όπως η JADE⁴³ ή η ZEUS⁴⁴ είναι αρκετά κλιμακώσιμα (scalable) ώστε να αποκρίνονται ικανοποιητικά σε σχετικά τεράστιους αριθμούς πρακτόρων και ανταλασσομένων μηνυμάτων [92].

Η σημασιολογική πύλη με μουσεία της Φιλανδίας [93] αποτελεί ένα αντιπροσωπευτικό παράδειγμα για την παρουσίαση ετερογενών μουσειακών συλλογών στο Σημασιολογικό Ιστό. Σε αυτήν παρουσιάζεται πώς το περιεχόμενο των μουσείων μπορεί να δημιουργήσει μια σημασιολογικά εμπλουτισμένη συλλογή στο Διαδίκτυο. Επιπλέον, στο [94] παρουσιάζεται ένα σύστημα για την αναζήτηση σε δυναμικά μεταβαλλόμενες θεματικές περιοχές. Ωστόσο, πολλές από τις σύγχρονες προσεγγίσεις δεν ενσωματώνουν ευφυΐα στο σύστημά τους. Πολλές χρησιμοποιούν είτε αναζήτηση απλών λέξεων-κλειδιών (keywords) σε RDF γράφους είτε άλλες συμβατικές τεχνικές ανάκτησης πληροφορίας (IR) [95, 96].

Η γλώσσα που χρησιμοποιείται για την ανταλλαγή μηνυμάτων που εμπεριέχουν σημασιολογία μεταξύ πρακτόρων λογισμικού είναι η SL⁴⁵ (Semantic Language), μια γλώσσα περιεχομένου της οποίας η σύνταξη ορίζεται από την FIPA και μπορεί να χρησιμοποιηθεί σε συνδυασμό με την FIPA ACL. Η ACL αποτελεί το προτεινόμενο πρότυπο από την FIPA για την επικοινωνία πρακτόρων.

3.7.2 Συστήματα με επίγνωση περιβάλλοντος

Ο όρος Συστήματα με Επίγνωση του Περιβάλλοντος (Context-Aware Systems) συμπεριλαμβάνει τα συστήματα που αντιλαμβάνονται το περιβάλλον στο οποίο δραστηριοποιούνται και μεταβάλλουν τη δραστηριότητά τους σύμφωνα με τις συνθήκες που επικρατούν. Σύμφωνα με τον A. Dey [97], πληροφορία σχετική με το περιβάλλον ενός συστήματος είναι οποιαδήποτε πληροφορία η οποία μπορεί να χρησιμοποιηθεί για να χαρακτηρίσει την κατάσταση μιας οντότητας. Μια οντότητα μπορεί να είναι ένας άνθρωπος, ένας τόπος, ένα αντικείμενο, ακόμη και το ίδιο το σύστημα. Στις οντότητες μπορεί να συμπεριλαμβάνεται ο χρήστης του συστήματος αλλά και το σύστημα το ίδιο. Ένα σύστημα θεωρείται ότι έχει επίγνωση του περιβάλλοντος αν είναι σε θέση να εξάγει, να μεταφράσει και να χρησιμοποιήσει πληροφορία του περιβάλλοντός του και να προσαρμόσει τη λειτουργικότητά του ανάλογα με την πληροφορία που δέχεται.

Η πρόκληση στα συστήματα αυτά έγκειται στην πολυπλοκότητα που απαιτείται για τη σύλληψη, αποθήκευση, διαχείριση, επεξεργασία και αναπαράσταση της πληροφορίας. Στις άμεσες ανάγκες παρόμοιων συστημάτων είναι η καθιέρωση ενός κοινού προτύπου

⁴³Java Agent DEveloper Framework: jade.tilab.com

⁴⁴Εργαλειοθήκη ZEUS: labs.bt.com/projects/agents/zeus

⁴⁵FIPA SL Content Language Specification: www.fipa.org/specs/fipa00008

στην αναπαράσταση της πληροφορίας που συγκεντρώνεται ώστε να μπορεί η πληροφορία να επαναχρησιμοποιείται και να διαμοιράζεται και σε άλλα συστήματα. Προκειμένου να διασφαλιστεί αυτή η αναγκαία συντακτική και σημασιολογική διαλειτουργικότητα, έχουν προταθεί αρκετές προσεγγίσεις, όπως για παράδειγμα το Context Toolkit [98, 99], το HP's Cooltown [100], το Intelligent Room project, αλλά καμία από αυτές δεν πρότεινε ένα κοινό τρόπο αποθήκευσης των δεδομένων, ένα κοινό λεξικό το οποίο όλοι να μπορούν να κατανοούν και στο οποίο να μπορούν όλοι να συνεισφέρουν.

Το κενό καλύπτεται από την πρόσφατη ανάπτυξη των τεχνολογιών του Σημασιολογικού Ιστού. Η ανάπτυξη σε γλώσσες περιγραφής, ερωτημάτων και σε παρεμφερείς τεχνολογίες παρέχει πλέον ένα κοινό, αξιόπιστο και ευέλικτο πλαίσιο για τη διαχείριση και διαμοιρασμό της πληροφορίας. Χωρίς ομοιογένεια στην πληροφορία, ένα σύστημα για παράδειγμα παρακολούθησης χώρου είναι καταδικασμένο να είναι μονολιθικό, δύσχρηστο και γρήγορα να θεωρείται παρωχημένο. Στο [101] οι συγγραφείς ισχυρίζονται ότι φορμαλισμοί με ανεπαρκώς ορισμένη σημασιολογία που υιοθετούνται κατά περίπτωση δυσχεραίνουν εξαιρετικά την ενοποίηση και/ή την ολοκλήρωση της πληροφορίας.

Όμως, οι έννοιες του κόσμου μπορούν να περιγραφούν λεπτομερώς με τη χρήση τεχνολογιών Σημασιολογικού Ιστού. Η δυναμικότητα αυτής της προσέγγισης βασίζεται στο ότι ένα σημασιολογικά εμπλουτισμένο μοντέλο του κόσμου μπορεί να υφίσταται επεξεργασία από υπηρεσίες συλλογιστικής οι οποίες προσθέτουν ευφυΐα εξάγοντας έμμεση γνώση. Σύμφωνα με τους A. Toninelli et al. [102], η πληροφορία για το περιβάλλον που εμπεριέχεται στα συστήματα με επίγνωση περιβάλλοντος μπορεί να οργανωθεί λογικά στα τμήματα που περιγράφουν την κατάσταση των πόρων (resource part), τις οντότητες που αλληλεπιδρούν με τους πόρους (actor part) και τις συνθήκες του περιβάλλοντος (environment part). Είναι φανερό ότι η εκφραστικότητα που παρέχεται από την Περιγραφική Λογική μπορεί να χρησιμοποιηθεί ώστε να μοντελοποιήσει την πληροφορία αυτή.

Με το θέμα της σημασιολογικής περιγραφής της πληροφορίας που αφορά το περιβάλλον των εφαρμογών (contextual information) ασχολείται το ευρωπαϊκό έργο CHIL. Ο στόχος του έργου αυτού είναι η παροχή ευφών υπηρεσιών ώστε να υποστηρίξουν την αλληλεπίδραση ανθρώπου-υπολογιστή. Στο [103] παρουσιάζεται μια οντολογία για τη διαχείριση διατάξεων αισθητήρων και ενεργοποιητών. Επίσης, αντίστοιχη προσπάθεια γίνεται στο έργο KaOS [104], στο οποίο χρησιμοποιούνται οντολογίες Περιγραφικής Λογικής για την αναπαράσταση πληροφορίας του περιβάλλοντος και αυτές χρησιμοποιούνται για τη διαχείριση του συστήματος. Αντίστοιχα, στο έργο Rei [105], χρησιμοποιούνται οντολογίες RDF(S) ή OWL

Lite για την αναπαράσταση εννοιών του περιβάλλοντος. Οι οντολογίες βασίζονται στην Reī οντολογία που έχει αναπτυχθεί για το σκοπό αυτό.

Επιπλέον, στο έργο SmartResource [106] παρουσιάζεται ένα πλαίσιο για την επεξεργασία σε δυο στάδια με στόχο την μετατροπή των υπάρχοντων XML δεδομένων σε εμπλουτισμένη σημασιολογικά πληροφορία η οποία εκφράζεται σε RDF. Καμία από τις δύο αυτές προσεγγίσεις όμως δεν αντιμετωπίζει το πρόβλημα της επεξεργασίας των δεδομένων σε *πραγματικό χρόνο*.

Συνήθως, για την σύλληψη της κατάστασης του συστήματος απαιτούνται διατάξεις αισθητήρων και κατάλληλο λογισμικό. Χρειάζεται επίσης η πληροφορία που συγκεντρώνεται να εμφανίζει συντακτική και σημασιολογική ομοιογένεια προκειμένου να εξασφαλίζεται τόσο η ολοκλήρωσή της όσο και η διαλειτουργικότητα μεταξύ των επιμέρους τμημάτων του συστήματος. Στην ενότητα 5 εξετάζεται ο τρόπος με τον οποίον επιτυγχάνεται μια προσέγγιση η οποία βασίζεται στις τεχνολογίες του Σημασιολογικού Ιστού, παρέχει ολοκλήρωση ετερογενών πηγών δεδομένων και ευφυή εκμετάλλευση της πληροφορίας.

Χρειάζεται επίσης να σημειωθεί ότι τα συστήματα με επίγνωση περιβάλλοντος είναι συνήθως πολύπλοκα στο βαθμό που να καθιστούν αδύνατες διαδικασίες επισημείωσης αν δεν υποστηρίζονται από υψηλό βαθμό αυτοματοποίησης. Είναι αναμφισβήτητη η χρησιμότητα ενός μεσισμικού το οποίο θα ομογενοποιεί την επεξεργασία και αναπαράσταση των δεδομένων σε επίπεδο υποδομής προκειμένου να είναι δυνατή η συνολική διαχείριση σε υψηλότερο επίπεδο. Ο σημασιολογικός εμπλουτισμός της συλλεγόμενης πληροφορίας και η εξαγωγή συμπερασμάτων αποτελεί μια από τις σύγχρονες ερευνητικές προκλήσεις [107].

Δυο ενδιαφέροντα χαρακτηριστικά, την ύπαρξη των οποίων μπορεί να παρατηρηθεί τόσο στις Βάσεις Γνώσεως όσο και στα συστήματα με επίγνωση περιβάλλοντος είναι η μονοτονικότητα (monotonicity) και η προϋπόθεση ανοιχτού κόσμου. Ένα σύστημα είναι μονοτονικό (monotonic) όταν η νέα πληροφορία που εισάγεται δεν αναιρεί την ήδη υπάρχουσα ενώ η θεώρηση ανοιχτού (και κλειστού) κόσμου παρουσιάζεται στην ενότητα 4.1.5. Επιπλέον, στα συστήματα με επίγνωση περιβάλλοντος σημαντικό παράγοντα αποτελεί η διατήρηση της ιστορικότητας ούτως ώστε η πληροφορία που το σύστημα συγκεντρώνει για το περιβάλλον του να μπορεί να είναι προσβάσιμη, να επιτρέπει ερωτήματα και αναζητήσεις και να παρέχει αξιοποιήσιμη λειτουργικότητα. Επιπρόσθετα, ειδικότερα στα συστήματα πολυμεσικού περιεχομένου, η επεξεργασία πραγματικού χρόνου αποτελεί σημαντικό ζήτημα καθώς ο μεγάλος όγκος των δεδομένων απαιτείται να συνοδεύεται από μια διασφάλιση της κλιμακωσιμότητας του συστήματος επεξεργασίας. Στο κεφάλαιο 5 αναλύεται διεξοδικότερα το πρόβλημα της επεξεργασίας

πραγματικού χρόνου, σε παραλληλία με την ενσωμάτωση ευφυΐας κατά την επεξεργασία της εισερχόμενης πληροφορίας.

3.7.3 Σημασιολογικές Υπηρεσίες Διαδικτύου

Οι Υπηρεσίες Διαδικτύου είναι στην ουσία εφαρμογές οι οποίες εκθέτουν τη λειτουργικότητά τους στο Διαδίκτυο ώστε να μπορεί ο καθένας να τις χρησιμοποιήσει. Οι Σημασιολογικές Υπηρεσίες Διαδικτύου προσφέρουν περισσότερη εκφραστικότητα στην περιγραφή αυτών των υπηρεσιών με σκοπό να διευκολύνεται η καλύτερη διαφήμιση των υπηρεσιών, η αυτοματοποιημένη ανακάλυψή τους από τρίτους και η αυτοματοποιημένη σύνθεση και διαλειτουργικότητα μεταξύ τους.

Η αυτόματη ανακάλυψη υπηρεσιών Ιστού αναφέρεται στην αυτοματοποιημένη διαδικασία εύρεσης της κατάλληλης Υπηρεσίας Διαδικτύου προκειμένου να περατωθεί κάποιος στόχος του χρήστη. Αν για παράδειγμα ότι κάποιος χρήστης χρειάζεται να οργανώσει ένα ταξίδι διακοπών, με τις υπάρχουσες τεχνολογίες θα πρέπει να επιλέξει χειροκίνητα τις υπηρεσίες που τον εξυπηρετούν – αεροπορικά εισιτήρια, διαμονή, ενοικίαση αυτοκινήτου – και να τις συνδυάσει χειροκίνητα. Με την προσθήκη ευφυέστερης περιγραφής τους, θα μπορεί κανείς να αναζητήσει τη βέλτιστη λύση χωρίς την άμεση παρέμβασή του [108].

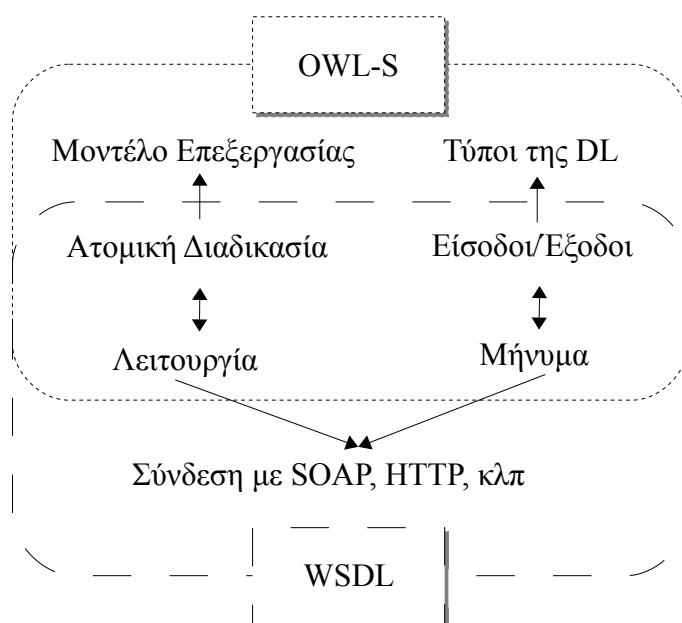
Η έρευνα στην περιοχή των Σημασιολογικών Υπηρεσιών Διαδικτύου προσανατολίζεται στην ανάπτυξη κάποιου προτύπου περιγραφής των υπηρεσιών. Τα μειονεκτήματα των Υπηρεσιών Διαδικτύου με τη σημερινή τους μορφή είναι ότι η σημερινή τεχνολογία – UDDI, WSDL και SOAP – παρέχει περιορισμένη υποστήριξη στην αυτοματοποίηση διαδικασιών και ειδικότερα στην αναγνώριση, παραμετροποίηση και συνδυασμό διαφορετικών Υπηρεσιών Διαδικτύου. Οι Σημασιολογικές Υπηρεσίες Διαδικτύου προσθέτουν περισσότερη ευφυΐα σε σχέση με τις σημερινές Υπηρεσίες Διαδικτύου.

Η πρώτη απόπειρα για τη δημιουργία μιας γλώσσας περιγραφικότερης της WSDL είναι η DAML-S. Πρόκειται για μια προσέγγιση κατά την οποία στην περιγραφή μιας υπηρεσίας να ενσωματώνεται πληροφορία για το πώς λειτουργεί η υπηρεσία, όχι μόνο τι προσφέρει. Η βασική προσέγγιση είναι η αντικατάσταση της UDDI στην περιγραφή των υπηρεσιών από οντολογία [109] περιγραφής ανώτερου επιπέδου. Η εξέλιξη της DAML-S είναι η OWL-S⁴⁶ κατά τον ίδιο τρόπο με τον οποίο η OWL είναι εξέλιξη της DAML. Η OWL-S έχει υποβληθεί στο W3C για προτυποποίηση⁴⁷, δεν έχει προκύψει ως πρότυπο όμως ακόμη.

⁴⁶OWL-S: www.daml.org/services/owl-s/1.0/

⁴⁷Πρόταση για την OWL-S στο W3C: www.w3.org/Submission/OWL-S/

Όπως φαίνεται και στην εικόνα 6, η σημασιολογική περιγραφή της υπηρεσίας βασίζεται στις υπάρχουσες τεχνολογίες, δεν τις αντικαθιστά. Ένας προγραμματιστής Υπηρεσιών Διαδικτύου μπορεί να εκμεταλλευτεί τις επιπλέον δυνατότητες της OWL-S χωρίς να μεταβάλλει την υπάρχουσα λειτουργικότητα, χρειάζεται μόνο να περιγράψει τις αντιστοιχίσεις των μηνυμάτων και των διαδικασιών της υπηρεσίας με μηνύματα και διαδικασίες της OWL-S.



Εικόνα 6: Περιγραφή Υπηρεσιών Διαδικτύου με OWL-S

Επίσης, το Semantic Web Services Framework (SWSF) έχει προταθεί στο W3C για προτυποποίηση στις 9-9-05⁴⁸ δεν έχει προκύψει ακόμη όμως ως πρότυπο. Το έγγραφο περιγράφει την προτεινόμενη γλώσσα περιγραφής υπηρεσιών διαδικτύου (SWSL) και την προτεινόμενη οντολογία περιγραφής υπηρεσιών διαδικτύου (SWSO). Η προτεινόμενη γλώσσα (SWSL) αποτελείται από δυο γλώσσες, την SWSL-FOL και την SWSL-Rules. Η πρώτη είναι βασισμένη στον Κατηγορικό Λογισμό Πρώτης Τάξης και χρησιμοποιείται για να εκφράσει τις έννοιες της υπηρεσίας. Η δεύτερη δηλώνει τους κανόνες που χρησιμοποιούνται για την εκτέλεση του reasoning που γίνεται για την υποστήριξη της SWSL-FOL. Γενικά, η SWSL είναι μια γλώσσα γενικής χρήσης, έχει σχεδιαστεί όμως ούτως ώστε να καλύπτει τα περισσότερα θέματα των Υπηρεσιών Διαδικτύου.

⁴⁸SWSF: www.w3.org/Submission/SWSF/

Το πρόβλημα που παρουσιάζεται με τις Υπηρεσίες Διαδικτύου αποτελείται από δυο σκέλη. Αφενός δεν υπάρχει εδραιωμένο πρότυπο για την εξαγωγή σημασιολογικής περιγραφής τους. Αφετέρου, παρουσιάζεται το πρόβλημα που παρουσιάζεται και με τις συμβατικές Υπηρεσίες Διαδικτύου: οι εταιρείες είναι απρόθυμες στο να δημοσιοποιήσουν λειτουργίες που χρησιμοποιούν για εξυπηρέτηση δικών τους αναγκών. Για το λόγο αυτό, οι αναζητήσεις σε UDDI καταχωρήσεις παρουσιάζουν αμφίβολα και ελάχιστα αξιοποιήσιμα αποτελέσματα – σε δικτυακούς τόπους όπως το `www.webservices.net` και το `www.xmethods.net`. Για παράδειγμα, μια Υπηρεσία Διαδικτύου για νομισματικές ισοτιμίες ή για κράτηση σε ξενοδοχείο σπάνια γίνεται διαθέσιμη χωρίς αντίτιμο.

3.7.4 Μηχανές αναζήτησης

Το πιο ισχυρό εργαλείο στα χέρια των χρηστών του Διαδικτύου είναι οι μηχανές αναζήτησης. Χωρίς αυτές, το Διαδίκτυο στη σημερινή του μορφή θα ήταν πολύ μειωμένης χρηστικότητας. Ο Σημασιολογικός Ιστός αναμένεται να αλλάξει τον τρόπο με τον οποίο όχι μόνο διαχειρίζεται η πληροφορία, αλλά και τον τρόπο με τον οποίο πραγματοποιούνται αναζητήσεις σε αυτή. Μέχρι σήμερα, η κοινή πρακτική για οτιδήποτε αναζητείται στο Διαδίκτυο είναι η χρήση μηχανής αναζήτησης. Σπάνια οι χρήστες θα εισάγουν χειροκίνητα μια διεύθυνση διαδικτύου στον browser, στη συντριπτική πλειοψηφία των περιπτώσεων θα καταφύγουν στο Google, το Yahoo!, το MSN ή κάποια άλλη μηχανή αναζήτησης. Η τεχνολογία των μηχανών αναζήτησης στη σημερινή τους μορφή βασίζεται στη χρήση ειδικών τεχνικών ανάκτησης πληροφορίας (Information Retrieval).

Χρειάζεται να τονιστεί ότι οι μηχανές αναζήτησης ως ιδιωτικές εταιρείες δεν είναι πολύ πρόθυμες στο να κοινοποιούν τον τρόπο εργασίας τους και να αποκαλύπτουν τα μυστικά τους. Επομένως, παρουσιάζονται στη συνέχεια μερικά από τα στοιχεία τα οποία οι ίδιες οι μηχανές αναζήτησης αφήνουν να γνωστοποιηθούν.

Η κύρια λειτουργία των μηχανών αναζήτησης είναι η *διάσχιση* του Διαδικτύου (crawling). Κάθε μηχανή αναζήτησης έχει το δικό της πρόγραμμα – γνωστό ως spider ή bot – το οποίο επισκέπτεται τη μια σελίδα μετά την άλλη και αποστέλλει τα αποτελέσματα στη μηχανή αναζήτησης. Εδώ δυο σημεία είναι ενδιαφέροντα. Πρώτον, ότι αυτός είναι ο λόγος για τον οποίο μπορεί να επιστραφεί αποτέλεσμα που δε λειτουργεί από τη μηχανή αναζήτησης καθώς μπορεί το spider να μην έχει ενημερωθεί για τυχόν αλλαγές στην κάθε σελίδα. Δεύτερον, κάθε αναζήτηση δεν πραγματοποιείται στο σύνολο του Διαδικτύου αλλά στις σελίδες που έχουν επισκεφτεί τα spiders.

Η δεύτερη λειτουργία είναι το *indexing* κειμένου. Οι μηχανές αναζήτησης δεν ασχολούνται με τα γραφικά, τα video και τα scripts κάθε σελίδας. Όχι τουλάχιστον όσον

αφορά την καθημερινή αναζήτηση καθώς η αναζήτηση για περιεχόμενο τέτοιας μορφής προσφέρεται ως extra λειτουργία (πχ Google Images). Όσον αφορά το κείμενο, κάθε spider μηχανής που θα περάσει από μια σελίδα θα στείλει το περιεχόμενο της σελίδας σε μορφή καθαρού κειμένου. Το κείμενο θα πρέπει να καταχωρηθεί στο ευρετήριο (να γίνει indexed). Καταχωρείται στους servers της μηχανής αναζήτησης μαζί με εκατομμύρια άλλες εγγραφές πάνω στις οποίες υπολογίζονται τα ερωτήματα. Αναφορικά, το google μέχρι τον Ιούλιο του 2008 έχει αποθηκεύσει στους καταλόγους του περισσότερες από ένα τρισεκατομμύριο σελίδες⁴⁹.

Η τρίτη λειτουργία είναι η *επεξεργασία ερωτημάτων*. Όταν μια μηχανή αναζήτησης δεχτεί ένα ερώτημα, κάποια λέξη-κλειδί, τότε θα αντλήσει από τους servers της όλες τις σελίδες που περιέχουν τη λέξη αυτή. Πλην όμως, θα πρέπει να τα ταξινομήσει ανάλογα με το πόσο σχετικό κρίνεται το κάθε αποτέλεσμα σε σχέση με το ερώτημα, ώστε να παρουσιάσει στο χρήστη πρώτα τα πιο σχετικά αποτελέσματα.

Παρά τις διευκολύνσεις που προσφέρουν οι μηχανές αναζήτησης, πάσχουν από συγκεκριμένα προβλήματα. Ένα πρόβλημα είναι ότι ο μεγάλος όγκος πληροφορίας ο οποίος δεν είναι διαθέσιμος υπό τη μορφή κειμένου – είναι για παράδειγμα αποθηκευμένος σε Βάση Δεδομένων – δεν είναι προσβάσιμος. Η πληροφορία αυτή συχνά αναφέρεται ως Άγνωστο ή Βαθύ Διαδίκτυο (Deep Web) [1] σε αντίθεση με το Γνωστό Διαδίκτυο (Surface Web) το οποίο περιλαμβάνει τις στατικές HTML σελίδες.

Το σημαντικότερο πρόβλημα είναι όμως ότι η τακτική που χρησιμοποιείται είναι η αναζήτηση ίδιων συμβολοσειρών μέσα σε κείμενα. Αν κάποιος θέλει να αναζητήσει για παράδειγμα, πληροφορία για τους φοιτητές ενός τμήματος, θα εισάγει τη λέξη 'φοιτητές' στην αναζήτησή του. Με αυτόν τον τρόπο όμως θα χάσει σελίδες στις οποίες μπορεί να μην αναφέρεται η λέξη 'φοιτητές' αλλά μόνο τα ονόματα των φοιτητών για τους οποίους υπάρχει ενδιαφέρον. Απουσιάζει εν γένει η σημασιολογική πληροφορία από το Διαδίκτυο. Επιπλέον, κατά την αναζήτηση δε λαμβάνονται υπόψη ιεραρχίες και δομές που πιθανόν να υπάρχουν σε Δικτυακούς τόπους.

Αρκετές προσεγγίσεις έχουν γίνει στην επίλυση του προβλήματος αυτού, με κύρια ιδέα την προσθήκη σημασιολογικών μεταδεδομένων στην πληροφορία και εκμετάλλευση των σχέσεων αυτών προκειμένου να παραχθούν ακριβέστερα αποτελέσματα [110]. Στο προηγούμενο παράδειγμα, αν υπήρχε μια ιδιότητα της σελίδας των φοιτητών που να την κατέτασσε στη γενικότερη έννοια των φοιτητών, η σελίδα θα μπορούσε να επιστραφεί στο χρήστη, ακόμα κι αν δεν υπήρχε η λέξη "φοιτητές" στη σελίδα. Το σημαντικότερο πρόβλημα αυτών είναι όμως ότι ενώ βελτιώνουν τη συμπεριφορά των σημερινών μηχανών αναζήτησης, απαιτούν από το χρήστη ένα σχετικό βαθμό εξοικείωσης με το χώρο,

⁴⁹googleblog.blogspot.com/2008/07/we-knew-web-was-big.html

γεγονός που καθιστά την πρακτική αυτή μειωμένης χρησιμότητας για την πλειοψηφία των χρηστών.

Μια από τις σημαντικότερες εφαρμογές που έχουν αναπτυχθεί για το Σημασιολογικό Ιστό είναι το Swoogle⁵⁰, το οποίο παρουσιάζεται στο [79]. Το Swoogle αναλύει και αποθηκεύει έγγραφα σημασιολογίας και προσφέρει μηχανή αναζήτησης με τρόπο πολύ παρόμοιο του Google. Τα έγγραφα που αναγνωρίζει είναι αυτά που καταλήγουν σε .rdf, .owl, .daml και .n3. Λειτουργεί όπως και οι μηχανές αναζήτησης που περιγράψαμε προηγουμένως με τη διαφορά ότι οι σελίδες στις οποίες κινείται είναι έγγραφα του Σημασιολογικού Ιστού.

Η SHOE [111] είναι μια από τις πρώτες προσπάθειες για δημιουργία μηχανής αναζήτησης και παρέχει φόρμες για το σχηματισμό ερωτημάτων. Ωστόσο η μηχανή αυτή είναι χρήσιμη μόνο σε κάποιον ο οποίος είναι εξοικειωμένος με τις οντολογίες στις οποίες πραγματοποιείται η αναζήτηση. Οι χρήστες χωρίς την απαραίτητη γνώση αδυνατούν ακόμη και να κατανοήσουν τις φόρμες αυτές.

Η AquaLog [112] επίσης επιτρέπει την αναζήτηση ενσωματώνοντας τεχνικές επεξεργασίας φυσικής γλώσσας για τη μετατροπή των ερωτημάτων σε τριάδες.

Η ORAKEL [113] επίσης επεξεργάζεται τη φυσική γλώσσα προκειμένου να προσφέρει ένα μοντέλο για πρόσβαση σε Βάσεις Γνώσεως F-Λογικής.

Στη μηχανή αναζήτησης TAP [114], τα ερωτήματα είναι σημασιολογικής φύσης βασισμένα σε λέξεις-κλειδιά. Η διαδικασία αποτίμησης των ερωτημάτων αποτελείται από δύο βήματα. Πρώτα βρίσκεται ένα instance που να περιέχει τη λέξη κλειδί και στη συνέχεια βρίσκονται τα instances που συνδέονται με αυτά που περιέχουν τη λέξη κλειδί.

Η μηχανή αναζήτησης Corese (Conceptual Resource Search Engine) [115] είναι μια προσέγγιση που βασίζεται σε RDF(S).

Το SemSearch [116] προτείνει ένα τρόπο αναζήτησης σε πραγματικά δεδομένα αυτή τη φορά. Η διαφορά του από το Google είναι ότι ο χρήστης καλείται να πραγματοποιήσει αναζητήσεις εισάγοντας λέξεις στη μορφή “subject:keyword1 and/or keyword2 and/or keyword3 ...”. Για παράδειγμα, αν επιθυμεί να βρει νέα που να αφορούν μεταπτυχιακούς φοιτητές, μπορεί να εισάγει “news:postgraduate students”.

Επίσης μηχανή αναζήτησης για το Σημασιολογικό Ιστό είναι ο Falcon-S⁵¹, ο οποίος συγκεντρώνεται σε σημασιολογική αναζήτηση πληροφορίας σε αγώνες ποδοσφαίρου. Ο Sindice⁵² και ο Watson⁵³ είναι επίσης μηχανές αναζήτησης του Σημασιολογικού Ιστού

⁵⁰Swoogle: swoogle.umbc.edu

⁵¹Falcon-S: www.falcons.com.cn/saquery.jsp

⁵²Sindice: The Semantic Web Index: www.sindice.com

⁵³Watson: watson.kmi.open.ac.uk/WatsonWUI/

που προσφέρουν λειτουργικότητα ανάλογη του Swoogle.

Σημειώνεται ωστόσο, ότι πλήθος αποτελεσμάτων μπορεί να βρει κανείς και μέσω συμβατικών μηχανών αναζήτησης, από το Google για παράδειγμα με τη χρήση τελεστών όπως `filetype:owl` ή `filetype:rdf`. Η υποστήριξη σημασιολογίας και μικροπροτύπων από τις συμβατικές μηχανές αναζήτησης βρίσκεται ακόμη σε πρώιμο στάδιο. Τέλος, όπως αναφέρεται στην ενότητα 3.8, το εμπορικό μέλλον του Σημασιολογικού Ιστού είναι συνδεδεμένο και με την αποδοχή του από τις συμβατικές μηχανές αναζήτησης.

3.7.5 Πολυμεσικά δεδομένα

Πολλή προσπάθεια καταβάλλεται τελευταία στην κατεύθυνση της ενοποίησης των τεχνολογιών του Σημασιολογικού Ιστού με τα πολυμεσικά δεδομένα. Το πρόβλημα που προκύπτει είναι στη διαχείριση πολυμεσικής πληροφορίας καθώς στη φυσική τους μορφή τα δεδομένα αυτά δεν είναι κατανοητά από τον άνθρωπο. Για παράδειγμα, σε ένα αρχείο βίνεο, για παράδειγμα `mpeg`, η μορφή αποθήκευσης περιέχει δεδομένα για την κάθε εικόνα που αφορούν χρώματα και συντεταγμένες και δεδομένα για τον ήχο που αφορούν ένταση, συχνότητα `kok`. Δεν προβλέπεται η ένταξη στο αρχείο πληροφορίας που να αφορά το πού εξελίσσεται η σκηνή, τι προβάλλεται ή ποιοι εμφανίζονται.

Είναι επομένως απαραίτητη η προσθήκη μεταδεδομένων στην πολυμεσική πληροφορία ώστε να είναι εφικτή η κατηγοριοποίηση του υλικού και η πραγματοποίηση ευφυών αναζητήσεων πάνω σε αυτό. Το πρότυπο `MPEG7` επιτρέπει την αποθήκευση επισημειώσεων (`annotations`) σε αυτό, η δυνατότητα αυτή όμως είναι σε στοιχειώδες επίπεδο. Το επιθυμητό θα ήταν η επισημείωση να προσφέρονταν σε τρία επίπεδα: χαμηλό επίπεδο όπως για παράδειγμα χρώμα, υψηλό επίπεδο όπως αντικείμενα και ακόμη πιο υψηλό επίπεδο με σημασιολογία, επίπεδο που θα περιέχει πληροφορία όπως για παράδειγμα ομιλία, συνέδριο ή πολιτισμικό γεγονός.

Πολλά από τα χαρακτηριστικά αυτά μπορούν να εξαχθούν με αλγόριθμους από τομείς έρευνας όπως η όραση υπολογιστών ή η αναγνώριση προτύπων. Για τα υψηλότερου επιπέδου μεταδεδομένα είναι δυνατή – και εφαρμόζεται σε μεγάλο βαθμό – η καταχώρηση απλών λέξεων κλειδιών οι οποίες αποτελούν και το αντικείμενο αναζήτησης. Η διαδικασία αυτή όμως είναι συνήθως αρκετά επίπονη και δεν καλύπτει όλες τις πτυχές ενδιαφέροντος. Για τη διευκόλυνση της προσθήκης σημασιολογίας στα πολυμεσικά δεδομένα είναι αναγκαία η εξαγωγή πληροφορίας από ήδη καταχωρημένη πληροφορία. Στην κατεύθυνση αυτή διεξάγεται έρευνα και τμήμα αυτής παρατίθεται στη συνέχεια.

Στο έργο AceMedia⁵⁴ η έρευνα εστιάζεται στη δημιουργία οντολογικής υποδομής για την κατάλληλη περιγραφή πολυμεσικής πληροφορίας [117, 118]. Η σημασιολογική επισημείωση χωρίζεται σε χαμηλού επιπέδου όπως χρώματα, σχήματα και υψηλού επιπέδου όπως αν μια σκηνή εξελίσσεται σε εξωτερικό ή εσωτερικό χώρο. Στο [119] παρουσιάζεται μια οντολογία που αντιπροσωπεύει τη δομή του MPEG7 σε RDFS. Επιπλέον, ένα εργαλείο σημασιολογικής επισημείωσης, το OntoMat Annotizer⁵⁵ είναι ένα από τα κύρια αποτελέσματα της έρευνας αυτής. Ο M-Ontomat Annotizer [118, 120, 121] αποτελεί τμήμα του πλαισίου CREAM [122], συνδέει χαμηλού επιπέδου MPEG-7 οπτικούς περιγραφείς (visual descriptors) με οντολογίες RDF(S) και αποτελεί ένα από τα σημαντικότερα αποτελέσματα του έργου AceMedia.

Στο έργο M-Advantage (FP6-027131, 22.03.2005) γίνεται προσπάθεια για την περιγραφή πολυμεσικού περιεχομένου με τη χρήση OWL οντολογιών. Ο στόχος είναι η δημιουργία κεντρικής OWL οντολογίας η οποία να συμπεριλαμβάνει πλήρως το σχήμα περιγραφής πολυμεσικής πληροφορίας (Multimedia Description Schema, MDS) του MPEG7. Για τη διαδικασία της επισημείωσης, στην περίπτωση που αυτή δεν ενσωματώνεται στο περιεχόμενο, χρησιμοποιούνται τεχνολογίες όπως αναγνώριση φωνής για την εξαγωγή σημασιολογικής πληροφορίας.

Το έργο SCHEMA⁵⁶ ασχολείται με την ανάλυση πολυμεσικής πληροφορίας μέσω ανεξάρτητων modules τα οποία ανήκουν σε τρεις κατηγορίες:

1. Ανάλυση της εικόνας σε μικρότερες περιοχές για τις οποίες είναι δυνατή η εξαγωγή πληροφορίας χαμηλότερου επιπέδου, ολοκλήρωση σημασιολογίας υψηλού επιπέδου και indexing and retrieval άλλων μορφών (όπως audio based indexing and retrieval).
2. Εξαγωγή πληροφορίας υψηλότερου επιπέδου ώστε να παράγεται γνώση για παράδειγμα αν μια εικόνα περιέχει ένα πρόσωπο ή όχι ή αν περιέχει κάποιο συγκεκριμένο γεγονός όπως για παράδειγμα αν σημειώθηκε τέρμα σε αγώνα ποδοσφαίρου.
3. Επεξεργασία άλλων δεδομένων που μπορεί να συνοδεύουν την εικόνα, όπως ήχος και κείμενο.

Στο MUDPY [123] έχει παραχθεί μια απλή, γενική, ανεξάρτητη πλατφόρμα ελεύθερου κώδικα core engine υπό το όνομα MUD η οποία είναι εύκολη στην κατανόηση και εύκολα επεκτάσιμη. Το έργο BOEMIE [124] προτείνει μια προσέγγιση στην ανάπτυξη οντολογιών η οποία βασίζεται σε semantics που εξάγονται από διάφορα modules και

⁵⁴AceMedia: www.acemedia.org

⁵⁵OntoMat-Annotizer: annotation.semanticweb.org/ontomat/

⁵⁶SCHEMA: www.iti.gr/SCHEMA

στόχο έχει την κατά το δυνατό αυτοματοποίηση της διαδικασίας που χρειάζεται για τη συγγραφή οντολογίας. Το έργο Kowari⁵⁷ παρέχει πρόσβαση σε ontology datastores με τη χρήση JRDF και SOAP. Στο έργο MUSCLE⁵⁸ ερευνάται η αυτόματη εξαγωγή σημασιολογίας υψηλού επιπέδου. Το σύστημα επισημείωσης AKTive Media [125] είναι επίσης βασισμένο σε οντολογία.

Η προσέγγιση της χειροκίνητης επισημείωσης συνήθως συμπληρώνεται από τεχνικές ημιαυτόματης εξαγωγής μεταδεδομένων. Αυτή η κατηγορία περιέχει εργαλεία όπως το Vannotea [126] που επισημαίνει συλλογές από εικόνες, video, ήχο, ή τρισδιάστατα αντικείμενα (MPEG-2, JPEG2000 και Direct 3D).

3.8 Συμπεράσματα

Υπάρχει πλήθος τεχνολογιών για τη διαχείριση και την επεξεργασία σημασιολογικής πληροφορίας. Η σημασιολογία, όπως φαίνεται στο κεφάλαιο αυτό μπορεί να συνοδεύει πλήθος άλλων “οικοσυστημάτων”. Η έλευση προτύπων όπως το RDF και η SPARQL επιτρέπει τη δημιουργία εργαλείων που να είναι σε θέση να διαχειριστούν σημασιολογική πληροφορία και να την ενσωματώσουν σε άλλες τεχνολογίες του Διαδικτύου. Το ερώτημα όμως που παραμένει είναι το γιατί περίπου μια δεκαετία έρευνας πάνω στο Σημασιολογικό Ιστό, το όνομά του δεν έχει συνδεθεί ακόμη με τις υπηρεσίες που παρέχονται στον τελικό χρήστη. Ο Tim Berners-Lee σε πρόσφατο άρθρο του διαπιστώνει ότι η απλή αυτή ιδέα παραμένει σε μεγάλο βαθμό μη υλοποιημένη [13].

Παράλληλα με την τεχνολογική υποδομή η οποία έχει διανύσει αρκετό δρόμο, υπάρχει επίσης και πλήθος πληροφορίας η οποία έχει αποτυπωθεί κατά τρόπο ώστε να αποκλείει αμφισημίες. Το έργο dbpedia για παράδειγμα (dbpedia.org) αριθμεί μέχρι στιγμής περισσότερες από 100 εκατομμύρια τριάδες οι οποίες βασίζονται στις περιγραφές της wikipedia.

Το πρόβλημα εντοπίζεται στην *αξιοποίηση* της σημασιολογικής περιγραφής. Ουσιαστικά, εντοπίζεται στο επόμενο βήμα μετά την προσθήκη σημασιολογικών μεταδεδομένων στα δεδομένα. Χρειάζεται η δημιουργία εφαρμογών που να αναγνωρίζουν και να εκμεταλλεύονται σημασιολογική πληροφορία προκειμένου να προσφέρουν στον τελικό χρήστη πιο πλούσιες υπηρεσίες.

Διαπιστώνεται μια ανεπάρκεια στην επιχειρηματολογία για τη χρησιμότητα του Σημασιολογικού Ιστού, ή αλλιώς, μια περιορισμένη *αξιοποίηση* της σημασιολογικής πληροφορίας. Δεν είναι σαφές στην πλειονότητα των εφαρμογών που παρουσιάζονται

⁵⁷Kowari: www.kowari.org

⁵⁸Muscle: www.muscle-noe.org

το πώς μπορεί να αξιοποιηθεί η επιπλέον πληροφορία. Επομένως, το πρόβλημα συνοψίζεται στο εξής: Ενώ χρειάζεται επιπλέον προσπάθεια σε σχέση με τις συμβατικές πρακτικές για τον εμπλουτισμό της πληροφορίας με σημασιολογία: αυτοματοποίηση για τους λόγους που αναφέρθηκαν, εμπειρία σε διάφορους γνωστικούς τομείς κλπ, ο σχεδιαστής συστημάτων ή ο υπεύθυνος μιας εταιρείας δεν έχει πειστεί ακόμη ότι η προσθήκη σημασιολογίας θα του αποδώσει καλύτερα αποτελέσματα για παράδειγμα στην αναζήτηση. Ενώ πολλά τμήματα του Σημασιολογικού Ιστού έχουν αναπτυχθεί και ωριμάσει, η συνολική ιδέα του οράματος δεν έχει υλοποιηθεί ακόμη [127]. Το πρόβλημα που υπάρχει μέχρι σήμερα και αποτελεί τροχοπέδη στην ευρύτερη εξάπλωση του Σημασιολογικού Ιστού εντοπίζεται στην απουσία κινήτρου για το σημασιολογικό εμπλουτισμό της πληροφορίας στους διαχειριστές της.

Σε μεμονωμένα παραδείγματα όπως για παράδειγμα στο [128] περιγράφεται η διευκόλυνση που παρέχει στις online κινήσεις ενός blogger η ύπαρξη μικροπροτύπων που εξασφαλίζουν ότι τα δεδομένα της είναι συνδεδεμένα. Η καθιέρωση του Web 2.0 ως όρου δίνει πεδίο ανάπτυξης στο Σημασιολογικό Ιστό όσον αφορά την αξιοποίηση της σημασιολογικής πληροφορίας τόσο από μηχανές αναζήτησης όσο και από τον άνθρωπο. Κατά το 2007, ο αριθμός των τριάδων που υπάρχουν ως διασυνδεδεμένα δεδομένα στο Διαδίκτυο ξεπερνά τις 2 δισ.[44]. Υπάρχει επομένως σημασιολογικό περιεχόμενο, κάτι που απουσίαζε μέχρι σήμερα. Έτσι, με αφετηρία το περιεχόμενο αυτό αλλά και τα μικροπρότυπα που παρέχονται (ενότητα 2.5) μπορεί ένας χρήστης να επισημειώσει το περιεχόμενο που δημοσιοποιεί στο Διαδίκτυο. Αντίστοιχα, μπορεί και να διαπιστώσει την εγκυρότητα της πληροφορίας στην οποία περιηγείται προκειμένου να είναι σίγουρος για την εγκυρότητά της.

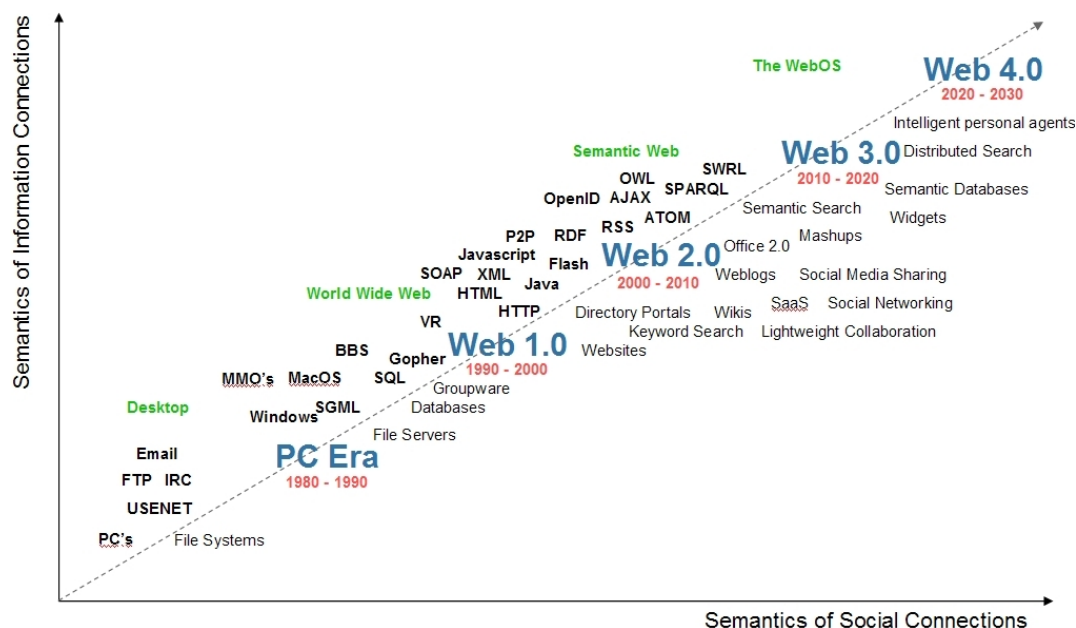
Το Web 3.0 αναμένεται να είναι συνδεδεμένο με το Σημασιολογικό Ιστό. Ήδη, μηχανές αναζήτησης αξιοποιούν σημασιολογική πληροφορία η οποία μπορεί να περιέχεται σε ιστοσελίδες. Το Yahoo! για παράδειγμα, προσφέρει το SearchMonkey, μια επέκταση στη μηχανή αναζήτησης η οποία αναγνωρίζει αρκετά μικροπρότυπα του Σημασιολογικού Ιστού⁵⁹ (εικόνα 8). Αντίστοιχα, υπάρχουν επεκτάσεις για τους περιηγητές Διαδικτύου (browsers), όπως το Semantic Radar⁶⁰ τα οποία αναγνωρίζουν σημασιολογικό περιεχόμενο. Παρέχεται επομένως το κίνητρο στην ανάπτυξη εφαρμογών που να υποστηρίζουν και να εκμεταλλεύονται το Σημασιολογικό Ιστό.

Η κατεύθυνση στην οποία φαίνεται να κινείται η ανάπτυξη του Σημασιολογικού Ιστού είναι επομένως η προσθήκη επισημειώσεων στις σελίδες του Διαδικτύου, γεγονός που θα μετατρέψει το Web 3.0 σε πλήθος διασυνδεδεμένων, επισημειωμένων σημασιολογικά

⁵⁹developer.yahoo.com/searchmonkey/smguides/profile_vocab.html

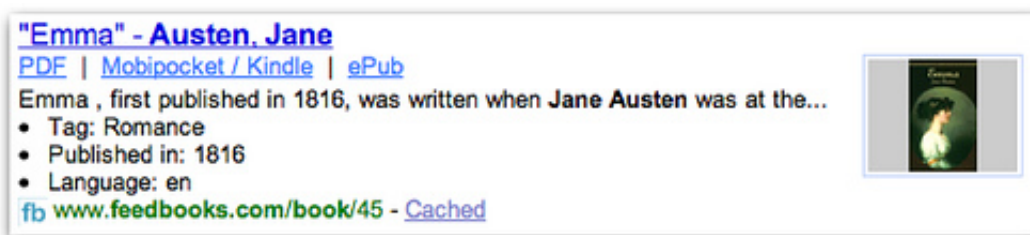
⁶⁰Semantic Radar: sioc-project.org/firefox

3 Διαχείριση, επεξεργασία και αξιοποίηση σημασιολογικής πληροφορίας



Εικόνα 7: Ιστορία του Διαδικτύου

δεδομένων [44]. Τα δεδομένα αυτά θα ενέχουν περισσότερη αξιοπιστία από τα σημερινά και θα προσθέτουν στην εμπειρία του τελικού χρήστη δίνοντάς του τη δυνατότητα για ευφυέστερες αναζητήσεις και παρέχοντας ακριβέστερα αποτελέσματα στις αναζητήσεις αυτές. Περισσότερο από ποτέ, όπως σχηματικά απεικονίζεται και στην εικόνα 7 ο στόχος του Σημασιολογικού Ιστού φαίνεται να υλοποιείται στο άμεσο μέλλον⁶¹.



Εικόνα 8: Αποτέλεσμα σημασιολογικά εμπλουτισμένο με RDFa σε μηχανή αναζήτησης

Σχετικά με το εμπορικό μέλλον του Σημασιολογικού Ιστού, φαίνεται πως το ισχυρότερο κίνητρο θα δοθεί για την ευρύτερη εξάπλωσή του όταν η σημασιολογική

⁶¹ Πηγή: Radar networks & Nova Spivack, 2007 (www.radarnetworks.com)

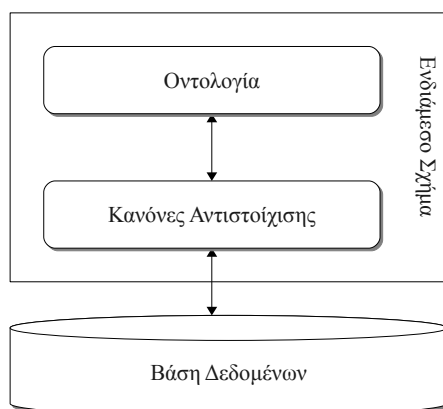
πληροφορία γίνει εκμεταλλεύσιμη από μηχανές αναζήτησης. Όταν η σημασιολογική πληροφορία χρησιμοποιηθεί από τις μηχανές αναζήτησης για να δίνει πιο πλούσια αποτελέσματα, όπως στην εικόνα 8 (πηγή: [129]) θα δοθεί κίνητρο σε εταιρείες για επισημείωση του online περιεχομένου τους. Ωστόσο, η υποστήριξη RDFa ή μικροπροτύπων από μηχανές αναζήτησης βρίσκεται ακόμη σε πρώιμο στάδιο.

Το αποτέλεσμα της εικόνας 8 θα είναι πιο ελκυστικό ανάμεσα στα αποτελέσματα της αναζήτησης του χρήστη και αναμένεται πιθανώς να προσελκύσει περισσότερους επισκέπτες, γεγονός που θα δίνει συγκριτικό πλεονέκτημα στο δικτυακό τόπο. Αν επιπλέον, ανάμεσα στις κινήσεις που χρειάζεται να κάνει κανείς για να παράξει μια σελίδα πιο φιλική [130] στις μηχανές αναζήτησης συμπεριληφθεί και η προσθήκη σημασιολογίας, τότε το μέλλον του Σημασιολογικού Ιστού καθίσταται ευοίωνο.

4 Συνεργασία οντολογιών με Βάσεις Δεδομένων

4.1 Εισαγωγή

Το κεφάλαιο αυτό αναλύει το πρόβλημα της αντιστοίχισης και συνεργασίας οντολογιών και Βάσεων Δεδομένων. Η ανάπτυξη μηχανισμών που να επιτρέπουν την αντιστοίχιση αυτή επιλύει ταυτόχρονα δύο προβλήματα. Αφενός, η πλειοψηφία των δεδομένων τα οποία υπάρχουν στο Διαδίκτυο προέρχονται από συμβατικές σχεσιακές Βάσεις Δεδομένων, οι οποίες δεν ενσωματώνουν σημασιολογική πληροφορία. Αφετέρου, οι οντολογικές περιγραφές με μεγάλο αριθμό individuals πάσχουν από κλιμακωσιμότητα. Στο [131] παρουσιάζεται μια προσέγγιση η οποία είναι σε θέση να διαχειρίζεται μέχρι 100.000 individuals, αριθμός ο οποίος δε μπορεί να προσεγγίσει τις επιδόσεις των σύγχρονων Βάσεων Δεδομένων⁶². Σχηματική απεικόνιση του προβλήματος της συνεργασίας των οντολογιών με τις Βάσεις Δεδομένων παρουσιάζεται στην εικόνα 9.



Εικόνα 9: Συνεργασία οντολογιών με Βάσεις Δεδομένων

Το πρόβλημα της αντιστοίχισης Βάσεων Δεδομένων και οντολογιών μπορεί γενικότερα να θεωρηθεί ως ειδική περίπτωση του προβλήματος ολοκλήρωσης της πληροφορίας (ενότητα 1.2). Στην ειδικότερη αυτή περίπτωση ολοκλήρωσης της πληροφορίας δεν υπάρχει συγκεκριμένη καθιερωμένη προσέγγιση λόγω της ετερογένειας που διακρίνει τα δυο πεδία. Οι δυσκολίες που εντοπίζονται οφείλονται κατά κύριο λόγο στη διαφοροποίηση των σκοπών για τους οποίους έχουν σχεδιαστεί και εξυπηρετούν τα δύο πεδία, αλλά και των μεθοδολογιών ανάπτυξης και χρήσης τους. Ενδεικτικά,

⁶²Η Oracle 11g μπορεί να επεκταθεί μέχρι 8 exabytes (10^{18} bytes) σε μέγεθος [132]

ένα σχήμα Βάσης Δεδομένων δεν είναι συνηθισμένο να διαμοιράζεται μεταξύ πολλών χρηστών και να επαναχρησιμοποιείται για διαφορετικούς σκοπούς, αντίθετα σχεδιάζεται και χρησιμοποιείται για να εξυπηρετεί συγκεκριμένους σκοπούς. Αντίθετα, μια οντολογία είναι εξ' ορισμού επαναχρησιμοποιήσιμη και διαμοιράσιμη. Επίσης σημαντική διαφορά αποτελεί η προσέγγιση ανάπτυξης. Η ανάπτυξη μιας οντολογίας είναι συχνά αποτέλεσμα συντονισμένης προσπάθειας ομάδας ανθρώπων ενώ το σχήμα μιας Βάσης Δεδομένων σπάνια προκύπτει από συνεργασία [30]. Επιπλέον, το σχήμα της Βάσης Δεδομένων δεν παρέχει συγκεκριμένη σημασιολογία σε αντίθεση με την οντολογία.

Η έρευνα στο χώρο των Βάσεων Δεδομένων από τη σκοπιά του Σημασιολογικού Ιστού χωρίζεται σε δυο υποκατηγορίες. Από τη μια πλευρά υπάρχει η προσπάθεια συνεργασίας Βάσεων Δεδομένων και οντολογιών, από την άλλη υπάρχει η προσπάθεια μεταπήδησης (migration) από τις Βάσεις Δεδομένων στις βασισμένες σε οντολογίες Βάσεις Γνώσεως. Στην πρώτη περίπτωση, προσπάθεια καταβάλλεται ώστε να προσφέρονται προσεγγίσεις στην αντιστοίχιση δεδομένων αποθηκευμένων σε Βάση Δεδομένων με έννοιες που περιγράφονται σε οντολογίες. Στη δεύτερη, γίνεται προσπάθεια ώστε να επιλέγονται σύνολα από τη Βάση Δεδομένων και να μεταφέρονται στην οντολογία και αντίστροφα. Όλα τα εργαλεία τα οποία παρουσιάζονται εμπίπτουν στην πρώτη κατηγορία. Εξάιρεση αποτελεί το Relational.OWL [133] το οποίο προτείνει απευθείας εξαγωγή της σχεσιακής Βάσης Δεδομένων σε πρότυπο RDF ή OWL και την μετέπειτα επεξεργασία του αποτελέσματος από reasoner. Στα υπόλοιπα εργαλεία ο στόχος είναι η ολοκλήρωση μέσω της χρήσης εργαλείου αντιστοίχισης.

Το κίνητρο για την παρούσα έρευνα είναι δυσπόστατο. Αφενός, υπάρχει επιτακτική ανάγκη για ουσιαστική ολοκλήρωση των τεχνολογιών του Σημασιολογικού Ιστού με τις υπάρχουσες τεχνολογίες: υπάρχει τεράστιος όγκος δεδομένων του Διαδικτύου ο οποίος δεν είναι προσβάσιμος από τις μηχανές αναζήτησης καθώς η πληροφορία παράγεται δυναμικά κατόπιν αιτημάτων των χρηστών [1, 2]. Αφετέρου, η αποτελεσματικότητα των υφιστάμενων προσεγγίσεων εμφανίζεται περιορισμένη. Επιπλέον, σημειώνεται ότι η παρούσα έρευνα αποτελεί αντικείμενο το οποίο απουσίαζε από την υπάρχουσα βιβλιογραφία. Δεδομένης της μη ωρίμανσης των σχετικών τεχνολογιών και της απουσίας συγκεκριμένων προτύπων, κρίθηκε αναγκαία μια μελέτη σε βάθος και καταγραφή των υπάρχουσών προσεγγίσεων που μαζί συνθέτουν την τρέχουσα αιχμή της έρευνας στο χώρο της συνεργασίας του Σημασιολογικού Ιστού και των Βάσεων Δεδομένων.

4.1.1 Αντιστοίχιση και ολοκλήρωση δεδομένων

Στο [134] παρέχεται ένας ορισμός για την αντιστοίχιση: Μια αντιστοίχιση είναι ένας προσδιορισμός ενός μηχανισμού με τον οποίο στοιχεία ενός μοντέλου το οποίο

συμμορφώνεται σε ένα μετα-μοντέλο μετασχηματίζονται σε στοιχεία ενός άλλου μοντέλου, το οποίο συμμορφώνεται με ένα διαφορετικό – ή μπορεί και το ίδιο – μετα-μοντέλο. Μια αντιστοίχιση μπορεί να δηλώνεται ως ένα σύνολο σχέσεων, περιορισμών, κανόνων, προτύπων με παραμέτρους που καθορίζονται κατά τη διάρκεια της αντιστοίχισης, ή με άλλες μορφές που δεν έχουν καθοριστεί ακόμη.

Η έννοια της αντιστοίχισης συνδέεται στενά με την έννοια της ολοκλήρωσης. Για την ολοκλήρωση των δεδομένων (της πληροφορίας) έγινε λόγος στην ενότητα 1.2. Επιπρόσθετα, ένα Σύστημα Ολοκλήρωσης Δεδομένων (Data Integration System) \mathcal{I} ορίζεται ως μια τριάδα

$$\mathcal{I} := \langle \mathcal{G}, \mathcal{S}, \mathcal{M} \rangle$$

όπου

- \mathcal{G} είναι το καθολικό σχήμα,
- \mathcal{S} είναι το σχήμα των πηγών και
- \mathcal{M} είναι η αντιστοίχιση μεταξύ του \mathcal{G} και των \mathcal{S} .

Κατά την προσέγγιση LAV system, κάθε δήλωση στο \mathcal{M} αντιστοιχίζει ένα στοιχείο του σχήματος πηγής \mathcal{S} σε ένα ερώτημα (μια όψη) στο καθολικό σχήμα \mathcal{G} . Κατά την GAV προσέγγιση ισχύει το αντίθετο: κάθε δήλωση στο \mathcal{M} αντιστοιχίζει ένα στοιχείο του συνολικού σχήματος \mathcal{G} σε ένα ερώτημα (μια όψη) στο σχήμα πηγής \mathcal{S} . Το καθολικό σχήμα \mathcal{G} αποτελεί μια εναρμονισμένη όψη του ετερογενούς συνόλου των πηγών δεδομένων. Γενικά, ο στόχος παραμένει ο ίδιος: η ενοποίηση των υποκείμενων πηγών δεδομένων υπό ένα κοινό σχήμα.

Με βάση τον ορισμό των συστημάτων ολοκλήρωσης δεδομένων, ορίζεται και η έννοια της αντιστοίχισης. Μια αντιστοίχιση m από ένα σχήμα \mathcal{S} σε ένα σχήμα \mathcal{T} είναι μια δήλωση της μορφής $Q^S \rightsquigarrow Q^T$ όπου Q^S είναι ένα ερώτημα στο \mathcal{S} και Q^T είναι ένα ερώτημα στο \mathcal{T} [135].

4.1.2 Προτυποποίηση συστημάτων με τη χρήση Τεχνικών Χώρων

Κατά το σχεδιασμό συστημάτων Διαδικτύου, οι μηχανικοί λογισμικού συνήθως καλούνται να επιλέξουν τις περιβάλλουσες έννοιες του χώρου στον οποίο χρειάζεται να κινηθούν ώστε να αντιμετωπίσουν αποδοτικά το πρόβλημα ανά χείρας.

Προκειμένου να αποτυπωθεί το συγκεκριμένο πεδίο δράσης, έχει προταθεί ο όρος Τεχνικός Χώρος (Technical Space). Ένας τεχνικός χώρος είναι ένας γενικότερος εννοιολογικός χώρος ο οποίος αποτελείται από ένα σύνολο αλληλο-συσχετιζόμενων εννοιών, γλωσσών, εργαλείων και δυνατοτήτων [136].

Πριν τη λήψη απόφασης για την υιοθέτηση συγκεκριμένου τεχνικού χώρου, ή συνδυασμού αυτών, ο σχεδιαστής χρειάζεται να λάβει υπόψη τα υπέρ και τα κατά του κάθε χώρου. Συγκεκριμένα, τα σημεία ενδιαφέροντος είναι η εκφραστικότητα των παρεχόμενων γλωσσών, η ευκολία στη διαλειτουργικότητα μεταξύ διαφορετικών τεχνικών χώρων, το στάδιο προτυποποίησης των τεχνολογιών καθώς επίσης και μια σειρά από τεχνικές διευκολύνσεις, όπως για παράδειγμα η ασφάλεια, η ανοχή σε σφάλματα ή η φιλικότητα προς το χρήστη.

Παρά το γεγονός ότι είναι δύσκολο κανείς να δώσει ακριβή περιγραφή του τι ακριβώς είναι ένας τεχνικός χώρος, μπορεί εύκολα να τους αναγνωρίσει.

1. Τεχνολογίες βασισμένες σε XML (XML TS)
2. Model-Driven Architecture όπως ορίζεται από το Object Management Group (OMG) (MDA TS)
3. Τεχνικός Χώρος Συστημάτων Διαχείρισης Βάσεων Δεδομένων (DBMS TS)
4. Τεχνικός Χώρος οντολογιών (Ontology TS)

Αυτά είναι μερικά μόνο παραδείγματα από τους πιο ευρέως διαδεδομένους Τεχνικούς Χώρους. Το κεφάλαιο αυτό εξετάζει πιο λεπτομερώς τους τεχνικούς χώρους των οντολογιών και των Βάσεων Δεδομένων. Αξίζει να σημειωθεί ότι μια προσέγγιση μπορεί να εκτείνεται σε περισσότερους του ενός Τεχνικούς Χώρους, όπως για παράδειγμα στο [137] όπου οι συγγραφείς παρουσιάζουν μια προσέγγιση για σημασιολογική ολοκλήρωση των γλωσσών περιγραφής μετα-μοντέλων.

4.1.3 Σχεσιακό μοντέλο

Σύμφωνα με το [46], μια σχεσιακή Βάση Δεδομένων είναι μια συλλογή από διαφορετικές σχέσεις (relations). Μια σχέση αποτελείται από ένα σχήμα και ένα στιγμιότυπο σχέσης. Το σχήμα της Βάσης Δεδομένων αποτελείται από τα σχήματα των σχέσεων ενώ το στιγμιότυπο της Βάσης περιέχει τα στιγμιότυπα των σχέσεων. Τα στιγμιότυπα των σχέσεων είναι στην ουσία σύνολα εγγραφών τα οποία ονομάζονται πλειάδες (tuples). Πρόκειται στην ουσία για τους πίνακες της Βάσης Δεδομένων στους οποίους κάθε πλειάδα είναι και μια γραμμή του πίνακα. Όλες οι πλειάδες μέσα σε μια σχέση έχουν τον ίδιο αριθμό πεδίων. Το πεδίο ορισμού για κάθε πεδίο είναι ουσιαστικά ο τύπος του πεδίου δηλαδή ακολουθία χαρακτήρων, ακέραιος αριθμός, boolean ή κάποιος άλλος τύπος γνωστός και από τις γλώσσες προγραμματισμού.

Προκειμένου να υπάρχει πρόσβαση στα περιεχόμενα σχεσιακών Βάσεων Δεδομένων, πολλές γλώσσες ερωτημάτων έχουν αναπτυχθεί, η SQL όμως αποτελεί πλέον την πρακτικότερη επιλογή. Η είσοδος και το αποτέλεσμα των SQL ερωτημάτων είναι σχέσεις.

Τα ερωτήματα αποτιμώνται πάνω στα στιγμιότυπα των σχέσεων της Βάσης Δεδομένων και παράγουν στιγμιότυπα σχέσεων εξόδου. Η SQL είναι μια σχεσιακά πλήρης γλώσσα – εφόσον είναι και υπερσύνολο της σχεσιακής άλγεβρας – καθώς κάθε ερώτημα που μπορεί να διατυπωθεί σε σχεσιακή άλγεβρα μπορεί επίσης να διατυπωθεί και σε SQL. Με άλλα λόγια, δεν υπάρχει συνδυασμός υποσυνόλων ή πλειάδων που να είναι αποθηκευμένα σε Βάση Δεδομένων και να μην είναι δυνατό να ανακτηθούν με SQL ερωτήματα.

4.1.4 Αντιστοίχιση σχεσιακών Βάσεων Δεδομένων με οντολογίες

Σύμφωνα με το [134], μια *αντιστοίχιση* είναι ο καθορισμός ενός μηχανισμού σύμφωνα με τον οποίο τα στοιχεία ενός μοντέλου το οποίο υπακούει σε ένα μετα-μοντέλο είναι δυνατό να μετασχηματιστούν σε στοιχεία ενός άλλου μοντέλου, το οποίο συμβαδίζει με ένα άλλο (πιθανότατα το ίδιο) μετα-μοντέλο. Μια αντιστοίχιση μπορεί να εκφράζεται ως ένα σύνολο από συσχετισμούς, περιορισμούς, κανόνες, πρότυπα (templates) με παραμέτρους που καθορίζονται κατά τη διαδικασία της αντιστοίχισης.

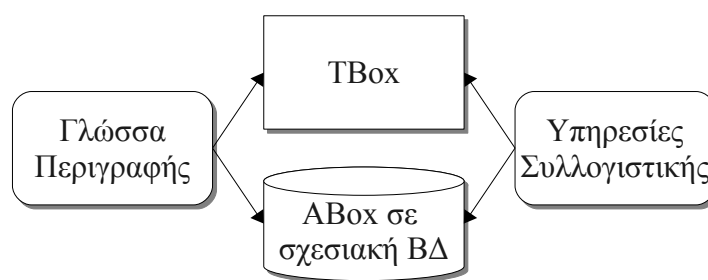
Από μια απλοποιημένη οπτική γωνία, θα μπορούσαμε να ισχυριστούμε ότι το σχήμα μιας Βάσης Δεδομένων μπορεί να συγκριθεί με το TBox μιας Βάσης Γνώσεως ενώ το στιγμιότυπο που περιέχει και τα πραγματικά δεδομένα, με το ABox. Θα μπορούσαμε να ισχυριστούμε ότι οι Βάσεις Δεδομένων και οι Βάσεις Γνώσεως εξυπηρετούν τον ίδιο σκοπό καθώς στην πράξη και οι δυο χρησιμοποιούνται για την αποθήκευση και διατήρηση ενός μοντέλου και κάποιων δεδομένων σχετικά με ένα πεδίο ενδιαφέροντος [138].

Υπάρχουν ωστόσο σημαντικές διαφορές. Οι Βάσεις Δεδομένων είναι σε θέση να διαχειρίζονται τεράστια μοντέλα από σχετικά απλά συνδεδεμένα δεδομένα ενώ οι Βάσεις Γνώσεως προορίζονται για τη διαχείριση μικρότερου όγκου αλλά αυξημένης πολυπλοκότητας δεδομένων. Η βασικότερη διαφορά όμως είναι ότι οι Βάσεις Γνώσεως μπορούν να απαντήσουν και σε ερωτήματα που αφορούν γνώση η οποία δεν καταχωρήθηκε ρητά αλλά προκύπτει επαγωγικά από τα δεδομένα τους. Σε αντίθεση, οι Βάσεις Δεδομένων γνωρίζουν μόνο τα δεδομένα που καταχωρούνται χωρίς να προβαίνουν σε περαιτέρω επεξεργασία τους.

Επομένως, με την αντιστοίχιση Βάσεων Δεδομένων σε Βάσεις Γνώσεως επιτυγχάνεται η προσθήκη συνεπαγόμενης γνώσεως στις Βάσεις Δεδομένων χρησιμοποιώντας την τερμινολογία που περιγράφεται στο TBox της Βάσης Γνώσεως. Από διαφορετική θεώρηση θα μπορούσαμε να ισχυριστούμε ότι στην ουσία πρόκειται για την αποθήκευση του ABox της Βάσης Γνώσεως μέσα σε μια Βάση Δεδομένων [139]. Η προσέγγιση αυτή παρουσιάζεται σχηματικά στην εικόνα 10. Η βασική ιδέα είναι η διατήρηση του ABox της οντολογίας ξεχωριστά από την οντολογία, σε σχεσιακή Βάση Δεδομένων. Η Βάση Γνώσεως που προκύπτει θα μπορούσε να χαρακτηριστεί ως

υβριδική αφού μέσα στην οντολογία περιέχονται μόνο αναφορές για την πραγματική θέση των δεδομένων και όχι τα ίδια τα δεδομένα.

Επίσης, πολλά εργαλεία έχουν προταθεί όπως τα Sesame [56], Corese [115], Instance Store [131], 3Store [140], Kowari [141, 142], τα οποία αναλαμβάνουν την αποθήκευση οντολογίας σε Βάση Δεδομένων, δεν αφήνουν όμως στο χρήστη την επιλογή αποθήκευσης στη Βάση· ο χρήστης μπορεί μόνο να αλληλεπιδράσει με το επίπεδο της οντολογίας.



Εικόνα 10: Αντιστοίχιση σχεσιακών Βάσεων Δεδομένων με οντολογίες

4.1.5 Θεώρηση ανοιχτού και κλειστού κόσμου

Ανάμεσα στις διαφορές μεταξύ των Βάσεων Δεδομένων και των Βάσεων Γνώσεως είναι η θεώρηση ανοιχτού κόσμου (open-world semantics) που ισχύει στις Βάσεις Γνώσεως σε αντίθεση με τη θεώρηση κλειστού κόσμου (closed-world semantics) που ισχύει στις Βάσεις Δεδομένων – και την XML. Στις Βάσεις Δεδομένων, η θεώρηση κλειστού κόσμου επιβάλλει ότι η αντίληψη ενός συστήματος για τον κόσμο περιορίζεται σε ό,τι έχει δηλωθεί ρητά σε αυτό. Οτιδήποτε δεν έχει δηλωθεί θεωρείται μη αληθές. Αντίθετα, στη θεώρηση ανοιχτού κόσμου (open-world semantics), οτιδήποτε δεν έχει δηλωθεί θεωρείται άγνωστο και όχι εσφαλμένο. Η ιδιότητα αυτή συνδέεται στενά με την *μονοτονικότητα* του Κατηγορικού Λογισμού Πρώτης Τάξης: Η προσθήκη νέων δεδομένων στο σύστημα δεν αναιρεί τα προηγούμενα συμπεράσματα. Μια Βάση Γνώσεως θα δώσει απάντηση ότι κάτι είναι άγνωστο αν δεν το γνωρίζει. Έστω το εξής παράδειγμα⁶³:

Δήλωση: Ο John είναι πολίτης της Γαλλίας.

Ερώτηση: Είναι ο John πολίτης του Καναδά;

⁶³Πηγή: en.wikipedia.org/wiki/Open_world_assumption

Απάντηση, θεώρηση κλειστού κόσμου (XML, SQL): Όχι

Απάντηση, θεώρηση ανοιχτού κόσμου (RDF, OWL): Άγνωστο, καθώς ο John θα μπορούσε να έχει διπλή υπηκοότητα.

4.1.6 Θεωρητικοί περιορισμοί

Μπορεί κανείς εύκολα να παρατηρήσει ότι οι DDLs (Data Definition Languages) της SQL ενσωματώνουν εκφραστικότητα η οποία δεν παρουσιάζεται στις γλώσσες περιγραφής της Περιγραφικής Λογικής και αντίστροφα. Αναφορικά:

- Το σχεσιακό μοντέλο μπορεί να περιέχει επιπλέον περιορισμούς: DEFAULT, NOT NULL, UNIQUE, AUTO INCREMENT
- Μπορεί κανείς να ορίσει κανόνες (triggers) μέσα από την SQL: ON INSERT, ON UPDATE, ON DELETE
- Stored Procedures
- Transactions (START TRANSACTION, COMMIT, ROLLBACK)
- Θέματα ασφάλειας όσον αφορά την αποθήκευση και ανάκτηση οντολογιών δεν αντιμετωπίζονται ακόμα από τις οντολογίες

Θα μπορούσε κανείς να πει ότι δεν είναι δίκαιο να συγκρίνονται μεταξύ τους οι γλώσσες SQL και SPARQL (ενότητα 2.7.2) διότι η πρώτη αποτελεί μια ώριμη, προτυποποιημένη και ευρέως διαδεδομένη γλώσσα ενώ η δεύτερη βρίσκεται ακόμα υπό σχεδιασμό ακόμα και όσον αφορά τον επίσημο ορισμό της (τα formal semantics της). Επιπλέον, η SPARQL κατά τον επίσημο ορισμό της παρουσιάζει περιορισμούς και ελλείψεις στη συμπεριφορά της, οι οποίοι δεν εμφανίζονται στην SQL. Σημειώνεται ότι οι περιορισμοί αυτοί αφορούν τον επίσημο ορισμό της SPARQL και όχι τις υλοποιήσεις της, όπως η ARQ για παράδειγμα οι οποίες επεκτείνουν τη λειτουργικότητά τους πέρα από τον επίσημο ορισμό της SPARQL.

4.2 Ανασκόπηση βιβλιογραφίας

Το έργο *MOMIS* [143, 144] ασχολείται με το πρόβλημα της ενοποίησης ετερογενών πηγών πληροφορίας. Η προσέγγιση που ακολουθείται είναι η global-as-view (GAV). Αποτελείται από μια συλλογή εργαλείων τα οποία παρέχουν πρόσβαση σε πληροφορία η οποία βρίσκεται αποθηκευμένη σε συστήματα Βάσεων Δεδομένων ή απλά συστήματα αρχείων (filesystems) ή απλές πηγές ημιδομημένης πληροφορίας. Το *MOMIS* χρησιμοποιεί την *ODL₁₃*, μια object-oriented γλώσσα η οποία προτείνεται από το ODMG

(Object Database Management Group) με στόχο να αναπαραστήσει τα semantics τα οποία καθορίζουν το ενιαίο σχήμα.

Τα βασικά χαρακτηριστικά της αρχιτεκτονικής του MOMIS είναι

- οι wrappers οι οποίοι διαχειρίζονται όλες τις τοπικές πηγές πληροφορίας
- ο mediator ο οποίος περιέχει έναν ενιαίο (global) ontology builder
- ο διαχειριστής ερωτημάτων (query manager)
- το ARTEMIS (Analysis and Reconciliation Tool Environment for Multiple Information Sources), ένα εργαλείο το οποίο αναλαμβάνει την classification των τοπικών ODL_{I3} κλάσεων με στόχο τη σύνθεση των global ODL_{I3} κλάσεων
- το εργαλείο υπό το όνομα ODB-Tools Engine το οποίο βασίζεται στην Περιγραφική Λογική OLCD (Object Language with Complements allowing Descriptive cycles) και αναλαμβάνει να συμπεράνει νέες σχέσεις μεταξύ των τοπικών ODL_{I3} κλάσεων και συνεισφέρει στη δημιουργία ενός κοινού Θησαυρού.

Οι wrappers του συστήματος τοποθετούνται εννοιολογικά πάνω από τις πηγές πληροφορίας και είναι υπεύθυνοι για τη “μετάφραση” του σχήματος των πηγών σε γλώσσα ODL_{I3} . Αναλαμβάνουν επιπλέον τη μετάφραση των ερωτημάτων τα οποία υποβάλλονται σε OQL_{I3} γλώσσα σε ερωτήματα τα οποία είναι δυνατό να εκτελεστούν και να αποτιμηθούν από τους αντίστοιχους τοπικούς επεξεργαστές ερωτημάτων. Η μετάφραση των σχημάτων των πηγών σε ODL_{I3} βασίζεται στους εξής (μάλλον απλοϊκούς) κανόνες:

- Μια σχέση Βάσεων Δεδομένων (ένας πίνακας της Βάσης) αντιστοιχίζεται σε μια κλάση ODL_{I3}
- Για κάθε ιδιότητα σχέσης στην Βάση Δεδομένων, ορίζεται μια ιδιότητα στην αντίστοιχη ODL_{I3} κλάση.

Ένα γραφικό περιβάλλον που ονομάζεται SI-Designer υποστηρίζει τη διαδικασία ολοκλήρωσης. Συγκεκριμένα, το εργαλείο αυτό επαληθεύει τις σχέσεις που προέκυψαν από το ARTEMIS και δημιουργεί νέες με χρήση εργαλείων ODB. Επιπλέον, το περιβάλλον του SI-Designer έχει τη δυνατότητα σύνδεσης με την οντολογία WordNet για την άντληση επιπρόσθετων σημασιολογικών σχέσεων. Ο χρήστης αλληλεπιδρά με το σύστημα σε δύο στάδια. Στο πρώτο στάδιο της διαδικασίας, με χρήση εργαλείων ODB προσθέτει χειροκίνητα σχέσεις που δεν βρέθηκαν αυτόματα. Στο στάδιο επεξεργασίας της τελικής οντολογίας, χρησιμοποιώντας την οντολογία WordNet, ο χρήστης παρέχει επιπρόσθετες πληροφορίες για τις σχέσεις μεταξύ των αντικειμένων.

Το *Clio* [145, 146] αποτελεί ένα εργαλείο το οποίο είναι ικανό να συμπεραίνει αντιστοιχίσεις από ένα σύνολο σχέσεων Βάσεων Δεδομένων (πίνακες) και/ή δεδομένα

XML, σε ένα άλλο. Με μικρές αλλαγές, ωστόσο, μπορεί να εφαρμοστεί και σε σχήματα οντολογιών. Το Clio βασίζεται στον αλγόριθμο data chase [147] για τη δημιουργία των αντιστοιχίσεων. Οι αντιστοιχίσεις αυτές αναπαριστώνται με τη χρήση της εσωτερικής notation-mapping γλώσσας στις οποίες τις δυνατότητες περιλαμβάνεται η διαχείριση συναρτήσεων Skolem.

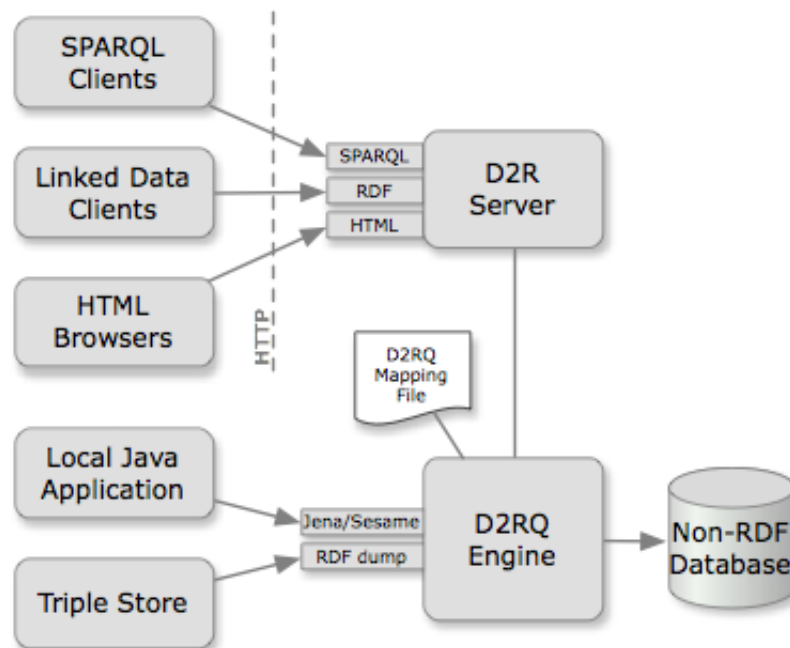
Στον πυρήνα του συστήματος βρίσκονται τα components δημιουργίας αντιστοιχίσεων και δημιουργίας ερωτημάτων. Το component δημιουργίας αντιστοιχίσεων δέχεται ως είσοδο αντιστοιχίσεις μεταξύ των σχημάτων πηγών και των τελικών σχημάτων και παράγει μια αντιστοίχιση σχημάτων η οποία αποτελείται από ένα σύνολο λογικών αντιστοιχίσεων (declarative assertions στην πράξη) οι οποίες παρέχουν μια μετάφραση των δεδομένων αντιστοιχίσεων. Το component δημιουργίας ερωτημάτων έχει το ρόλο τού να μετατρέπει σύνολο λογικών αντιστοιχίσεων σε ένα εκτελέσιμο ερώτημα. Η παραγωγή των ερωτημάτων αποτελείται από ένα generic module, ανεξάρτητο από κάθε συγκεκριμένη γλώσσα εκτέλεσης ερωτημάτων. Στην περίπτωση του Clio, οι γλώσσες αυτές περιλαμβάνουν τις SQL, SQL/XML, XQuery και XSLT. Κατά τη φάση του σχεδιασμού, ο χρήστης μπορεί ανά πάσα στιγμή να επεξεργαστεί τις αντιστοιχίσεις μεταξύ των σχημάτων μέσω της γραφικής διεπαφής που παρέχει το Clio.

Οι Stojanovic at al. παρουσιάζουν στο [148] το *KAONREVERSE*, μια προσέγγιση reverse engineering για την ημιαυτόματη αντιστοίχιση σχήματος και δεδομένων σε οντολογική περιγραφή. Αρχικά, συγκεντρώνεται η πληροφορία από το σχήμα της σχεσιακής Βάσης Δεδομένων και εφαρμόζοντας ένα σύνολο κανόνων αντιστοίχισης πραγματοποιούν ελέγχους συνέπειας και καταλήγουν με ένα σύνολο αντιστοιχίσεων. Ελέγχεται αν το σύνολο των οντοτήτων της Βάσης Δεδομένων αντιστοιχεί σε οντότητες της οντολογίας. Τελικά, πραγματοποιείται μετανάστευση (migration) των δεδομένων στην οντολογία. Η μετανάστευση αυτή συμπεριλαμβάνει τα δεδομένα της Βάσης Δεδομένων, τα οποία αντιστοιχίζονται σε άτομα της οντολογίας.

Το D2R Map παρουσιάστηκε στο [149], βασίζεται σε XML σύνταξη και προσφέρει το μέσο για την αντιστοίχιση οντολογικών εννοιών σε τμήματα της Βάσης Δεδομένων. Το αποτέλεσμα είναι να συμπεριλαμβάνονται SQL ερωτήματα στους κανόνες αντιστοίχισης. Πιο συγκεκριμένα, στους κανόνες μπορούν να συμπεριλαμβάνονται τα SQL ερωτήματα που περιγράφουν τα σύνολα, προσδιορισμοί GROUP BY αλλά και URI patterns για τη δημιουργία εννοιών στην οντολογία. Πρόκειται για ένα ισχυρό και ευέλικτο εργαλείο. Τα αποτελέσματα των αντιστοιχίσεων μπορούν να εξαχθούν σε RDF, N3 ή μοντέλα Jena. Το D2R μπορεί να διαχειριστεί αρκετά σύνθετες δομές πινάκων στη Βάση Δεδομένων, παρέχει όμως μειωμένη εκφραστικότητα.

Το D2RQ [150] βασίζεται στο D2R Map, διατηρεί στοιχεία όπως το ClassMap το

οποίο δηλώνει αντιστοιχίσεις, αλλά η σύνταξή του διαφοροποιείται ελαφρώς. Το D2RQ είναι υλοποιημένο ως γράφος της Jena (ενότητα 3.4.1) και η προσέγγισή του συνίσταται στην ολοκλήρωση σχεσιακών Βάσεων Δεδομένων υπό έναν εικονικό γράφο RDF που επιτρέπει μόνο ανάγνωση. Με τη χρήση του D2RQ μια εφαρμογή μπορεί να αποστείλει ερωτήματα στη βάση της πληροφορίας με τη χρήση RDQL. Το D2RQ γράφει εκ νέου τα υποβληθέντα RDQL ερωτήματα σε SQL μέσω κλήσεων της προγραμματιστικής διεπαφής που παρέχει η Jena. Τα αποτελέσματα που επιστρέφει η αποτίμηση των ερωτημάτων μετασχηματίζονται σε τριάδες RDF οι οποίες στη συνέχεια υφίστανται επεξεργασία από τα ανώτερα επίπεδα της Jena. Σχηματική αναπαράσταση της αρχιτεκτονικής του D2RQ παρουσιάζεται στην εικόνα 11 (Πηγή: www.w3.org/2007/03/RdfRDB/papers/d2rq-positionpaper/).



Εικόνα 11: Αρχιτεκτονική του D2RQ

Το αρνητικό σημείο του D2RQ, όπως και στην πλειονότητα των εργαλείων είναι ότι δεν υποστηρίζεται αυτόματη προσαρμογή σε αλλαγές του σχήματος της Βάσης ή της οντολογίας. Εφόσον οι αντιστοιχίσεις εισάγονται χειροκίνητα, ο επανέλεγχος μετά από κάθε αλλαγή καθίσταται απαραίτητος. Επομένως, σε περιβάλλοντα συχνών αλλαγών απειλείται η βιωσιμότητα του εγχειρήματος.

Το RDF Gateway⁶⁴ αποτελεί εμπορικό προϊόν το οποίο παρέχει παρόμοιες δυνατότητες. Αναλαμβάνει τη διασύνδεση Βάσεων Δεδομένων με το Σημασιολογικό Ιστό μέσω της διεπαφής SQL Data Service που παρέχει. Η SQL Data Service μεταφράζει από την RDFQL, μια γλώσσα βασισμένη σε RDF που έχουν δημιουργήσει οι προγραμματιστές της IntelliDimension, σε SQL και επιστρέφει τα αποτελέσματα σε RDF δεδομένα. Ωστόσο, χρησιμοποιεί μηχανή Βάσης Δεδομένων στο παρασκήνιο η οποία αποθηκεύει τα δεδομένα σε έναν (μόνο!) πίνακα Βάσης Δεδομένων. Η επιλογή αυτή δημιουργεί αρκετά ερωτηματικά σε σχέση με την απόδοση του συστήματος καθώς για την αποτίμηση των ερωτημάτων χρειάζεται να αναζητηθεί ολόκληρη η Βάση Δεδομένων.

Το eD2R (extended D2R) [151] αποτελεί επέκταση του D2R MAP, προσθέτοντας στοιχεία επιπλέον πράξεων και συνθηκών εκφράζοντας με συναρτήσεις πιο πολύπλοκους μετασχηματισμούς. Χρησιμοποιεί επίσης τεχνικές αναζήτησης λέξεων κλειδιών και επεξεργασίας φυσικής γλώσσας.

Η R_2O (Relational to Ontology) [152] είναι επίσης μια επεκτάσιμη δηλωτική γλώσσα βασισμένη σε XML η οποία επιτρέπει την περιγραφή αντιστοιχίσεων μεταξύ σχημάτων σχεσιακών Βάσεων Δεδομένων και οντολογιών σε RDF ή OWL. Παρόμοια με το D2R MAP, η R_2O επιτρέπει τη δήλωση αντιστοιχίσεων μεταξύ χαρακτηριστικών των δυο μοντέλων. Η R_2O είναι μια ανεξάρτητη από το σύστημα διαχείρισης Βάσεων Δεδομένων υψηλού επιπέδου γλώσσα η οποία συνεργάζεται με οποιαδήποτε Βάση Δεδομένων υλοποιεί το πρότυπο SQL. Μια αντιστοιχίση στην R_2O περιγράφει τον τρόπο δημιουργίας των ατόμων στην οντολογία βασισμένη στα δεδομένα που υπάρχουν αποθηκευμένα στη Βάση Δεδομένων. Η προσέγγιση των δημιουργών της συνοψίζεται στη δημιουργία ενός εγγράφου περιγραφής των αντιστοιχίσεων. Οι αντιστοιχίσεις αυτές επεξεργάζονται στη συνέχεια αυτόματα από τον “επεξεργαστή αντιστοιχίσεων” προκειμένου να παραχθούν τα άτομα της οντολογίας.

Το Ontomat Application Framework το οποίο παρουσιάστηκε στο [82] αποτελεί μια από τις πρώτες προσεγγίσεις για ενσωμάτωση σημασιολογίας σε Βάση Δεδομένων. Ουσιαστικά πρόκειται για ένα γραφικό εργαλείο βασισμένο σε αρθρωτή αρχιτεκτονική, τμήμα της οποίας αποτελεί το *Ontomat Reverse* [2] το οποίο παρέχει τα μέσα για μια ημιαντόματη υλοποίηση των αντιστοιχίσεων. Ο χρήστης παρέχει ένα σύνολο από κανόνες αντιστοιχίσης μεταξύ εννοιών της οντολογίας και της Βάσης Δεδομένων οι οποίες υποστηρίζουν διασύνδεση μέσω JDBC. Υποστηρίζει μόνο RDF γράφους και αντιστοιχίσεις μεταξύ πινάκων της Βάσης Δεδομένων που βρίσκονται στον εξυπηρετητή και κλάσεων των οντολογιών που βρίσκονται στους πελάτες. Για την αυτοματοποίηση των αντιστοιχίσεων, το *Ontomat Reverse* πραγματοποιεί αυτόματα αντιστοιχίσεις όταν

⁶⁴RDF Gateway: www.intellidimension.com

εντοπίζει λεκτική ομοιότητα μεταξύ των ονομάτων των πινάκων της Βάσης Δεδομένων και των ονομάτων των κλάσεων της οντολογίας. Παρόμοια, αντιστοιχίζονται αυτόματα τα πεδία των πινάκων της Βάσης σε ιδιότητες τύπου δεδομένων (datatype properties) της οντολογίας και τα ξένα κλειδιά σε ιδιότητες αντικειμένου (object properties). Ο χρήστης διατηρεί την ελευθερία διόρθωσης των αυτόματα παραγόμενων αντιστοιχίσεων.

Μια επίσης πρόσφατη προσέγγιση στο πρόβλημα της ολοκλήρωσης ετερογενών πηγών δεδομένων προτείνουν οι Dou και LePendu στο [153] μέσω της αρχιτεκτονικής *OntoGrate* κατά την οποία το σχήμα της σχεσιακής Βάσης Δεδομένων μετασχηματίζεται σε οντολογική αναπαράσταση ενώ ο χρήστης έχει τη δυνατότητα να ορίσει αντιστοιχίσεις σε οντολογικό επίπεδο χρησιμοποιώντας γεφυρωτικά αξιώματα (bridge-axioms). Η προσέγγιση αυτή στοχεύει στη μοντελοποίηση τόσο του σχήματος της Βάσης Δεδομένων όσο και της οντολογίας μέσω της Web-PDDL γλώσσας οντολογίας η οποία υπάγεται στις γλώσσες Κατηγορικού Λογισμού Πρώτης Τάξης. Επομένως, οι αντιστοιχίσεις, τα ερωτήματα που υποβάλλονται αλλά και τα αποτελέσματα που επιστρέφονται περιγράφονται σε όρους της Web-PDDL μέσω wrappers που διατίθενται για κάθε πιθανή αντιστοίχιση: από έννοιες της οντολογίας και από SQL σε Web-PDDL. Συγκεκριμένα για την περίπτωση μετάφρασης από SQL σε Web-PDDL έχει αναπτυχθεί η γλώσσα PDDSQL η οποία παρουσιάζεται στο [154] με σύνταξη παρόμοια της Lisp. Η ίδια ομάδα έχει επίσης αναπτύξει τον “μεταφραστή” PDDOWL ο οποίος αναλαμβάνει επίσης τη μετάφραση από OWL-QL σε Web-PDDL.

Μια ακόμη προσέγγιση δίνουν οι Laborda και Conrad [133] οι οποίοι εξετάζουν την *Relational.OWL* σαν γλώσσα δήλωσης αντιστοιχίσεων και την SPARQL σαν γλώσσα υποβολής ερωτημάτων. Η τεχνική που χρησιμοποιείται είναι η έκφραση εννοιών της Βάσης Δεδομένων σε έννοιες της οντολογίας όπως `Table` και `Column` και η αυτόματη πρόταση πιθανών αντιστοιχίσεων μεταξύ αυτών των κλάσεων. Το αποτέλεσμα είναι η αποθήκευση των περιεχομένων της Βάσης Δεδομένων σε μια οντολογία με συγκεκριμένη ορολογία. Η προσέγγιση αυτή υστερεί στο ότι για ογκώδεις Βάσεις Δεδομένων, η αποθήκευσή τους σε RDF/OWL θα τις καθιστούσε μη πρακτικές.

Το *MAPONTO* [155] είναι ένα ημι-αυτοματοποιημένο εργαλείο μέσω του οποίου ο χρήστης μπορεί να ορίσει σημασιολογικές αντιστοιχίσεις μεταξύ του σχήματος μιας Βάσης Δεδομένων και μιας οντολογίας. Οι αντιστοιχίσεις αυτές εκφράζονται ως φόρμουλες λογικής. Το εργαλείο είναι υλοποιημένο ως επέκταση του Protégé. Ο χρήστης παρέχει απλές αντιστοιχίσεις μεταξύ των στοιχείων του σχήματος της Βάσης Δεδομένων και της οντολογίας και παρέχει μια λίστα από υποψήφιους κανόνες για την αντιστοίχιση. Η κύρια ιδέα του MAPONTO είναι η αναπαράσταση της οντολογίας ως γράφου αποτελούμενου από τις έννοιες της οντολογίας ως κόμβους και τις ιδιότητες ως

ακμές⁶⁵. Στη συνέχεια, το εργαλείο αναλαμβάνει τη εύρεση του ελάχιστου συνεκτικού δένδρου (minimum spanning tree, Steiner tree) το οποίο θα συνδέει τις έννοιες οι οποίες έχουν ιδιότητες τύπου δεδομένων (datatype properties) που να αντιστοιχούν σε πίνακες της Βάσης. Αξίζει να σημειωθεί πως οι συγγραφείς δεν αναφέρουν τι χρειάζεται να γίνει όταν το σχήμα της Βάσης υποστεί αλλαγές. Προφανώς δεν υπάρχει αυτοματοποιημένος τρόπος ενσωμάτωσης των αλλαγών. Η αλλαγή στα σχήματα των πηγών ή του καθολικού σχήματος συνοδεύεται από επιπρόσθετη πολυπλοκότητα για την οποία έχουν προταθεί αρκετές προσεγγίσεις [156].

Επιπλέον, ανάμεσα στα εργαλεία που αντιμετωπίζουν το πρόβλημα της αντιστοίχισης εντάσσεται και το *Vis-A-Vis*, το οποίο αποτελεί συνεισφορά της παρούσας διατριβής και αναλύεται διεξοδικά στην ενότητα 4.3.

4.2.1 Ανάλυση των προσεγγίσεων

Στη συνέχεια, παρουσιάζεται μια συγκριτική ανάλυση των πιο σημαντικών προσεγγίσεων. Στον πίνακα 11 δίνεται η γλώσσα οντολογίας, το Σύστημα Διαχείρισης της Βάσης Δεδομένων (ΣΔΒΔ) και η σημασιολογική γλώσσα ερωτημάτων που υποστηρίζεται από το κάθε εργαλείο. Στον πίνακα 12 συνεχίζεται η παρουσίαση των εργαλείων με την παράθεση για το κάθε ένα της μεθοδολογίας και των τεχνικών που χρησιμοποιούνται, των δομικών συστατικών που αντιστοιχίζονται και των σημασιολογικών ελέγχων συνέπειας που πραγματοποιούνται. Η ανάλυση ολοκληρώνεται στον πίνακα 13 με την παρουσίαση του επιπέδου αυτοματοποίησης που παρέχεται καθώς και της μεθόδου αλληλεπίδρασης με το χρήστη.

<i>Εργαλείο</i>	<i>Γλώσσα Οντολογίας</i>	<i>ΣΔΒΔ</i>	<i>Γλώσσα ερωτημάτων</i>
D2RQ	RDF, DAML+OIL	Οποιοδήποτε με JDBC ή ODBC	RDQL
D2R MAP	RDF	Οποιοδήποτε με JDBC ή ODBC	-
Clio	N/A	Οποιοδήποτε	N/A
MOMIS	<i>ODL_{I3}</i>	Οποιοδήποτε	<i>OQL_{I3}</i>

⁶⁵MAPONTO: www.cs.toronto.edu/semanticweb/maponto

4 Συνεργασία οντολογιών με Βάσεις Δεδομένων

R ₂ O	RDF/OWL	Οποιοδήποτε υποστηρίζει SQL	Όχι
OntoGrate	Web-PDDL	Οποιοδήποτε υποστηρίζει SQL	Web-PDDL
MAPONTO	OWL	Οποιοδήποτε υποστηρίζει SQL	Όχι
Relational.OWL	RDF/OWL	DB2, MySQL, Oracle	Οποιαδήποτε γλώσσα μπορεί να υποβάλει ερωτήματα σε οντολογία OWL
Vis-A-Vis	RDF/OWL	MySQL	RDQL

Πίνακας 11: Κατηγοριοποίηση προσεγγίσεων (πίνακας 1 από 3)

<i>Εργαλείο</i>	<i>Μεθοδολογία-Τεχνικές</i>	<i>Αντιστοιχιζόμενα Στοιχεία</i>	<i>Έλεγχοι συνέπειας</i>
D2RQ	Γλώσσα για περιγραφή αντιστοιχίσεων	Πίνακες, Στήλες, Κλειδιά	Ναι, μέσω Jena API
D2R MAP	Γλώσσα βασισμένη σε XML	Πίνακες, Στήλες, Κλειδιά	Όχι
Clio	Αλγόριθμος Data chase [147]. Περιγραφή των αντιστοιχίσεων σε SQL, XQuery ή XSLT	Πίνακες, Στήλες, Κλειδιά	Όχι
MOMIS	Affinity calculus, τεχνικές clustering, χρήση WordNet	Πίνακες, Στήλες	Ναι, μέσω των ODB-Tools

4 Συνεργασία οντολογιών με Βάσεις Δεδομένων

R ₂ O	Η οντολογία συμπληρώνεται με άτομα σύμφωνα με τις αντιστοιχίσεις που δηλώνει ο χρήστης	Πίνακες, Στήλες, Πρωτεύοντα και Δευτερεύοντα κλειδιά	Όχι
OntoGrate	Αντιστοιχίσεις που περιγράφονται ως γεφυρωτικά αξιώματα (bridging axioms)	Πίνακες, Στήλες, Πρωτεύοντα και Δευτερεύοντα κλειδιά	Ναι, μέσω του OntoEngine reasoner
MAPONTO	Shortest path μεταξύ των εννοιών της οντολογίας	Πίνακες και Στήλες	Όχι
Relational.OWL	Δημιουργία μιας κλάσης για κάθε στοιχείο της Βάσης Δεδομένων που αντιστοιχίζεται	Πίνακες, Στήλες, Κλειδιά, datatypes	Όχι, η οντολογία χρησιμοποιεί την OWL Full
Vis-A-Vis	Δημιουργία ενός SQL ερωτήματος για την ανάκτηση συνόλου αποτελεσμάτων από τη Βάση Δεδομένων	Οποιοδήποτε υποσύνολο αποτελεσμάτων της Βάσης Δεδομένων	Ναι, μέσω reasoner για την οντολογία, και κλάσεων-υποκλάσεων και ξένων κλειδιών για τα δεδομένα που αντιστοιχίζονται

Πίνακας 12: Κατηγοριοποίηση προσεγγίσεων (πίνακας 2 από 3)

<i>Εργαλείο</i>	<i>Επίπεδο αυτοματοποίησης</i>	<i>Αλληλεπίδραση με το χρήστη</i>
D2RQ	Χειροκίνητο ή αυτόματο	Χωρίς γραφική διεπαφή, ο χρήστης δηλώνει τις αντιστοιχίσεις μέσω της γλώσσας που παρέχεται

4 Συνεργασία οντολογιών με Βάσεις Δεδομένων

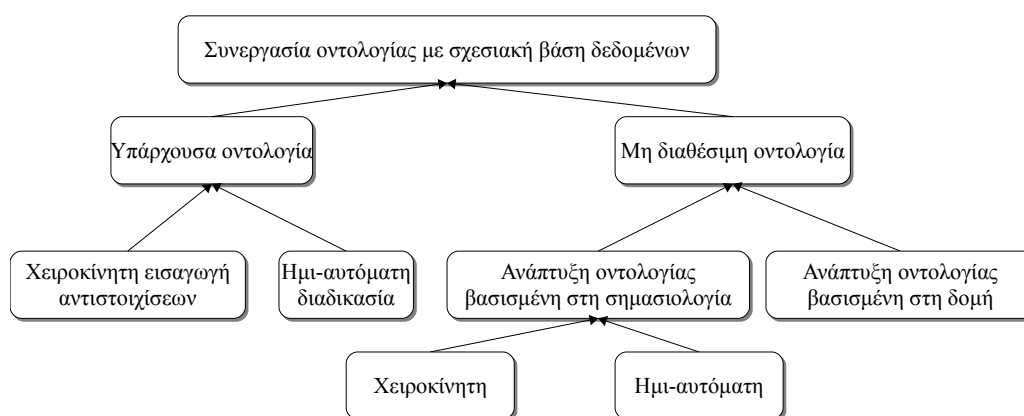
D2R MAP	Χειροκίνητο	Χωρίς γραφική διεπαφή, ο χρήστης δηλώνει τις αντιστοιχίσεις μέσω της γλώσσας που παρέχεται
Clio	Ημιαυτόματο	Γραφική διεπαφή που επιτρέπει στο χρήστη τη διαχείριση αντιστοιχίσεων μεταξύ εννοιών και μεταξύ σχημάτων πηγής και στόχου
MOMIS	Ημιαυτόματο	Γραφική διεπαφή η οποία επιτρέπει παρέμβαση του χρήστη
R ₂ O	Χειροκίνητο	Χωρίς γραφική διεπαφή, ο χρήστης δηλώνει τις αντιστοιχίσεις μέσω της δηλωτικής γλώσσας που παρέχεται
OntoGrate	Χειροκίνητο	Διεπαφή ερωτημάτων
MAPONTO	Ημιαυτόματο	Ο χρήστης παρέχει τις αντιστοιχίσεις
Relational.OWL	Αυτόματο	Καμία, η διαδικασία είναι πλήρως αυτοματοποιημένη
Vis-A-Vis	Χειροκίνητο	Γραφική διεπαφή που επιτρέπει στο χρήστη τη διαχείριση αντιστοιχίσεων μεταξύ εννοιών και μεταξύ σχημάτων πηγής και στόχου

Πίνακας 13: Κατηγοριοποίηση προσεγγίσεων (πίνακας 3 από 3)

Όπως μπορεί να γίνει αντιληπτό από τους πίνακες 11, 12 και 13, στη βιβλιογραφία υπάρχει ένα ευρύ φάσμα προσεγγίσεων στο ζήτημα της συνεργασίας μεταξύ οντολογιών και Βάσεων Δεδομένων, αυθαίρετης μεθοδολογίας και πληθώρας τεχνικών χαρακτηριστικών. Το γεγονός αυτό απλά τονίζει το ότι το πεδίο είναι εξελισσόμενο και ότι ακόμα απουσιάζει η καθιέρωση κοινών πρακτικών. Με στόχο τη σύνοψη με πιο συμπαγή τρόπο των εν λόγω προσεγγίσεων, παρουσιάζεται στη συνέχεια μια απλή ταξινόμηση στην οποία κατηγοριοποιούνται οι προσεγγίσεις αυτές. Χρειάζεται ωστόσο να τονιστεί ότι ακόμη και σε κάθε κατηγορία τα εργαλεία παρουσιάζουν ουσιαστική ετερογένεια μεταξύ τους σε αρκετούς τομείς.

Το σύνολο των προσεγγίσεων στο πεδίο της συνεργασίας μεταξύ οντολογιών και Βάσεων Δεδομένων μπορεί να διαχωριστεί σε δυο ευρείες κατηγορίες σύμφωνα με το αν η οντολογία προϋπάρχει ή ο χρήστης επιθυμεί να δημιουργήσει μια καινούρια.

Η πρώτη κατηγορία περιλαμβάνει εργαλεία όπως τα KAON-Reverse, R₂O, Ontomat Reverse, MAPONTO και Vis-A-Vis. Τα εργαλεία αυτά μπορούν να διαχωριστούν περαιτέρω ανάλογα με το αν οι αντιστοιχίσεις χρειάζεται να οριστούν ρητά από τον άνθρωπο (R₂O) ή η διαδικασία υποβοηθείται. Στη δεύτερη περίπτωση ανήκουν τα υπόλοιπα εργαλεία στα οποία η ημιαυτόματη διαδικασία υποβοηθείται από την αυτόματη ανακάλυψη αντιστοιχίσεων με τη χρήση ευριστικών, μετρήσεων λεκτικής εγγύτητας (lexical proximity) και άλλων τεχνικών.



Εικόνα 12: Απλή ταξινόμηση των προσεγγίσεων για την αντιστοίχιση Βάσεων Δεδομένων με οντολογίες

Η δεύτερη κατηγορία προσεγγίσεων δε θεωρεί δεδομένη την ύπαρξη οντολογίας αλλά αντίθετα την παράγει βασισμένη στο σχήμα της Βάσης Δεδομένων. Στην κατηγορία αυτή συμπεριλαμβάνονται εργαλεία όπως το MOMIS, το D2RQ, το OntoGrate, και η Relational.OWL.

Μερικά από τα εργαλεία αυτά χρησιμοποιούν το παραγόμενο οντολογικό μοντέλο απλά ως μια ενοποίηση μεταξύ δυο ή περισσότερων σχημάτων Βάσεων Δεδομένων ενώ για τα υπόλοιπα ο αιώτερος στόχος είναι η δημιουργία οντολογίας η οποία θα επιτρέπει την προσθήκη σημασιολογίας στην πληροφορία που υπάρχει αποθηκευμένη στη Βάση Δεδομένων. Ανεξαρτήτως σκοπιμότητας ωστόσο, μπορούν να διακριθούν δύο περιπτώσεις στην δημιουργία του οντολογικού μοντέλου: η προσανατολισμένη στη δομή και η προσανατολισμένη στη σημασιολογία.

Η προσανατολισμένη στη δομή δημιουργία οντολογικού μοντέλου βασίζεται στον εντοπισμό εννοιών και σχέσεων μεταξύ τους στο σχήμα της Βάσης Δεδομένων και την κατασκευή των αντίστοιχων κλάσεων και ιδιοτήτων στην οντολογία. Ο εντοπισμός

αυτός μπορεί να δηλώνεται από το χρήστη οπότε και η δημιουργία της οντολογίας είναι χειροκίνητη ή να υποβοηθείται από ημιαυτόματη διαδικασία όπως στην περίπτωση του MOMIS και του OntoGrate όπου το σύστημα προτείνει συγκεκριμένες αντιστοιχίσεις τις οποίες ο χρήστης μπορεί να δεχτεί, να τις απορρίψει ή να προτείνει νέες αντιστοιχίσεις.

Αντίθετα, τα εργαλεία της κατηγορίας στην οποία η δημιουργία οντολογικού μοντέλου βασίζεται στη δομή της Βάσης Δεδομένων, αντιστοιχίζουν πίνακες ή ιδιότητες της Βάσης Δεδομένων σε προϋπάρχουσες κλάσεις και ιδιότητες της οντολογίας, αγνοώντας επομένως τη σημασιολογία που μπορεί να περιέχεται στο σχήμα της Βάσης Δεδομένων. Στην κατηγορία αυτή συμπεριλαμβάνεται το Relational.OWL.

4.2.2 Μελέτη εφικτότητας (Proof of concept)

Προκειμένου να δοθεί μια καθαρή εικόνα του κατά πόσο οι θεωρητικές προσεγγίσεις είναι εφικτές στην πράξη, δημιουργήθηκε μια Βάση Δεδομένων (travel) και τμήμα των περιεχομένων της αντιστοιχήθηκε σε μια οντολογία που περιέχει πληροφορία σχετική με ταξίδια⁶⁶ χρησιμοποιώντας τα εργαλεία που είναι διαθέσιμα για χρήση. Το σχήμα της Βάσης Δεδομένων συμπεριλαμβάνει μεταξύ άλλων τους πίνακες *cities*, *hotels*, *museums* και *activities* και δεδομένα που σχετίζονται σημασιολογικά με το πεδίο της οντολογίας. Το παράδειγμα βασίζεται σε αυτό που περιγράφεται στο [139], το οποίο περιγράφει μια ακόμα προσέγγιση στο ζήτημα της συνεργασίας μεταξύ Βάσεων Δεδομένων και οντολογιών, η οποία αποτελεί συνεισφορά της διατριβής και αναλύεται διεξοδικότερα στην ενότητα 4.3.

Το D2RQ είναι ένα εργαλείο το οποίο στην πράξη δέχεται ως είσοδο μια Βάση Δεδομένων κι ένα αρχείο αντιστοιχίσεων (mapping document) και εξάγει μια οντολογία. Ένα ενδιαφέρον χαρακτηριστικό του D2RQ είναι η αυτοματοποίηση της δημιουργίας του αρχείου αντιστοιχίσεων εφόσον ο χρήστης το επιθυμεί. Παρουσιάζεται στη συνέχεια, σε N3 μορφοποίηση το τμήμα του έγγραφου αντιστοίχισης το οποίο αναφέρεται στον πίνακα *cities*, ο οποίος διαθέτει τρεις στήλες: *id* (πρωτεύον κλειδί), *name* και *isCapital* με *null* ως προκαθορισμένη τιμή.

```
map:cities a d2rq:ClassMap;
d2rq:dataStorage map:database;
d2rq:uriPattern "cities/@@cities.id@@";
d2rq:class vocab:cities; .
map:cities__label a d2rq:PropertyBridge;
d2rq:belongsToClassMap map:cities;
```

⁶⁶Η οντολογία είναι διαθέσιμη στο protege.cim3.net/file/pub/ontologies/travel/travel.owl

```
d2rq:property rdfs:label;  
d2rq:pattern "cities #@@cities.id@" ;  
map:cities_id a d2rq:PropertyBridge;  
d2rq:belongsToClassMap map:cities;  
d2rq:property vocab:cities_id;  
d2rq:column "cities.id";  
d2rq:datatype xsd:int; .
```

Το D2RQ δημιουργεί μια κλάση οντολογίας με το όνομα `cities` καθώς και τις ιδιότητες τύπου δεδομένων (`datatype properties`), `cities_id`, `cities_name` και `cities_isCapital` – οι αντιστοιχίσεις των `cities_name` και `cities_isCapital` δεν απεικονίζονται εδώ – οι οποίες αντιστοιχίζονται στις στήλες του πίνακα `cities`.

Επιπλέον, τα δεδομένα που βρίσκονται αποθηκευμένα στη Βάση χρησιμοποιούνται ώστε να παραχθούν άτομα της οντολογίας. Για αυτό το λόγο χρησιμοποιούνται URI patterns για τον καθορισμό των αντιστοιχίσεων. Ένα στιγμιότυπο της κλάσης `cities` παρουσιάζεται στη συνέχεια, σε N3 notation.

```
<http://localhost/cities/6> a vocab:cities ;  
rdfs:label "cities #6" ;  
vocab:cities_id "6"^^xsd:int ;  
vocab:cities_name "Patras" .
```

Η Relational.OWL παρουσιάζει ομοιότητα με την D2RQ στο ότι η οντολογία δε χρειάζεται να προϋπάρχει. Αντίθετα, η οντολογία συγγράφεται από το εργαλείο σύμφωνα με προκαθορισμένους κανόνες και τη χρήση συγκεκριμένων κλάσεων και ιδιοτήτων. Υπόψη λαμβάνεται μόνο η δομή της Βάσης Δεδομένων, όπως και στην περίπτωση του D2RQ όταν εκτελείται αυτόματα, χωρίς την παρέμβαση χρήστη. Παρουσιάζεται στη συνέχεια το τμήμα της RDF οντολογίας το οποίο αναφέρεται στον πίνακα `cities`.

```
<rdf:Description rdf:ID="Cities">  
<rdf:type rdf:resource="&dbs;Table"/>  
<dbs:isIdentifiedBy rdf:resource="#PK0"/>  
<dbs:hasColumn rdf:resource="#Cities.id"/>  
<dbs:hasColumn rdf:resource="#Cities.name"/>  
<dbs:hasColumn rdf:resource="#Cities.isCapital"/>  
</rdf:Description>  
<rdf:Description rdf:ID="Cities.id">  
<rdf:type rdf:resource="&dbs;Column"/>  
<rdfs:domain rdf:resource="#Cities"/>
```

4 Συνεργασία οντολογιών με Βάσεις Δεδομένων

```
<rdfs:range rdf:resource="&xsd;integer"/>
<dbpedia:length>6</dbpedia:length>
</rdf:Description>
<rdf:Description rdf:ID="Cities.name">
<rdf:type rdf:resource="&dbpedia;Column"/>
<rdfs:domain rdf:resource="#Cities"/>
<rdfs:range rdf:resource="&xsd;string"/>
</rdf:Description>
<rdf:Description rdf:ID="Cities.isCapital">
<rdf:type rdf:resource="&dbpedia;Column"/>
<rdfs:domain rdf:resource="#Cities"/>
<rdfs:range rdf:resource="&xsd;integer"/>
<dbpedia:length>3</dbpedia:length>
</rdf:Description>
<rdf:Description rdf:about="PK0">
<rdf:type rdf:resource="&dbpedia;PrimaryKey"/>
<dbpedia:hasColumn rdf:resource="#Cities.id"/>
</rdf:Description>
```

Το αποτέλεσμα της Relational.OWL εμφανίζει ομοιότητες με αυτό της D2RQ όσον αφορά τις κλάσεις και τις ιδιότητες που δημιουργούνται. Ακριβέστερα, δημιουργείται μια κλάση ανά πίνακα της Βάσης Δεδομένων και μια ιδιότητα ανά στήλη. Επιπρόσθετα, δημιουργείται μια κλάση η οποία περιγράφει το πρωτεύον κλειδί κάθε πίνακα και μια κλάση ακόμη για κάθε ξένο κλειδί. Οι κλάσεις αυτές είναι προκαθορισμένες στην Relational.OWL και με αυτόν τον τρόπο επιτυγχάνεται η δημιουργία της οντολογίας που περιγράφει το σχήμα της Βάσης Δεδομένων ανεξάρτητα από την πολυπλοκότητά του. Μια δεύτερη οντολογία, συνδεδεμένη με την οντολογία του σχήματος δημιουργείται, η οποία χρησιμεύει για την αποθήκευση των ατόμων που δημιουργούνται από τη Βάση Δεδομένων. Παρατίθεται στη συνέχεια ένα τμήμα της αντιστοίχισης του πίνακα `cities`, όπου το namespace `j.0` αναφέρεται στην οντολογία σχήματος.

```
<j.0:Cities>
  <j.0:Cities.id>6</j.0:Cities.id>
  <j.0:Cities.name>Patras</j.0:Cities.name>
</j.0:Cities>
```

Η R₂O διαφοροποιείται από τις υπόλοιπες προσεγγίσεις καθώς οι αντιστοιχίσεις που παράγει απαιτούν την προϋπαρξη της οντολογίας. Απαιτεί επιπλέον την περιγραφή του σχήματος της Βάσης Δεδομένων όπως και την περιγραφή των αντιστοιχίσεων, επιβάλλοντας επιπλέον εργασία από την πλευρά του χρήστη. Το επόμενο παράδειγμα

Περιγραφή του σχήματος της ΒΔ στο R ₂ O	Περιγραφή των αντιστοιχίσεων στο R ₂ O
<pre> dbschema-desc name Travel has-table name Cities keycol-desc name id columnType integer nonkeycol-desc name name columnType string nonkeycol-desc name isCapital columnType integer </pre>	<pre> ontology http://www.example.org/ travel.owl#conceptmap-def name City identified-by Cities.id uri-as concat arg-restriction on-param string1 has-value string "City_" arg-restriction has-column Cities.name </pre>

Πίνακας 14: Αντιστοίχιση οντολογίας και Βάσης Δεδομένων με τη χρήση του R₂O

απεικονίζει την αντιστοίχιση του πίνακα `cities` στην κλάση `City` της οντολογίας. Ορίζεται επιπλέον το URI των ατόμων που θα απαρτίσουν την οντολογία. Το σχήμα της Βάσης Δεδομένων και ο καθορισμός των αντιστοιχίσεων θα έχουν τη μορφή που εμφανίζεται στον πίνακα 14

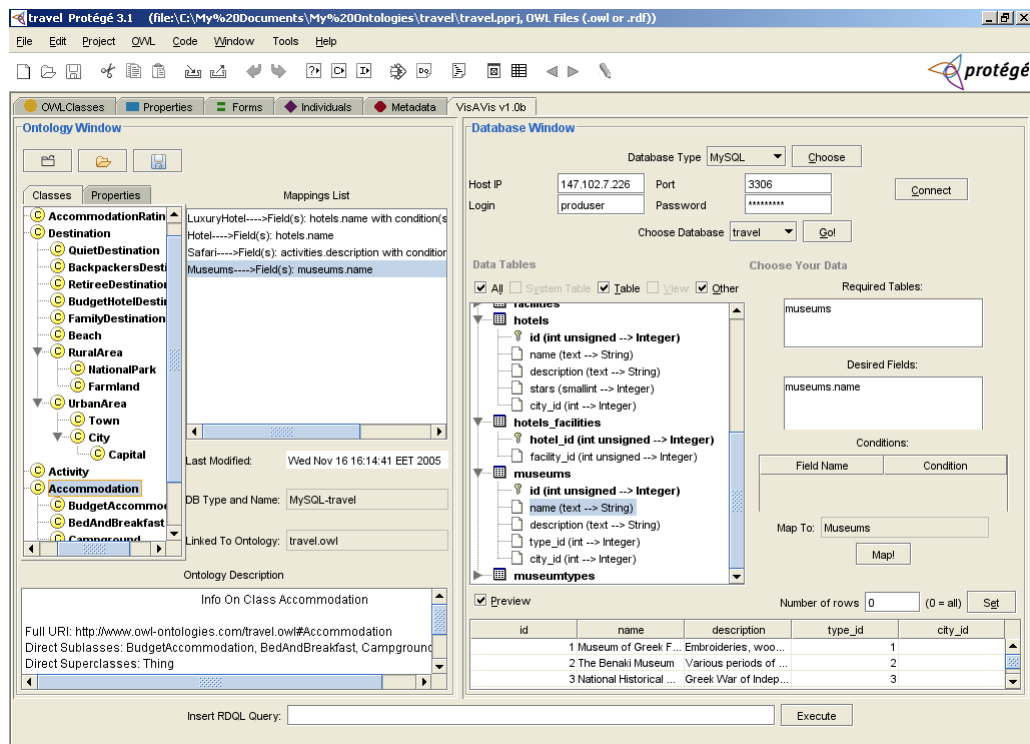
4.3 Σχεδιασμός και υλοποίηση ενός πρωτότυπου συστήματος (Vis-A-Vis)

Στην ενότητα αυτή αλλά και στη συνέχεια του κεφαλαίου αναλύεται η συνεισφορά της παρούσας διατριβής στην μελέτη του προβλήματος και προτείνεται μια πρωτότυπη προσέγγιση στο πρόβλημα της αντιστοίχισης οντολογιών με Βάσεις Δεδομένων. Το όνομα της προσέγγισης, `Vis-A-Vis`, είναι δανεισμένο από την γαλλική γλώσσα στην οποία σημαίνει κυριολεκτικά Πρόσωπο με Πρόσωπο. Το εργαλείο `Vis-A-Vis` [139] υλοποιήθηκε ως επέκταση του `Protégé` η οποία επιτρέπει την αντιστοίχιση μεταξύ οντολογιών και σχεσιακών Βάσεων Δεδομένων. Η ιδέα που υπάρχει πίσω από το `Vis-A-Vis` είναι η προσθήκη μιας ιδιότητας τύπου δεδομένων (`datatype property`) στις κλάσεις της οντολογίας για κάθε υποσύνολο της Βάσης Δεδομένων που αντιστοιχίζεται στην οντολογία. Η επιπλέον αυτή ιδιότητα, με το όνομα `queryString`, στην ουσία περιέχει το SQL ερώτημα το οποίο ανακτά το σύνολο της πληροφορίας που έχει ορίσει ο χρήστης.

Η αντιστοίχιση πραγματοποιείται επιλέγοντας από τη Βάση Δεδομένων το σύνολο των δεδομένων (`dataset`) το οποίο αντιστοιχεί σε μια κλάση. Μια καινούρια ιδιότητα

4 Συνεργασία οντολογιών με Βάσεις Δεδομένων

προστίθεται στην κλάση η οποία περιέχει το SQL ερώτημα το οποίο επιστρέφει το επιθυμητό υποσύνολο δεδομένων. Το εργαλείο επίσης εκτελεί ένα σύνολο από ελέγχους συνέπειας οι οποίοι διασφαλίζουν τη συνέπεια των αντιστοιχίσεων. Το αποτέλεσμα είναι να διασφαλίζεται η συνέπεια τόσο στις αντιστοιχίσεις όσο και στα αποτελέσματα που επιστρέφονται μετά την υποβολή ερωτημάτων. Στιγμιότυπο από τη λειτουργία του προγράμματος παρατίθεται στην εικόνα 13.



Εικόνα 13: Visavis

4.3.1 Κεντρική ιδέα του συστήματος

Υπάρχουν γενικότερα αρκετές περιπτώσεις και αρκετοί τρόποι απεικόνισης μιας κλάσης μιας οντολογίας σε στοιχεία μιας Βάσης Δεδομένων. Εκτός από την απλή περίπτωση αντιστοίχισης ενός ολόκληρου πίνακα μιας Βάσης Δεδομένων σε μια κλάση της οντολογίας, η οποία θα μπορούσε να αντιμετωπιστεί εύκολα, υπάρχει και η ανάγκη αντιστοίχισης συγκεκριμένων στηλών ή γραμμών ενός ή περισσότερων πινάκων σε μια κλάση, χρησιμοποιώντας και συνδυάζοντας με αυτόν τον τρόπο τις αλγεβρικές πράξεις της προβολής, της επιλογής και της συνένωσης.

Αυτή η ανάγκη οδήγησε στη χρήση ερωτημάτων σε γλώσσα Βάσης Δεδομένων (SQL ερωτημάτων), τα αποτελέσματα της εκτέλεσης των οποίων αποτελούν το σύνολο των πραγματικών δεδομένων που αντιστοιχίζονται σε κάθε κλάση της οντολογίας. Με αυτόν τον τρόπο, καλύπτονται οι περισσότερες περιπτώσεις αντιστοιχίσεων. Υπενθυμίζεται ότι η SQL, ως υπερσύνολο της σχεσιακής άλγεβρας είναι μια σχεσιακά πλήρης γλώσσα, γεγονός που συνεπάγεται ότι δεν υπάρχει συνδυασμός υποσυνόλων ή πλειάδων που να είναι αποθηκευμένα σε Βάση Δεδομένων και να μην είναι δυνατό να ανακτηθούν με SQL ερωτήματα.

Βασικό σημείο της προσέγγισης είναι ο τρόπος αποθήκευσης των αντιστοιχίσεων. Ένας τρόπος υλοποίησης θα χρησιμοποιούσε γλώσσες επισήμανσης (XML, XSL κ.ά.) για την αποθήκευση των αντιστοιχίσεων σε διαφορετικό αρχείο από αυτό της οντολογίας. Στην τρέχουσα περίπτωση, επιλέχθηκε η αποθήκευση των αντιστοιχίσεων σε ένα ξεχωριστό OWL αρχείο, το οποίο περιέχει ένα αντίγραφο της αρχικής οντολογίας. Συνεπώς, το αρχείο που περιέχει την οντολογία αναφοράς παραμένει αναλλοίωτο, ενώ το αντίγραφο της εμπλουτίζεται με τις αντιστοιχίσεις.

Επίσης, ένα ακόμα σημαντικό πλεονέκτημα της χρήσης SQL ερωτήματος είναι το γεγονός ότι η μετατροπή (προσθήκη, αφαίρεση) δεδομένων από τη Βάση συνεπάγεται άμεση ενημέρωση και των δεδομένων που έχουν αντιστοιχηθεί σε κάποια κλάση. Αυτό είναι λογικό, εφόσον για να βρεθούν τα δεδομένα που αντιστοιχούν, μια συγκεκριμένη χρονική στιγμή, σε μια κλάση, πρέπει την ίδια στιγμή να εκτελεστεί το συσχετισμένο με αυτήν την κλάση SQL ερώτημα, με αποτέλεσμα να επιστραφούν δεδομένα που βρίσκονται εκείνη τη δεδομένη στιγμή στη Βάση Δεδομένων. Τέλος, αξίζει να αναφερθεί ότι η ενσωμάτωση μόνο του SQL ερωτήματος για την επίτευξη της αντιστοίχισης έχει ως αποτέλεσμα τη λειτουργία της εφαρμογής με όσο το δυνατόν μικρότερο κόστος αποθήκευσης.

Όσον αφορά στην επιλογή των δεδομένων, στόχος είναι η δημιουργία ενός ερωτήματος σε γλώσσα Βάσης Δεδομένων (SQL Query) το οποίο θα επιστρέφει τα επιθυμητά δεδομένα. Η αντιστοίχιση γίνεται δεκτή, εκτός αν οι κανόνες και οι περιορισμοί της οντολογίας δεν την επιτρέπουν. Οι έλεγχοι συνέπειας που πραγματοποιούνται συνίστανται σε σύγκριση των δεδομένων που έχουν αντιστοιχηθεί σε ξένες μεταξύ τους κλάσεις, υποκλάσεις και υπερκλάσεις.

4.3.2 Έλεγχοι συνέπειας

Κατά το στάδιο της αντιστοίχισης, πραγματοποιούνται έλεγχοι συνέπειας (consistency checks) προκειμένου να καταχωρηθεί μια αντιστοίχιση ως αποδεκτή ή όχι. Οι περιπτώσεις που εξετάζονται είναι ο έλεγχος των δεδομένων που απεικονίζονται σε ξένες

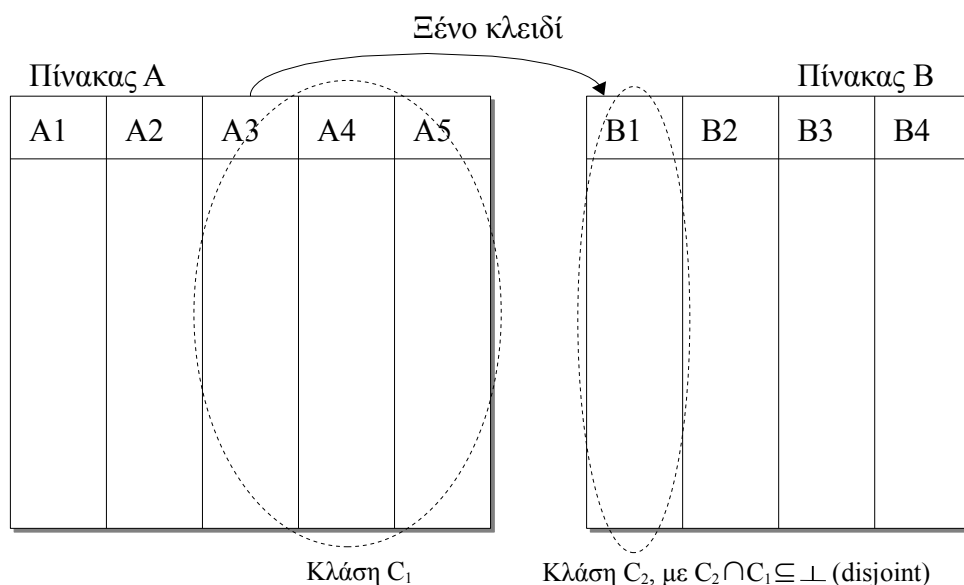
μεταξύ τους κλάσεις, καθώς και ο έλεγχος των δεδομένων που απεικονίζονται σε κλάσεις που συνδέονται με μια σχέση υπερκλάσης/υποκλάσης.

Έλεγχος των ξένων κλάσεων. Ο πρώτος έλεγχος που επιτελείται προκειμένου να εξακριβωθεί αν μια δεδομένη αντιστοίχιση είναι σημασιολογικά συνεπής είναι ο έλεγχος των δεδομένων που έχουν αντιστοιχηθεί σε κλάσεις ξένες προς την τρέχουσα απεικονιζόμενη κλάση. Δύο κλάσεις ονομάζονται ξένες ή μη επικαλυπτόμενες (dis-joint), όταν δεν έχουν κοινά στοιχεία. Συνεπώς, είναι λογικό τα δεδομένα στα οποία αντιστοιχούνται δύο ξένες μεταξύ τους κλάσεις να είναι διαφορετικά και να μην επικαλύπτονται.

Αρχικά, ανακτώνται όλες οι ξένες κλάσεις της υπό έλεγχο κλάσης έτσι ώστε να συγκριθούν τα δεδομένα που έχουν αντιστοιχηθεί σε αυτές με αυτά που πρόκειται να αντιστοιχηθούν στην υπό έλεγχο κλάση. Τα δεδομένα που έχουν απεικονιστεί στις ξένες κλάσεις της υπό έλεγχο κλάσης συγκρίνονται ένα προς ένα με τα δεδομένα που πρόκειται να απεικονιστούν στην υπό έλεγχο κλάση και αν βρεθεί έστω και ένα κοινό υποσύνολο των δεδομένων μεταξύ των δύο ομάδων δεδομένων, η αντιστοίχιση απορρίπτεται. Συγκεκριμένα, για να θεωρηθούν δύο δεδομένα ταυτόσημα, πρέπει να έχουν την ίδια τιμή, να ανήκουν στην ίδια στήλη και στον ίδιο πίνακα. Συνεπώς, η εύρεση τμημάτων δεδομένων με τις ίδιες τιμές δε συνεπάγεται ταυτότητα αυτών των δεδομένων, αφού πρέπει να ελεγχθούν παράλληλα και τα ονόματα των στηλών και των πινάκων στα οποία αυτά ανήκουν.

Σημειώνεται ότι υπάρχει περίπτωση δύο τμήματα δεδομένων να θεωρηθούν ταυτόσημα, ακόμα και αν έχουν τις ίδιες τιμές αλλά ανήκουν σε διαφορετικές στήλες. Κάτι τέτοιο συναντάται όταν μια στήλη είναι ξένο κλειδί σε μια άλλη στήλη, οπότε τα δεδομένα που περιέχονται σε αυτήν είναι ουσιαστικά τα ίδια. Επομένως, εκτός από τον έλεγχο τιμής των δεδομένων, ελέγχεται και αν η στήλη ή οι στήλες στις οποίες ανήκουν τα δεδομένα προς αντιστοίχιση με την υπό έλεγχο κλάση, αποτελούν ξένο κλειδί σε μια στήλη άλλου πίνακα η οποία περιέχει δεδομένα που έχουν ήδη απεικονιστεί σε μια ξένη προς αυτή κλάση. Επίσης, χρειάζεται να εξεταστεί και η αντίστροφη περίπτωση. Συγκεκριμένα, ελέγχεται αν κάποια στήλη που περιέχει δεδομένα που έχουν απεικονιστεί σε μια ξένη κλάση προς την υπό έλεγχο κλάση αποτελεί ξένο κλειδί σε μια στήλη στην οποία περιέχονται δεδομένα που πρόκειται να αντιστοιχηθούν στην υπό έλεγχο κλάση. Η περίπτωση αυτή απεικονίζεται στην εικόνα 14.

Αξίζει να αναφερθεί ότι ο παραπάνω έλεγχος δεν είναι αρκετός για να εξασφαλιστεί η σημασιολογική συνέπεια της αντιστοίχισης, σε ό,τι αφορά στις ξένες κλάσεις. Είναι απαραίτητο να επεκταθεί ο έλεγχος και στα ανώτερα επίπεδα ιεραρχίας κλάσεων και να εξεταστούν οι ξένες κλάσεις της υπερκλάσης της υπό έλεγχο κλάσης.

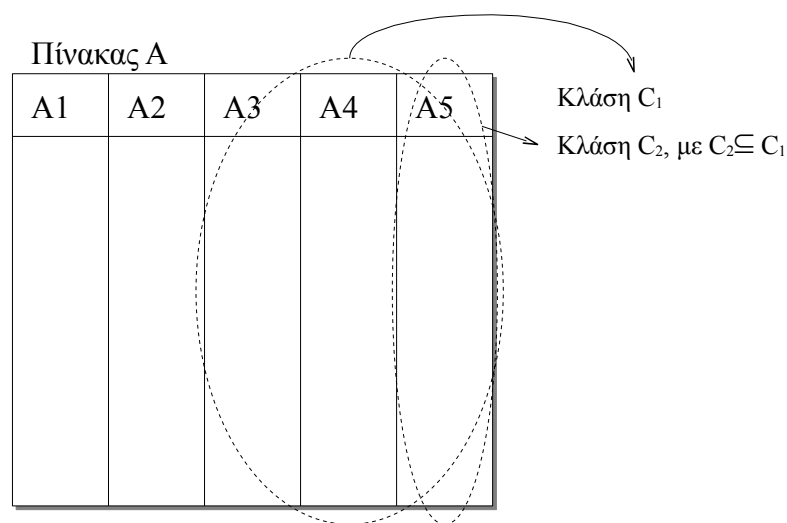


Εικόνα 14: Μη αποδεκτή αντιστοίχιση σε περίπτωση ξένων μεταξύ τους κλάσεων

Στην περίπτωση που μια ξένη κλάση της υπό έλεγχο κλάσης δεν έχει απεικονιστεί σε κάποια δεδομένα, ο έλεγχος επεκτείνεται στις υποκλάσεις της, οι οποίες λόγω κληρονομικότητας είναι και αυτές ξένες με την υπό έλεγχο κλάση. Ο έλεγχος γίνεται για κάθε μία από τις υποκλάσεις στις οποίες έχουν ήδη αντιστοιχηθεί κάποια δεδομένα, ενώ στην περίπτωση που κάποια υποκλάση δεν έχει απεικονιστεί σε κάποια δεδομένα, ο έλεγχος επεκτείνεται και σε χαμηλότερο επίπεδο και ούτω καθεξής.

Συμπερασματικά, μια αντιστοίχιση ενός συνόλου δεδομένων σε μία κλάση γίνεται δεκτή μόνο αν δε βρεθεί μία ξένη σε αυτήν κλάση, η οποία να έχει αντιστοιχηθεί σε τμήμα δεδομένων που πρόκειται να αντιστοιχηθεί στην υπό έλεγχο κλάση.

Έλεγχος υπερκλάσεων. Ο επόμενος έλεγχος που επιτελείται για την εξασφάλιση της σημασιολογικής ορθότητας μιας αντιστοίχισης δεδομένων σε μια κλάση εξετάζει τις υπερκλάσεις αυτής της κλάσης και τα δεδομένα στα οποία αυτές έχουν τυχόν αντιστοιχηθεί. Σε αυτό το σημείο, υπενθυμίζεται ότι αν μια κλάση C_1 είναι υπερκλάση μιας κλάσης C_2 , αυτό σημαίνει ότι όλα τα άτομα που ανήκουν στην κλάση C_2 θα ανήκουν υποχρεωτικά και στην κλάση C_1 . Με βάση τον παραπάνω ορισμό, είναι λογικό ότι τα δεδομένα που πρόκειται να αντιστοιχηθούν σε μια κλάση πρέπει να αποτελούν υποσύνολο των δεδομένων που έχουν αντιστοιχηθεί σε όλες τις υπερκλάσεις αυτής της κλάσης. Αν αυτό δεν ισχύει, η αντιστοίχιση απορρίπτεται. Η εικόνα 15 απεικονίζει τον έλεγχο συνέπειας στην περίπτωση αυτή.

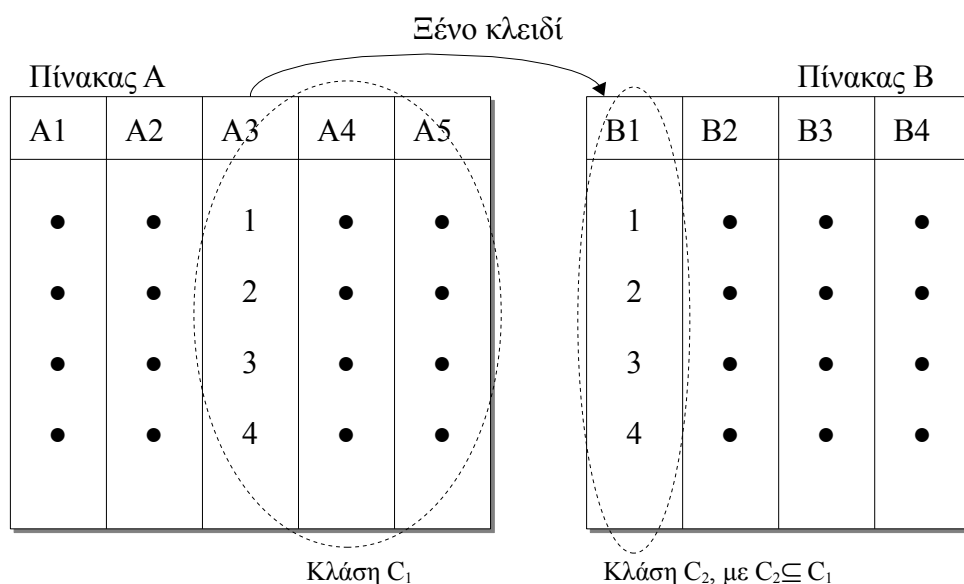


Εικόνα 15: Αποδεκτή αντιστοίχιση σε περίπτωση σχέσης υπερκλάσης-υποκλάσης

Αρχικά, ανακτώνται όλες οι άμεσες υπερκλάσεις της υπό έλεγχο κλάσης προκειμένου να συγκριθούν τα δεδομένα που έχουν αντιστοιχηθεί σε αυτές με τα δεδομένα που πρόκειται να αντιστοιχηθούν στην υπό έλεγχο κλάση. Σημειώνεται ότι άμεσες υπερκλάσεις μιας κλάσης είναι αυτές που ιεραρχικά βρίσκονται στο αμέσως ανώτερο επίπεδο από το επίπεδο τη κλάσης αυτής. Τα δεδομένα που πρόκειται να απεικονιστούν στην υπό έλεγχο κλάση εξετάζονται για να εξακριβωθεί αν υπάγονται στο σύνολο των δεδομένων που έχουν αντιστοιχηθεί σε μια άμεση υπερκλάση της υπό έλεγχο κλάσης. Αυτή η διαδικασία εκτελείται για όλες τις άμεσες υπερκλάσεις της υπό έλεγχο κλάσης. Αν βρεθεί τμήμα δεδομένων που πρόκειται να αντιστοιχηθεί στην υπό έλεγχο κλάση και που δε βρίσκεται μεταξύ των δεδομένων που έχουν αντιστοιχηθεί σε κάποια άμεση υπερκλάση της υπό έλεγχο κλάσης, η αντιστοίχιση απορρίπτεται.

Όπως και στον έλεγχο ξένων κλάσεων, για να θεωρηθούν δύο δεδομένα ταυτόσημα, χρειάζεται να έχουν την ίδια τιμή, να ανήκουν στην ίδια στήλη και στον ίδιο πίνακα. Εξαιρέση αποτελεί η περίπτωση στην οποία τα δύο δεδομένα έχουν την ίδια τιμή και ανήκουν σε στήλες, η μία εκ των οποίων είναι ξένο κλειδί στην άλλη, οπότε και πάλι τα δεδομένα αυτά θεωρούνται ταυτόσημα. Αυτή η περίπτωση φαίνεται στην εικόνα 16, όπου η κλάση C_2 έχει αντιστοιχηθεί στη στήλη B1, ενώ η C_1 , η οποία είναι υπερκλάση της C_2 , έχει αντιστοιχηθεί στις στήλες A3, A4, A5. Εκ πρώτης όψεως, η αντιστοίχιση δε θα έπρεπε να επιτραπεί, όμως δεδομένου του ότι η στήλη A3 είναι ξένο κλειδί στη B1, περιέχει τις ίδιες τιμές με αυτή και οι τιμές αυτές αναφέρονται σημασιολογικά στην

ίδια έννοια, θεωρείται ότι τα δεδομένα που έχουν αντιστοιχηθεί στην C_2 (στήλη B1) περιέχονται εξ ολοκλήρου στα δεδομένα που έχουν αντιστοιχηθεί στην C_1 (στήλες A3, A4, A5). Συνεπώς, η αντιστοίχιση είναι σημασιολογικά αποδεκτή.



Εικόνα 16: Αποδεκτή αντιστοίχιση σε περίπτωση ξένου κλειδιού

Στην περίπτωση που μια άμεση υπερκλάση της υπό έλεγχο κλάσης δεν έχει απεικονιστεί σε κάποια δεδομένα, τότε ο έλεγχος πρέπει να μεταφερθεί στο αμέσως ανώτερο επίπεδο, ελέγχοντας όλες τις άμεσες υπερκλάσεις αυτής της μη αντιστοιχημένης κλάσης και ούτω καθεξής.

Συμπερασματικά, για να γίνει αποδεκτή μια αντιστοίχιση δεδομένων σε μια κλάση, πρέπει τα δεδομένα αυτά να είναι υποσύνολο των δεδομένων που έχουν αντιστοιχηθεί στις υπερκλάσεις αυτής της κλάσης.

Έλεγχος υποκλάσεων. Ο τελευταίος έλεγχος σημασιολογικής ορθότητας μιας αντιστοίχισης δεδομένων με μια κλάση μιας οντολογίας αναφέρεται στη σύγκριση των δεδομένων που πρόκειται να αντιστοιχηθούν στην υπό έλεγχο κλάση με τα δεδομένα που έχουν αντιστοιχηθεί στις υποκλάσεις της. Όπως και προηγουμένως, τονίζεται ότι μια κλάση C_2 είναι υποκλάση μιας κλάσης C_1 , όταν όλα τα άτομα που ανήκουν στην κλάση C_2 ανήκουν και στην κλάση C_1 . Συνεπώς, για να γίνει αποδεκτή μια αντιστοίχιση, θα πρέπει τα δεδομένα που πρόκειται να αντιστοιχηθούν σε μια κλάση να αποτελούν υπερσύνολο των δεδομένων ή, με άλλα λόγια, να περιέχουν όλα τα δεδομένα που έχουν

ήδη απεικονιστεί σε όλες τις υποκλάσεις της υπό έλεγχο κλάσης. Ουσιαστικά, ο έλεγχος αυτός εφαρμόζει τους ίδιους κανόνες με τον έλεγχο υπερκλάσεων και για αυτόν το λόγο, το σχηματικό παράδειγμα που δόθηκε στην εικόνα 15 ισχύει και για το συγκεκριμένο είδος ελέγχου.

Ο έλεγχος των υποκλάσεων της υπό έλεγχο κλάσης είναι αντίστοιχος του ελέγχου υπερκλάσεων με τη μόνη διαφορά ότι, αυτή τη φορά, θα πρέπει τα δεδομένα που έχουν αντιστοιχηθεί στις υποκλάσεις μιας κλάσης να αποτελούν υποσύνολο των δεδομένων που πρόκειται να αντιστοιχηθούν σε αυτήν την κλάση. Συνεπώς, όπως και στον έλεγχο υπερκλάσεων, ανακτώνται όλες οι άμεσες υποκλάσεις της υπό έλεγχο κλάσης, δηλαδή όλες οι υποκλάσεις που βρίσκονται ιεραρχικά στο αμέσως κατώτερο επίπεδο προκειμένου να συγκριθούν τα δεδομένα που έχουν αντιστοιχηθεί σε αυτές, με τα δεδομένα που πρόκειται να αντιστοιχηθούν στην υπό έλεγχο κλάση. Τα δεδομένα που έχουν αντιστοιχηθεί σε καθεμία από τις άμεσες υποκλάσεις της υπό έλεγχο κλάσης συγκρίνονται ένα προς ένα με τα δεδομένα που πρόκειται να αντιστοιχηθούν στην υπό έλεγχο κλάση, προκειμένου να εξασφαλιστεί ότι τα πρώτα αποτελούν υποσύνολο των δεύτερων. Αν βρεθεί τμήμα δεδομένων που έχει ήδη απεικονιστεί σε κάποια υποκλάση, αλλά όχι και στην υπό έλεγχο κλάση, τότε η εν λόγω απεικόνιση είναι σημασιολογικά απορριπτέα.

Δύο σύνολα δεδομένων, όπως και στους δύο παραπάνω ελέγχους, θεωρούνται ταυτόσημα, όταν έχουν τις ίδιες τιμές και ανήκουν στην ίδια στήλη και στον ίδιο πίνακα ή όταν έχουν τις ίδιες τιμές και ανήκουν σε στήλες, η μία εκ των οποίων είναι ξένο κλειδί στην άλλη.

Στην περίπτωση που μια άμεση υποκλάση της υπό έλεγχο κλάσης δεν έχει απεικονιστεί σε κάποια δεδομένα, τότε ο έλεγχος μεταφέρεται ένα επίπεδο χαμηλότερα, εξετάζοντας τις υποκλάσεις της μη αντιστοιχηθείσας κλάσης και ούτω καθεξής.

Συνοπτικά, για να γίνει αποδεκτή μια αντιστοίχιση, θα πρέπει τα δεδομένα που πρόκειται να αντιστοιχηθούν στην υπό έλεγχο κλάση να περιέχουν όλα τα δεδομένα που έχουν ήδη αντιστοιχηθεί σε όλες τις υποκλάσεις της.

4.3.3 Εκτέλεση σημασιολογικού ερωτήματος

Ο χρήστης μπορεί να εισάγει ένα σημασιολογικό ερώτημα (σε RDQL) προκειμένου να ανακτήσει τα ονόματα των κλάσεων που δίνουν απάντηση στο ερώτημα, αλλά και τα δεδομένα που έχουν αντιστοιχηθεί σε αυτές. Αρχικά, εκτελείται το ερώτημα και επιστρέφονται τα αποτελέσματά του. Στη συνέχεια, γίνεται έλεγχος αν κάποια από τις κλάσεις που έχουν αντιστοιχηθεί σε δεδομένα της Βάσης περιλαμβάνεται μεταξύ των αποτελεσμάτων του σημασιολογικού ερωτήματος. Αν ναι, τα δεδομένα ανακτώνται

από τη Βάση για τις κλάσεις αυτές μέσω της τιμής της ιδιότητας `queryString`. Τα αποτελέσματα κάθε SQL ερωτήματος οργανώνονται σε πίνακες και όλοι αυτοί οι πίνακες μαζί με το όνομα της αντίστοιχης κλάσης της οντολογίας παρουσιάζονται στο χρήστη.

4.4 Σχολιασμός

Ο Σημασιολογικός Ιστός μπορεί να θεωρηθεί ως ένα ανοιχτό και καταναμημένο σύστημα. Σε αυτά τα συστήματα δεν είναι δυνατό να αποφευχθεί η ετερογένεια. Σύμφωνα με το [157], η ετερογένεια μπορεί να υφίσταται σε τέσσερα επίπεδα: συντακτικό, τερμινολογικό, εννοιολογικό και semiotic/pragmatic. Αρκετές έρευνες έχουν διεξαχθεί στο πεδίο της συνεργασίας μεταξύ οντολογιών [131, 158] έχοντας ως όραμα τη δημιουργία ενός Ιστού ο οποίος θα παρέχει πληθώρα ανθρωποκεντρικών υπηρεσιών σε αντίθεση με τις σημερινές προσεγγίσεις. Χρειάζεται να ληφθεί υπόψη ότι η διαλειτουργικότητα μεταξύ των προϊόντων των διαφόρων μεθοδολογιών αντιστοίχισης είναι ένα σημαντικό καθήκον. Σε πρώτο στάδιο, μέθοδοι και εργαλεία χρειάζεται να παραχθούν ώστε να επιτευχθεί διαλειτουργικότητα σε ανώτερο σημασιολογικό επίπεδο. Επιπλέον, τα διασυνδεδεμένα δεδομένα χρειάζεται να συνδεθούν μεταξύ τους ώστε να επιτευχθεί ο απώτερος στόχος του Σημασιολογικού Ιστού.

Η παρούσα έρευνα φανερώνει ότι το πεδίο της αντιστοίχισης Βάσεων Δεδομένων με οντολογίες είναι αρκετά ενεργό. Σημαντικές και διαδεδομένες τάσεις στο χώρο όπως το blogging, τα RSS, η DBPedia, το RDFa, ή τα FOAF και SIOC δίκτυα (ενότητα 2.5) παρέχουν στο Διαδίκτυο πολύ υλικό το οποίο ανανεώνεται συνεχώς. Επιπλέον, η διάδοση του Σημασιολογικού Ιστού κατέστησε δυνατό το να παράγει κανείς οντολογίες σε πολύ μεγαλύτερο βαθμό απ' ό,τι οι μηχανικοί του Διαδικτύου θεωρούσαν πιθανό [159].

Η κοινότητα του Σημασιολογικού Ιστού έχει παράξει μέχρι στιγμής πολυάριθμες οντολογίες⁶⁷ οι οποίες καθιστούν τη διαχείρισή τους τόσο επίπονη διαδικασία όσο και η ανάπτυξή τους. Παρ' όλα αυτά, η σύγχρονη κατάσταση φανερώνει ότι τα εργαλεία αντιστοίχισης παραμένουν σε πρώιμο στάδιο.

Επίσης, δε βρέθηκε προγραμματιστική διεπαφή η οποία να επιτρέπει την σύνταξη αυθαίρετων κανόνων από τους προγραμματιστές, με μόνη εξαίρεση το D2RQ [160]. Επιπλέον, δεν υπάρχει μεσισμικό το οποίο να επιτρέπει τη διαφανή διαλειτουργικότητα μεταξύ των δυο επιπέδων.

Σημαντικό είναι επίσης το ζήτημα της ασφάλειας. Οι σχεσιακές Βάσεις Δεδομένων είναι ευρύτατα διαδεδομένες σε κρίσιμες εφαρμογές πραγματικού κόσμου όπου η

⁶⁷Μέχρι σήμερα το Swoogle [79] έχει καταμετρήσει περισσότερες από 10,000 οντολογίες. Για πιο ακριβή και ενημερωμένα στατιστικά ο αναγνώστης παραπέμπεται στο swoogle.umbc.edu/index.php?option=com_swoogle_stats

ασφάλεια είναι παράγοντας υψηλής βαρύτητας. Οι καθιερωμένες πρακτικές έχουν επηρεάσει την τεχνολογία των Βάσεων Δεδομένων ούτως ώστε να παρέχεται υψηλό επίπεδο ασφάλειας τόσο στα δικαιώματα των χρηστών όσο και στην αποθήκευση των δεδομένων. Θα ήταν επιθυμητή μια διαπίστωση της ύπαρξης εφαρμογών που να τονίζουν το θέμα της ασφάλειας στα δύο αυτά επίπεδα. Πρώτον, θα ήταν χρήσιμο να υπάρχει μηχανισμός προστασίας του περιεχομένου της οντολογίας κατά την αποθήκευσή του και δεύτερον, θα ήταν επιθυμητή η πρόσβαση στα δεδομένα της οντολογίας μέσω μηχανισμού ταυτοποίησης, για παράδειγμα με συνθηματικό και κωδικό ή με προσωπικό κλειδί. Μεταξύ των υπό διερεύνηση εργαλείων, μόνο το D2RQ εμφανίζει στοιχεία που αντιμετωπίζουν στοιχεία ασφάλειας, επιτρέποντας τις υπό συνθήκη αντιστοιχίσεις (conditional mappings). Με τη χρήση των υπό συνθήκη αντιστοιχίσεων, μόνο τα δεδομένα που πληρούν συγκεκριμένες συνθήκες είναι προσβάσιμα από την οντολογία, λαμβάνοντας υπόψη κατά τον τρόπο αυτό την εμπιστευτικότητα των δεδομένων.

4.5 Πληρότητα, εγκυρότητα και απόδοση

Ανάμεσα στα σημαντικά ζητήματα που χρειάζεται να αναλυθούν είναι η πρακτική αξία των αποτελεσμάτων. Από θεωρητικής απόψεως, το αποτέλεσμα εξετάζεται κατά πόσο είναι sound and complete. Ένα σύστημα είναι sound όταν κάθε απάντηση που επιστρέφεται είναι έγκυρη και complete όταν μπορεί να δώσει απαντήσεις για όλο το πεδίο γνώσεώς του. Παρόλα αυτά, από πρακτικής άποψης, είναι γνωστό ότι ακόμη και διαδικασίες συλλογιστικής σε OWL Lite οντολογία μπορεί να εμφανίζουν υψηλή worst-case πολυπλοκότητα [161]. Επομένως, χρειάζεται κατά το σχεδιασμό και την υλοποίηση τέτοιων συστημάτων να λαμβάνονται υπόψη οι περιορισμοί που τίθενται. Χρειάζεται να ισοσταθμιστεί η απόδοση με την πληρότητα και την εγκυρότητα.

Γενικά, τα αποτελέσματα που επιστρέφονται στο χρήστη, αναμένονται συνήθως σε λιγότερο από δευτερόλεπτο. Οι χρήστες προτιμούν ατελή αποτελέσματα σε εύλογο χρόνο – λιγότερο από ένα δευτερόλεπτο – παρά μια πλήρη λίστα αποτελεσμάτων για τα οποία θα χρειαστεί να περιμένουν. Οι Dou et. al [154] ισχυρίζονται ότι μια αξιόπιστη (sound) αλλά ατελής (incomplete) προσέγγιση μοιάζει να είναι ικανοποιητική.

Όσον αφορά το ζήτημα της επεκτασιμότητας, η πιο διαδεδομένη λύση για λήψη πειραματικών αποτελεσμάτων είναι το Lehigh University Benchmark (LUBM) [162] το οποίο παρά την κριτική που έχει δεχθεί [163] παραμένει η πιο πρακτική λύση. Στο [161], οι συγγραφείς ισχυρίζονται ότι προκειμένου ένα σύστημα το οποίο βασίζεται σε υπηρεσίες συλλογιστικής να αντεπεξέλθει σε ανάγκες πραγματικού κόσμου, χρειάζεται να παρέχει επαρκώς εκφραστική γλώσσα ερωτημάτων όπως επίσης και ευέλικτη και αποτελεσματική

διεπαφή επικοινωνίας.

4.6 Συμπεράσματα, παρατηρήσεις και προτάσεις

Στην ενότητα αυτή ορίζονται οι απαιτήσεις που χρειάζεται να ληφθούν υπόψη κατά το σχεδιασμό των εργαλείων της επόμενης γενιάς, σύμφωνα με τα πορίσματα της παρούσας έρευνας στο χώρο και κατ' αναλογία με το [10] όπου οι Uren et al. θέτουν τις απαιτήσεις για τη σημασιολογική επισημείωση περιεχομένου (content annotation).

Προκειμένου μια αντιστοίχιση οντολογίας και Βάσης Δεδομένων να ικανοποιήσει ανάγκες πραγματικού κόσμου, χρειάζεται κανείς να λάβει υπόψη αρκετούς παράγοντες. Ο πρώτος παράγοντας είναι ότι για μια εταιρεία που εξετάζει την πιθανή υιοθέτηση μιας αντίστοιχης τεχνολογίας χρειάζεται να αντισταθμιστούν τα πλεονεκτήματα και τα μειονεκτήματα που κερδίζονται από μια τέτοια προσέγγιση. Χρειάζεται καταρχήν να γίνει μια τυπική ανάλυση, πολύ σημαντική στη λήψη αποφάσεων σχετικά με το κέρδος που μπορεί να αποκομισθεί. Ανάμεσα στα προφανή πλεονεκτήματα που αποκομίζονται είναι ότι η υποβολή ερωτημάτων στο σύστημα θα είναι εμπλουτισμένη με τεχνολογίες Σημασιολογικού Ιστού και επομένως θα εμπεριέχει ευφυΐα η οποία θα επιτρέπει ερωτήματα ανώτερου σημασιολογικού επιπέδου.

Οστόσο, ένα σύστημα που ενσωματώνει σημασιολογία, δε μπορεί να βασίζεται σε μια και μόνο οντολογία. Χρειάζεται να ληφθεί υπόψη ότι ένα μεγάλης έκτασης έργο χρειάζεται οντολογίες με αρκετά λεπτομερείς περιγραφές των διάφορων θεματικών περιοχών που μπορεί να καλύπτει. Μια μεγάλη σε όγκο οντολογία συνήθως είναι άβολη και δύσχρηστη τόσο κατά το σχεδιασμό όσο και κατά τη συντήρηση, όπως για παράδειγμα η οντολογία Cyc (ενότητα 3.5). Λεπτομερείς περιγραφές εννοιολογικών μοντέλων είναι πιθανό να εκτείνονται σε περισσότερα του ενός πεδία γνώσεως και επομένως απαιτούν συμβολή ειδικών σε αρκετούς χώρους προκειμένου να περιγράψουν επαρκώς και με επιτυχία τον κάθε γνωσιακό χώρο.

Η συνεργασία μεταξύ των διαφορετικών τεχνολογιών, προκειμένου να ανταγωνιστούν καθιερωμένες και ευρέως διαδεδομένες πρακτικές όπως το Model Driven Engineering (MDE) χρειάζεται να αντιμετωπίζει ζητήματα όπως τα παρακάτω. Η ανάγκη για την παροχή απαιτήσεων πηγάζει από την παρατήρηση ότι οι υπάρχουσες προσεγγίσεις βασίζονται σε προσεγγίσεις χωρίς κοινό παρονομαστή με αποτέλεσμα ο χώρος να μην έχει ακόμα διαμορφωθεί επαρκώς. Απουσιάζει επίσης η ενοποίηση των προσεγγίσεων. Επομένως, μια περιεκτική αλλά όχι πλήρης λίστα των πτυχών του προβλήματος αφορά τα εξής:

1. *Δυναμική προσαρμογή* στις αλλαγές. Κάθε δήλωση αντιστοίχισης θα πρέπει να

μεταβάλλεται δυναμικά καθώς είναι πιθανό η δήλωση αυτή να υποβάλλεται σε συχνές αλλαγές. Η ιδέα βασίζεται στο ότι ένα περιβάλλον εργασίας θα πρέπει να αντιλαμβάνεται τυχόν αλλαγές τόσο στην ιεραρχία της οντολογίας όσο και στο σχήμα της Βάσης και να προσαρμόζεται έχοντας την ελάχιστη ανθρώπινη παρέμβαση. Εργαλεία όπως το D2R MAP, το D2RQ και η Relational.OWL παρέχουν τη λειτουργικότητα αυτόματης παραγωγής αντιστοιχίσεων, χρειάζεται ωστόσο να υπογραμμιστεί η απουσία ενός (run-time) περιβάλλοντος το οποίο θα αναγνωρίζει αυτόματα τις αλλαγές στις πηγές δεδομένων και θα προσαρμόζεται αντανακλώντας τις αλλαγές στο προϊόν της αντιστοίχισης.

2. Υποστήριξη *συνεργασίας* για την παραγωγή των αντιστοιχίσεων. Η πολυπλοκότητα όμως της φύσης του προβλήματος σε συνδυασμό με την κατανεμημένη μορφή των συστημάτων του Διαδικτύου υπογραμμίζει την ανάγκη για συνεργασία. Ειδικότερα όταν ο κάθε συνεργάτης διαθέτει γνώση σε διαφορετικούς τομείς, καθίσταται ακόμη πιο επιτακτική η ανάγκη για την παροχή κοινής πλατφόρμας συνεργασίας. Αναφέρθηκε ήδη ότι η συγγραφή οντολογίας δεν αποτελεί γραμμική διαδικασία και ότι συνήθως χρειάζεται τη συμβολή ατόμων με εμπειρία σε διαφορετικούς χώρους (ενότητα 3.2). Η ανάγκη για την υποστήριξη συνεργασίας τονίζεται όταν πρόκειται για τη δήλωση αντιστοιχίσεων καθώς οι συγγραφείς/δημιουργοί χρειάζεται να είναι γνώστες των μοντέλων και των δυο πεδίων περιγραφής της γνώσεως. Επομένως, η απουσία υποστήριξης συνεργασίας δυσχεραίνει τη βιωσιμότητα του εγχειρήματος.

Τα εργαλεία που εξετάστηκαν, σχεδόν στο σύνολό τους δε διαθέτουν υποστήριξη συνεργασίας για την ανάπτυξη των αντιστοιχίσεων. Μόνο τα εργαλεία D2RQ, D2R MAP και R₂O παρέχουν περιγραφικά λεξικά μέσω των οποίων δηλώνονται οι αντιστοιχίσεις επιτρέποντας με τον τρόπο αυτό τη συνεργασία. Επίσης, το εμπορικό εργαλείο RDF Gateway επιτρέπει η διαδικασία αντιστοίχισης να δίνεται μέσα από κώδικα υποστηρίζοντας τη συνεργασία μεταξύ ατόμων σύμφωνα με τις πρακτικές της ανάπτυξης κώδικα.

3. Συμμόρφωση με διεθνώς αναγνωρισμένα *πρότυπα*. Το αποτέλεσμα της σημασιολογικής επισημείωσης επιβάλλεται να υπακούει σε πρότυπα όπως το RDF, το KIF, η OWL ή η SQL. Επιπλέον, χρειάζεται να υποστηρίζονται μορφοποιήσεις όπως η RDF/XML, N3/Notation3 ή Turtle. Στο [163], οι συγγραφείς υποστηρίζουν πως η ταχύτητα των συστημάτων δεν επηρεάζεται μόνο από τον τρόπο με τον οποίο τα δεδομένα μεταδίδονται αλλά και από τον τρόπο με τον οποίο αυτά είναι αποθηκευμένα. Τα εργαλεία τα οποία εξετάστηκαν, υποστηρίζουν στο σύνολό τους προτυποποιημένες γλώσσες, θα ήταν επιθυμητές ωστόσο πιο πλούσιες λύσεις. Στον πίνακα 11, οι στήλες Γλώσσα Οντολογίας και Γλώσσα Ερωτημάτων συγκεντρώνουν την εκφραστικότητα των υπό εξέταση εργαλείων.

4. Ο *έλεγχος εκδόσεων* αποτελεί σημαντική διαδικασία, ειδικά στην περίπτωση

συνεργασίας ομάδων. Η διαδικασία συγγραφής τόσο της οντολογίας όσο και της Βάσης Δεδομένων – πόσο μάλλον της αντιστοίχισης – δεν είναι γραμμική διαδικασία και συχνά απαιτεί τη συνεισφορά ειδικών σε κάθε χώρο. Επομένως, θα ήταν επιθυμητό τα εργαλεία να παρέχουν λειτουργικότητα παρόμοια με το CVS ή το Microsoft Visual SourceSafe (VSS) τα οποία είναι διαδεδομένα μεταξύ των προγραμματιστών. Τα εργαλεία τα οποία παρέχουν γραφική και όχι προγραμματιστική διεπαφή δυστυχώς περιορίζονται στις δυνατότητες του γραφικού περιβάλλοντος. Αντίθετα, τα μη γραφικά εργαλεία όπως τα D2RQ, D2R MAP, R₂O ή τα APIs όπως το RDF Gateway παρέχουν δυνατότητα παράλληλης επεξεργασίας επιλύοντας το ζήτημα του ελέγχου εκδόσεων το οποίο καθίσταται αρκετά σημαντικό σε εμπορικά περιβάλλοντα.

5. Το *επίπεδο αυτοματοποίησης* κατά τη φάση σχεδιασμού είναι μια ιδιαίτερα σημαντική πτυχή του προβλήματος καθώς η χειροκίνητη δήλωση αντιστοιχίσεων αποτελεί μια χρονοβόρα και επιρρεπή σε λάθη διαδικασία. Μια εταιρεία είναι γενικότερα διστακτική στο να ξοδέψει πόρους προκειμένου να επιτύχει αμφίβολα αποτελέσματα. Λαμβάνοντας υπόψη τα δυο αυτά επιχειρήματα διαπιστώνεται ότι ο αυτοματισμός κατά το μέγιστο εφικτό αποτελεί ανάγκη καθώς δεν υποστηρίζονται επαρκείς μηχανισμοί αυτοματοποίησης. Η ομώνυμη στήλη του πίνακα 13 αποτυπώνει το επίπεδο αυτοματοποίησης που παρέχεται από τα εργαλεία που περιγράφηκαν. Ανάμεσα στα υπό διερεύνηση εργαλεία, μόνο το D2RQ και το Relational.OWL επιτρέπουν μέχρι ενός σημείου αυτοματοποίηση. Επιπλέον, στο κεφάλαιο 5 παρουσιάζεται μια προσέγγιση για την πλήρως αυτοματοποιημένη σημασιολογική επισημείωση πληροφορίας βασισμένη σε κανόνες.

6. *Πληρότητα*. Οποιοδήποτε σύστημα θα έπρεπε να παρέχει ελευθερία στο χρήστη να συλλάβει οποιοδήποτε υποσύνολο της υποκείμενης πληροφορίας. Με απλά λόγια, οποιοδήποτε υποσύνολο της οντολογίας θα έπρεπε να είναι δυνατό να αντιστοιχηθεί σε οποιοδήποτε σύνολο της Βάσης Δεδομένων. Δυστυχώς, η πλειονότητα των εργαλείων παρέχουν αντιστοιχίσεις μεταξύ πινάκων και στηλών της Βάσης Δεδομένων, όπως αποτυπώνεται στη στήλη 'Αντιστοιχιζόμενα Στοιχεία' του πίνακα 12. Θα ήταν επιθυμητή η ύπαρξη εργαλείων που να επιτρέπουν τη σύλληψη γενικότερων υποσυνόλων όπως για παράδειγμα επιτυγχάνει το Vis-A-Vis (ενότητα 4.3) με την ενσωμάτωση αυθαίρετων SQL ερωτημάτων.

7. *Επαναχρησιμοποίηση των αντιστοιχίσεων*. Το αποτέλεσμα της διαδικασίας αντιστοίχισης θα ήταν επιθυμητό να είναι εύκολα επεκτάσιμο και συνδυάσιμο με άλλες αντιστοιχίσεις. Οι συγγραφείς θα έπρεπε να είναι σε θέση να επαναχρησιμοποιήσουν και να συνενώσουν τμήματα της δουλειάς τους. Προτείνεται επομένως η καθιέρωση ενός προτύπου στο χώρο κατ' αντιστοιχία με το XMI για τα έγγραφα XML ή το KIF

για τις οντολογίες, το οποίο θα χρησιμοποιείται για τη συνεργασία μεταξύ ατόμων του χώρου. Παρ' όλα αυτά, τα εργαλεία που βασίζονται σε γραφική διεπαφή (Clio, MOMIS, MAPONTO) στερούνται της δυναμικής αυτής. Τα υπόλοιπα εργαλεία επιτρέπουν την περαιτέρω επεξεργασία των αντιστοιχίσεων ανεξάρτητα από το αν αυτές παρήχθησαν αυτόματα ή χειροκίνητα. Η απουσία όμως ενός κοινού και γενικότερα αποδεκτού προτύπου περιορίζει το αποτέλεσμα στις δυνατότητες των συγκεκριμένων εργαλείων.

Χρειάζεται επιπλέον να τονιστεί ότι η κοινότητα του Σημασιολογικού Ιστού θα μπορούσε και θα έπρεπε να δανειστεί ιδέες από την κοινότητα των Βάσεων Δεδομένων. Στο χώρο των Βάσεων Δεδομένων υπάρχει ώριμη τεχνογνωσία καθώς και χρόνια εμπειρίας μέσα στα οποία έχουν επιλυθεί ουσιαστικά ζητήματα όπως η ασφάλεια, η βελτιστοποίηση σε σημαντικό βαθμό των διαδικασιών αποθήκευσης των δεδομένων, υποβολής ερωτημάτων και διαχείρισης των αλληλεπιδράσεων των χρηστών. Επιπλέον, υπάρχει τεράστιος όγκος πληροφορίας ο οποίος εξυπηρετεί επιτυχώς το σκοπό του και η κοινότητα που τον συντηρεί θα είναι μάλλον απρόθυμη για μεταπήδηση σε νέες, λιγότερο ώριμες τεχνολογίες. Για το λόγο αυτό, είναι προτιμώμενη η ολοκλήρωση των υπάρχοντων τεχνολογικών υποδομών με το Σημασιολογικό Ιστό και όχι η μεταπήδηση σε αυτόν.

Στα μειονεκτήματα της χρήσης οντολογίας συμπεριλαμβάνεται η δυσκολία στη βιωσιμότητα του όλου εγχειρήματος μετά την συγγραφή των αντιστοιχίσεων. Δε μπορεί κανείς να διαφωνήσει με το γεγονός ότι σε σενάρια πραγματικού κόσμου, η οργάνωση των αντιστοιχίσεων μεταξύ Βάσεων Δεδομένων και οντολογιών αποτελεί επιπλέον φόρτο εργασίας ο οποίος κοστίζει σε πόρους και επιπλέον παρουσιάζει αμφίβολα αποτελέσματα καθώς αποτελεί μια χρονοβόρα και επιρρεπή σε σφάλματα διαδικασία. Ο υπεύθυνος για τη δημιουργία του συστήματος αντιστοίχισης θα χρειαστεί συχνά να λάβει σημαντικές, μη τετριμμένες αποφάσεις και θα χρειαστεί η εμπειρία και στους δυο τομείς, τόσο του Σημασιολογικού Ιστού όσο και στις σχεσιακές Βάσεις Δεδομένων αλλά και κατ' επέκταση στα μοντέλα του προβλήματος που καλείται να επιλύσει.

Επιπλέον, οι οντολογίες και οι Βάσεις Δεδομένων είναι σύνηθες να υποβάλλονται σε αλλαγές οι οποίες μπορούν εύκολα να καταστήσουν ξεπερασμένες κάποιες δηλώσεις αντιστοίχισης ή ακόμα χειρότερα, εσφαλμένες. Τίθεται τέλος το ερώτημα: ποιος, πώς και πόσο συχνά θα πραγματοποιεί τις αντιστοιχίσεις; Χωρίς κάποιου είδους συντήρηση, μια στατική αντιστοίχιση δε θα ικανοποιήσει ανάγκες πραγματικού κόσμου.

5 Προσθήκη σημασιολογίας σε πραγματικό χρόνο και καταναμημένο περιβάλλον

5.1 Εισαγωγή

Το κεφάλαιο αυτό κινείται στο ευρύτερο πλαίσιο του σημασιολογικού εμπλουτισμού της πληροφορίας σε συστήματα πραγματικού χρόνου. Ειδικότερα, η έρευνα επικεντρώνεται στη διερεύνηση των αλλαγών και προσθηκών που απαιτούνται τόσο σε τεχνικό όσο και σε θεωρητικό επίπεδο ώστε τα πληροφοριακά συστήματα με επίγνωση περιβάλλοντος (context-aware) να ενσωματώνουν, να εκμεταλλεύονται και να αξιοποιούν σε πραγματικό χρόνο σημασιολογική πληροφορία.

Η ουσιαστική διαφορά εδώ σε σχέση με την ολοκλήρωση Βάσεων Δεδομένων έγκειται στον παράγοντα του χρόνου. Τυπικά, τα συστήματα με επίγνωση περιβάλλοντος διαχειρίζονται μεγάλο όγκο πληροφορίας με πρωτογενή δεδομένα προερχόμενα από αισθητήρες. Ενώ είναι εύκολη μια χαμηλού επιπέδου online επεξεργασία, η οποία θα μπορεί ενδεχομένως να εξυπηρετεί έναν περιορισμένο αριθμό απαιτήσεων, γείρονται σημαντικές δυσκολίες όταν η πληροφορία αυτή πρόκειται να ολοκληρωθεί και αξιοποιηθεί σε σημασιολογικό επίπεδο.

Ειδικότερα, μελετώνται τα προβλήματα που προκύπτουν από τη σύγχρονη και ασύγχρονη επεξεργασία της πληροφορίας που εισρέει σε ένα σύστημα με επίγνωση περιβάλλοντος μέσω ενός δικτύου αισθητήρων σχηματίζοντας σε πραγματικό χρόνο μια σημασιολογική Βάση Γνώσεως, η οποία είναι ικανή να απαντά σε ερωτήματα υψηλότερου σημασιολογικού επιπέδου. Ιδιαίτερη αναφορά γίνεται στις διαδικασίες πραγματικού χρόνου που πραγματοποιούνται ώστε το σύστημα να καταστεί δυνατό να παρέχει υποστήριξη στην εξαγωγή συμπερασμάτων και τη λήψη αποφάσεων. Παρουσιάζεται στο κεφάλαιο αυτό μια νέα προσέγγιση στην ομογενοποίηση και ολοκλήρωση πληροφορίας που προέρχεται από ένα καταναμημένο δίκτυο αισθητήρων. Το ιδιαίτερο χαρακτηριστικό της προτεινόμενης προσέγγισης είναι ο ευφυής σχεδιασμός της αρχιτεκτονικής η οποία είναι βασισμένη σε μεσισμικό στο οποία μπορούν να ενσωματωθούν χαμηλότερου επιπέδου συνιστώσες όπως αισθητήρες, αλγόριθμοι εξαγωγής χαρακτηριστικών και πηγές δεδομένων, και ανώτερου επιπέδου όπως οντολογίες συγκεκριμένες για κάθε εφαρμογή και κανόνες για την σύγχρονη και ασύγχρονη επεξεργασία της εισρέουσας πληροφορίας.

Για τη μελέτη εφικτότητας στην προσέγγιση που προτείνεται και αναλύεται στο κεφάλαιο αυτό, δυο σενάρια εφαρμογών παρουσιάζονται: ένα σενάριο εφαρμογής για την ασφάλεια χώρων και ένα για την υποστήριξη συνεδριάσεων. Αναλύεται σε εννοιολογικό και τεχνικό επίπεδο ο τρόπος με τον οποίον επιτυγχάνεται η ρύθμιση του μεσισμικού

και των συνιστωσών που απαρτίζουν το περιβάλλον του, το αποτέλεσμα υποβάλλεται σε μετρήσεις απόδοσης και αναλύονται τέλος τα αποτελέσματα που προκύπτουν από την έρευνα αυτή.

Το κεφάλαιο είναι δομημένο ως εξής: καταρχήν εισάγονται και προσδιορίζονται αναλυτικότερα οι έννοιες που απασχολούν την παρούσα έρευνα, όπως η φύση των δεδομένων που χρησιμοποιούνται, η έννοια του μεσισμικού, των συστημάτων πραγματικού χρόνου, και οι έννοιες της σύγχρονης και ασύγχρονης επεξεργασίας πληροφορίας. Στην ενότητα 5.2 αναλύεται διεξοδικά η προτεινόμενη προσέγγιση, η 5.3 παρουσιάζει τα δυο σενάρια λειτουργίας του μεσισμικού σε εργαστηριακές συνθήκες, και στην 5.4 συγκεντρώνονται τα αποτελέσματα των μετρήσεων απόδοσης. Στην ενότητα 5.5 γίνεται αναφορά σε πιθανές αιτίες μη ομαλής λειτουργίας ή μείωσης της απόδοσης της προτεινόμενης προσέγγισης και τα αντίστοιχα μέτρα που λαμβάνονται για την αντιμετώπισή τους. Στην ενότητα 5.6 παρατίθενται τα συμπεράσματα που προκύπτουν από την παρούσα έρευνα, ενώ στην 5.7 δίνονται πιθανές μελλοντικές κατευθύνσεις στις οποίες μπορεί να κινηθεί η έρευνα στο χώρο.

5.1.1 Δεδομένα και Πληροφορία

Καταρχήν, χρειάζεται να προσδιοριστούν τα δεδομένα για τα οποία γίνεται λόγος στη συνέχεια. Όπως αναφέρθηκε και στην ενότητα 1.1, ο διαχωρισμός της *πληροφορίας* από τα *δεδομένα* υπόκειται στην οπτική γωνία του χρήστη του συστήματος. Στην ανάλυση ροών video για παράδειγμα με στόχο την εξαγωγή περιγραφής του περιεχομένου του, τα δεδομένα μπορεί να είναι τα καρέ του video ενώ η πληροφορία να είναι κάποια παραγόμενη επισημείωση.

Για την παρούσα εργασία, τα δεδομένα τα οποία χρησιμοποιούνται πειραματικά είναι κατ' ουσίαν μεταδεδομένα πολυμεσικής πληροφορίας, σε ημιδομημένη μορφή (XML), τα οποία προέρχονται από ετερογενείς καταναμημένες πηγές. Με τη χρήση μεσισμικού, τα δεδομένα αυτά εισρέουν στο σύστημα, εμπλουτίζονται, και μετατρέπονται σε σημασιολογική πληροφορία σχηματίζοντας μια Βάση Γνώσεως η οποία είναι ικανή να απαντά σε ερωτήματα υψηλότερου σημασιολογικού επιπέδου.

Τα δεδομένα αυτά διαχωρίζονται σε δυο ειδών: στατικά και πραγματικού χρόνου. Στα *στατικά* δεδομένα περιλαμβάνονται αρχεία ημιδομημένων δεδομένων όπως XML, RDF ή OWL ή πηγές δομημένων δεδομένων όπως οι σχεσιακές Βάσεις Δεδομένων. Η πρόσβαση στα στατικά δεδομένα μπορεί να γίνεται είτε τοπικά είτε από απομακρυσμένο εξυπηρετητή. Στα δεδομένα *πραγματικού χρόνου* περιλαμβάνονται εκείνα που παράγονται από αυτόματες διαδικασίες όπως για παράδειγμα από διατάξεις ανίχνευσης κίνησης, αισθητήρες, κάμερες που καταγράφουν το οπτικό τους πεδίο ή ακόμη και από λογισμικό

το οποίο καταγράφει κινήσεις χρηστών σε μια ιστοσελίδα. Το κοινό χαρακτηριστικό των δεδομένων πραγματικού χρόνου είναι ότι η ροή τους προς το σύστημα είναι ακαθόριστη, απρόβλεπτη και επηρεάζεται από εξωτερικούς παράγοντες.

Πρέπει να τονιστεί ότι τα δεδομένα δε χρειάζεται να περιέχουν καθεαυτό πληροφορία αλλά μπορεί να έχουν το ρόλο μεταδεδομένων, να αποτελούν απλά περιγραφή άλλων δεδομένων. Τυπική περίπτωση στην οποία είναι αναγκαία η ύπαρξη μεταδεδομένων είναι στα δεδομένα ροής εικόνας και ήχου, τα γεωγραφικά δεδομένα, και τα πολυμεσικά δεδομένα γενικότερα. Το ιδιαίτερο χαρακτηριστικό των περιπτώσεων αυτών είναι ότι η πληροφορία στην κωδικοποιημένη της μορφή, δεν είναι κατανοητή από τον άνθρωπο. Για παράδειγμα, μια γεωγραφική Βάση Δεδομένων περιέχει αριθμούς που εκφράζουν συνιστώσες, συντεταγμένες κλπ, δεν είναι όμως κατανοητή στο ανθρώπινο μάτι χωρίς κάποιες μορφής οπτικοποίηση. Το ίδιο ισχύει για την πολυμεσική πληροφορία γενικότερα. Είναι άμεση ανάγκη στην περίπτωση της πολυμεσικής πληροφορίας να συνοδεύεται από μεταδεδομένα. Χωρίς αυτά η χρηστικότητα της μειώνεται κατακόρυφα, όπως αναλύθηκε και στην ενότητα 3.7.5.

Η ανάγκη αυτή της ύπαρξης μεταδεδομένων επιτείνεται στην περίπτωση των συστημάτων που χρειάζεται να έχουν επίγνωση του περιβάλλοντός τους. Ειδικά στην περίπτωση που η επίγνωση του περιβάλλοντος επηρεάζει τη λήψη αποφάσεων ή αφορά θέματα ασφάλειας είναι βασική η διαχείριση της γνώσεως που μπορεί να συσσωρεύεται στο σύστημα. Εκτός από την κατηγοριοποίηση, την ανάκτηση και τη διατήρηση εκδόσεων είναι επίσης σημαντικό σε τέτοια συστήματα να υποστηρίζονται και επίπεδα αφαίρεσης με στόχο την πληρέστερη περιγραφή και κατανόηση από χρήστες με διαφορετικές αρμοδιότητες.

Στην προτεινόμενη υλοποίηση, προαπαιτείται η ύπαρξη μεταδεδομένων τα οποία έχουν παραχθεί με συμβατικές μεθόδους, όπως για παράδειγμα μεταδεδομένα αρχείων video, ήχου και εικόνων, log files εξυπηρετητών ή οποιεσδήποτε ροές μηνυμάτων από Υπηρεσίες Διαδικτύου. Τα μεταδεδομένα εμπεριέχουν επισημείωση σε δεδομένα εξωτερικά του παρουσιαζόμενου συστήματος. Η μόνη απαίτηση από τα μεταδεδομένα αυτά είναι η ύπαρξη δομής, η οποία να τα συνοδεύει (για παράδειγμα XML σύνταξη) ή να τα ακολουθεί (σχεσιακές Βάσεις Δεδομένων).

5.1.2 Μεσισμικό

Η γενικότερη έννοια του μεσισμικού (middleware) αναφέρεται σε τμήματα λογισμικού τα οποία αναλαμβάνουν να διασυνδέσουν άλλα τμήματα λογισμικού ή εφαρμογές παρέχοντας τη μεταξύ τους διαλειτουργικότητα. Εν προκειμένω, το μεσισμικό που αναπτύχθηκε καθιστά ικανή την επικοινωνία μεταξύ εφαρμογών πραγματικού

κόσμου και ενός δικτύου αισθητήρων το οποίο συλλέγει την πληροφορία των συμβάντων του κόσμου. Επομένως, το εν λόγω μεσισμικό παρέχει υπηρεσίες προς δύο κατευθύνσεις: προς το δίκτυο αισθητήρων το οποίο συλλέγει την πληροφορία και προς τις εφαρμογές που βασίζονται στο μεσισμικό για την αξιοποίηση της πληροφορίας αυτής. Επιπλέον, το μεσισμικό είναι υπεύθυνο για τη διαχείριση και επεξεργασία της πληροφορίας σε σημασιολογικό επίπεδο. Το μεσισμικό που προτείνεται ως πιλοτική εφαρμογή για τη διαχείριση και τον εμπλουτισμό όλης αυτής της πληροφορίας βασίζεται σε τεχνολογίες Σημασιολογικού Ιστού. Αναλυτική περιγραφή της εσωτερικής αρχιτεκτονικής του μεσισμικού αλλά και ενός συστήματος που βασίζεται στο μεσισμικό δίνεται στην ενότητα 5.2.2.

5.1.3 Επεξεργασία πραγματικού χρόνου

Ένα σύστημα λογισμικού το οποίο δέχεται, επεξεργάζεται και παράγει δεδομένα, ορίζεται ως *πραγματικού χρόνου* όταν ικανοποιεί συγκεκριμένους χρονικούς περιορισμούς (deadlines) προκειμένου να παράξει απάντηση και να αποφύγει πιθανή αποτυχία [164]. Επιπρόσθετα, ένα σύστημα πραγματικού χρόνου εμφανίζει λογική συνέπεια τόσο στα αποτελέσματα όσο και στον χρόνο παραγωγής τους (timeliness). Ο χρόνος μεταξύ της εμφάνισης των δεδομένων στο σύστημα και της ολοκλήρωσης της παραγωγής των αντίστοιχων αποτελεσμάτων που το αφορούν ονομάζεται *χρόνος απόκρισης* (latency).

Μια σημαντική έννοια στα συστήματα πραγματικού χρόνου είναι η έννοια του γεγονότος. Ένα γεγονός είναι ένα συμβάν το οποίο έχει ως αποτέλεσμα τη μεταβολή της ροής εκτέλεσης ενός προγράμματος. Τα γεγονότα διακρίνονται σε δύο ειδών: σύγχρονα και ασύγχρονα. Τα *σύγχρονα* γεγονότα είναι αυτά τα οποία προκύπτουν σε προβλέψιμους χρόνους ενώ *ασύγχρονα* είναι τα γεγονότα τα οποία προκύπτουν σε μη προβλέψιμους χρόνους και προκαλούνται συνήθως από εξωτερικές πηγές του συστήματος. Με βάση τους χρόνους κατά τους οποίους τα γεγονότα αυτά υφίστανται επεξεργασία ορίζεται αντίστοιχα και η επεξεργασία της πληροφορίας σε σύγχρονη και ασύγχρονη, όπως αναλύεται στην 5.1.4.

Μια συνηθισμένη παρερμηνεία σχετικά με τα συστήματα πραγματικού χρόνου αφορά την αντίληψη ότι τα συστήματα αυτά χρειάζεται να έχουν πολύ μικρό χρόνο απόκρισης, π.χ. της τάξης χιλιοστών του δευτερολέπτου για κάθε γεγονός. Η απαίτηση όμως για το χαρακτηρισμό ενός συστήματος ως πραγματικού χρόνου είναι ο χρόνος απόκρισης να υπακούει σε συγκεκριμένες προθεσμίες και να είναι προβλέψιμος (ντετερμινιστικός).

Η βασική διαφορά των συστημάτων πραγματικού χρόνου από τα συστήματα μη πραγματικού χρόνου είναι η έμφαση των πρώτων στην πρόβλεψη του χρόνου απόκρισης και της προσπάθειας για τη μείωσή του. Αυτό επιχειρείται και στις μετρήσεις που έχουν

ληφθεί πάνω στο προτεινόμενο σύστημα: στα γραφήματα των μετρήσεων στην ενότητα 5.4, το μέγεθος που ενδιαφέρει είναι ο χρόνος απόκρισης.

Διαισθητικά, κάθε σύστημα μπορεί να χαρακτηριστεί ως σύστημα πραγματικού χρόνου απλά με την προσθήκη προθεσμιών στις λειτουργίες του. Λόγω της θεώρησης αυτής, ο ορισμός ενός συστήματος πραγματικού χρόνου περιορίζεται ώστε να λαμβάνει υπόψη τις προθεσμίες που έχουν τεθεί. Σύμφωνα με τους E. Dougherty και Ph. Laplante [164], τα συστήματα πραγματικού χρόνου χωρίζονται σε τρία είδη:

- **Hard real-time:** η αποτυχία σε μια προθεσμία για την επεξεργασία των δεδομένων οδηγεί σε πλήρη αποτυχία του συστήματος.
- **Firm real-time:** μπορεί να προκύψει αποτυχία σε συγκεκριμένο αριθμό προθεσμιών χωρίς πλήρη αποτυχία του συστήματος.
- **Soft real-time:** η αποτυχία σε προθεσμίες μπορεί να οδηγήσει σε μείωση της απόδοσης του συστήματος αλλά όχι σε αποτυχία.

Συνηθίζεται, ο όρος “συστήματα πραγματικού χρόνου” να αναφέρεται στα συστήματα *hard real-time*, εκτός αν διαφοροποιηθεί ρητά. Η σύμβαση αυτή ακολουθείται και στην παρούσα προτεινόμενη αρχιτεκτονική, με την έννοια του ότι οι προθεσμίες τηρούνται και η αποτυχία του συστήματος αποφεύγεται.

5.1.4 Σύγχρονη και ασύγχρονη επεξεργασία πληροφορίας

Οι έννοιες σύγχρονη και ασύγχρονη επεξεργασία αφορούν τα συστήματα πραγματικού χρόνου. Κατά την *σύγχρονη επεξεργασία πληροφορίας*, κάθε νέο γεγονός του περιβάλλοντος που γίνεται αντιληπτό από το σύστημα υφίσταται επεξεργασία άμα τη γενέσει του. Ως αποτέλεσμα, η προσέγγιση αυτή είναι επιθυμητή όταν ο χρόνος απόκρισης είναι σημαντική παράμετρος. Για παράδειγμα, σε εφαρμογές ασφάλειας, ένα ανεπιθύμητο γεγονός του περιβάλλοντος χρειάζεται να γίνει αντιληπτό σε προβλέψιμο και σχετικά μικρό χρόνο από το σύστημα προκειμένου να λάβουν χώρα τυχόν ενέργειες για την αντιμετώπισή του.

Αντίθετα, κατά την *ασύγχρονη επεξεργασία*, τα γεγονότα του περιβάλλοντος μπορεί να υποστούν επεξεργασία σε χρόνο μη σχετιζόμενο με τη χρονική στιγμή κατά την οποία έλαβαν χώρα. Ασύγχρονη επεξεργασία θεωρείται για παράδειγμα μια εντολή δημιουργίας ευρετηρίων σε μια Βάση Δεδομένων ή η δημιουργία ενός κύβου OLAP για την υποστήριξη εξόρυξης δεδομένων (*data mining*).

Κατά την προτεινόμενη προσέγγιση, τα γεγονότα που εισρέουν στο σύστημα λαμβάνουν χώρα ασύγχρονα, σε χρόνους μη προβλέψιμους, καθώς το μεσισμικό λειτουργεί ως εξυπηρετητής μηνυμάτων από εξωτερικές πηγές. Η *επεξεργασία* όμως

των γεγονότων είναι σύγχρονη καθώς συγκεκριμένοι χρονικοί περιορισμοί τίθενται όσον αφορά το χρόνο επεξεργασίας της πληροφορίας που προκύπτει.

Το ενδιαφέρον σημείο της σύγχρονης επεξεργασίας στην παρούσα αρχιτεκτονική εντοπίζεται στο ότι η απόκριση του συστήματος γίνεται πιο αργή όσο πιο πολλή πληροφορία συσσωρεύεται και θέτει σαφώς όρια στην επεκτασιμότητα. Εφόσον κάθε νέο γεγονός που προστίθεται στο σύστημα χρειάζεται να λάβει υπόψη όλη την υπάρχουσα πληροφορία του συστήματος, είναι αναμενόμενο ο χρόνος απόκρισης να αυξάνει ανάλογα με τον όγκο της υπάρχουσας πληροφορίας και να περιορίζει δραστικά την επεκτασιμότητα του συστήματος.

Για το λόγο αυτό παρουσιάζεται στη συνέχεια η πρόταση για *σύγχρονο* διαχωρισμό της Βάσης Γνώσεως σε προσωρινή και μόνιμη (Εικόνα 17). Η πρώτη επεξεργάζεται την πληροφορία που προκύπτει και την αποθηκεύει προσωρινά (buffer) και η δεύτερη χρησιμοποιείται για την αποθήκευση της μόνιμης πληροφορίας (persistent), η οποία έχει ήδη υποστεί επεξεργασία. Συγκεκριμένες συνθήκες πυροδοτούν την μεταφορά της συσσωρευμένης πληροφορίας από την προσωρινή στη μόνιμη Βάση Γνώσεως. Οι συνθήκες αυτές μπορεί να ορίζονται αυθαίρετα, να εξαρτώνται από το χρόνο ή από τον όγκο των δεδομένων. Ο στόχος του διαχωρισμού αυτού είναι η αύξηση της ταχύτητας στην επεξεργασία των νεοεισερχομένων δεδομένων αλλά και η δημιουργία μιας πλήρους Βάσης Γνώσεως στην οποία καταγράφεται όλο το ιστορικό του συστήματος σε λειτουργία.

Η μεταφορά της πληροφορίας από την προσωρινή Βάση Γνώσεως μπορεί να γίνεται και ασύγχρονα, υπό τη μορφή δαίμονα (daemon): λογισμικού που δρα στο παρασκήνιο. Το λογισμικό αυτό μπορεί να ρυθμιστεί ώστε να έχει την ίδια λειτουργικότητα με το λογισμικό που αναλαμβάνει σύγχρονες λειτουργίες. Στην προτεινόμενη προσέγγιση, έχει υλοποιηθεί ρυθμιζόμενος daemon ο οποίος να δρα ασύγχρονα, οι μετρήσεις στην ενότητα 5.4 όμως παρουσιάζουν τη λειτουργία του συστήματος με σύγχρονες διαδικασίες.

5.2 Εξαγωγή σημασιολογικής πληροφορίας από πολυμεσική πληροφορία

Η ενότητα αυτή περιγράφει την αρχιτεκτονική του μεσισμικού, τόσο ως προς τις εξωτερικές διεπαφές του με τις πηγές δεδομένων και τις εφαρμογές, όσο και ως προς τις εσωτερικές διεπαφές των επιμέρους συνιστωσών λογισμικού που το απαρτίζουν. Χρειάζεται όμως στην αρχή μια επισκόπηση της σχετικής βιβλιογραφίας ούτως ώστε να αποσαφηνιστεί το πρόβλημα, οι λύσεις που έχουν προταθεί και αντίστοιχα, η προσφορά της παρούσας εργασίας.

5.2.1 Σχετικές Εργασίες

Σύμφωνα με τους Stamou et al. [165], η πληροφορία που μπορεί να μεταφέρει ένα πολυμεσικό αρχείο μπορεί να αναπαρασταθεί και να αναλυθεί με βάση τρία επίπεδα αφαιρέσεως: υπο-συμβολικό (subsymbolic), συμβολικό (symbolic) και λογικό (logical) επίπεδο.

Το πρώτο επίπεδο στοχεύει στα ακατέργαστα πολυμεσικά δεδομένα τα οποία βρίσκονται συνήθως αποθηκευμένα με τη χρήση ευρέως διαδεδομένων προτύπων για ήχο και εικόνα. Στο επίπεδο αυτό έχουν αναπτυχθεί πολυάριθμοι αλγόριθμοι για την αναγνώριση και παρακολούθηση συγκεκριμένων χαρακτηριστικών όπως για παράδειγμα ένα αντικείμενο ή ένα ανθρώπινο πρόσωπο σε μια ροή εικόνας ή ομιλία σε μια ροή ήχου. Σχετικά με την αναγνώριση προσώπου για παράδειγμα, μπορεί κανείς να χρησιμοποιήσει τον αλγόριθμο των Viola-Jones ενώ για την παρακολούθηση προσώπων έναν από τους αλγόριθμους PCA [166], LDA [167] ή ICA [168]. Σχετικά με την παρακολούθηση αντικειμένων ή προσώπων, οι ανάγκες μιας εφαρμογής μπορεί να καλυφθούν από την υλοποίηση ενός αλγόριθμου όπως ο Mean-shift [169], ο Camshift [170] ή από ένα Kalman filter [171]. Για τον εντοπισμό ομιλίας υπάρχουν στη βιβλιογραφία αλγόριθμοι όπως αυτός των J. Sohn et al. [172]. Επιπλέον, αρκετές μέθοδοι έχουν επίσης προταθεί για την αναγνώριση και την εξαγωγή γεγονότων ενδιαφέροντος σε πολυμεσικές ροές πληροφορίας. Στο [173] για παράδειγμα, παρουσιάζεται ένας μηχανισμός επισημείωσης αγώνων καλαθοσφαίρισης ενώ αντίστοιχα στα [174, 175], το άθλημα ενδιαφέροντος είναι το μπίτζμπολ.

Παρά το γεγονός ότι οι προσεγγίσεις αυτές αποτελούν στο σύνολό τους ικανές λύσεις, η επισημείωση πολυμεσικού περιεχομένου στο επίπεδο αυτό δεν είναι συνήθως κατάλληλη για περαιτέρω επεξεργασία. Αυτό το επίπεδο επισημείωσης δε διευκολύνει την προσπάθεια για ολοκλήρωση σε ανώτερο επίπεδο ετερογενούς πολυμεσικής πληροφορίας προερχόμενης από κατανεμημένες πηγές. Το πρόβλημα αυτό αντιμετωπίζεται με την υιοθέτηση προτύπων όπως τα MPEG-7, MPEG-21, Visual Resource Association (VRA), International Press Communications Council (IPTC), NewsML, τα οποία λειτουργούν στο συμβολικό επίπεδο. Ο στόχος των προτύπων αυτών είναι η παροχή ομοιογένειας στις επισημειώσεις με στόχο το συνδυασμό των αποτελεσμάτων που προκύπτουν από επισημείωση κατανεμημένων πηγών.

Ωστόσο, το πρόβλημα που παραμένει ακόμη και με την επισημείωση στο συμβολικό επίπεδο είναι ότι η επισημείωση αυτή παρουσιάζει μεν ομοιογένεια αλλά αυτή είναι μόνο συντακτική και δεν μεταφέρει σημασιολογία. Επιπλέον, οι επισημειώσεις σε κάθε περίπτωση είναι στενά συνδεδεμένες με τα χαρακτηριστικά του κάθε λεξιλογίου και του αντίστοιχου τύπου αρχείου πολυμέσων. Επομένως, προκειμένου να καταστεί εφικτή η

ολοκλήρωση και η διαλειτουργικότητα σε λογικό επίπεδο, χρειάζεται η προσθήκη ενός επιπέδου σημασιολογίας.

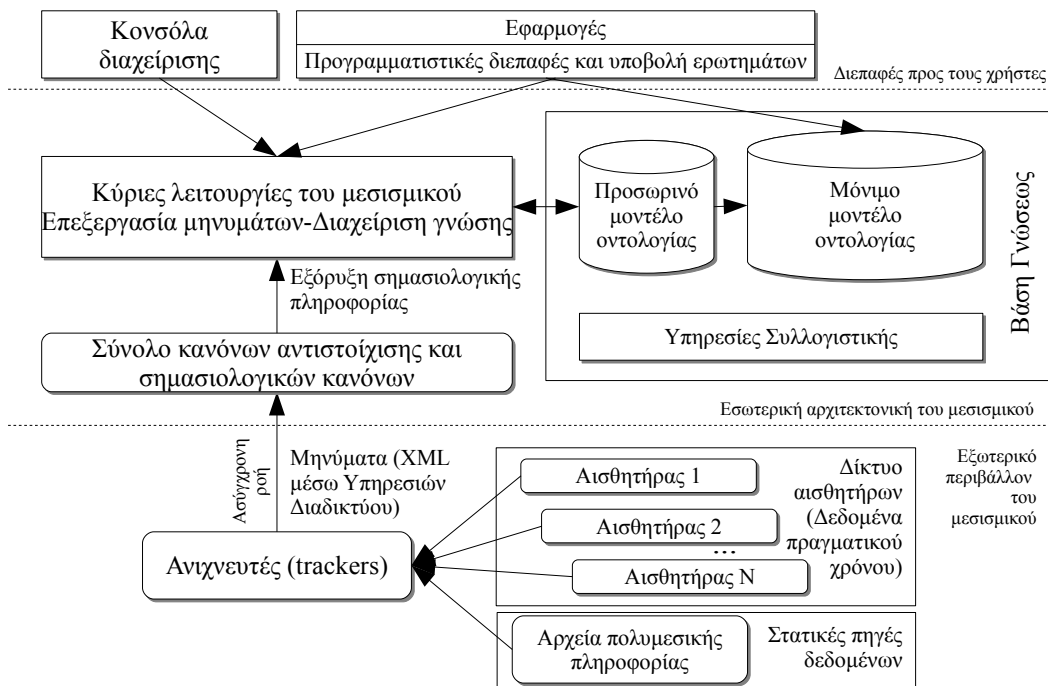
Το τρίτο επίπεδο προσφέρει σημασιολογική επισημείωση της πληροφορίας χρησιμοποιώντας τεχνολογίες Σημασιολογικού Ιστού. Τα εργαλεία που λειτουργούν σ' αυτό το επίπεδο αναλύονται διεξοδικότερα στην ενότητα 3.7.5.

Σε σχέση με τις παραπάνω προσεγγίσεις, η καινοτομία της παρούσας προσέγγισης βασίζεται στην ευελιξία και την προσαρμοστικότητα που μπορεί να προσφέρει ένα σύστημα βασισμένο σε μεσισμικό, σε αντίθεση με τις κατά περίπτωση λύσεις: Το μεσισμικό που παρουσιάζεται στο κεφάλαιο αυτό επιτρέπει την αφαίρεση (abstraction) των αποτελεσμάτων σε κάθε επίπεδο επισημείωσης. Συγκεκριμένα, trackers μπορούν να ενσωματωθούν στο μεσισμικό, λειτουργώντας στο πρώτο (υπο-συμβολικό) επίπεδο, λεξιλόγια μπορούν να επιλεγθούν με στόχο να παρέχουν τις απαραίτητες διεπαφές στα μηνύματα που αποστέλλονται (συμβολικό επίπεδο) και, τέλος, κανόνες μπορούν να οριστούν οι οποίοι να επισημαίνουν σημασιολογικά την παραγόμενη πληροφορία και να παράγουν μια Βάση Γνώσεως στο τρίτο (λογικό) επίπεδο. Επιπλέον, η προσέγγιση και η μελέτη του προβλήματος εστιάζει στα προβλήματα που προκύπτουν από την εφαρμογή σε συστήματα πραγματικού χρόνου, μια πτυχή του προβλήματος που δε μελετήθηκε από καμία από τις προηγούμενες προσεγγίσεις.

5.2.2 Αρχιτεκτονική του μεσισμικού

Η αρχιτεκτονική του μεσισμικού βασίζεται σε τμήματα (modules) τα οποία εγγυώνται επεκτασιμότητα, προσαρμοστικότητα και ανεξαρτησία από τεχνολογικές προσεγγίσεις. Αυτά χωρίζονται στα παρακάτω επιμέρους τμήματα και παρουσιάζονται στην εικόνα 17.

- Οι *αισθητήρες* (sensors) είναι υπεύθυνοι για τη σύλληψη γεγονότων που λαμβάνουν χώρα στον πραγματικό κόσμο. Οι αισθητήρες είναι συνδεδεμένοι με ανιχνευτές οι οποίοι επεξεργάζονται σε πρώτο στάδιο τα δεδομένα των αισθητήρων. Κάθε αισθητήρας ελέγχεται από έναν ελεγκτή (sensor controller) ο οποίος ρυθμίζει ανάλογα τον αισθητήρα με βάση τις εντολές που εισάγει ο χρήστης. Αυτός ο ελεγκτής είναι προσαρμοσμένος για κάθε αισθητήρα γιατί οι εντολές του χρήστη μετατρέπονται σε εντολές χαμηλού επιπέδου προς την συσκευή, εντολές οι οποίες είναι ιδιαίτερες για κάθε τύπο συσκευής και διαφορετικές για κάθε κατασκευαστή.
- Οι *ανιχνευτές* (trackers) αποτελούν αλγόριθμους αναγνώρισης χαρακτηριστικών χαμηλού επιπέδου και είναι υπεύθυνοι για την παραγωγή των μηνυμάτων που αποστέλλονται στο μεσισμικό σχετικά με τα γεγονότα του περιβάλλοντος. Τα μηνύματα αυτά περιέχουν μεταδεδομένα της πρωτογενούς πολυμεσικής



Εικόνα 17: Γενική αρχιτεκτονική συστήματος αυτόματης εξαγωγής σημασιολογίας

πληροφορίας που συλλέγεται από τους αισθητήρες.

- Το *σύνολο κανόνων* εμπεριέχει την ευφυΐα στην επεξεργασία των μηνυμάτων που εισέρχονται από εξωτερικές πηγές. Οι κανόνες είναι δυο ειδών *Κανόνες αντιστοίχισης* και *Σημασιολογικοί κανόνες*. Οι κανόνες, η πλήρης σύνταξη των οποίων παρατίθεται στα παραρτήματα Α και Β, αναλύονται περισσότερο στην ενότητα 5.2.3, ενώ παραδείγματα κανόνων παρουσιάζονται στην ανάλυση των εφαρμογών της ενότητας 5.3. Επιπλέον, στην ενότητα 5.3.3 αναλύεται το πώς οι κανόνες αυτοί συνεργάζονται για το διαχωρισμό και τη σύντηξη των μηνυμάτων που λαμβάνει το μεσισμικό.
- Οι *κύριες λειτουργίες του μεσισμικού* αποτελούν την προγραμματιστική διεπαφή (API) με τις εφαρμογές που βασίζονται στο μεσισμικό και με το υποσύστημα εξαγωγής χαρακτηριστικών. Η διεπαφή παρέχεται μέσω της χρήσης Υπηρεσιών Διαδικτύου (Web Services). Μέσω της διεπαφής αυτής ο χρήστης μπορεί να διαχειριστεί απομακρυσμένα τα σύνολα των κανόνων, τις ενέργειες τις οποίες μπορεί να εκτελέσει το μεσισμικό, τα πρότυπα μηνυμάτων τα οποία αποτυπώνουν τη δομή σύμφωνα με την οποία χρειάζεται να συμμορφώνονται τα εισερχόμενα στο μεσισμικό μηνύματα, να υποβάλλει ερωτήματα σε ανώτερο σημασιολογικό

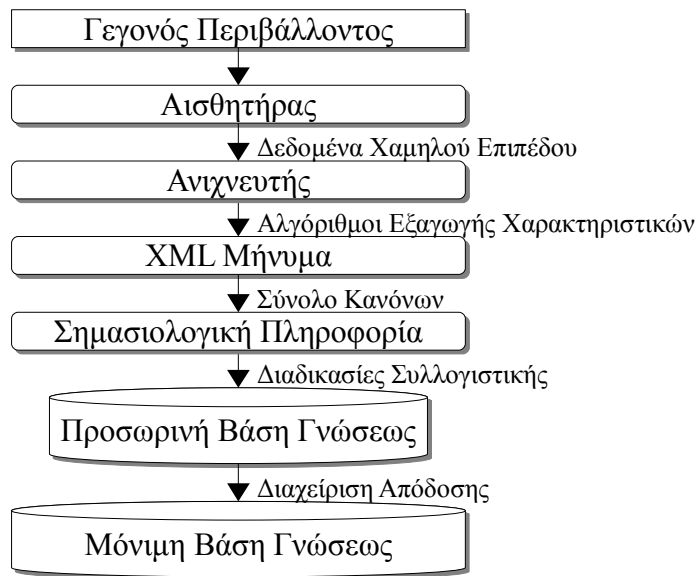
επίπεδο κ.ο.κ. Αναλυτικότερα, το API παρουσιάζεται στην 5.2.5.

- Το *μοντέλο οντολογίας* περιγράφει τις έννοιες του κόσμου που αφορούν το προς επίλυση πρόβλημα. Το προσωρινό και το μόνιμο μοντέλο οντολογίας ακολουθούν την ίδια δομή (TBox). Η διαφορά έγκειται στα ABox (ενότητα 1.7) των δυο οντολογιών, τα οποία περιέχουν τα individuals των οντολογιών. Το προσωρινό μοντέλο οντολογίας περιέχει ένα υποσύνολο των individuals, τα πιο πρόσφατα όπως αναλύεται στη συνέχεια, ενώ το μόνιμο μοντέλο οντολογίας περιέχει το σύνολο της γνώσης που συσσωρεύεται.
- Οι *υπηρεσίες συλλογιστικής* (reasoning) δύνανται να εξάγουν γνώση η οποία δεν έχει πιθανότατα δοθεί κατηγορηματικά (explicitly) στο σύστημα (ενότητα 3.6). Οι υπηρεσίες συλλογιστικής σε συνδυασμό με την οντολογία συνθέτουν τη Βάση Γνώσεως (εικόνα 2).

Προκειμένου να αποσαφηνιστεί περισσότερο ο τρόπος με τον οποίο λειτουργεί το μεσισμικό, αποτυπώνεται στην εικόνα 18 η πορεία επεξεργασίας των εισερχόμενων μηνυμάτων. Οι ανιχνευτές (trackers) είναι οι πρώτοι που επεξεργάζονται τα ακατέργαστα δεδομένα. Μόλις αρχίσουν την επεξεργασία, παράγουν τα μηνύματα που περιέχουν τις περιγραφές των γνωρισμάτων που αναγνωρίζονται από τους αισθητήρες. Τα μηνύματα αυτά δεν περιορίζονται σε συγκεκριμένη περιγραφή γλώσσας, αλλά μπορεί να είναι αυθαίρετα καλώς δομημένα XML. Μέσω αυτών των μηνυμάτων, τα δεδομένα μεταφέρονται στον κεντρικό εξυπηρετητή του μεσισμικού και. Μέσω της επεξεργασίας που υφίστανται από τους κανόνες για τους οποίους έχει ρυθμιστεί το μεσισμικό, μια Βάση Γνώσεως δημιουργείται με όλα τα χαρακτηριστικά γνωρίσματα ενδιαφέροντος συγκεντρώνοντας όλα τα μηνύματα που αναγνωρίζονται από τους αισθητήρες.

Ένα σχεδιαστικό ζήτημα το οποίο χρειάστηκε να επιλυθεί κατά το σχεδιασμό της αρχιτεκτονικής αφορά το πώς επιτυγχάνεται η αλληλεπίδραση μεταξύ των επί μέρους καταναμημένων τμημάτων. Στις προσεγγίσεις που προτείνονται στη βιβλιογραφία περιλαμβάνονται οι αυτόνομοι πράκτορες λογισμικού και οι Υπηρεσίες Διαδικτύου. Οι αυτόνομοι πράκτορες λογισμικού έχουν το πλεονέκτημα ότι είναι εντελώς ανεξάρτητοι από το σύστημα στο οποίο δραστηριοποιούνται και προσφέρουν ευέλικτα και επεκτάσιμα συστήματα. Δεν έχουν τύχει ωστόσο ευρείας αποδοχής όπως αναλύθηκε στην ενότητα 3.7.1. Αντίθετα, οι Υπηρεσίες Διαδικτύου είναι αντικείμενο έρευνας για το οποίο έγινε εκτεταμένη αναφορά στην ενότητα 3.7.3.

Και οι δυο προσεγγίσεις μπορούν να οδηγήσουν σε ευέλικτες, καταναμημένες αρχιτεκτονικές. Προσφέρουν τη δυνατότητα αυτοματοποίησης των ενεργειών και προσθήκης ευφυΐας στο σύστημα που υλοποιούν. Επίσης κοινό τους χαρακτηριστικό είναι ότι επιτρέπουν τη συνένωση συστημάτων δημιουργώντας υποδομές που υποστηρίζουν τη



Εικόνα 18: Επεξεργασία εισερχόμενων μηνυμάτων

συνεργασία μεταξύ ετερογενών τμημάτων. Τέλος, μπορούν και οι δυο προσεγγίσεις να αποκτήσουν το ρόλο διεπαφής μεσισμικού με το περιβάλλον του. Η διεπαφή των κλήσεων προς το μεσισμικό μπορεί να είναι υλοποιημένη με RMI, CORBA, RPC, J2EE ή Web Services.

Η παρούσα εργασία ακολουθεί προσέγγιση βασισμένη σε Υπηρεσίες Διαδικτύου και επομένως, οι κλήσεις προς το λογισμικό μπορεί να γίνονται από οποιοδήποτε σύστημα είναι συμβατό με WSDL όπως για παράδειγμα το Windows Communication Foundation των Windows Vista ή το Axis και το CXF της Apache.

Επιπλέον, η αποθήκευση της οντολογίας και των αποθηκευμένων μεταδεδομένων δεν περιορίζεται σε συγκεκριμένες τεχνολογίες. Η JDBC συνδεσιμότητα με τη Βάση Δεδομένων επιτρέπει για παράδειγμα χρήση MySQL, PostgreSQL, Oracle, ακόμη και Apache Derby. Αντίστοιχα και οι εφαρμογές πελάτη δεν περιορίζονται σε συγκεκριμένη πλατφόρμα. Λόγω της επικοινωνίας με Υπηρεσίες Διαδικτύου, μπορεί ο προγραμματιστής να επιλέξει οποιαδήποτε γλώσσα διαθέτει βιβλιοθήκες που υλοποιούν τα πρότυπα των Υπηρεσιών Διαδικτύου.

Ανάμεσα στα πλεονεκτήματα της συγκεκριμένης προσέγγισης είναι επίσης η πρακτικά απεριόριστη ελευθερία στην περιγραφή των δεδομένων. Η αμοιβαία προκαθορισμένη διεπαφή που χρησιμοποιείται στη διασύνδεση με εξωτερικές πηγές δεδομένων και αισθητήρων μπορεί να κάνει χρήση λεξιλογίων που έχουν αναπτυχθεί

προκειμένου να περιγράψουν συγκεκριμένους τομείς ενδιαφέροντος, όπως η SensorML⁶⁸ ή η MPEG-7 DDL [176]. Απεριόριστη ελευθερία παρέχεται επίσης και στη σημασιολογική περιγραφή της πληροφορίας αφού ούτε στην οντολογία που χρησιμοποιείται υπάρχει κάποια δέσμευση. Μπορεί μια εφαρμογή να χρησιμοποιήσει οποιοδήποτε από τα λεξικά για τα οποία έγινε λόγος στην ενότητα 2.5 αλλά και πιο στοχευμένες υλοποιήσεις όπως η βασισμένη σε OWL VEMML [177] ή το XMP της Adobe⁶⁹.

Δίνεται η δυνατότητα εκ των υστέρων επεξεργασίας και εκμετάλλευσης της Βάσης Γνώσεως που δημιουργείται μέσω υποβολής σημασιολογικών ερωτημάτων υψηλότερου επιπέδου. Επιπλέον, η συγκεκριμένη προσέγγιση στην αποθήκευση της γνώσης επιτρέπει διασύνδεση με άλλες πηγές, καθώς η Βάση Γνώσεως υπακούει σε διαδεδομένα πρότυπα.

Χρειάζεται να υπογραμμιστεί ότι το σύστημα δεν περιορίζεται μόνο στη χρήση για πολυμεσικό περιεχόμενο. Μπορεί επίσης να χρησιμοποιηθεί για όλες τις περιπτώσεις όπου η επίγνωση του περιβάλλοντος σε συνδυασμό με την επισημείωση περιεχομένου είναι κρίσιμη, παραδείγματος χάριν για αισθητήρες που δεν επεξεργάζονται απαραίτητα πολυμεσικό περιεχόμενο όπως αισθητήρες θερμοκρασίας ή υγρασίας. Βήματα προς αυτή την κατεύθυνση παρουσιάζονται στο [107].

5.2.3 Κανόνες αντιστοίχισης και Σημασιολογικοί κανόνες

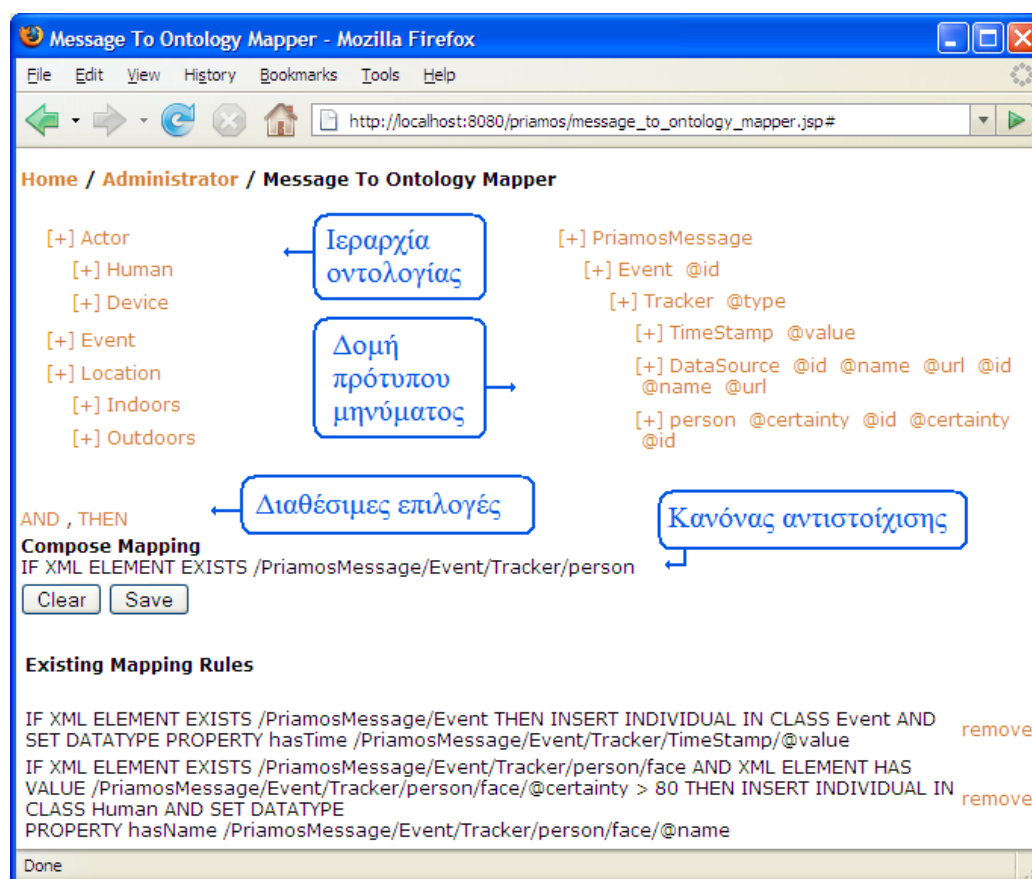
Για την υποστήριξη της προσθήκης σημασιολογίας, δυο ειδών κανόνες έχουν χρησιμοποιηθεί, οι κανόνες *αντιστοίχισης* οι οποίοι αναλαμβάνουν την αντιστοίχιση XML μηνυμάτων σε έννοιες και ιδιότητες της οντολογίας, και οι *σημασιολογικοί* κανόνες οι οποίοι επεξεργάζονται τη σημασιολογική πληροφορία με στόχο είτε την εξαγωγή περαιτέρω γνώσεως είτε την πυροδότηση ενεργειών από το μεσισμικό. Στα παραρτήματα Α και Β παρατίθεται η πλήρης επίσημη (formal) σύνταξη των γλωσσών κανόνων που αναπτύχθηκαν στα πλαίσια της παρούσας έρευνας και αποτελούν συνεισφορά της διατριβής.

Στην εικόνα 19 απεικονίζεται η γραφική διεπαφή που αναπτύχθηκε με στόχο τη διευκόλυνση στη συγγραφή κανόνων αντιστοίχισης. Παρέχεται η δυνατότητα στο χρήστη να ορίσει κανόνες σύμφωνα με τη γλώσσα των κανόνων αντιστοίχισης. Ο χρήστης αντιστοιχίζει τα στοιχεία των εισερχόμενων μηνυμάτων περιγράφοντας στο μεσισμικό πώς αυτά θα αποθηκεύονται στην Βάση Γνώσεως που δημιουργείται.

Στην εικόνα 20 απεικονίζεται η γραφική διεπαφή για τη συγγραφή σημασιολογικών

⁶⁸OpenGIS Sensor Model Language (SensorML): www.opengeospatial.org/standards/sensorml

⁶⁹eXtensible Metadata Platform (XMP): www.adobe.com/products/xmp/

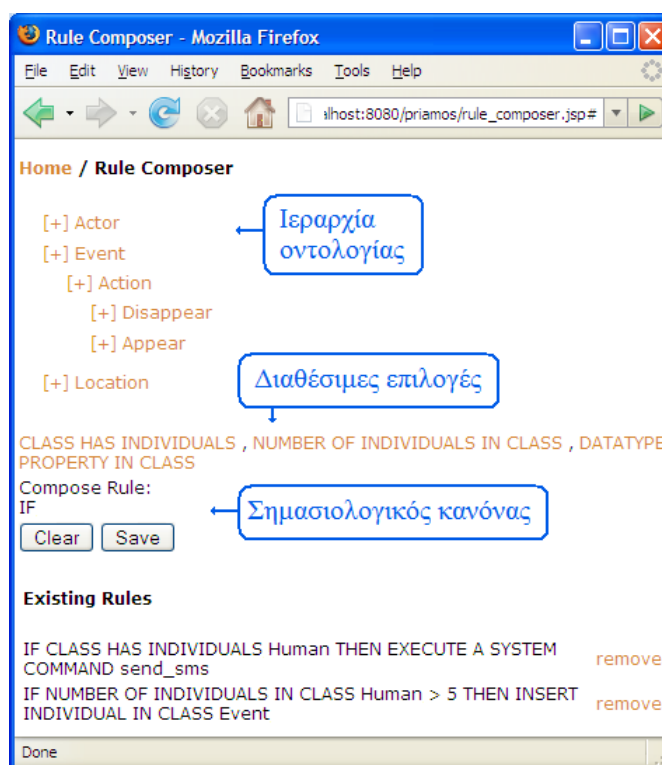


Εικόνα 19: Σύνταξη Κανόνων αντιστοίχισης

κανόνων. Ο χρήστης μπορεί να ορίσει τις συνθήκες που επιθυμεί προκειμένου να πυροδοτούνται ενέργειες από το μεσισμικό.

Εξετάστηκε αρχικά η χρήση της SWRL ή άλλης γλώσσας σύνταξης κανόνων. Διαπιστώθηκε όμως ελλιπής προγραμματιστική υποστήριξη, γεγονός που οφείλεται στην τρέχουσα ανάπτυξη των τεχνολογιών που απαρτίζουν το Σημασιολογικό Ιστό αλλά και στην απουσία καθιερωμένων προτύπων. Επομένως, κρίθηκε αναγκαία η ανάπτυξη συγκεκριμένης γλώσσας ορισμού κανόνων.

Παράλληλα με την ανάπτυξη των γλωσσών για τη συγγραφή κανόνων, εξετάστηκε και το ενδεχόμενο πλήρους αυτοματοποίησης της διαδικασίας επεξεργασίας των μηνυμάτων. Συγκεκριμένα, εξετάστηκε το ενδεχόμενο να μη χρειάζεται συγγραφή κανόνων και τα εισερχόμενα μηνύματα να αποθηκεύονται αυτόματα στην οντολογία. Το πρόβλημα αυτό αφορά την αυτόματη αντιστοίχιση των αρχείων XML σε κλάσεις και ιδιότητες της



Εικόνα 20: Σύνταξη Σημασιολογικών κανόνων

οντολογίας. Πολλές προτάσεις υπάρχουν στη βιβλιογραφία σχετικά με την αυτόματη εισαγωγή αρχείων XML σε RDF όπως για παράδειγμα με τη χρήση XSLT και το πρόβλημα κρίνεται πλέον τετριμμένο από ερευνητικής απόψεως. Επομένως, η διατήρηση των κανόνων κρίθηκε ως η βέλτιστη λύση προκειμένου το μεσοστικό να είναι πιο προσαρμόσιμο στις ανάγκες των εφαρμογών.

5.2.4 Ερωτήματα σημασιολογικού επιπέδου

Η δημιουργία μιας Βάσης Γνώσεως δεν παρουσιάζει πρακτικό ενδιαφέρον αν δεν παρέχεται ταυτόχρονα ένας μηχανισμός υποβολής σημασιολογικών ερωτημάτων. Για την υποβολή ευφώνων ερωτημάτων, τα οποία θα αξιοποιούν τον όγκο της σημασιολογικής πληροφορίας, ακολουθείται η εξής προσέγγιση: Αρχικά, γίνεται από το χρήστη ο σχηματισμός των ερωτημάτων. Το σύνηθες πρόβλημα των μηχανών ερωτημάτων του Σημασιολογικού Ιστού είναι ότι απαιτείται από το χρήστη η εκ των προτέρων γνώση των δυνατοτήτων του συστήματος [116]. Προκειμένου ο χρήστης να εκμεταλλευτεί τις

πλήρεις δυνατότητες θα πρέπει να είναι εξοικειωμένος με το μοντέλο του κόσμου όπως είναι αποθηκευμένο στη Βάση Γνώσεως. Στη συνέχεια, τα ερωτήματα χρειάζεται να υποβληθούν στη Βάση και να υπολογιστεί το σύνολο των συμβατών δεδομένων. Το ερώτημα του χρήστη θα μετατρέπεται σε ερώτημα πάνω στον RDF γράφο – η γλώσσα υλοποίησης είναι η ARQ, υλοποίηση της SPARQL η οποία αναλύεται στην ενότητα 2.7.2. Τέλος, τα αποτελέσματα ταξινομούνται και παρουσιάζονται στο χρήστη του συστήματος.

Το βασικότερο πρόβλημα στην υποβολή σημασιολογικών ερωτημάτων εντοπίζεται στο ότι δεν είναι σκόπιμη η απαίτηση εξοικείωσης από τον τελικό χρήστη. Δεν έχει νόημα η παροχή μιας υπηρεσίας η οποία θα είναι δυσνόητη ή στην καλύτερη περίπτωση δύσχρηστη. Χρειάζεται μια προσέγγιση η οποία θα ενσωματώνει στο ακέραιο τις δυνατότητες των υπαρχουσών γλωσσών ερωτημάτων του Σημασιολογικού Ιστού, θα προσφέρει ευελιξία και κατά το δυνατό, απλότητα. Με βάση το σκεπτικό αυτό, η παρούσα εργασία βασίζεται σε προ-αποθηκευμένα πρότυπα ερωτημάτων (query templates) για την κάθε εφαρμογή τα οποία θα ικανοποιούν πιθανά σενάρια. Για παράδειγμα, σε ένα σενάριο παρακολούθησης χώρων (ενότητα 5.3.1), μερικά από τα ερωτήματα που μπορεί να είναι προ-αποθηκευμένα είναι τα εξής:

- Πότε έγινε αντιληπτό για τελευταία φορά το άτομο X ή κάποιος από τους συναδέλφους του;
- Πότε βρέθηκαν στον ίδιο χώρο τα άτομα μιας κατηγορίας X και μιας κατηγορίας Y ταυτόχρονα;
- Υπήρξε στο χώρο παρακολούθησης άτομο εκτός ωρών εργασίας χωρίς την επίβλεψη υπεύθυνου;

Τα παραδείγματα αυτά σημασιολογικών ερωτημάτων είναι δύσκολο να επιτευχθούν δίχως τη χρήση τεχνολογιών Σημασιολογικού Ιστού, γεγονός που τεκμηριώνει τον ισχυρισμό ότι η προσθήκη σημασιολογίας απελευθερώνει πλήθος δυνατοτήτων [11]. Επιπλέον, με βάση τα προ αποθηκευμένα πρότυπα ερωτημάτων, ο χρήστης μπορεί να δημιουργήσει παραλλαγές ερωτημάτων ή να τα προσαρμόσει στις ανάγκες του κάθε ερωτήματος με στόχο να πάρει τα επιθυμητά αποτελέσματα.

Τα αποτελέσματα που επιστρέφονται στο χρήστη είναι τα μεταδεδομένα (η επισημείωση) της αρχικής πολυμεσικής πληροφορίας αλλά και η ίδια η πολυμεσική πληροφορία καθώς στα μεταδεδομένα διατηρείται δείκτης στην τοποθεσία των δεδομένων αυτών.

5.2.5 Χρήστες συστήματος και παρεχόμενη λειτουργικότητα

Καταρχήν, χρειάζεται να οριστούν οι κατηγορίες χρηστών στους οποίους απευθύνεται το υπό περιγραφή σύστημα. Στην πρώτη κατηγορία εμπίπτει το γενικό κοινό, οι τελικοί χρήστες των εφαρμογών οι οποίες αναπτύσσονται με βάση το μεσισμικό. Οι γενικοί χρήστες δε χρειάζεται να έχουν εξοικείωση με τις τεχνολογίες Σημασιολογικού Ιστού ούτε χρειάζεται να έχουν γνώση της αρχιτεκτονικής και των τεχνικών λεπτομερειών του μεσισμικού. Οι δυνατότητες αλληλεπίδρασης που τους παρέχονται περιορίζονται στην υποβολή ερωτημάτων και τη λήψη ειδοποιήσεων από το μεσισμικό.

Η δεύτερη και σημαντικότερη κατηγορία αφορά τους προγραμματιστές/δημιουργούς λογισμικού οι οποίοι αποτελούν τους κύριους χρήστες του μεσισμικού. Είναι επίσης αυτοί οι οποίοι ωφελούνται περισσότερο από την ύπαρξη ενός μεσισμικού το οποίο παρέχει την υποδομή για την αυτοματοποιημένη ολοκλήρωση ετερογενών πηγών πληροφορίας και τον σημασιολογικό εμπλουτισμό της. Η κύρια δυνατότητα που παρέχεται στους προγραμματιστές είναι η δυνατότητα να χρησιμοποιήσουν προηγμένα χαρακτηριστικά με γενικό τρόπο. Μπορούν να διασυνδέσουν αλγόριθμους εξαγωγής χαρακτηριστικών με επαναχρησιμοποιήσιμο τρόπο, διαφανή ως προς τις τεχνολογίες εξαγωγής χαρακτηριστικών και να συνθέσουν πολύπλοκα και ευφυή σενάρια χρήσης βασισμένα σε καθορισμένη σημασιολογία.

Σε τεχνικό επίπεδο, οι λειτουργίες που παρέχονται προς τους προγραμματιστές είναι οι εξής:

- Εισαγωγή και επεξεργασία οντολογιών. Οι οντολογίες που εισάγονται αποθηκεύονται σε Βάση Δεδομένων σε RDF/XML μορφοποίηση αλλά και σε Jena persistent model (triplestore). Η λογική που ακολουθείται είναι ότι το μοντέλο της οντολογίας που εισάγεται είναι χωρίς άτομα – πρόκειται ουσιαστικά για το TBox της οντολογίας – και δεν παρουσιάζει περιττή πολυπλοκότητα στις έννοιες που περιγράφει. Το μοντέλο που αποθηκεύεται στη Βάση Δεδομένων μπορεί πρακτικά να είναι κατά πολύ μεγαλύτερο καθώς το μεσισμικό θα προσθέτει στο ABox άτομα με βάση τους κανόνες που έχουν οριστεί.
- Εισαγωγή και επεξεργασία προτύπων μηνυμάτων. Τα πρότυπα αυτά μηνύματα είναι τα μηνύματα που δέχεται το μεσισμικό από το υποσύστημα εξαγωγής χαρακτηριστικών και μπορεί να είναι οποιοδήποτε καλά δομημένο XML. Τα πρότυπα των μηνυμάτων αυτών χρησιμεύουν για τη σύνταξη των κανόνων αντιστοίχισης μεταξύ των κόμβων του δένδρου του XML και των κλάσεων και ιδιοτήτων της οντολογίας. Οι κανόνες αυτοί είναι της μορφής “if συνθήκη then ενέργεια”.

- Εισαγωγή, επεξεργασία και εκτέλεση σημασιολογικών ερωτημάτων. Τα πρότυπα ερωτημάτων δημιουργούνται σύμφωνα με τη γλώσσα SPARQL και αποτελούν τα ερωτήματα τα οποία παρέχονται προς τους τελικούς χρήστες οι οποίοι μπορούν να τα εκτελέσουν αλλά και να δημιουργήσουν άλλα σύμφωνα με τις ανάγκες τους. Ο λόγος για τον οποίο παρέχονται πρότυπα ερωτημάτων είναι ότι ενώ χρειάζεται αυτά να είναι δομημένα σύμφωνα με το λεξιλόγιο της οντολογίας, δεν πρέπει να υπάρχει απαίτηση από τους χρήστες να είναι εξοικειωμένοι με αυτό.
- Διαχείριση των κανόνων (σημασιολογικών και αντιστοίχισης) οι οποίοι καθορίζουν τη συμπεριφορά του μεσιμικού στα εισερχόμενα μηνύματα.
- Διαχείριση των ενεργειών τις οποίες μπορεί να εκτελέσει το μεσιμικό. Οι ενέργειες περιλαμβάνουν εκτέλεση τοπικών αρχείων ή αποστολή μηνυμάτων Υπηρεσιών Διαδικτύου. Η υλοποίηση βασίζεται σε Apache/Axis2.
- Ρύθμιση της κατάστασης λειτουργίας του μεσιμικού. Μέσω των συναρτήσεων αυτών δηλώνεται αν το μεσιμικό θα επεξεργάζεται ή όχι τα μηνύματα που δέχεται.

Πρότυπα μηνυμάτων	Σημασιολογικά ερωτήματα
<code>insertMessageTemplate</code>	<code>getQueryNames</code>
<code>removeMessageTemplate</code>	<code>insertQuery</code>
<code>getMessageTemplateId</code>	<code>removeQuery</code>
<code>getMessageTemplateNames</code>	<code>getQueryContents</code>
Μοντέλα οντολογιών	Κανόνες
<code>insertOntology</code>	<code>insertMappingRule</code>
<code>removeOntology</code>	<code>insertRule</code>
<code>getOntologyId</code>	<code>removeMappingRule</code>
<code>getOntologyNames</code>	<code>removeRule</code>
<code>getOntologyContents</code>	<code>getMappingRules</code>
	<code>getRules</code>
(Απ)ενεργοποίηση μεσιμικού	Ενέργειες μεσιμικού
<code>turnOnMiddleware</code>	<code>insertAction</code>
<code>turnOffMiddleware</code>	<code>removeAction</code>
<code>getMiddlewareStatus</code>	<code>getActionPathToExecutable</code>
<code>getActionNames</code>	
Εισερχόμενα μηνύματα	
<code>priamosMessage</code>	

Πίνακας 15: Σύνοψη των προγραμματιστικών διεπαφών του μεσιμικού

Προκειμένου το μεσιμικό να είναι προσαρμόσιμο στις ανάγκες των εκάστοτε

εφαρμογών, ένα σύνολο από συναρτήσεις παρέχεται τις οποίες οι προγραμματιστές μπορούν να καλούν απομακρυσμένα, μέσω Υπηρεσιών Διαδικτύου. Μέσω της διεπαφής αυτής, σύνοψη της οποίας παρέχεται στον πίνακα 15 είναι δυνατή η απομακρυσμένη μοντελοποίηση και διαχείριση της συμπεριφοράς του μεσισμικού.

5.3 Παράδειγμα επεξεργασίας πραγματικού χρόνου

Στην ενότητα αυτή παρουσιάζεται ένα παράδειγμα ευφυούς παρακολούθησης χώρου βασισμένο στην προτεινόμενη αρχιτεκτονική. Ο στόχος του παραδείγματος είναι να αποσαφηνιστεί καλύτερα η λειτουργικότητα που παρέχεται με τη χρήση ενός μεσισμικού για την επεξεργασία των εισερχόμενων μηνυμάτων και το σημασιολογικό εμπλουτισμό της πληροφορίας που συγκεντρώνεται στο σύστημα. Επιπλέον, παρουσιάζεται ο τρόπος με τον οποίο ενσωματώνεται και αξιοποιείται ευφυΐα στις εφαρμογές που μπορεί να αναπτυχθούν με βάση το μεσισμικό⁷⁰.

5.3.1 Παρακολούθηση χώρων

Στην ενότητα αυτή παρουσιάζεται ένα σενάριο παρακολούθησης χώρων για το οποίο μπορεί να γίνει χρήση της προτεινόμενης αρχιτεκτονικής. Κατά το σενάριο αυτό, τοποθετούνται σε ένα χώρο κάμερες, οι οποίες είναι συνδεδεμένες με υπολογιστή στον οποίο εκτελούνται οι αλγόριθμοι αναγνώρισης και παρακολούθησης προσώπων. Το μεσισμικό ρυθμίζεται με συγκεκριμένη οντολογία (παρατίθεται στο παράρτημα Ε), με συγκεκριμένο πρότυπο μηνυμάτων στο οποίο χρειάζεται να συμμορφώνονται όλα τα μηνύματα που αποστέλλονται στο μεσισμικό από τον υπολογιστή που ελέγχει τις κάμερες, με συγκεκριμένα σύνολα κανόνων αντιστοίχισης και σημασιολογικών και συγκεκριμένες ενέργειες που μπορεί να πραγματοποιήσει. Σημειώνεται ότι η συγγραφή της οντολογίας πραγματοποιήθηκε με βάση την εργασία που παρουσιάζεται στο [102] και στην οποία οι συγγραφείς έχουν ακολουθήσει την προσέγγιση *resource, actor, environment* προκειμένου να περιγράψουν το μοντέλο γνώσης ενός συστήματος με επίγνωση περιβάλλοντος.

Κατά τη λειτουργία του σεναρίου, εξετάζονται πρώτα τα άτομα που εισέρχονται στο δωμάτιο – στο χώρο αντίληψης της κάμερας. Οι κάμερες αποτυπώνουν την αντίληψή τους για το χώρο σε μορφή video η οποία υφίσταται επεξεργασία από τους

⁷⁰Η εν λόγω έρευνα διεξήχθη στα πλαίσια του ερευνητικού έργου ΠΡΙΑΜΟΣ (ΠΡοσαρμοστικά συστήματα πραγματικού χρόνου για εξόρυξη σημασιολογίας και ευφυείς διεπαφές - επίδειξη σε εφαρμογές ασφάλειας και επικοινωνίας του πολίτη), το οποίο χρηματοδοτήθηκε από τη Γενική Γραμματεία Έρευνας και Τεχνολογίας (ΓΓΕΤ), Επιχειρησιακό Πρόγραμμα “Κοινωνία της Πληροφορίας”, μέτρο 3.3: “Επεξεργασία εικόνων, ήχου και γλώσσας”.

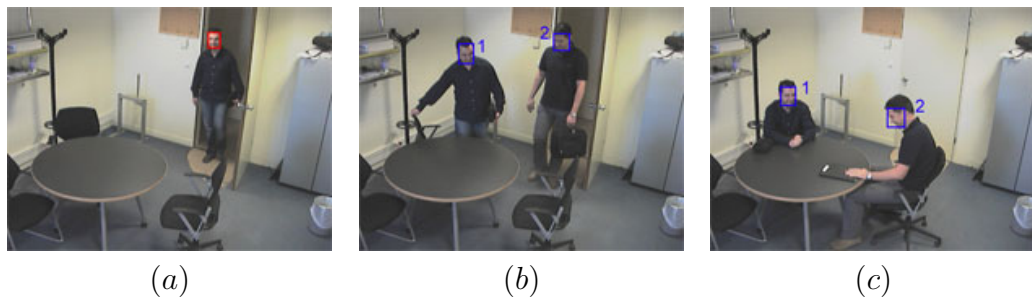
ανιχνευτές (trackers). Οι ανιχνευτές παράγουν απλή περιγραφή XML η οποία περιέχει τετριμμένη πληροφορία όπως ο αύξων αριθμός του μηνύματος, ο χρόνος, η πηγή αισθητήρα, οι συντεταγμένες ενός προσώπου αν υπάρχει, και ένα δείκτη στην τοποθεσία της πρωτογενούς πληροφορίας, για παράδειγμα ένα URL στην εικόνα που αποτυπώθηκε από την κάμερα. Για παράδειγμα, το παρακάτω μήνυμα αποστέλλεται από μια κάμερα τοποθετημένη κοντά στην οροφή ενός δωματίου η οποία καταγράφει την κίνηση ατόμων στο δωμάτιο.

```
<Event id="5712">
  <Tracker type="FaceTracker">
    <DataSource id="3" name="CeilingCamera"
      url="http://localhost/seq_0077.jpg"/>
    <person id="1" certainty="100">
      <location2d datasourceId="3" x="429" y="46"/>
      <face dbpersonid="10" name="John" certainty="91"/>
    </person>
  </Tracker>
</Event>
```

Σημειώνεται ότι στο μήνυμα διατηρείται δείκτης στην αρχική πληροφορία προκειμένου να είναι δυνατή η μεταγενέστερη ανάκτησή της: η διαδρομή /Event/Tracker/Datasource/@url στο παραπάνω XML αντιστοιχεί στην εικόνα η οποία επεξεργάστηκε προκειμένου να παραχθεί το μήνυμα.

Για τις ανάγκες του παραδείγματος, στο σύστημα των καμερών ενσωματώνονται δυο αλγόριθμοι: ο αλγόριθμος Viola-Jones για αναγνώριση προσώπων [178] και ο αλγόριθμος Camshift [170] για παρακολούθηση προσώπων. Ο αλγόριθμος Viola-Jones έχει ρυθμιστεί ώστε να αναγνωρίζει ένα συγκεκριμένο σύνολο ανθρώπων. Το video που παράγεται από τις κάμερες αναλύεται σε πραγματικό χρόνο. Στην εικόνα 21 εμφανίζονται τρία στιγμιότυπα από το σενάριο παρακολούθησης χώρων. Στην 21(a) παρουσιάζεται το ότι ο αλγόριθμος Viola-Jones αναγνωρίζει ανθρώπινο πρόσωπο τη στιγμή που αυτό εισέρχεται στο δωμάτιο. Στις 21(b) και 21(c) παρουσιάζεται ο αλγόριθμος Camshift ο οποίος παρακολουθεί την κίνηση των προσώπων. Επιπλέον, οι αλγόριθμοι αποτυπώνουν σε XML την αντίληψή τους σύμφωνα με το πρότυπο μηνυμάτων που περιγράφηκε προηγουμένως, το οποίο και αποστέλλουν με Υπηρεσίες Διαδικτύου στο μεσισμικό.

Στο οντολογικό μοντέλο του μεσισμικού υπάρχουν ήδη ορισμένες κατηγορίες από γνωστά (Known) και άγνωστα (Unknown) πρόσωπα σύμφωνα με το αν είναι δυνατή η αναγνώρισή τους ή όχι από το σύστημα εξαγωγής χαρακτηριστικών. Ανάλογα με το αν το πρόσωπο αναγνωρίζεται ή όχι, εντάσσεται αντίστοιχα ως άτομο σε μια από αυτές τις κλάσεις. Ένας κανόνας που το επιτυγχάνει είναι ο ακόλουθος:



Εικόνα 21: Στιγμιότυπα της εφαρμογής παρακολούθησης χώρων

```
if, /Event/Tracker/person/face/@dbpersonid, eq, 10, then,
insert individual in class, Professor,
and set datatype property, hasXLocation,
/Event/Tracker/person/location2d/@x
```

Σημειώνεται ότι χάρη απλότητας ο παραπάνω κανόνας δεν παρουσιάζεται σε ολοκληρωμένη μορφή. Σε συνθήκες πραγματικού κόσμου θα τροποποιούταν επίσης και μερικές άλλες ιδιότητες του κάθε ατόμου, όπως οι συντεταγμένες του και ενδεχομένως μερικές ιδιότητες αντικειμένου (object properties) όπως το γεγονός στο οποίο συμμετέχει. Προβλέπεται επίσης η δυνατότητα διατήρησης καταφυγίου όσον αφορά τη βεβαιότητα των μηνυμάτων που λαμβάνονται από το μεσισμικό. Για παράδειγμα, ένα νέο πρόσωπο μπορεί να γίνεται αποδεκτό όταν η βεβαιότητα (η διαδρομή `/Event/Tracker/person/face/@certainty` στο παραπάνω XML) είναι μεγαλύτερη του 90%.

Με παρόμοιο τρόπο, το σύστημα μπορεί να εκπαιδευτεί ώστε να αναγνωρίζει τα πρόσωπα που και ο αλγόριθμος αναγνώρισης προσώπων αναγνωρίζει, αντιστοιχίζοντας το αναγνωριστικό τους (id 10 στο παραπάνω παράδειγμα) με την κατάλληλη κλάση της οντολογίας. Η εφαρμογή του κανόνα θα οδηγήσει στην εισαγωγή στην οντολογία μιας τριάδας της ακόλουθης μορφής αν το άτομο έχει αναγνωριστεί για παράδειγμα ως Professor:

```
default:John      rdf:type      default:Professor ;
                  default:hasXLocation  "429"^^xsd:int .
```

Αν ο αλγόριθμος δεν αναγνωρίσει το άτομο, η τριάδα που θα εισαχθεί θα είναι της μορφής:

```
default:Unknown_1  rdf:type      default:Unknown ;
                   default:hasXLocation  "429"^^xsd:int .
```

Ένας άλλος κανόνας θα μπορούσε να εισάγει άτομα στην κλάση `Student` κ.ο.κ., ούτως ώστε το σύστημα να έχει πλήρη γνώση του κόσμου.

Στη συνέχεια εκτελούνται οι Σημασιολογικοί κανόνες. Με βάση τη γλώσσα σύνταξης η οποία δημιουργήθηκε (πλήρης περιγραφή της στο παράρτημα Β), δηλώνονται οι ενέργειες τις οποίες μπορεί να εκτελέσει το μεσισμικό βασισμένο μόνο στην κατάσταση της οντολογίας.

i) Δεν επιτρέπεται σε φοιτητή (άτομο της κλάσης `Student`) να βρίσκεται στον επιβλεπόμενο χώρο χωρίς τη συνοδεία μέλους του προσωπικού (άτομο της κλάσης `Staff`). Ο κανόνας αυτός μπορεί να επεκταθεί ώστε να καλύπτει και άλλα σενάρια συνοδείας, η χρήση των κλάσεων `Student` και `Staff` δεν είναι δεσμευτική.

```
if, class has individuals, Unknown, and,
SPARQL query does not have results,
SELECT ?x ?y WHERE {
    ?x rdf:type default:Staff .
    ?x default:hasTime ?time1 .
    ?y rdf:type default:Student .
    ?y default:hasTime ?time2
FILTER (?time1 = ?time2) }
then,
alert ("Unknown unattended person detected")
```

ii) Σε ώρες εργασίας, μόνο γνωστά άτομα επιτρέπονται.

```
if, SPARQL query has results,
SELECT ?x ?y WHERE {
    ?x rdf:type default:Unknown .
    ?x :hasTime ?y .
FILTER (?y > "20:00:00"^^xsd:time || ?y < "08:00:00"^^xsd:time) }
then,
alert ("Unknown person detected during working hours")
```

iii) Κατά τις βραδινές ώρες (για παράδειγμα μετά τις 23:00 και πριν τις 7:00) κανένας δεν πρέπει να εισέλθει στο δωμάτιο.

```
if, datatype property in class, Actor, gt, "23:00:00", and,
datatype property in class, Actor, lt, "07:00:00",
then,
alert ("Person Detected during High-Security Level Time")
```


Στις παραπάνω περιπτώσεις, `alert` είναι μια εντολή η οποία θα μπορούσε να ανάψει τα φώτα, να ενεργοποιήσει ένα συναγερμό, να αποστείλει ένα email, ένα SMS ή ένα μήνυμα Υπηρεσιών Διαδικτύου. Ένας υπεύθυνος φύλαξης του χώρου θα μπορούσε για παράδειγμα να ειδοποιηθεί με ένα SMS στο κινητό του τηλέφωνο.

Τα πλεονεκτήματα της προσέγγισης συνοψίζονται στην ενσωμάτωση ευφυΐας, της εκφραστικότητα η οποία παρέχεται στην περιγραφή της επιθυμητής συμπεριφοράς του συστήματος ενώ παράλληλα γίνεται επεξεργασία πραγματικού χρόνου σε μεγάλο όγκο εισερχομένων μηνυμάτων. Επιπρόσθετα, μια Βάση Γνώσεως δημιουργείται η οποία επιτρέπει μελλοντική αξιοποίηση της συγκεντρωμένης πληροφορίας.

Η χρήση του Σημασιολογικού Ιστού δικαιολογείται λόγω της διευκόλυνσης που παρέχει στην περιγραφή σύνθετων σεναρίων και στην επαναχρησιμοποίηση την οποία καθιστά δυνατή. Για παράδειγμα, στον κανόνα (i), αντί των κλάσεων `Staff` και `Student`, θα μπορούσαν να χρησιμοποιηθούν οι κλάσεις `Adult` και `Minor` ή οποιεσδήποτε κλάσεις `Class_A` και `Class_B` των οποίων η συνύπαρξη θα ήταν επιθυμητή.

Επιπρόσθετα, ένας απλός κανόνας ο οποίος ελέγχει την ύπαρξη ενός φοιτητή (η δήλωση `if, class, has individuals, Student`), σε συνδυασμό με υπηρεσίες συλλογιστικής θα συνυπολογίσει μεταξύ των ατόμων της κλάσης `Student` και άτομα των κλάσεων `Postgraduate` και `Undergraduate`, χωρίς να χρειαστεί να δηλωθούν επιπλέον κανόνες. Το ίδιο ισχύει και για την κλάση `Staff`, στην οποία επίσης συνυπολογίζονται τα άτομα των υποκλάσεών της. Με αυτόν τον τρόπο, ο κανόνας (i) συλλαμβάνει μια σειρά συνδυασμών συνυπαρχόντων ατόμων από κλάσεις οι οποίες συνδέονται μεταξύ τους με σχέση κλάσης/υποκλάσης.

Για το παράδειγμα, βασιζόμαστε κυρίως στη σχέση κλάσης/υποκλάσης αλλά η χρήση υπηρεσιών συλλογιστικής απελευθερώνει όλη την εκφραστικότητα της OWL DL όσον αφορά την περιγραφή του κόσμου.

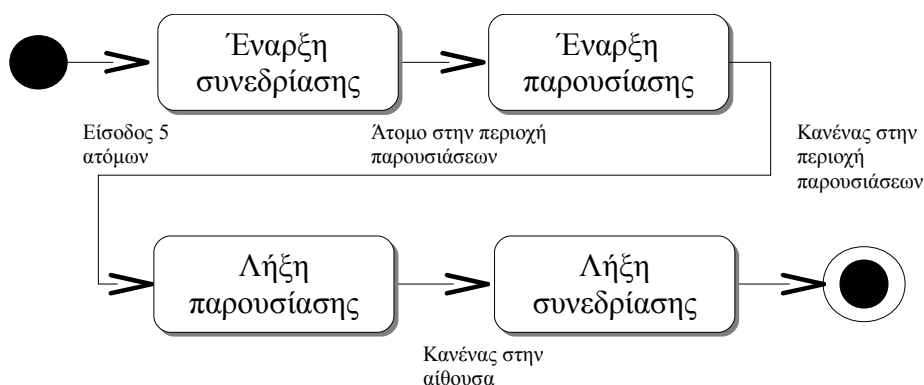
5.3.2 Εφαρμογή υποστήριξης συνεδριάσεων

Ένα απλό σενάριο για την επίδειξη μιας εφαρμογής υλοποιημένης με βάση το μεσισμικό για την χρήση της σε συνεδριακούς χώρους ή χώρους πραγματοποίησης διαλέξεων γενικότερα, αποτελεί το εξής:

Σε ένα χώρο πραγματοποιούνται συνεδριάσεις αποτελούμενες από συγκεκριμένα άτομα. Τα άτομα αναμένονται να εισέλθουν όλα στο χώρο πριν ξεκινήσει η συνεδρίαση. Οι χρήστες είναι ήδη εγγεγραμμένοι στο σύστημα και αναγνωρίζονται από αυτό κατά την είσοδό τους και κάθονται στο χώρο που προσδιορίζεται για το κοινό. Μετά την είσοδο όλων των συνέδρων, ο ομιλητής μεταβαίνει στο χώρο παρουσιάσεων και ξεκινάει την παρουσίασή του. Η έναρξη της ομιλίας του, που εντοπίζεται από το σύστημα σηματοδοτεί

και την έναρξη της παρουσιάσής του. Έπειτα συνεχίζεται η παρουσίαση μέχρι που ο ομιλητής επιστρέφει στο χώρο του κοινού. Η απομάκρυνση του ομιλητή από το χώρο παρουσιάσεων σηματοδοτεί και το τέλος της παρουσιάσης. Τέλος όταν αποχωρήσουν όλοι από τη αίθουσα τελειώνει η συνεδρίαση. Όπως παρατηρείται, στο σενάριο αυτό υπάρχει ένας αριθμός από διακριτές καταστάσεις στις οποίες μεταβαίνει το σύστημα.

Ορίζοντας επομένως τα γεγονότα που οδηγούν από τη μία κατάσταση στην άλλη μπορεί να παρακολουθηθεί η διαδικασία της συνεδρίασης. Οι καταστάσεις είναι οι εξής: έναρξη συνεδρίασης, έναρξη παρουσίασης, λήξη παρουσίασης, και λήξη συνεδρίασης. Τα γεγονότα που καθορίζουν τους πιθανούς τρόπους μετάβασης από τη μία κατάσταση στην άλλη παρουσιάζονται στην εικόνα 22 η οποία μοντελοποιεί την κατάσταση του συστήματος.



Εικόνα 22: Διάγραμμα καταστάσεων εφαρμογής υποστήριξης συνεδριάσεων

Οι κανόνες που υποστηρίζουν το σενάριο της εφαρμογής υποστήριξης συνεδριάσεων είναι οι εξής:

1. Ένας κανόνας αντιστοίχισης ο οποίος αναλαμβάνει να μετασχηματίσει την ημιδομημένη πληροφορία που αποστέλλεται στο μεσοσμικό μέσω Υπηρεσιών Διαδικτύου σε σημασιολογικά εμπλουτισμένη. Αναλυτικότερα, η γνώση για το κάθε άτομο που γίνεται αντιληπτό από τις κάμερες αποτυπώνεται σε τριάδες RDF και αποθηκεύεται στη Βάση Γνώσεως του συστήματος.
2. Οκτώ σημασιολογικοί κανόνες (τέσσερα ζεύγη) οι οποίοι ρυθμίζουν τις ενέργειες του μεσοσμικού. Το πρώτο ζεύγος κανόνων καθορίζει την μετάβαση από την αρχική κατάσταση στην κατάσταση έναρξης συνεδρίασης, το δεύτερο καθορίζει τη μετάβαση από την έναρξη συνεδρίασης στην έναρξη παρουσίασης, το τρίτο τη

μετάβαση στην κατάσταση λήξης παρουσίασης και το τέταρτο τη μετάβαση στη λήξη συνεδρίασης.

Ουσιαστικά, οι κανόνες αυτοί υλοποιούν το μοντέλο που απεικονίζεται στην εικόνα 22 και παρατίθενται στο παράρτημα Δ.

Η οντολογία η οποία υποστηρίζει την εφαρμογή είναι στοιχειώδης ως προς την πολυπλοκότητα που ενέχει. Μπορεί ωστόσο να συνδυαστεί με τους αλγόριθμους παρακολούθησης και αναγνώρισης προσώπου. Η βασική καινοτομία είναι η αυτοματοποίηση, η επεξεργασία πραγματικού χρόνου ενώ παράλληλα η δυνατότητα απάντησης ευφύστερων ερωτημάτων, βασισμένων όχι σε απλές λέξεις-κλειδιά. Η Βάση Γνώσεως που δημιουργείται είναι ικανή να απαντήσει σε ερωτήματα της μορφής:

- Πότε έκανε παρουσίαση ένα συγκεκριμένο πρόσωπο;
- Πότε έκανε τελευταία φορά παρουσίαση προσκεκλημένος;

5.3.3 Διαχωρισμός και Σύντηξη Μηνυμάτων

Προκειμένου να αποσαφηνιστεί η διαδικασία που ακολουθείται στην επεξεργασία των μηνυμάτων που εισέρχονται στο μεσισμικό, παρατίθεται στην ενότητα αυτή ένα λεπτομερές παράδειγμα στο οποίο αποσαφηνίζεται ο τρόπος με τον οποίο τα μηνύματα καταρχήν διαχωρίζονται ανάλογα με την πηγή προέλευσης του καθενός και στη συνέχεια συντήκονται ώστε να δημιουργηθεί η σημασιολογικά εμπλουτισμένη Βάση Γνώσεως. Έστω το παρακάτω παράδειγμα, κατά το οποίο ένας Body Tracker είναι σε λειτουργία στην οροφή ενός δωματίου κι ένας Face Tracker ο οποίος αντικρύζει την είσοδο του δωματίου. Ο διαχωρισμός των μηνυμάτων βασίζεται καταρχήν στη διαφοροποίηση των μηνυμάτων: ένα πρότυπο μηνύματος ανά tracker. Το πρότυπο μηνύματος του Body Tracker μπορεί να είναι της μορφής:

```
<Event id="5712">  
  <Tracker type="BodyTracker">  
  ...
```

Αντίστοιχα, το πρότυπο μηνύματος για τον Face Tracker μπορεί να είναι της μορφής:

```
<Event id="5713">  
  <Tracker type="FaceTracker">  
  ...
```

Τα μηνύματα αυτά αποστέλλονται στο μεσισμικό με τη χρήση Υπηρεσιών Διαδικτύου. Οι κανόνες αντιστοίχισης είναι υπεύθυνοι καταρχήν για το διαχωρισμό των μηνυμάτων αυτών. Επομένως, οι κανόνες αντιστοίχισης που εφαρμόζονται στα μηνύματα που προέρχονται από τον Body Tracker μπορεί να έχουν τη μορφή:

```
if,xml element has value,/Event/Tracker/@type,"BodyTracker",and,...
then,
insert individual in class,Person,
named after,/Event/Tracker/person/@id,
and set datatype property,hasTime,/Event/Tracker/TimeStamp/@value,...
```

Αντίστοιχα, οι κανόνες αντιστοίχισης που εφαρμόζονται στα μηνύματα που προέρχονται από τον Face Tracker μπορεί να έχουν τη μορφή:

```
if,xml element has value,/Event/Tracker/@type,"FaceTracker",and,...
then,
insert individual in class,Professor,
named after,/Event/Tracker/person/face/@name,
and set datatype property,hasTime,/Event/Tracker/@timestamp,...
```

Η εκτέλεση αυτών των κανόνων αντιστοίχισης θα οδηγήσει στην αντίστοιχη εισαγωγή τριάδων στην οντολογία ως εξής:

```
default:Person_1      rdf:type      default:Person ;
                      default:hasTime  "11:28:34"^^xsd:time .
```

και

```
default:Professor_John  rdf:type      default:Professor ;
                       default:hasTime  "11:28:34"^^xsd:time .
```

Στη συνέχεια, προκειμένου να συντηχθούν τα μηνύματα αυτά, οι Σημασιολογικοί κανόνες που εκτελούνται για το κάθε εισερχόμενο μήνυμα μπορούν να λάβουν υπόψη τα individuals τα οποία ανήκουν σε διαφορετικές κλάσεις. Ένα παράδειγμα Σημασιολογικού κανόνα που να συνδυάζει τα μηνύματα αυτά είναι το εξής:

```
if,SPARQL query has results,
SELECT ?x ?y WHERE {
    ?x rdf:type default:Professor .
    ?x :hasTime ?t1.
    ?y rdf:type default:Person .
```

```
?y hasTime ?t2
FILTER (?t1 = ?t2) }
then,
alert ("There is a Professor in the entrance of the room")
```

Στην περίπτωση αυτή, μια ειδοποίηση (alert) παράγεται όταν ένας καθηγητής εισέρχεται στο δωμάτιο. Η ειδοποίηση βασίζεται στα μηνύματα και των δυο trackers: του Body Tracker και του Face Tracker και παράγεται όταν και οι δυο trackers αναγνωρίσουν ένα σώμα (body) κι ένα πρόσωπο (face) αντίστοιχα, μειώνοντας την πιθανότητα λάθους. Επομένως, μπορεί να διαπιστωθεί πως οι κανόνες αντιστοίχισης είναι υπεύθυνοι για το διαχωρισμό των μηνυμάτων ενώ οι Σημασιολογικοί κανόνες είναι υπεύθυνοι για τη σύντηξη των μηνυμάτων.

Σημειώνεται ότι μια συνεργασία μεταξύ των trackers δεν είναι απαραίτητη. Στην περίπτωση αυτή δε χρειάζεται σύντηξη των μηνυμάτων. Στην εφαρμογή Υποστήριξης Συνεδριάσεων για παράδειγμα (ενότητα 5.3.2), ο Panoramic tracker ο οποίος χρειάζεται ώστε να παρακολουθεί την κατάσταση και την πορεία της συνεδρίασης μπορεί να συνδυαστεί με έναν Face Tracker ο οποίος να χαιρετά κάθε άτομο το οποίο εισέρχεται στο δωμάτιο. Σε μια τέτοια περίπτωση η σύντηξη των μηνυμάτων δεν είναι απαραίτητη χωρίς ωστόσο να μειώνεται η χρηστικότητα της εφαρμογής.

5.4 Μετρήσεις απόδοσης στην επεξεργασία πραγματικού χρόνου

Η απόδοση του προτεινόμενου συστήματος επηρεάζεται από τους εξής παράγοντες:

1. Το συνολικό αριθμό των εισερχομένων μηνυμάτων από τη στιγμή έναρξης λειτουργίας του συστήματος. Λόγω του ότι για την επεξεργασία και εισαγωγή ενός νέου μηνύματος στο σύστημα είναι πιθανό να επηρεάζεται μεγάλο τμήμα της υπάρχουσας Βάσης Γνώσεως, είναι πιθανό να αυξάνει ο χρόνος απόκρισης του συστήματος σε σχέση με τον αριθμό των μηνυμάτων που έχουν ήδη εισέλθει στο σύστημα.
2. Το ρυθμό προσέλευσης των μηνυμάτων, ο οποίος περιορίζεται από το χρόνο επεξεργασίας ενός μηνύματος.
3. Την πολυπλοκότητα στις έννοιες του οντολογικού μοντέλου κόσμου. Η αντίδραση του συστήματος είναι διαφορετική όσο οι έννοιες της οντολογίας αυξάνουν σε αλληλεξαρτήσεις όπως η κληρονομικότητα και οι μεταβατικές ιδιότητες.
4. Τον όγκο του μοντέλου κόσμου (της οντολογίας). Η προσθήκη ενός νέου μηνύματος στη Βάση Γνώσεως θα εκτελείται για περισσότερο χρόνο καθώς το

σχήμα της οντολογίας (TBox) μπορεί να κυμαίνεται από μερικές κλάσεις μέχρι αρκετές χιλιάδες κλάσεων.

5. Το σύνολο των κανόνων που εφαρμόζονται σε κάθε μήνυμα. Συγκεκριμένα, η απόδοση του συστήματος ελαττώνεται όσο αυξάνει ο αριθμός των κανόνων που επηρεάζουν τη Βάση Γνώσεως.
6. Την ετερογένεια των χρησιμοποιούμενων τεχνολογιών. Για παράδειγμα, μια διαφορετική Βάση Δεδομένων ή ένας διαφορετικός εξυπηρετητής υπηρεσιών συλλογιστικής μεταβάλλουν την απόδοση του συστήματος. Στην εικόνα 25(a) απεικονίζεται η διαφοροποίηση στην απόδοση με τη χρήση διαφορετικού εξυπηρετητή υπηρεσιών συλλογιστικής.

Ανάμεσα στα μεγέθη που μπορεί να μετρηθούν όσον αφορά την απόδοση του συστήματος, ιδιαίτερο ενδιαφέρον παρουσιάζει το χρονικό διάστημα που μεσολαβεί από την αποδοχή ενός μηνύματος από το σύστημα μέχρι την επεξεργασία του (χρόνος απόκρισης).

Όσον αφορά την ακρίβεια και την πληρότητα των αποτελεσμάτων (soundness and completeness), ένα σύστημα θεωρείται ολοκληρωμένο όταν αυτό είναι ταυτόχρονα ακριβές και πλήρες. Ένα σύστημα είναι ακριβές όταν κάθε απάντηση που επιστρέφεται είναι έγκυρη και πλήρες όταν μπορεί να επιστρέψει απαντήσεις για όλη την έκταση της γνώσεώς του. Δύσκολα όμως επιτυγχάνεται η δημιουργία ενός τέτοιου συστήματος, ειδικά στην περίπτωση του Σημασιολογικού Ιστού όπου είναι γνωστό ότι ακόμα και μια σχετικά μικρή OWL Lite οντολογία μπορεί να έχει υψηλή worst-case πολυπλοκότητα [161]. Χρειάζεται να λαμβάνονται υπόψη οι περιορισμοί που τίθενται σε παρόμοια συστήματα όσον αφορά την ακρίβεια και την πληρότητα (ενότητα 1.3) και να παρέχεται εξισορρόπηση κατά το ζητούμενο μεταξύ των δυο αυτών παραγόντων. Γενικά, ωστόσο, οι χρήστες προτιμούν μη πλήρη αποτελέσματα μέσα σε λογικό χρόνο παρά πλήρη αποτελέσματα σε μεγαλύτερο χρόνο [154].

Στην παρούσα περίπτωση, οι απαντήσεις που επιστρέφονται από το σύστημα είναι πλήρεις λόγω της χρήσης της SPARQL. Η ακρίβεια του συστήματος εξαρτάται από την ακρίβεια η οποία παρέχεται από τους αλγόριθμους εξαγωγής χαρακτηριστικών που ενσωματώνονται στους ανιχνευτές. Η ακρίβεια της γνώσης του μεσισμικού για τον εξωτερικό κόσμο, η οποία αποτυπώνεται στα μηνύματα που δέχεται, εξαρτάται από την ακρίβεια των αλγορίθμων αυτών.

Όσον αφορά την ακρίβεια στην επισημείωση των δεδομένων, χρειάζεται να υπογραμμιστεί ότι η συμπεριφορά του μεσισμικού – και κατ' επέκταση των εφαρμογών που βασίζονται σε αυτό – καθορίζεται από τους ανιχνευτές (trackers) που έχουν επιλεγεί. Οι trackers που έχουν επιλεγεί για τους σκοπούς των μετρήσεων έχουν περιγραφεί στις

εργασίες [179] και [180] όπου τα αποτελέσματά τους παρουσιάζονται διεξοδικότερα.

5.4.1 Πειραματικές μετρήσεις

Το μεσισμικό εξετάστηκε σχετικά με την κλιμακωσιμότητα που μπορεί να παράσχει. Οι μετρήσεις πραγματοποιήθηκαν σε υπολογιστή με λειτουργικό σύστημα Linux, επεξεργαστή με 2 πυρήνες Intel συχνότητας 2.0GHz ο καθένας και μνήμη RAM 3GB (ορατή από το λειτουργικό). Οι σχέσεις μεταξύ των πειραματικών αποτελεσμάτων ωστόσο είναι ανεξάρτητες της hardware υποδομής. Το πειραματικό περιβάλλον περιλαμβάνει τις εξής συνιστώσες: Μια κάμερα η οποία είναι συνδεδεμένη με υπολογιστή στον οποίο αποστέλλει τη ροή video που καταγράφει, έναν αλγόριθμο ανίχνευσης και έναν παρακολούθησης προσώπων ο οποίος εκτελείται στον υπολογιστή αυτόν, και το σύστημα του μεσισμικού.

Κατά τη διάρκεια της πραγματικής λειτουργίας του μεσισμικού, καταγράφηκε το αποτέλεσμα των αλγορίθμων και συγκεκριμένα τα αρχεία XML που παρήχθησαν όπως και οι εικόνες οι οποίες διαδοχικά αποτελούν το video που καταγράφηκε. Στη συνέχεια και προκειμένου να μελετηθούν και να βελτιωθούν τα πειραματικά αποτελέσματα, έγινε αναπαραγωγή τους, εξομοιώνοντας το ρόλο της κάμερας και αποστέλλοντας τα μηνύματα μέσω Υπηρεσιών Διαδικτύου στο μεσισμικό.

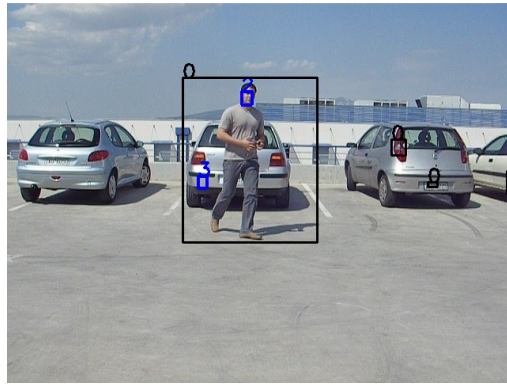
Αρχικά για τις μετρήσεις έγινε προσπάθεια επεξεργασίας σε videos που κατέγραψαν κινήσεις σε εξωτερικό χώρο (εικόνες 23(a), 23(b) και 23(c)). Τα αποτελέσματα όμως περιείχαν πολλή εσφαλμένη πληροφορία, λόγω των υλοποιήσεων των trackers. Επειδή σε εξωτερικά περιβάλλοντα υπάρχει κίνηση η οποία παραπλανά τους αλγόριθμους (κίνηση φύλλων λόγω ακόμη κι ανεπαίσθητου ανέμου, άτομα ή αυτοκίνητα στο πεδίο αντίληψης της κάμερας, φωτισμοί), το πόρισμα το οποίο εξάγεται συνιστά τη χρήση σε εσωτερικούς χώρους στους οποίους τα αποτελέσματα είναι πολύ καλύτερα λόγω του ελεγχόμενου περιβάλλοντος. Στην εικόνα 23(a) χαρακτηριστικά, ο ανιχνευτής προσώπων δεν ήταν σε θέση να διακρίνει το προσκίνηνο από το παρασκήνιο λόγω της κίνησης των φύλλων παρότι η κάμερα ήταν σταθερή.

Ακόμα και σε εσωτερικούς χώρους, οι αλγόριθμοι οι οποίοι χρησιμοποιήθηκαν στις πρώτες πειραματικές προσπάθειες εμφάνισαν πλήθος λαθών. Όπως φαίνεται στις εικόνες 23(e) και (f), μετά από αρκετή ώρα λειτουργίας ο αλγόριθμος εσφαλμένα αναγνωρίζει άτομα στο δωμάτιο, στην 23(f) μάλιστα έχει ήδη αναγνωρίσει 41 άτομα ενώ εμφανίζονται μόνο 2. Τα λάθη αυτά αποτέλεσαν την βάση των πειραμάτων καθώς η πληροφορία που αποτυπώνεται στα XML μηνύματα των trackers εξομοιώνει στην ουσία δραστήρια περιβάλλοντα με πλήθος ανθρώπων.

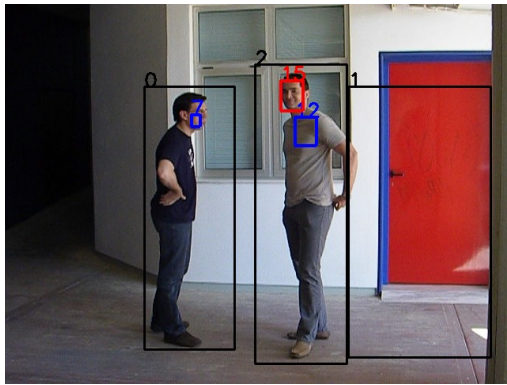
Αρχικά, υπήρξαν προβλήματα στην απόκριση του συστήματος λόγω ραγδαίας



(a)



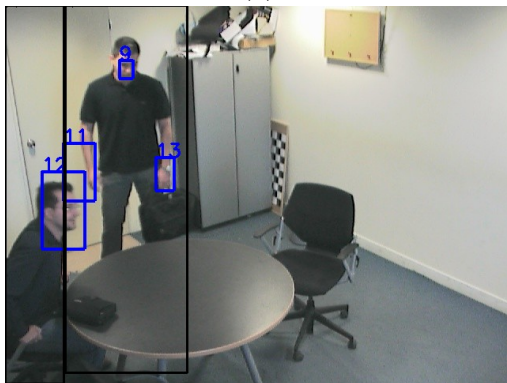
(b)



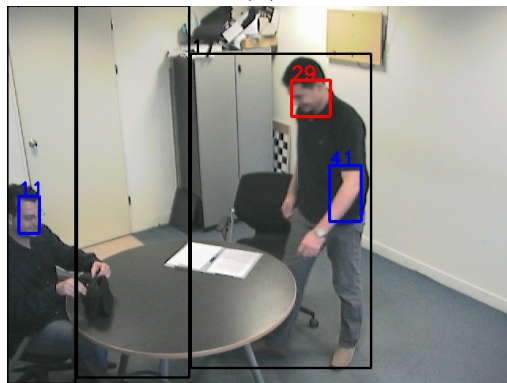
(c)



(d)

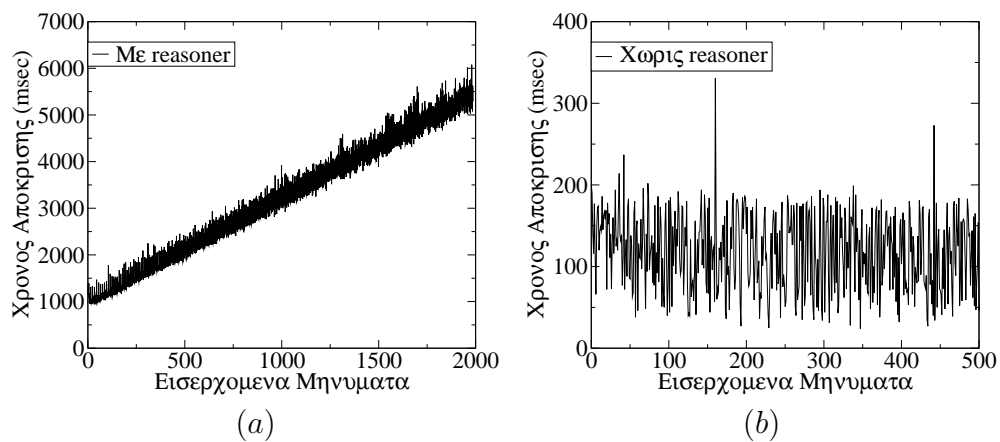


(e)



(f)

Εικόνα 23: Στιγμιότυπα πειραματικών δεδομένων



Εικόνα 24: Στην (a) φαίνεται η συμπεριφορά του μεσιμικού με υποστήριξη από υπηρεσίες συλλογιστικής. Στην (b) φαίνεται πως χωρίς τη χρήση υπηρεσιών συλλογιστικής ο χρόνος απόκρισης είναι αμελητέος και δεν εξαρτάται από το μέγεθος της Βάσης Γνώσεως

αύξησης του όγκου της Βάσης Γνώσεως. Έστω το απλό παράδειγμα το οποίο ουσιαστικά περιγράφει την εξής λειτουργικότητα: Για κάθε μήνυμα που λαμβάνει το μεσιμικό, ένα άτομο εισάγεται στην οντολογία. Τα αποτελέσματα παρόμοιας χρήσης εμφανίζονται στο γράφημα 24(a). Πολύ γρήγορα, το σύστημα καθίσταται μη αποδοτικό καθώς όπως φαίνεται, ο χρόνος επεξεργασίας κάθε μηνύματος γρήγορα αυξάνει πέρα από αποδεκτά όρια.

Αντίθετα, ο χρόνος απόκρισης στην επεξεργασία του κάθε εισερχόμενου μηνύματος, όταν αφαιρεθεί η υποστήριξη από υπηρεσίες συλλογιστικής, παραμένει σε πολύ χαμηλά επίπεδα. Όπως εμφανίζεται και στο γράφημα 24(b), ο χρόνος απόκρισης γίνεται μικρότερος από 300 msec και παραμένει σχεδόν σταθερός με μικρές διακυμάνσεις αφού δεν επηρεάζεται από το μέγεθος του μοντέλου της οντολογίας που είναι αποθηκευμένο στη Βάση Γνώσεως. Συγκεκριμένα, ο χρόνος που χρειάζεται για την εισαγωγή ενός νέου γεγονότος στη Βάση Γνώσεως δεν επηρεάζεται από το μέγεθος ή την πολυπλοκότητα του μοντέλου της οντολογίας που είναι αποθηκευμένη, καθώς η εισαγωγή αυτή γίνεται χωρίς να χρειάζεται να αναγνωσθεί το μοντέλο. Στην πράξη, κάθε εισερχόμενο μήνυμα προκαλεί την εκτέλεση ενός απλού SQL INSERT ερωτήματος. Η προσέγγιση αυτή ωστόσο δεν ενσωματώνει και δεν προσφέρει εκμετάλλευση σημασιολογικής πληροφορίας.

Το πόρισμα που προκύπτει από τις πρώτες μετρήσεις στο μεσιμικό είναι ότι, αφενός, χρειάζεται ιδιαίτερη προσοχή στη σύνθεση των κανόνων ώστε να μην αυξάνεται ραγδαία ο όγκος της Βάσης Γνώσεως και, αφετέρου, χρειάζεται ένα είδος ελεγχόμενης συντήρησης

στην απόδοση του μεσισμικού.

Επιπλέον, εφόσον η υποστήριξη από υπηρεσίες συλλογιστικής κρίνεται απαραίτητη, έγινε διερεύνηση σχετικά με την απόδοση των εξυπηρετητών υπηρεσιών συλλογιστικής (reasoners) προκειμένου να χρησιμοποιηθεί ο βέλτιστος. Συγκεκριμένα, εξετάστηκε η απόδοση του συστήματος με υποστήριξη από τον FaCT++ και τον Pellet (αναλυτικότερη παρουσίαση των εξυπηρετητών έγινε στην ενότητα 3.6). Με τον τρόπο αυτό προκύπτει το γράφημα της εικόνας 25(a) στο οποίο παρουσιάζεται η απόδοση των reasoners αυτών και φαίνεται ότι ο Pellet έχει ελαφρώς καλύτερη απόδοση. Ο Racer και ο RacerPro δεν εξετάζονται λόγω του περιορισμού που θέτουν στον όγκο της οντολογίας. Ο αλγόριθμος τους χρειάζεται να φορτώσει το σύνολο της οντολογίας στη μνήμη του υπολογιστή, περιορισμός που θέτει χαμηλό όριο στην επεκτασιμότητα της Βάσης Γνώσεως.

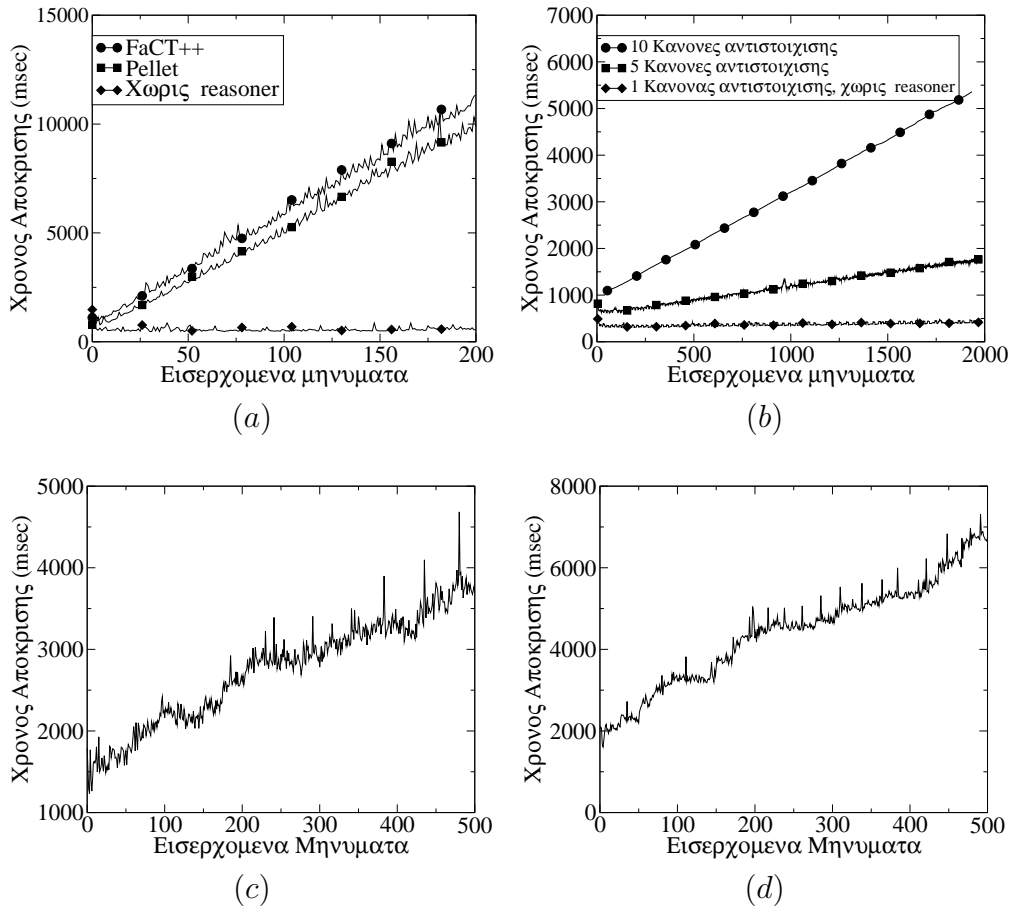
Στη συνέχεια, προκειμένου να διαπιστωθεί η συνάρτηση του χρόνου απόκρισης σε σχέση με το σύνολο των κανόνων που εφαρμόζεται σε κάθε μήνυμα, πραγματοποιήθηκαν οι μετρήσεις οι οποίες εμφανίζονται στο γράφημα 25(b) στην οποία το μεσισμικό εξετάζεται σε διαφορετικές συνθήκες φόρτου όσον αφορά τους κανόνες αντιστοίχισης. Οι κανόνες αυτοί σε κάθε περίπτωση και για κάθε εισερχόμενο μήνυμα εισάγουν 1, 5 και 10 άτομα στην οντολογία, αντίστοιχα.

Στα γραφήματα 25(c) και 25(d) απεικονίζεται ο χρόνος απόκρισης του μεσισμικού σε msec ανάλογα με τον αριθμό των εισερχόμενων μηνυμάτων στο μεσισμικό για τις δυο περιπτώσεις σεναρίων που αναλύθηκαν στην ενότητα 5.3. Όπως γίνεται αντιληπτό, ο στόχος της εφαρμογής εξυπηρετείται, πολύ σύντομα όμως τα συστήματα καθίστανται υπερβολικά αργά. Χρειάζεται επομένως ένα είδος βελτίωσης.

Η πρόταση για βελτίωση εστιάζει στην (σύγχρονη) επεξεργασία των μηνυμάτων με στόχο να διατηρείται ο χρόνος χαμηλότερα από κάποιο κατώφλι (threshold). Το επίπεδο αυτό μπορεί να ρυθμιστεί είτε εξετάζοντας το μέγεθος της οντολογίας και αφαιρώντας τις παλιότερες τριάδες, είτε εξετάζοντας το χρόνο επεξεργασίας ενός μηνύματος. Μια προσωρινή Βάση Δεδομένων διατηρείται, η οποία αναλαμβάνει την αποθήκευση και επεξεργασία των τελευταίων μηνυμάτων και μια μόνιμη η οποία εμπεριέχει όλο το ιστορικό των μηνυμάτων.

Ο χρόνος περιορισμού επιλέχθηκε αυθαίρετα, είναι όμως παραμετροποιήσιμος ανάλογα με τις ανάγκες της κάθε εφαρμογής. Αντίστοιχα, και το μέγεθος της οντολογίας είναι παραμετροποιήσιμο. Ο πρακτικός περιορισμός που προκύπτει από την προσέγγιση αυτή είναι το ότι δε μπορούν να δηλωθούν κανόνες οι οποίοι να λαμβάνουν υπόψη παλαιότερα μηνύματα.

Για την εφαρμογή παρακολούθησης χώρων, τα αποτελέσματα εμφανίζονται στα γραφήματα 26(a) και 26(b). Στο (a), το κατώφλι που τέθηκε για τον χρόνο επεξεργασίας



Εικόνα 25: Στις (a) και (b) παρουσιάζεται η απόδοση του μεσιμικού σε σχέση με τις υπηρεσίες συλλογιστικής και το σύνολο κανόνων, αντίστοιχα. Στις (c) και (d) παρουσιάζεται η απόδοση του μεσιμικού κατά τις εφαρμογές παρακολούθησης χώρων και υποστήριξης συνεδριάσεων, αντίστοιχα. Σε κάθε περίπτωση το σύστημα υποστηρίζεται από υπηρεσίες συλλογιστικής ενώ δε λαμβάνει χώρα κάποια διαχείριση του χρόνου επεξεργασίας των μηνυμάτων

του κάθε μηνύματος είναι στα 2” για το κάθε μήνυμα. Όπως εμφανίζεται, η buffer Βάση Γνώσεως μπορεί να μεταφερθεί στην μόνιμη Βάση όταν 50 άτομα περίπου έχουν κάνει την εμφάνισή τους. Παρόμοια αποτελέσματα εμφανίζονται και στο (b), όπου το κατώφλι αφορά τον αριθμό τριάδων της Βάσης Γνώσης οι οποίες περιορίζονται στις 180. Πρακτικά, το γεγονός αυτό σημαίνει ότι η μνήμη του συστήματος όσον αφορά τη λήψη αποφάσεων σε πραγματικό χρόνο είναι περιορισμένη στο κατώφλι το οποίο ορίστηκε.

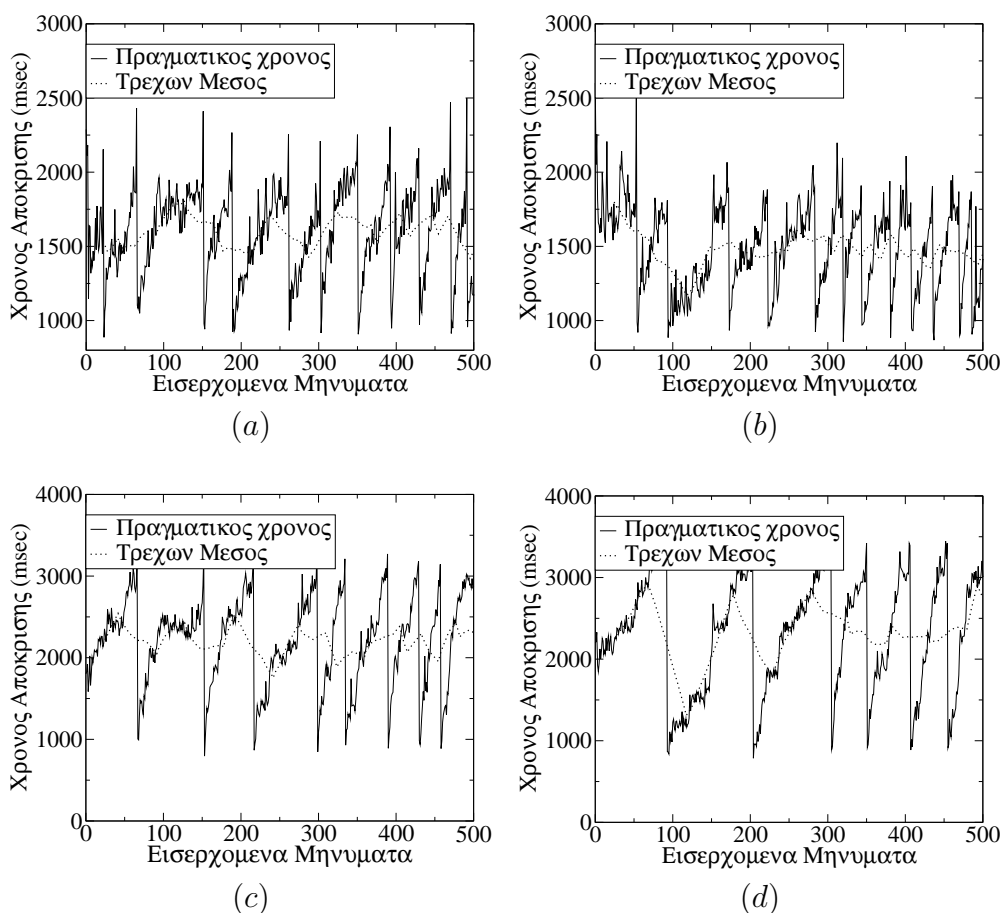
Αντίστοιχα, για την εφαρμογή της υποστήριξης συνεδριάσεων, τα αποτελέσματα παρουσιάζονται στα γραφήματα 26(c) και 26(d). Στο 26(c), το κατώφλι του χρόνου επεξεργασίας ορίζεται στα 3”, ενώ στο 26(d) ορίζεται κατώφλι στον αριθμό τριάδων σε 80. Η διαφορά στην απόδοση από το σενάριο παρακολούθησης χώρων οφείλεται στα μεγαλύτερα σύνολα κανόνων που εφαρμόζονται σε κάθε εισερχόμενο μήνυμα. Οι σημασιολογικοί κανόνες εδώ αυξάνονται, από τους 2 που υποστηρίζουν το σενάριο παρακολούθησης, σε 8 και παρατίθενται στο παράρτημα Δ.

Ο τρέχων μέσος στις μετρήσεις της εικόνας 26 αναφέρεται στο μέσο χρόνο επεξεργασίας ανά πάσα στιγμή των τελευταίων 50 μηνυμάτων. Ο στόχος της προσθήκης του τρέχοντος μέσου στις μετρήσεις είναι να αποσαφηνιστεί ότι παρά τις διακυμάνσεις στο χρόνο απόκρισης κατά τη λειτουργία του συστήματος, η συνολική συμπεριφορά παραμένει σχετικά σταθερή.

Χρειάζεται να αναφερθεί ότι στα πειράματα που εκτελέστηκαν με τη χρήση του εσωτερικού μηχανισμού κανόνων της Jena τα αποτελέσματα ήταν ιδιαίτερα αποθαρρυντικά σε σχέση με τα αποτελέσματα που προέκυψαν με τη χρήση εξωτερικού εξυπηρετητή υπηρεσιών συλλογιστικής και διαλειτουργικότητα DIG.

Ενδιαφέρον παρουσιάζει το να τονιστούν μερικοί από τους παράγοντες ο οποίοι μπορούν να βελτιώσουν την απόδοση του συστήματος. Καταρχήν, για τη βελτίωση της εκτέλεσης στα ερωτήματα ανάγνωσης από τη Βάση Δεδομένων, στην οποία διατηρεί το μοντέλο της οντολογίας η Jena, παρατηρείται ότι ο εσωτερικός μηχανισμός αποθήκευσης της Jena διατηρεί δείκτες (indices). Για πειραματικούς λόγους, έγινε απόπειρα αφαίρεσης των δεικτών αυτών με το σκεπτικό ότι η ενέργεια αυτή θα επιτάχυνε τις πολλαπλές εισαγωγές στη Βάση που πραγματοποιούνται κατά τη λειτουργία του μεσισμικού. Τα αποτελέσματα ωστόσο ήταν αρκετά αποθαρρυντικά καθώς το μοντέλο χρειάζεται να αναγνωσθεί από τη Βάση για κάθε εισερχόμενο μήνυμα.

Για κάθε εισερχόμενο μήνυμα έχει ενδιαφέρον να αναφερθεί ότι υπήρξε πρόβλημα ανανέωσης των ιδιοτήτων (properties) των ατόμων της οντολογίας. Λόγω της θεώρησης κλειστού κόσμου (ενότητα 4.1.5) ο επίσημος καθορισμός της γλώσσας ερωτημάτων (SPARQL, ενότητα 2.7.2) με την οποία γίνεται επεξεργασία του μοντέλου δεν επιτρέπει δηλώσεις τύπου DELETE ή UPDATE στον επίσημο ορισμό της [30]. Λόγω όμως

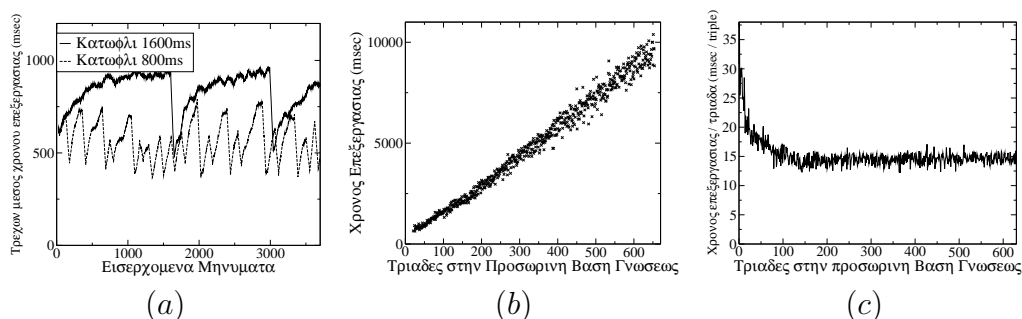


Εικόνα 26: Στις (a) και (b) παρουσιάζεται ο χρόνος απόκρισης της εφαρμογής παρακολούθησης χώρων, ενώ στις (c) και (d) ο χρόνος απόκρισης της εφαρμογής υποστήριξης συνεδριάσεων. Σε κάθε περίπτωση οι εφαρμογές υποστηρίζονται από υπηρεσίες συλλογιστικής και ο χρόνος απόκρισης διατηρείται σε συγκεκριμένο κατώφλι

της χρησιμότητας παρόμοιων εντολών, η λειτουργικότητα αυτή ενσωματώνεται στην προγραμματιστική διεπαφή Jena (από την έκδοση 2.5 και μετέπειτα) αλλά και στην ARQ, την υλοποίηση της SPARQL που αναπτύσσεται ανεξάρτητα αλλά χρησιμοποιείται από τη Jena. Έτσι γίνεται δυνατή η δήλωση εντολών "DELETE...WHERE...INSERT", όπως για παράδειγμα στον κανόνα 2b στο Παράρτημα Δ.

Με τον τρόπο αυτό αντιμετωπίζεται το φαινομενικά τετριμμένο πρόβλημα της ανανέωσης των ιδιοτήτων τύπου δεδομένων για ένα άτομο μιας κλάσης της οντολογίας. Συγκεκριμένα, στα παραδείγματα που προαναφέρθηκαν, για ένα άτομο που γίνεται αντιληπτό από τις κάμερες και κινείται στο χώρο οι συντεταγμένες του στο χώρο μεταβάλλονται διαρκώς. Επομένως οι ιδιότητες αυτές χρειάζεται να ανανεώνονται και

όχι να εισάγονται νέες για κάθε νέο στιγμιότυπο της θέσης του.



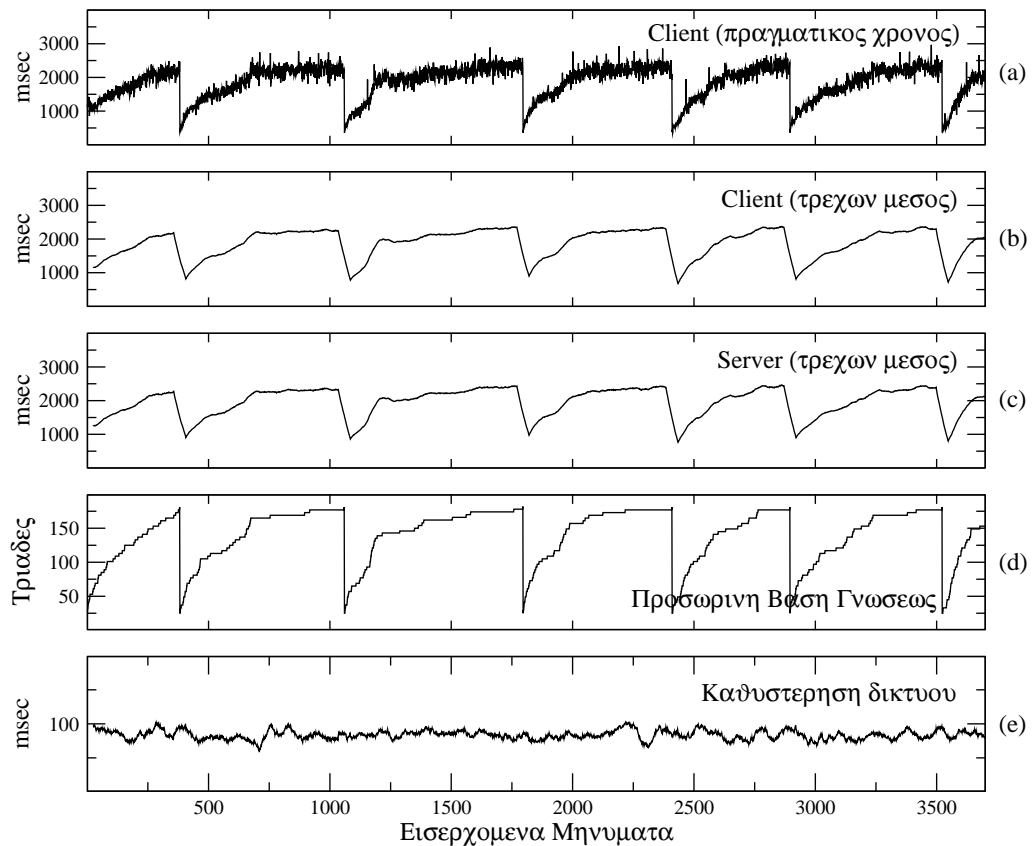
Εικόνα 27: Στην (a) παρατηρείται ότι μεγαλύτερα κατώφλια οδηγούν σε πιο συχνές διαδικασίες συντήρησης. Στην (b) φαίνεται το γεγονός ότι ο χρόνος επεξεργασίας του κάθε μηνύματος είναι ευθέως ανάλογος του αριθμού των τριάδων της προσωρινής Βάσης Γνώσεως. Στην (c) φαίνεται ότι ο χρόνος επεξεργασίας ανά τριάδα της προσωρινής Βάσης Γνώσεως είναι υψηλός για σχετικά μικρές οντολογίες, σταθεροποιείται όμως για μεγαλύτερες οντολογίες

Στην εικόνα 28 παρουσιάζεται η συμπεριφορά ενός μόνο πελάτη (client) – ο οποίος αντιστοιχεί σε έναν tracker – συνδεδεμένου στο μεσισμικό, ενώ το κατώφλι έχει οριστεί στις 160 τριάδες. Η εικόνα 28(a) αποτυπώνει τον πραγματικό χρόνο στον οποίο ο πελάτης λαμβάνει ειδοποίηση από το μεσισμικό ότι το μήνυμά του έχει υποστεί επεξεργασία και οι ενέργειες που το αφορούν έχουν ολοκληρωθεί.

Οι εικόνες 28(b) and 28(c) παρουσιάζουν τον τρέχοντα μέσο των τελευταίων 50 μηνυμάτων (όπως και στην εικόνα 26) με στόχο να τονιστεί η ομοιότητα στους χρόνους απόκρισης, εξομαλύνοντας τις τυχόν διακυμάνσεις. Η μοναδική καθυστέρηση (latency) η οποία οφείλεται στο δίκτυο εμφανίζεται στην εικόνα 28(e). Η καθυστέρηση αυτή είναι της τάξης των 100 msec και μπορεί να θεωρηθεί αμελητέα.

Χρειάζεται να τονιστεί ότι οι μετρήσεις συμφωνούν με την προηγούμενη παρατήρηση: ο χρόνος επεξεργασίας του κάθε μηνύματος είναι ευθέως ανάλογος του αριθμού των τριάδων της προσωρινής Βάσης Γνώσεως, όπως εμφανίζεται στην εικόνα 28(d).

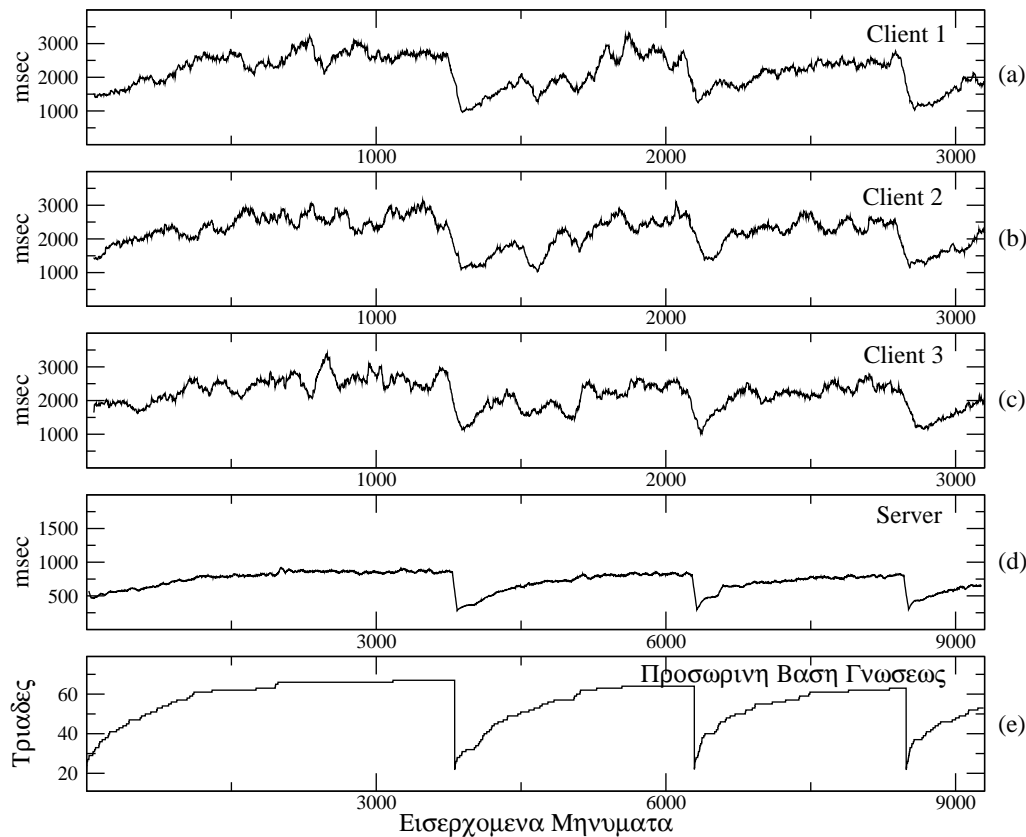
Η εικόνα 29 παρουσιάζει τρεις πελάτες σε παράλληλη λειτουργία πάνω στο ίδιο σύνολο δεδομένων. Κάθε πελάτης αποστέλλει 3000 μηνύματα στο μεσισμικό. Τα γραφήματα (a), (b) και (c) της εικόνας 29 παρουσιάζουν το χρόνο απόκρισης για τον κάθε πελάτη ενώ τα γραφήματα (d) και (e) παρουσιάζουν τη συμπεριφορά του εξυπηρετητή και τον αριθμό των τριάδων στην προσωρινή Βάση Γνώσεως. Ο εξυπηρετητής του μεσισμικού επεξεργάζεται 9000 μηνύματα, 3000 από κάθε πελάτη. Η συμπεριφορά του κάθε πελάτη παρατηρείται να είναι παρόμοια με αυτή του εξυπηρετητή. Ωστόσο, όπως



Εικόνα 28: Ένας πελάτης ο οποίος επεξεργάζεται offline δεδομένα. Παρατηρείται ότι η απόδοσή του είναι παρόμοια του εξυπηρετητή. Η απόδοση αυτή επηρεάζεται κυρίως από τον αριθμό των τριάδων στην προσωρινή Βάση Γνώσεως, ενώ η καθυστέρηση που οφείλεται σε άλλους παράγοντες μπορεί να θεωρηθεί αμελητέα.

ήταν αναμενόμενο, ο χρόνος που μεσολαβεί μέχρι ο κάθε πελάτης να λάβει ενημέρωση ότι το μήνυμά του έχει υποστεί επεξεργασία είναι μεγαλύτερος από το χρόνο που χρειάζεται ο εξυπηρετητής να επεξεργαστεί ένα μήνυμα. Το γεγονός αυτό οφείλεται στο συγχρονισμό των μηνυμάτων από τον εξυπηρετητή.

Ανάλογα πορίσματα προκύπτουν και από την εικόνα 30, στην οποία απεικονίζονται οι μετρήσεις στην απόδοση του συστήματος κατά την παράλληλη λειτουργία εννέα πελατών. Στην 30(a) απεικονίζεται ο χρόνος απόκρισης για έναν από τους εννέα πελάτες. Οι υπόλοιποι έχουν παραληφθεί καθώς τα αποτελέσματα που παρουσιάζουν είναι παρόμοια. Φαίνεται πως ο χρόνος απόκρισης αυτός είναι πολλαπλάσιος από τον χρόνο που χρειάζεται για την επεξεργασία των μηνυμάτων από τον εξυπηρετητή, όπως εμφανίζεται στην εικόνα 30(b). Σε κάθε περίπτωση ωστόσο, ο χρόνος αυτός εξαρτάται

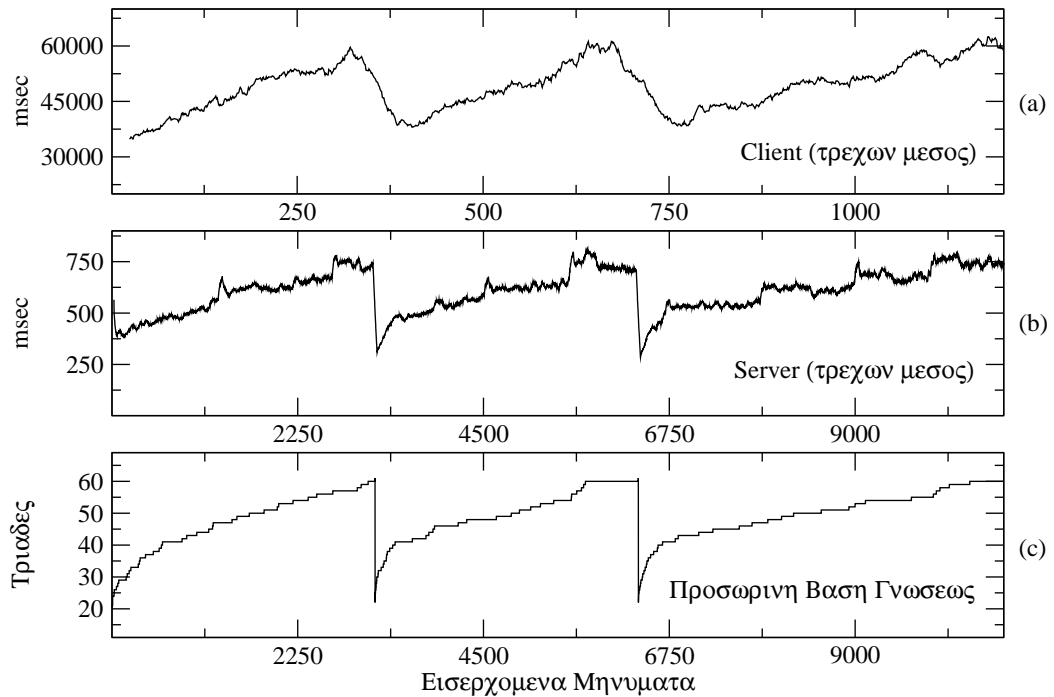


Εικόνα 29: Τρεις πελάτες σε παράλληλη επεξεργασία offline πληροφορίας

άμεσα από τον αριθμό των τριάδων που περιέχονται στην προσωρινή Βάση Γνώσεως του μεσισμικού (εικόνα 30(c)).

5.5 Αντιμετώπιση Σφαλμάτων και Ασυνεπειών

Σχετικά με τις πιθανές αποτυχίες που μπορεί να προκύψουν και να επηρεάσουν την απόδοση του μεσισμικού, μπορούμε να διακρίνουμε δυο περιπτώσεις. Καταρχήν, ένας ανιχνευτής (tracker) μπορεί να αποτύχει να δημιουργήσει ένα μήνυμα. Στην περίπτωση αυτή το μήνυμα δε θα φτάσει καν στο μεσισμικό. Στη δεύτερη περίπτωση ένας ανιχνευτής μπορεί να δημιουργήσει ένα εσφαλμένο μήνυμα (false positive) και να το αποστείλει στο μεσισμικό. Η δυνατότητα ωστόσο διόρθωσης του λάθους με σημασιολογικούς κανόνες διατηρείται. Για παράδειγμα, ένα πρόσωπο το οποίο εμφανίζεται και εξαφανίζεται ξαφνικά και δεν παρουσιάζει συνέχεια στην τροχιά του μπορεί να απορριφθεί με τη χρήση



Εικόνα 30: Εννέα πελάτες σε παράλληλη επεξεργασία offline πληροφορίας

σημασιολογικών κανόνων, μειώνοντας έτσι τον αριθμό των false positives.

Επιπλέον, με στόχο την αντιμετώπιση της ασυνέπειας που μπορεί να προκύψει μεταξύ μηνυμάτων από διαφορετικούς trackers, ένα κατώφλι βεβαιότητας μπορεί να τεθεί στη φάση εφαρμογής των κανόνων αντιστοίχισης. Επομένως, προκειμένου να ληφθεί υπόψη αυτό το κατώφλι βεβαιότητας, ο κανόνας που αναφέρεται για τον Face Tracker στην ενότητα 5.3.3, θα έχει την εξής μορφή:

```
if, xml element exists, /Event/Tracker/person, and
xml element has value, /Event/Tracker/person/face/@dbpersonid, eq, 10, and,
xml element has value, /Event/Tracker/person/face/@certainty, gt, 90,
then,
insert individual in class, Professor,
named after, /Event/Tracker/person/face/@name,
and set datatype property,
    hasXLocation, /Event/Tracker/person/location2d/@x,
and set datatype property,
    hasYLocation, /Event/Tracker/person/location2d/@y,
and set datatype property,
```

hasCertainty, /Event/Tracker/person/face/@certainty

...

Επίσης, στην περίπτωση που ένας ανιχνευτής παράγει υπερβολικά μεγάλο όγκο μηνυμάτων, τα γραφήματα της εικόνας 26 θα είναι πιο πυκνά αφού το μεσισμικό θα εκτελεί διαδικασίες συντήρησης πιο συχνά. Μια προσέγγιση για την αντιμετώπιση τέτοιου προβλήματος παρουσιάζεται στην εικόνα 27(a) όπου ένα υψηλότερο κατώφλι (σε τριάδες ή msec) οδηγεί σε λιγότερο συχνές διαδικασίες συντήρησης. Σε κάθε περίπτωση ωστόσο, η συμπεριφορά του μεσισμικού χαρακτηρίζεται από την έννοια του hard real-time (βλ. ενότητα 5.1.3): οι προθεσμίες τηρούνται πάντα.

5.6 Συμπεράσματα

Ένα βασικό συμπέρασμα του παρόντος κεφαλαίου είναι ότι η χρήση οντολογιών επιτρέπει τη μοντελοποίηση σεναρίων, συνόλων κανόνων και προτύπων μηνυμάτων. Το μεσισμικό παρουσιάζεται ως ικανή λύση στην αντιμετώπιση πλήθους προβλημάτων. Συγκεκριμένα, η αρχιτεκτονική που προτείνεται παρέχει την ευελιξία στην προσαρμογή σε μεγάλη ποικιλία από εφαρμογές. Επιπλέον, τα συστατικά του συστήματος είναι επαναχρησιμοποιήσιμα και υπακούουν σε καθορισμένα πρότυπα με αποτέλεσμα η παρούσα πρόταση να οδηγεί σε λύσεις που να μην αντιστοιχούν μόνο στην εκάστοτε περίπτωση αλλά να επιτρέπουν διαλειτουργικότητα με εφαρμογές από τρίτους, ολοκλήρωση σε ανώτερο επίπεδο και δυνατότητες επιπλέον αξιοποίησης της Βάσης Γνώσεως. Τα πειραματικά αποτελέσματα φανερώνουν ότι τα συστήματα που βασίζονται στην συμπερασματολογία (inference-based) μπορούν πλέον να χρησιμοποιηθούν για την επιτυχή υλοποίηση εφαρμογών όχι μόνο σε ερευνητικό επίπεδο αλλά και με απαιτήσεις πραγματικού κόσμου.

Ωστόσο, ορισμένα από τα ζητήματα που χρειάζεται να ληφθούν υπόψη κατά το σχεδιασμό εφαρμογών βασισμένων στο μεσισμικό είναι τα παρακάτω.

Για κάθε εισερχόμενο μήνυμα, χρειάζεται να αναγνωσθεί το μοντέλο της Βάσης Γνώσεως και να υποστηριχτεί η διαδικασία αυτή από τον εξυπηρετητή υπηρεσιών συλλογιστικής. Αυτό γίνεται προκειμένου το σύστημα να εκμεταλλεύεται σημασιολογική πληροφορία και να δίνει σωστές απαντήσεις, για παράδειγμα σχετικά με τον αριθμό των ατόμων μιας οντολογίας. Στην αντίθετη περίπτωση δεν ενσωματώνεται σημασιολογική πληροφορία και η όλη υλοποίηση θα μπορούσε να επιτευχθεί χωρίς τη χρήση τεχνολογιών Σημασιολογικού Ιστού και χωρίς την εκτέλεση αλγορίθμων εξαγωγής συμπερασματολογίας. Στην περίπτωση αυτή όμως δεν προστίθεται στο σύστημα επιπλέον γνώση στην ήδη υπάρχουσα και δεν γίνεται εκμετάλλευση δηλώσεων της μορφής `if`

class has individuals καθώς η ύπαρξη μελών (όπως και η *δυνατότητα* ύπαρξης μελών) σε μια κλάση είναι δυνατό να χρειάζεται *reasoner* ώστε να επαληθευτεί. Επιπλέον, τα άτομα της οντολογίας που θα επιστραφούν από ένα σχετικό ερώτημα θα είναι μόνο τα άτομα που έχουν δηλωθεί κατηγορηματικά (explicitly) ως άτομα και όχι όσα ανήκουν έμμεσα σε μια κλάση (ανήκουν για παράδειγμα σε υποκλάση αυτής ή ικανοποιούν τις συνθήκες υπαγωγής στην κλάση).

Οι κανόνες που έχουν εισαχθεί στο σύστημα υφίστανται επεξεργασία με τη σειρά, για κάθε εισερχόμενο μήνυμα. Η πρόκληση που χρειάζεται να αντιμετωπιστεί με τους κανόνες οι οποίοι δηλώνονται είναι ότι σε κάθε κανόνα δεν είναι δυνατόν να υπάρχουν δηλώσεις της μορφής: “Αν το προηγούμενο μήνυμα...” καθώς η ροή των μηνυμάτων είναι ασύγχρονη, προέρχεται από καταναμημένο σύστημα αισθητήρων και δε διασφαλίζεται απαραίτητα η χρονική τους συνέχεια. Η λογική τους χρειάζεται να είναι ανεξάρτητη των υπόλοιπων μηνυμάτων, δεν είναι δυνατή επομένως η δήλωση κανόνων όπως “Αν στο προηγούμενο μήνυμα ενεργοποιήθηκε ο κανόνας...” ή “Αν πρόσφατα ενεργοποιήθηκε ο κανόνας...”.

Η ιδιαιτερότητα αυτή καθιστά μη γραμμική τη διαδικασία σχεδιασμού μιας εφαρμογής βασισμένης στο μεσισμικό. Τα στάδια ανάπτυξης μιας εφαρμογής αφορούν – χωρίς κατ’ ανάγκη με αυτή τη σειρά – την ανάπτυξη της οντολογίας, των κανόνων, των προτύπων των μηνυμάτων και τον ορισμό των ανιχνευτών που θα ενσωματωθούν και, τέλος, των ενεργειών του μεσισμικού. Χρειάζεται παράλληλος σχεδιασμός όλων αυτών ούτως ώστε αφενός το σύστημα να καλύπτει τις ανάγκες του σεναρίου και αφετέρου να μην επηρεάζεται η απόδοσή του από περιττούς κανόνες, ή πολύ λεπτομερές μοντέλο οντολογίας.

Ο χρόνος απόκρισης είναι της τάξης των μερικών δευτερολέπτων, όπως φαίνεται και από τις μετρήσεις, γεγονός που αποδίδεται στη μη ωρίμανση των σχετικών τεχνολογιών, κάτι που περιορίζει τη χρήση μεσισμικού σε εφαρμογές που δε χρειάζονται άμεση απόκριση – σε χρόνο δηλαδή μικρότερο του ενός δευτερολέπτου όπως θα ήταν επιθυμητό για μια μηχανή αναζήτησης του Διαδικτύου. Το μεσισμικό μπορεί να χρησιμοποιηθεί όμως αποδοτικά σε εφαρμογές υποστήριξης συνεδριάσεων ή παρακολούθησης χώρων ή παρακολούθησης περιβάλλοντος όπου χρόνοι όπως μερικά δευτερόλεπτα είναι γενικότερα αποδεκτοί προκειμένου να εντοπιστεί ένα γεγονός.

Χρειάζεται να τονιστεί ότι στην περίπτωση της επεξεργασίας πραγματικού χρόνου, οι μετρήσεις δεν αφορούν τα ίδια στοιχεία που αφορούν την περίπτωση αντιστοίχισης Βάσης Δεδομένων με οντολογία. Σε συστήματα παρακολούθησης, χρόνοι όπως μερικά δευτερόλεπτα είναι γενικότερα αποδεκτοί προκειμένου να εντοπιστεί ένα γεγονός. Αντίθετα, σε αναζήτηση στο Διαδίκτυο, ο χρόνος χρειάζεται να είναι κάτω του ενός

δευτερολέπτου. Επιπλέον, σε κάθε περίπτωση η επιθυμητή συμπεριφορά της αναζήτησης διαφοροποιείται όσον αφορά την ακρίβεια και την πληρότητα. Ένα σύστημα ασφάλειας χρειάζεται να παρέχει έγκυρα και πλήρη (sound and complete) αποτελέσματα ενώ σε μια μηχανή αναζήτησης, μια έγκυρη αλλά όχι πλήρης προσέγγιση ικανοποιεί τους περισσότερους χρήστες [154].

Ο αριθμός των τριάδων στο προσωρινό μοντέλο της οντολογίας επηρεάζει άμεσα το χρόνο επεξεργασίας ενός μηνύματος από το μεσισμικό. Τα δυο αυτά μεγέθη είναι ευθέως ανάλογα όπως φαίνεται στην εικόνα 27(b). Επομένως, το κατώφλι μπορεί να τεθεί αυθαίρετα είτε σύμφωνα με το μέγιστο χρόνο που απαιτείται είτε σύμφωνα με το μέγιστο αριθμό τριάδων. Είναι προτιμητέος ωστόσο ο ορισμός του κατωφλιού σύμφωνα με τον αριθμό των τριάδων καθώς όπως φαίνεται στην εικόνα 28(a), ο χρόνος επεξεργασίας μπορεί να εμφανίσει απρόβλεπτες διακυμάνσεις (peaks) και υψηλή αστάθεια. Για παράδειγμα, στην εικόνα 27(a), ενώ ο μέσος χρόνος επεξεργασίας διατηρείται σχετικά χαμηλός, μια εργασία συντήρησης πυροδοτείται από την απλή καθυστέρηση στην επεξεργασία ενός μηνύματος.

Υψηλότερα κατώφλια οδηγούν σε μεγαλύτερα διαστήματα μεταξύ διαδοχικών διαδικασιών συντήρησης όπως επίσης και σε χειρότερη απόδοση, όπως φαίνεται στην εικόνα 27(a). Επιπλέον, υπάρχει σε κάθε περίπτωση ένα ελάχιστο όριο στο κατώφλι όπως γίνεται αντιληπτό από την εικόνα 27(c): Ένα χαμηλό κατώφλι δεν ευνοεί την απόδοση σε ταχύτητα του συστήματος καθώς εμφανίζεται υψηλός χρόνος επεξεργασίας ανά τριάδα του μοντέλου της προσωρινής οντολογίας.

Στην περίπτωση όπου μόνο ένας client είναι σε λειτουργία (εικόνα 28), η πιο σημαντική ιδιότητα προς μέτρηση είναι ο αριθμός των τριάδων στην οντολογία καθώς η επιπρόσθετη καθυστέρηση που οφείλεται στο δίκτυο ή σε άλλους παράγοντες μπορεί να θεωρηθεί αμελητέα. Στην περίπτωση περισσότερων clients σε παράλληλη λειτουργία, ο χρόνος που απαιτείται έως ώστε ο καθένας από αυτούς να λάβει απάντηση ανεβαίνει ανάλογα με τον αριθμό τους (εικόνες 29 και 30).

Σημειώνεται, τέλος, ότι η χρησιμότητα του συστήματος δεν ακυρώνεται στην περίπτωση που δεν υπάρχουν διαθέσιμοι αλγόριθμοι επεξεργασίας πραγματικού χρόνου. Ακόμη και σε μη πραγματικό χρόνο (offline), ένας tracker μπορεί να αποστείλει μηνύματα στο μεσισμικό επεξεργαζόμενος για παράδειγμα μια offline πηγή πολυμεσικών δεδομένων. Στην περίπτωση αυτή, η πληροφορία θα εξακολουθήσει να επισημαίνεται με τη διαφορά ότι δε θα έχει πλέον νόημα η δημιουργία ειδοποιήσεων (alerts) από το μεσισμικό.

5.7 Μελλοντική εργασία - προκλήσεις

Καταρχήν, μπορεί να εξεταστεί η ενσωμάτωση στο σύστημα διάφορων ειδών αισθητήρων οι οποίοι είναι ευρέως διαθέσιμοι όπως αισθητήρες Bluetooth, GPS ή RFID καθώς η υλοποιημένη υποδομή ήδη είναι σε θέση να υποστηρίξει εισερχόμενη πληροφορία από οποιοδήποτε αισθητήρα μπορεί να παρέχει Υπηρεσίες Διαδικτύου. Εκεί παρουσιάζει ενδιαφέρον η ολοκλήρωση με διαφορετικού τύπου αισθητήρες και επικοινωνία με προτυποποιημένες γλώσσες, όπως για παράδειγμα η Sensor Model Language (sensorML) ή η Observations & Measurements (O&M) που προτείνονται από το Open Geospatial Consortium⁷¹.

Επίσης πεδίο για έρευνα προσφέρεται στην υποστήριξη Σημασιολογικών Υπηρεσιών Διαδικτύου (ενότητα 3.7.3). Η σημασιολογική περιγραφή των υπηρεσιών μέσω των διεπαφών που προσφέρει το μεσισμικό – για παράδειγμα με το SWSF – μπορεί να οδηγήσει στην ανάπτυξη δικτυακών εφαρμογών με πλήρη σημασιολογική περιγραφή της λειτουργικότητας σε κάθε επίπεδο επεξεργασίας της πληροφορίας.

Οι κανόνες αποθηκεύονται στη Βάση Δεδομένων με τρόπο ο οποίος αναπτύχθηκε για να υποστηρίξει την προτεινόμενη προσέγγιση και που δεν υπακούει σε διεθνώς αναγνωρισμένα πρότυπα. Είναι αναγκαία η μελέτη σε τεχνικό και θεωρητικό επίπεδο για την ενσωμάτωση κανόνων που να υπακούουν σε ευρύτερα αναγνωρισμένα πρότυπα (ενότητα 2.6).

Επίσης, είναι δυνατή η επιπλέον βελτίωση των αποδόσεων μέσω της ενσωμάτωσης τεχνικών βελτίωσης Βάσεων Δεδομένων όπως η συστοιχία Βάσεων (clustering) ή η προσθήκη ευρετηρίων (indexing) των συχνά χρησιμοποιούμενων πεδίων των πινάκων.

Η αναζήτηση του χρήστη μπορεί να επεκταθεί ώστε να ενσωματώνει τυπικές προσεγγίσεις για αναζήτηση πλήρους κειμένου (Full-Text Search). Όπως έχει τονιστεί, η υιοθέτηση τεχνολογιών του Σημασιολογικού Ιστού χρειάζεται να διατηρεί τις δυνατότητες των συμβατικών προσεγγίσεων και να τις συμπληρώνει χωρίς να αφαιρεί τμήμα δυνατοτήτων, ειδικά μειώνονται οι λόγοι για την ευρύτερη καθιέρωση των τεχνολογιών αυτών.

Η χρήση μικροπροτύπων μπορεί να συμπεριληφθεί ούτως ώστε οι οντότητες της οντολογίας να περιγράφονται με τη χρήση ευρύτερα διαδεδομένων μικροπροτύπων (ενότητα 2.5). Ιδιαίτερα, η χρονική συνιστώσα θα μπορούσε να περιγράφεται με τη χρήση οντολογιών ειδικά ανεπτυγμένες για την περιγραφή του χρόνου. Επίσης, η χρήση μικροπροτύπων όπως το FOAF ή η DBPedia μπορεί να απελευθερώσει αρκετές δυνατότητες για την εξαγωγή στο Διαδίκτυο της Βάσης Γνώσεως που δημιουργείται.

⁷¹OpenGIS Specifications: www.opengeospatial.org/standards/is

Με τη χρήση κανόνων σε σημασιολογικό επίπεδο μπορεί να απορρίπτονται μηνύματα τα οποία δεν παρουσιάζουν συνέπεια στην περιγραφή τους. Για παράδειγμα, μια εσφαλμένη τιμή στα εισερχόμενα μηνύματα (false alarm) μπορεί να αποκοπεί και να μην εισαχθεί στη Βάση Γνώσεως. Όταν για παράδειγμα ένας αλγόριθμος παρακολούθησης προσώπων εσφαλμένα δώσει πληροφορία για τις συντεταγμένες του προσώπου με τρόπο ώστε αυτές να μην παρουσιάζουν συνέχεια, το μήνυμα που περιέχει την πληροφορία αυτή μπορεί ασφαλώς να απορριφθεί.

Επίσης, μια πιο ευέλικτη μηχανή κανόνων θα μπορούσε να υποστηρίξει ιεραρχία κανόνων. Στην παρούσα υλοποίηση δεν υπάρχει ιεραρχία, με την έννοια ότι σε κάθε εισερχόμενο μήνυμα στο μεσισμικό εφαρμόζεται το σύνολο των κανόνων αντιστοίχισης και των σημασιολογικών κανόνων. Ωστόσο, όσον αφορά τους κανόνες αντιστοίχισης, μπορεί να επιτευχθεί μια ιεραρχία με την προσθήκη επιπλέον συνθηκών στην αρχή του κάθε κανόνα. Στην περίπτωση αυτή το υπόλοιπο του κάθε κανόνα δε θα λαμβάνεται υπόψη. Στους σημασιολογικούς κανόνες όμως, ακόμα κι ένα φαινομενικά απλό ερώτημα όπως το “if, class has individuals,...”, μπορεί να είναι ακριβό στην αποτίμησή του. Επομένως, μια ιεραρχία στους σημασιολογικούς κανόνες θα μπορούσε να παρουσιάσει βελτιωμένα αποτελέσματα σε σχέση με αυτά της εικόνας 25(b). Σε κάθε περίπτωση ωστόσο, η σημαντικότερη ιδιότητα προς μέτρηση είναι ο αριθμός των τριάδων στην προσωρινή Βάση Γνώσεως (εικόνες 27(b), 28, 29 και 30).

Μπορούν να μελετηθούν δυνατότητες περαιτέρω βελτίωσης απόδοσης, όπως για παράδειγμα με δειγματοληψία στο επίπεδο του μεσισμικού. Σε μεγέθη που παρουσιάζουν συνέχεια, όπως για παράδειγμα η κίνηση ενός ανθρώπου σε ένα δωμάτιο, δεν είναι κρίσιμη η γνώση για τις συντεταγμένες του ανθρώπου ανά πάσα στιγμή. Εδώ θα μπορούσε να εφαρμοστεί δειγματοληψία, τόσο κατά την εισροή των μηνυμάτων στο μεσισμικό όσο και εκ των υστέρων, προκειμένου να μη διατηρείται περιττή πληροφορία.

Με παρόμοιο σκεπτικό, μπορεί επίσης να ρυθμιστεί ο ρυθμός δειγματοληψίας των μηνυμάτων σε συνδυασμό με προσωρινή αποθήκευση (caching). Αν για παράδειγμα μια κάμερα στέλνει μηνύματα με ρυθμό με τον οποίο δε μπορεί το μεσισμικό να συμβαδίσει, είναι δυνατό να μειωθεί ο ρυθμός επεξεργασίας και να αποθηκεύονται όλα τα μηνύματα στη Βάση Γνώσεως που δημιουργείται με στόχο όμως την επεξεργασία τους αργότερα. Με τον τρόπο αυτό το μεσισμικό μπορεί να προσαρμόζεται σε πηγές δεδομένων που μπορεί να παράγουν μηνύματα με μεγαλύτερο ρυθμό από ότι μπορεί να επεξεργαστεί το σύστημα.

Επιπλέον, μετρήσεις στο χρόνο απάντησης σε σημασιολογικά ερωτήματα σε σχέση με τον όγκο της Βάσης Γνώσεως θα παρουσίαζε ενδιαφέρον.

5.7.1 Ασφάλεια

Ιδιαίτερο ενδιαφέρον παρουσιάζει η προσθήκη ασφάλειας στο σύστημα, ειδικότερα σε ευαίσθητα περιβάλλοντα από αυτήν την άποψη. Στις περιπτώσεις αυτές, είναι υποχρεωτική η ύπαρξη δυνατοτήτων κρυπτογράφησης των δικτυακών δεδομένων. Έτσι αποτρέπεται η περίπτωση χρήσης ευαίσθητων οπτικοακουστικών δεδομένων κατά τη μεταφορά τους από εξωτερικές προς το σύστημα οντότητες. Είναι σημαντικό για τις περιπτώσεις αυτές το σύστημα να παρέχει ασφάλεια τόσο σε επίπεδο δικτύου όσο και σε επίπεδο πληροφοριών.

Ο τρόπος ασφάλισης των δεδομένων στο επίπεδο δικτύου ποικίλλει ανάλογα με το είδος της δικτυακής σύνδεσης. Έτσι, η προστασία μιας ροής δεδομένων από μια κάμερα ή ένα μικρόφωνο προς το σύστημα επεξεργασίας είναι διαφορετική από την προστασία των XML μηνυμάτων που ανταλλάσσονται μεταξύ του συστήματος εξαγωγής χαρακτηριστικών και μεσισμικού. Στις περισσότερες περιπτώσεις πολυμεσικών ροών, ενώ μια ελαφριά, συμμετρική κρυπτογράφηση θα ήταν ιδανική, αυτή δεν υποστηρίζεται από τον αισθητήρα. Πρέπει να ληφθεί ειδική μνεία, ώστε οι συνδέσεις αισθητήρων εσωτερικά στο σύστημα συλλογής πληροφοριών να μην μπορούν να χρησιμοποιηθούν από εξωτερικές προς το σύστημα οντότητες. Από το σημείο που η πληροφορία βρίσκεται πλέον στο σύστημα επεξεργασίας, μπορεί να οριστεί οποιαδήποτε μορφή κρυπτογράφησης θεωρείται αναγκαία για την εφαρμογή.

Πέρα από την ασφάλεια των δεδομένων σε επίπεδο δικτύου, είναι απαραίτητη η δυνατότητα εξακρίβωσης της αυθεντικότητας (authenticity) των πληροφοριών. Ιδιαίτερα σε κρίσιμα σενάρια εφαρμογής, είναι μεγάλος ο κίνδυνος κακόβουλης εισαγωγής ψευδών στοιχείων στο σύστημα, με σκοπό για παράδειγμα την απόκρυψη κάποιου γεγονότος ή την ακύρωση κάποιας δράσης του συστήματος. Τέτοιες περιπτώσεις αντιμετωπίζονται με τη χρήση ψηφιακών υπογραφών. Υπογραφές στο μεσισμικό έχουν νόημα μόνο στα μηνύματα XML που μεταφέρουν τη σημασιολογική πληροφορία. Υπάρχει επίσης η δυνατότητα υπογραφής των μηνυμάτων της διεπαφής του τελικού χρήστη. Το πρότυπο XML προβλέπει συγκεκριμένο τρόπο ψηφιακής υπογραφής μηνυμάτων, προτεινόμενο από το W3C⁷². Σύμφωνα με το πρότυπο αυτό, το οποίο καθορίζει το συντακτικό και τους αλγορίθμους κρυπτογράφησης XML δεδομένων, η βασική ιδέα στην οποία εδράζεται η κρυπτογραφία παραμένει, αλλάζοντας μόνο ως προς την υιοθέτηση μιας πρότυπης μορφής για αναπαράσταση και ανταλλαγή κρυπτογραφημένων XML δεδομένων. Η πρότυπη αυτή μορφή περιλαμβάνει ένα πρότυπο συντακτικό για αναπαράσταση του κρυπτογραφημένου περιεχομένου μέσα στο XML έγγραφο, καθώς και την απαιτούμενη πληροφορία για

⁷²XML Signature Syntax and Processing: www.w3.org/TR/xmlsig-core/

αποκρυπτογράφηση του περιεχομένου αυτό στο άκρο του παραλήπτη.

5.7.2 Δίκτυα αισθητήρων

Τα παραδείγματα που δόθηκαν στις ενότητες 5.3.1 και 5.3.2 μπορούν να επεκταθούν ώστε να συνδυαστούν και να προσαρμοστούν με τις ιδιαιτερότητες που εμφανίζονται σε δίκτυα μικροαισθητήρων. Η προσθήκη σημασιολογίας σε δίκτυα αισθητήρων μπορεί να χρησιμοποιηθεί για την αξιοποίηση των πρωτογενών δεδομένων των αισθητήρων.

Ένα δίκτυο αισθητήρων μπορεί να περιέχει αισθητήρες οι οποίοι μπορεί να καταγράφουν και να παρακολουθούν περιβαλλοντικά μεγέθη όπως θερμοκρασία, υγρασία, επιτάχυνση, πίεση ή και μεγέθη που αφορούν την υγεία ενός ατόμου όπως οι παλμοί της καρδιάς. Μπορεί να είναι ασύρματοι (wireless) αισθητήρες οι οποίοι γενικότερα αναφέρονται σε προϊόντα κινητής τηλεφωνίας με ευφυή χαρακτηριστικά και PDAs με ενσωματωμένους αισθητήρες τοποθεσίας και διεπαφές Bluetooth, GPS ή Wi-Fi. Εφαρμογές για τα κινητά αυτά μπορούν να γραφούν σε J2ME, Symbian C++ και C#.NET. Ως κανάλι επικοινωνίας μπορεί να επιλεγεί το Bluetooth, το WAP, το GPRS ή το Wi-Fi. Ένα δίκτυο αισθητήρων μπορεί επίσης να αποτελείται από audiovisual αισθητήρες και συγκεκριμένα, κάμερες ή μικρόφωνα.

Η πρόκληση στα συστήματα αυτά εντοπίζεται στην αποφυγή συμφόρησης στο δίκτυο και ειδικότερα (όταν οι αισθητήρες δεν είναι απλά παθητικοί και η αλληλεπίδραση είναι επιθυμητή) την χαμηλή κατανάλωση ενέργειας.

Τα δεδομένα από αυτούς τους αισθητήρες μπορούν να συναθροιστούν σε τεχνικό επίπεδο με βάση τεχνολογίες της TCP/IP στοίβας πρωτοκόλλων όπως τα Java/C sockets, οι http(s) συνδέσεις, οι Bluetooth serial drivers ή και Web Services. Πολλά πρωτόκολλα και σχήματα δρομολόγησης έχουν προταθεί για την συνάθροιση των δεδομένων αυτών, όπως για παράδειγμα η GSN⁷³, μια προσέγγιση η οποία παρέχει λογισμικό για τη συνάθροιση των δεδομένων αυτών σε συντακτικό επίπεδο.

Επιπλέον, για τη μελέτη σε θεωρητικό επίπεδο της συμπεριφοράς του δικτύου των αισθητήρων μέσω εξομίωσης παρέχεται ένα πλήθος εργαλείων όπως το ns-3⁷⁴, το OMNeT++⁷⁵, το NetTopo⁷⁶ ή το TOSSIM [181].

⁷³GSN, Middleware for Sensor Networks: gsn.sourceforge.net

⁷⁴The ns-3 network simulator: www.nsnam.org

⁷⁵OMNeT++simulation environment: www.omnetpp.org

⁷⁶NetTopo Simulator & Visualiser: www.semanticreality.org

6 Συμπεράσματα - Μελλοντική έρευνα

Στο κεφάλαιο αυτό γίνεται μια σύντομη σύνοψη της διατριβής και δίνονται τα σημαντικότερα συμπεράσματα που προκύπτουν από αυτήν. Επισημαίνονται στη συνέχεια τα κυριότερα καινοτομικά της στοιχεία και δίνονται, τέλος, κατευθύνσεις των οποίων η περαιτέρω διερεύνηση παρουσιάζει ενδιαφέρον.

6.1 Σύνοψη της διατριβής και συμπεράσματα

Στο κεφάλαιο 1 δίνεται η θεωρητική υποδομή των εννοιών που απασχολούν το αντικείμενο της διατριβής. Με βάση τις έννοιες αυτές θεμελιώνεται το *θεωρητικό* υπόβαθρο του Σημαιολογικού Ιστού πάνω στο οποίο βασίζεται η παρούσα έρευνα.

Στο κεφάλαιο 2 παρουσιάζεται το πώς οι ιδέες αυτές υλοποιούνται στην πράξη. Συγκεκριμένα, παρουσιάζεται το σύνολο των γλωσσών που έχουν αναπτυχθεί στα πλαίσια της έρευνας στο Σημαιολογικό Ιστό. Γίνεται αναφορά στις γλώσσες περιγραφής για τη σημασιολογική πληροφορία, στις γλώσσες σύνταξης κανόνων, στα μικροπρότυπα και τα λεξικά για την περιγραφή εννοιών του Σημαιολογικού Ιστού καθώς και στις γλώσσες ερωτημάτων. Το σύνολο των γλωσσών αυτών παρέχει το *τεχνολογικό* υπόβαθρο πάνω στο οποίο βασίζεται η σημερινή υλοποίηση των υπάρχοντων συστημάτων επεξεργασίας και αξιοποίησης της σημασιολογικής πληροφορίας.

Στο κεφάλαιο 3 συγκεντρώνεται η μελέτη της βιβλιογραφίας στο ευρύτερο πλαίσιο του Σημαιολογικού Ιστού ενώ παράλληλα διερευνάται η υιοθέτηση της τεχνολογίας αυτής σε συναφείς τομείς, όπως στα πολυμεσικά δεδομένα, τις Υπηρεσίες Διαδικτύου και τις μηχανές αναζήτησης. Το συμπέρασμα που προκύπτει από την μελέτη του χώρου είναι ότι πρόκειται για ερευνητικό τομέα με πλήθος δυνατοτήτων, ο οποίος διανύει πλέον το στάδιο της ωρίμανσής του. Υπάρχουν ολοκληρωμένα συστήματα σε ερευνητικό επίπεδο αλλά και εμπορικές μορφές προϊόντων τα οποία καλύπτουν ανάγκες πραγματικού κόσμου προσθέτοντας επιπλέον ευφυΐα σε σχέση με τα αντίστοιχα συμβατικά συστήματα. Η αδυναμία τους εντοπίζεται ωστόσο στην αξιοποίηση της πληροφορίας που παράγεται, γεγονός που διαπιστώνεται από την περιορισμένη εξάπλωση του Σημαιολογικού Ιστού στο Διαδίκτυο. Η πορεία των εξελίξεων διαφαίνεται πως οδηγείται στην προσθήκη αξιοπιστίας στη διαθέσιμη πληροφορία μέσω της δημοσιοποίησης διασυνδεδεμένων δεδομένων τα οποία θα στηρίζονται σε online Βάσεις Γνώσεως. Τα συμπεράσματα του κεφαλαίου αναλύονται περισσότερο στην ενότητα 3.8.

Ιδιαίτερο ενδιαφέρον υπάρχει στο πρόβλημα της συνεργασίας του Σημαιολογικού Ιστού με τις σαφώς πιο προηγμένες τεχνολογίες του χώρου των Βάσεων Δεδομένων. Το πρόσφατο αυτό πρόβλημα εξετάζεται διεξοδικά στο κεφάλαιο 4, όπου παρουσιάζονται

οι δυσκολίες που προκύπτουν και οι λύσεις που προτείνονται στη βιβλιογραφία. Στο κεφάλαιο αυτό αποτυπώνεται μια εκτενής επισκόπηση της βιβλιογραφίας, μια ταξινόμηση των προσεγγίσεων και μια πρωτότυπη λύση στο πρόβλημα (ενότητα 4.3). Το κεφάλαιο καταλήγει με τις παρατηρήσεις και τα συμπεράσματα που προκύπτουν καθώς και μια σειρά από προτάσεις (ενότητα 4.6) σχετικά με την αντιμετώπιση των προκλήσεων που τίθενται από το πρόβλημα της συνεργασίας των οντολογιών με τις Βάσεις Δεδομένων.

Στο κεφάλαιο 5 διερευνάται το πρόβλημα του σημασιολογικού εμπλουτισμού της πληροφορίας σε πραγματικό χρόνο και κατανεμημένο περιβάλλον. Οριοθετείται το πρόβλημα και αναλύονται οι επιμέρους συνιστώσες, οι οποίες συμβάλλουν στη δυσκολία αντιμετώπισής του. Στη συνέχεια παρουσιάζεται η ανάλυση και η υλοποίηση ενός συστήματος βασισμένου σε μεσιμικό, το οποίο είναι ικανό να υποστηρίξει ευφυείς εφαρμογές σε συστήματα με επίγνωση περιβάλλοντος. Παρουσιάζεται η λειτουργικότητα που προσφέρεται σύμφωνα με δύο σενάρια, αυτό της παρακολούθησης χώρων και αυτό της υποστήριξης συνεδριάσεων, το σύστημα υποβάλλεται σε μετρήσεις απόδοσης, γίνεται σχολιασμός για παράγοντες που μπορεί να επηρεάσουν την ομαλή λειτουργία του, και το κεφάλαιο καταλήγει με τα συμπεράσματα που προκύπτουν (ενότητα 5.6) και τις προτάσεις για μελλοντική εργασία.

6.2 Ανακεφαλαίωση των καινοτομικών στοιχείων της διατριβής

Στο σημείο αυτό κρίνεται απαραίτητη μια ανακεφαλαίωση των καινοτομικών στοιχείων που προκύπτουν από τη διατριβή αυτή. Τα στοιχεία αυτά αφορούν δύο συνιστώσες του ίδιου προβλήματος: του σημασιολογικού εμπλουτισμού της πληροφορίας.

Η πρώτη συνεισφορά της διατριβής αφορά στην ολοκλήρωση/αναβάθμιση Βάσεων Δεδομένων με στόχο τη δημιουργία μιας κατανεμημένης Βάσης Γνώσεως με ευφυή χαρακτηριστικά. Το πρόβλημα είναι επίκαιρο, βρίσκεται σε στάδιο επαρκούς ωρίμανσης, έχει προσφέρει μέχρι στιγμής πλήθος αποτελεσμάτων μέσα όμως από διαφορετικές και ετερογενείς προσεγγίσεις και εξετάζεται υπό το πρίσμα των τελευταίων τεχνολογικών εξελίξεων και συγκεκριμένα στο χώρο του αναπτυσσόμενου Σημασιολογικού Ιστού.

Η συνεισφορά της διατριβής εδώ εντοπίζεται στην ανάλυση του χώρου της συνεργασίας των Βάσεων Δεδομένων με τις οντολογίες, στη λεπτομερή καταγραφή και κατηγοριοποίηση των προσεγγίσεων που υπάρχουν στη βιβλιογραφία και μαζί συνθέτουν την αιχμή των εξελίξεων (ενότητα 4.2) καθώς και στην αναλυτική αποτύπωση των σχετικών συμπερασμάτων (ενότητα 4.6). Σημειώνεται ότι λόγω της δυναμικής των

εξελιξέων στο συγκεκριμένο χώρο, η καταγραφή, κατηγοριοποίηση και συγκριτική ανάλυση των σχετικών προσεγγίσεων αποτελεί καθοριστική συνεισφορά στη σχετική βιβλιογραφία, καθώς παρατηρήθηκε απουσία αντίστοιχων μελετών.

Επιπλέον, υλοποιείται και παρουσιάζεται μια πρωτότυπη προσέγγιση στην επίλυση του προβλήματος (ενότητα 4.3). Αξίζει να σημειωθεί ότι στην προσέγγιση αυτή έχει γίνει μέχρι στιγμής πλήθος αναφορών από ερευνητές που δε σχετίζονται με τους συγγραφείς [182–201].

Η διατριβή συνεισφέρει επίσης ουσιαστικά στο ζήτημα του σημασιολογικού εμπλουτισμού συστημάτων με επίγνωση περιβάλλοντος. Για την μελέτη του προβλήματος σχεδιάστηκε και υλοποιήθηκε μια πιλοτική εφαρμογή βασισμένη σε μεσισμικό η οποία στοχεύει στην επεξεργασία και το σημασιολογικό εμπλουτισμό της “ρέουσας” πληροφορίας, σε πραγματικό χρόνο και καταναμημένο περιβάλλον. Η προσέγγιση που παρουσιάζεται και αναλύεται διεξοδικά στο κεφάλαιο 5 αποτελεί συνεισφορά της διατριβής.

6.3 Θέματα για περαιτέρω διερεύνηση

Σε κάθε περίπτωση, η αξία και η χρησιμότητα μιας ερευνητικής εργασίας, εκτός από την ολοκληρωμένη συμβολή στο συναφές ερευνητικό πεδίο, έγκειται επίσης και στη δυνατότητα δημιουργικής αλληλεπίδρασης και επαναχρησιμοποίησης των ιδεών της από μελλοντικές ερευνητικές προσπάθειες. Σε συνδυασμό με τις προτάσεις για μελλοντική εργασία που δίνονται για τις δυο συνιστώσες της διατριβής, στις ενότητες 4.6 και 5.7 αντίστοιχα, περαιτέρω ερευνητικά θέματα για διερεύνηση μπορεί να αποτελούν τα παρακάτω:

1. *Αυτοματοποίηση*. Η αυτοματοποίηση κατά την φάση σημασιολογικού εμπλουτισμού είναι μια ιδιαίτερα σημαντική πτυχή του προβλήματος καθώς η χειροκίνητη προσθήκη σημασιολογίας αποτελεί μια χρονοβόρα και επιρρεπή σε λάθη διαδικασία καθώς απαιτεί υψηλό επίπεδο εξοικείωσης από τον χρήστη.

2. *Ασφάλεια*. Τα θέματα ασφάλειας που εγείρονται κατά τη δημιουργία της πληροφορίας, την πρόσβαση σε αυτή αλλά και την αποθήκευση και το διαμοιρασμό της χρήζουν ιδιαίτερης αντιμετώπισης (πρβλ. και ενότητα 5.7.1). Η εγκυρότητα, η αυθεντικότητα της πληροφορίας αλλά και η ταυτοποίηση των χρηστών κατά την πρόσβαση σε αυτή είναι ζητήματα ιδιαίτερα σημαντικά για κάθε σύγχρονο σύστημα που επεξεργάζεται πληροφορία.

3. *Ασάφεια* (fuzziness). Σε ανοιχτά και καταναμημένα περιβάλλοντα στα οποία η διαθέσιμη πληροφορία μπορεί να είναι ελλιπής, χρειάζεται στη λήψη αποφάσεων να

συνυπολογίζεται ο βαθμός αβεβαιότητας ή η ασάφεια των δεδομένων. Η ασάφεια στα πλαίσια του Σημασιολογικού Ιστού μπορεί να ενσωματώνεται σε επίπεδο υπηρεσιών συλλογιστικής ή εναλλακτικά να ενσωματώνεται στα μεταδεδομένα της πληροφορίας.

4. *Σημασιολογική διαλειτουργικότητα.* Η έννοια αυτή συνεπάγεται την προσθήκη σημασιολογίας στον τρόπο με τον οποίο η πληροφορία διαμοιράζεται και υφίσταται επεξεργασία μεταξύ δυο συστημάτων ούτως ώστε ένα τρίτο σύστημα να είναι σε θέση να κατανοεί και να εξάγει συμπεράσματα σε σχέση με την πληροφορία αυτή. Προκειμένου να επιτευχθεί ολοκλήρωση πληροφορίας σε ανώτερο επίπεδο είναι σημαντικό η διαμοιραζόμενη πληροφορία να υπακούει σε διεθνή πρότυπα τα οποία να εμπεριέχουν σημασιολογική περιγραφή της.

5. *Αξιοποίηση* σημασιολογικής πληροφορίας σε τομείς οι οποίοι μπορεί να ωφεληθούν. Ο Σημασιολογικός Ιστός δεν περιορίζεται στις ερευνητικές περιοχές οι οποίες εξετάστηκαν στην παρούσα διατριβή. Ερευνητικές περιοχές που αφορούν πληροφορία γεωγραφική, σχετική με την υγεία σε επίπεδο κατάστασης ασθενών ή σε επίπεδο περιγραφής ιατρικής και βιολογικής γνώσης, πληροφορία πολυμεσική στο χαμηλό επίπεδο κωδικοποίησης δεδομένων, πληροφορία καταχωρημένη από τους χρήστες ιστοτόπων κοινωνικής δικτύωσης ή και γενικότερα όπου επιτυγχάνεται η αναζήτηση προτύπων είναι περιοχές όπου ο σημασιολογικός εμπλουτισμός μπορεί να απελευθερώσει πλήθος νέων δυνατοτήτων.

Βιβλιογραφία

- [1] M. K. Bergman. The Deep Web: Surfacing Hidden Value. *Journal of Electronic Publishing*, 7(1), 2001.
- [2] R. Volz, S. Handschuh, S. Staab, and et al. Unveiling the Hidden Bride: Deep Annotation for Mapping and Migrating Legacy Data to the Semantic Web. *Journal of Web Semantics*, 1(2):187–206, 2004.
- [3] M. Uschold and M. Gruninger. Creating Semantically Integrated Communities on the World Wide Web. Abstract Invited Talk Semantic Web Workshop Co-located with WWW 2002, Honolulu, HI, 2002.
- [4] J. Park and S. Ram. Information Systems Interoperability: What Lies Beneath? *ACM Transactions on Information Systems*, 22(4):595–632, October 2004.
- [5] M. Friedman, A. Levy, and T. Millstein. Navigational Plans for Data Integration. In *Proceedings of the 16th National Conference on Artificial Intelligence (AAAI'99)*, pages 67–73. AAAI Press/The MIT Press, 1999.
- [6] B. Amann, C. Beeri, I. Fundulaki, and M. Scholl. Ontology-Based Integration of XML Web Resources. In *ISWC '02: Proceedings of the First International Semantic Web Conference on The Semantic Web*, pages 117–131, London, UK, 2002. Springer-Verlag. ISBN 3-540-43760-6.
- [7] I. Cruz, H. Xiao, and F. Hsu. An Ontology-based Framework for XML Semantic Integration. In *Eighth International Database Engineering and Applications Symposium (IDEAS'04)*, July 2004.
- [8] I. Manolescu, D. Florescu, D. Kossmann, F. Xhumari, and D. Olteanu. Agora: Living with XML and Relational. In *The VLDB Journal*, pages 623–626, 2000.
- [9] S. Cluet, P. Veltri, and D. Vodislav. Views in a Large Scale XML Repository. In *The VLDB Journal*, pages 271–280, 2001.
- [10] V. Uren, P. Cimiano, J. Iria, S. Handschuh, M. Vargas-Vera, E. Motta, and F. Ciravegna. Semantic Annotation for Knowledge Management: Requirements and a Survey of the State of the Art. *Journal of Web Semantics, Elsevier*, 4(1):14–28, January 2006.
- [11] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web - A New Form of Web Content that is Meaningful to Computers Will Unleash a Revolution of New Possibilities. *Scientific American*, 284(5):34–43, 2001.
- [12] S. Decker, S. Melnik, F. van Harmelen, D. Fensel, M. Klein, J. Broekstra, M. Erdmann, and I. Horrocks. The Semantic Web: The Roles of XML and RDF. *IEEE Internet Computing*, 4(5):63–74, October 2000.
- [13] N. Shadbolt, T. Berners-Lee, and W. Hall. The Semantic Web Revisited. *IEEE Intelligent Systems*, 21(3):96–101, 2006. ISSN 1541-1672.
- [14] R.J. Brachman and J.G. Schmolze. An Overview of the KL-ONE Knowledge representation System. *Cognitive Science*, 9:171–216, August 1985.
- [15] M. Schmidt-Schaub and G. Smolka. Attributive concept descriptions with complements. *Artificial Intelligence*, 48(1):1–26, 1991. ISSN 0004-3702.
- [16] D. Nardi and R. Brachman. *The Description Logic Handbook*, chapter An Introduction to Description Logics, pages 5–44. Cambridge University Press, 2002.
- [17] F. Baader and W. Nutt. *The Description Logic Handbook*, chapter Basic Description Logics,

- pages 47–100. Cambridge University Press, 2002.
- [18] A. Shaerf. *Query Answering in Concept-Based Knowledge Representation Systems: Algorithms, Complexity, and Semantic Issues*. PhD thesis, Dipartimento di Informatica e Sistemistica Univerita di Roma, 1994.
 - [19] A. Borgida. Description Logics in Data Management. *IEEE Transactions on Knowledge and Data Engineering*, 7(5):671–682, October 1995.
 - [20] B. Grosz, I. Horrocks, R. Volz, and S. Decker. Description Logic Programs: Combining Logic Programs with Description Logic. In *Proceedings of the Twelfth International World Wide Web Conference (WWW 2003)*, pages 48–57. ACM, 2003. ISBN 1-58113-680-3.
 - [21] R. MacGregor and R. Bates. The LOOM Knowledge Representation Language. In *Proceedings of the Knowledge-Based Systems Workshop*, 1987.
 - [22] I. Horrocks. Using an Expressive Description Logic: FaCT or Fiction? In Anthony G. Cohn, Leonard Schubert, and S. Shapiro, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the 6th International Conference (KR-98)*, pages 636–647. Morgan Kaufman, 1998.
 - [23] V. Haarslev and R. Möller. Description of the RACER System and its Applications. In *International Workshop on Description Logics (DL-2001)*, pages 131–141, Stanford, USA, August 2001.
 - [24] U. Hustadt, B. Motik, and U. Sattler. Reducing \mathcal{SHIQ}^- Description Logic to Disjunctive Datalog Programs. In *Proceedings of the 9th International Conference on Knowledge Representation and Reasoning (KR'04)*, 2004.
 - [25] E. Sirin, B. Parsia, B.C. Grau, A. Kalyanpur, and Y. Katz. Pellet: A practical OWL-DL reasoner. *Journal of Web Semantics: Science, Services and Agents on the World Wide Web*, 5(2):51–53, 2007.
 - [26] F. Baader, I. Horrocks, and U. Sattler. *Festschrift in honor of J. Siekmann*, chapter Description Logics as Ontology Languages for the Semantic Web. Lecture Notes in Artificial Intelligence. Springer, 2003.
 - [27] T. Gruber. Toward Principles for the Design of Ontologies Used for Knowledge Sharing. *International Journal of Human-Computer Studies*, 43(5–6):907–928, 1995.
 - [28] Michael R. Genesereth and Nils J. Nilsson. *Logical foundations of artificial intelligence*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1987. ISBN 0-934613-31-1.
 - [29] N. Guarino. Formal Ontology and Information Systems. In N. Guarino, editor, *Proceedings of the 1st International Conference on Formal Ontologies in Information Systems (FOIS'98)*, pages 3–15, Trento, Italy, June 1998. IOS Press.
 - [30] N. Konstantinou, D.E. Spanos, and N. Mitrou. Ontology and Database Mapping: A Survey of Current Implementations and Future Directions. *Journal of Web Engineering*, 7(1):1–24, March 2008.
 - [31] S. Decker, D. Fensel, F. van Harmelen, I. Horrocks, S. Melnik, M. Klein, and J. Broekstra. Knowledge Representation on the Web. In *Proceedings of the 2000 International Workshop on Description Logics (DL'00)*, 2000.
 - [32] D. Fensel, I. Horrocks, F. van Harmelen, S. Decker, M. Erdmann, and M. Klein. OIL in a nutshell. In *Knowledge Acquisition, Modeling, and Management, Proceedings of the European Knowledge Acquisition Conference (EKAW'00)*, 2000.
 - [33] D. Fensel, F. van Harmelen, I. Horrocks, D. McGuinness, and P. Patel-Schneider. OIL: An

- ontology infrastructure for the Semantic Web. *IEEE Intelligent Systems*, 2001.
- [34] I. Horrocks, P. Patel-Schneider, and F. van Harmelen. Reviewing the Design of DAML+OIL: An Ontology Language for the Semantic Web. In *Proceedings of the 18th National Conference on Artificial Intelligence (AAAI 2002)*, pages 792–797. AAAI Press, 2002. ISBN 0-26251-129-0.
- [35] R. Seeley. The Semantic Web: The OWL has landed. Available online at www.adtmag.com/article.asp?id=8144, August 2003. ADT Magazine.
- [36] I. Horrocks, P. Patel-Schneider, and F. van Harmelen. From *SHIQ* and RDF to OWL: The Making of a Web Ontology Language. *Journal of Web Semantics: Science, Services and Agents on the World Wide Web*, 1(1):7–26, 2003.
- [37] D. McGuinness and F. van Harmelen. OWL Web Ontology Language Overview. Available online at www.w3.org/TR/owl-features/, February 2004.
- [38] S. Bechhofer, F. van Harmelen, J. Hendler, I. Horrocks, D. McGuinness, P. Patel-Schneider, and L. Andrea Stein. OWL Web Ontology Language Reference. World Wide Web Consortium, Recommendation, available online at www.w3.org/TR/2004/REC-owl-ref-20040210, February 2004.
- [39] P. Patel-Schneider and I. Horrocks. OWL Web Ontology Language: Semantics and Abstract Syntax Section 3. Direct Model-Theoretic Semantics. Available online at www.w3.org/TR/owl-semantics/direct.html#3.1, 2004.
- [40] S. Auer, C. Bizer, J. Lehmann, G. Kobilarov, R. Cyganiak, and Z. Ives. DBpedia: A Nucleus for a Web of Open Data. In *The Semantic Web, 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference (ISWC+ASWC 2007)*, volume 4825 of *Lecture Notes in Computer Science*, Busan, Korea, November 2007. Springer.
- [41] J.G. Breslin, A. Harth, U. Bojars, and S. Decker. Towards Semantically-Interlinked Online Communities. In *Proceedings of the 2nd European Semantic Web Conference (ESWC'05)*, volume 3532 of *Lecture Notes in Computer Science*, pages 500–514, Heraklion, Greece, 2005.
- [42] F. Pan and J.R. Hobbs. Time in OWL-S. In *Proceedings of the AAAI Spring Symposium on Semantic Web Services*, 2004.
- [43] V. Milea, F. Frasinca, U. Kaymak, and T. di Noia. An OWL-Based Approach Towards Representing Time in Web Information Systems. In *The 4th International Workshop of Web Information Systems Modeling Workshop (WISM 2007)*, pages 791–802. Tapir Academic Press, 2007.
- [44] Christian Bizer, Tom Heath, Kingsley Idehen, and Tim Berners-Lee. Linked Data on the Web (LDOW2008). In *Proceedings of the 17th international conference on World Wide Web (WWW'08)*, pages 1265–1266, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-085-2.
- [45] R. Doorenbos. *Production Matching for Large Learning Systems*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, January 1995.
- [46] I. Horrocks, P. Patel-Schneider, H. Boley, S. Tabet, B. Groszof, and M. Dean. SWRL: A Semantic Web Rule Language Combining OWL and RuleML. World Wide Web Consortium, Member Submission, available online at www.w3.org/Submission/SWRL/, May 2004.
- [47] W. May, J.J. Alferes, and R. Amador. *Rules and Rule Markup Languages for the Seman-*

- tic Web*, volume 3791 of *Lecture Notes in Computer Science*, chapter Active Rules in the Semantic Web: Dealing with Language Heterogeneity, pages 30–44. Springer Berlin / Heidelberg, 2005.
- [48] D. Hirtle, H. Boley, B. Grosz, M. Kifer, M. Sintek, S. Tabet, and G. Wagner. Schema Specification of RuleML 0.91. Available online at www.ruleml.org/0.91/, 2006.
- [49] M.J. O’Connor, H. Knublauch, S.W. Tu, B. Grosz, M. Dean, W.E. Grosso, and M.A. Musen. Supporting Rule System Interoperability on the Semantic Web with SWRL. In *Fourth International Semantic Web Conference (ISWC’05)*, Galway, Ireland, 2005.
- [50] B. Parsia, E. Sirin, B. Cuenca Grau, E. Ruckhaus, and D. Hewlett. Cautiously Approaching SWRL. Available online at www.mindswap.org/papers/CautiousSWRL.pdf, February 2005.
- [51] P. Haase, J. Broekstra, A. Eberhart, and R. Volz. A Comparison of RDF Query Languages. In Sheila McIlraith, Dimitris Plexousakis, and Frank van Harmelen, editors, *The Semantic Web - ISWC 2004. Proceedings of the Third International Semantic Web Conference*, volume 3298 of *Lecture Notes in Computer Science*, Hiroshima, Japan, 2004. Springer-Verlag.
- [52] J. Bailey, F. Bry, T. Furche, and S. Schaffert. Web and Semantic Web Query Languages: A Survey. In Norbert Eisinger and Jan Mauszyski, editors, *Reasoning Web, First International Summer School*. Springer-Verlag, 2005.
- [53] R. Fikes, P. Hayes, and I. Horrocks. OWL-QL – A Language for Deductive Query Answering on the Semantic Web. *Journal of Web Semantics*, 2(1), 2005.
- [54] T. Berners-Lee. Cwm - A General Purpose Data Processor for the Semantic Web, 2008. Available online at www.w3.org/2000/10/swap/doc/cwm.html.
- [55] S. Decker, M. Sintek, A. Billig, N. Henze, P. Dolog, W. Nejdl, A. Harth, A. Leicher, S. Busse, J. Guy Suess, Z. Miklos, J.L. Ambite, M. Weathers, G. Neumann, and U. Zudun. TRIPLE - an RDF Rule Language with Context and Use Cases. In *W3C Workshop on Rule Languages for Interoperability*, Washington D.C., USA, April 2005.
- [56] J. Broekstra, A. Kampman, and F. van Harmelen. *Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential*, chapter Sesame: An Architecture for Storing and Querying RDF Data and Schema Information, pages 197–222. MIT Press, 2003.
- [57] J. Broekstra and A. Kampman. SeRQL: An RDF Query and Transformation Language (DRAFT), 2004.
- [58] A. Seaborne. RDQL - A Query Language for RDF. World Wide Web Consortium, Member Submission, available online at www.w3.org/Submission/2004/SUBM-RDQL-20040109, January 2004.
- [59] R. Oldakowski and C. Bizer. RAP: RDF API for PHP. Technical report, FU Berlin, 2004. www.wiwiss.fu-berlin.de/suhl/radek/pub/RAP-oldakowski.pdf.
- [60] E. Prud’hommeaux and A. Seaborne. SPARQL Query Language for RDF. World Wide Web Consortium, Recommendation, available online at www.w3.org/TR/rdf-sparql-query/, January 2008.
- [61] T. Liebig and O. Noppens. OntoTrack: Combining Browsing and Editing with Reasoning and Explaining for OWL Lite Ontologies. In *Proceedings of the 3rd International International Semantic Web Conference (ISWC’04)*, volume 3298 of *Lecture Notes in Computer Science*. Springer-Verlag, June 2004.
- [62] S. Bechofer, I. Horrocks, C. Goble, and R. Stevens. OilEd: A Reason-able Ontology Editor

- for the Semantic Web. In F. Baader, G. Brewka, and T. Eiter, editors, *KI 2001: Advances in Artificial Intelligence*, pages 396–408. Springer, 2001.
- [63] S. Auer. Powl - A Web Based Platform for Collaborative Semantic Web Development. In *Proceedings of 1st Workshop Scripting for the Semantic Web (SFSW'05)*, Heronissos, Greece, May 2005.
- [64] C. Bizer, R. Lee, and E. Pietriga. Fresnel - A Browser-Independent Presentation Vocabulary for RDF. In *Second International Workshop on Interaction Design and the Semantic Web (held in conjunction with the 4th International Semantic Web Conference)*, Galway, Ireland, November 2005.
- [65] J. Cardoso. The semantic web vision: Where are we? *IEEE Intelligent Systems*, 22(5): 84–88, September/October 2007.
- [66] B. Cuenca Grau, B. Parsia, and E. Sirin. Combining OWL ontologies using \mathcal{E} -Connections. *Web Semantics: Science, Services and Agents on the World Wide Web*, 4(1):40–59, January 2006.
- [67] A. Kalyanpur, B. Parsia, and J. Hendler. A Tool for Working with Web Ontologies. *Proceedings of the International Journal on Semantic Web and Information Systems*, 1(1), 2005.
- [68] Y. Yesilada, S. Bechhofer, and B. Horan. Personalised Dynamic Links on the Web. In *Proceedings of the First International Workshop on Semantic Media Adaptation and Personalization (SMAP'06)*, pages 7–12, Washington, DC, USA, 2006. IEEE Computer Society. ISBN 0-7695-2692-6.
- [69] R. Baumgartner, S. Flesca, and G. Gottlob. Visual Web Information Extraction with Lixto. In *The VLDB Journal*, pages 119–128, 2001.
- [70] A. Hogue and D. Karger. Thresher: Automating the Unwrapping of Semantic Content from the World Wide Web. In *Proceedings of the 14th International Conference on World Wide Web (WWW'05)*, pages 86–95, New York, NY, USA, 2005. ACM. ISBN 1-59593-046-9.
- [71] J. Heflin and J. Hendler. A portrait of the Semantic Web in action. *IEEE Intelligent Systems*, 16(2):54–59, 2001.
- [72] M. Vargas-Vera, E. Motta, J. Domingue, M. Lanzoni, A. Stutt, and F. Ciravegna. MnM: Ontology Driven Tool for Semantic Markup. In *European Conference on Artificial Intelligence (ECAI 2002)*, 2002.
- [73] F. Ciravegna, A. Dingli, D. Petrelli, and Y. Wilks. User-System Cooperation in Document Annotation Based on Information. In *Proceedings of the 13th International Conference on Knowledge Engineering and KM (EKAW02)*, Sigüenza, Spain, 2002.
- [74] J. Iria, F. Ciravegna, Ph. Cimiano, A. Lavelli, E. Motta, L. Gilardoni, and E. Mönch. Integrating Information Extraction, Ontology Learning and Semantic Browsing into Organizational Knowledge Processes. In *Proc. of the EKAW Workshop on the Application of Language and Semantic Technologies to support Knowledge Management Processes, at the 14th International Conference on Knowledge Engineering and Knowledge Management*, October 2004.
- [75] O. Etzioni, M. Cafarella, D. Downey, A.-M. Popescu, T. Shaked, S. Soderland, D. Weld, and A. Yates. Unsupervised Named-Entity Extraction from the Web: An Experimental Study. *Artificial Intelligence*, 165(1):91–134, 2005. ISSN 0004-3702.
- [76] J. Carroll, I. Dickinson, C. Dollin, D. Reynolds, A. Seaborne, and K. Wilkinson. Jena: Implementing the Semantic Web Recommendations. Technical Report HPL-2003-146,

- Hewlett-Packard, December 2003.
- [77] I. Niles and A. Pease. Towards a Standard Upper Ontology. In C. Welty and B. Smith, editors, *Proceedings of 2nd International Conference on Formal Ontology in Information Systems*, 2001.
 - [78] K. Ahrens, Ch. Siaw-Fong, and H. Chu-Ren. From Lexical Semantics to Conceptual Metaphors: Mapping Principle Verification with WordNet and SUMO. In *Proceedings of the 5th Chinese Lexical Semantics Workshop (CLSW-5)*, Singapore, 2004.
 - [79] L. Ding, T. Finin, A. Joshi, R. Pan, R. Scott Cost, Y. Peng, P. Reddivari, V. Doshi, and J. Sachs. Swoogle: A Search and Metadata Engine for the Semantic Web. In *Proceedings of the 13th ACM Conference on Information and Knowledge Management*, 2004.
 - [80] I. Niles and A. Pease. Linking Lexicons and Ontologies: Mapping WordNet to the Suggested Upper Merged Ontology. In *Proceedings of the International Conference on Information and Knowledge Engineering (IKE 2003)*, volume 1, pages 412–416. CSREA Press, 2003.
 - [81] F. Donini, M. Lenzerini, D. Nardi, and A. Schaerf. Reasoning in Description Logics. In Gerhard Brewka, editor, *Principles of Knowledge Representation*, pages 191–236. CSLI Publications, Stanford, California, 1996.
 - [82] E. Bozsak, M. Ehrig, S. Handschuh, A. Hotho, A. Maedche, B. Motik, D. Oberle, C. Schmitz, S. Staab, L. Stojanovic, N. Stojanovic, R. Studer, G. Stumme, Y. Sure, J. Tane, R. Volz, and V. Zacharias. KAON – Towards a Large Scale Semantic Web. In K. Bauknecht, A. Min Tjoa, and G. Quirchmayr, editors, *Proceedings of the third International Conference on E-Commerce and Web Technologies (EC-Web2002)*, pages 304–313, London, UK, 2002. Springer-Verlag. ISBN 3-540-44137-9.
 - [83] S. Greco. Binding Propagation Techniques for the Optimization of Bound Disjunctive Queries. *IEEE Transactions on Knowledge and Data Engineering*, 15(2):368–385, 2003. ISSN 1041-4347.
 - [84] B. Motik and U. Sattler. A Comparison of Reasoning Techniques for Querying Large Description Logic ABoxes. In Miki Hermann and Andrei Voronkov, editors, *Proceedings of the 13th Int. Conference on Logic for Programming Artificial Intelligence and Reasoning (LPAR'06)*, volume 4246 of *LNCS*, pages 227–241, Phnom Penh, Cambodia, November 13–17 2006. Springer.
 - [85] D. Tsarkov and I. Horrocks. *Automated Reasoning*, volume 4130 of *Lecture Notes in Computer Science*, chapter FaCT++ Description Logic Reasoner: System Description, pages 292–297. Springer Berlin / Heidelberg, October 2006.
 - [86] R. Fikes, J. Jenkins, and Q. Zhou. Including Domain-Specific Reasoners with Reusable Ontologies. In *Proceedings of the 2003 International Conference on Information and Knowledge Engineering*, Las Vegas, Nevada, USA, June 23–26 2003.
 - [87] Kopena J and W. Regli. DAMLJessKB: A Tool for Reasoning with the Semantic Web. *IEEE Intelligent Systems*, 18(3):74–77, 2003.
 - [88] C. Hewitt. Viewing Control Structures as Patterns of Passing Messages. *Artificial Intelligence*, 8(3):323–364, 1977.
 - [89] S. Franklin and A. Graesser. Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents. In *Intelligent Agents III. Agent Theories, Architectures and Languages (ATAL'96)*, volume 1193, Berlin, Germany, 1996. Springer-Verlag.

- [90] Roberto A. Flores-Mendez. Towards a Standardization of Multi-Agent System Framework. *Crossroads*, 5(4):18–24, 1999.
- [91] J. Müller and B. Bauer. *Agent-Oriented Software Engineering III*, volume 2585 of *Lecture Notes in Computer Science*, chapter Agent-Oriented Software Technologies: Flaws and Remedies, pages 210–227. Springer Berlin / Heidelberg, January 2003.
- [92] K. Chmiel, M. Gawinecki, P. Kaczmarek, M. Szymczak, and M. Paprzycki. Efficiency of JADE agent platform. In *Proceedings of the 3rd International Symposium on Parallel and Distributed Computing*, 2004.
- [93] E. Hyvönen, E. Mäkelä, M. Salminen, A. Valo, K. Viljanen, S. Saarela, M. Junnila, and S. Kettula. MuseumFinland – Finnish Museums on the Semantic Web. *Web Semantics: Science, Services and Agents on the World Wide Web*, 3(2–3):224–241, October 2005.
- [94] E. Solidakis, N. Konstantinou, E.S. Pashou, A. Papakonstantinou, and N. Mitrou. A Decentralized Multi-Agent Ontology-Based System for Information Retrieval. In *SWAMM'06, on the 15th World Wide Web Conference (WWW'06)*, Edinburgh, May 2006.
- [95] J.L. Dinos and J.F. Vega-Riveros. A Document Ontology and Agent-Based RDF Metadata Retrieval. In *Nineteenth National Conference on Artificial Intelligence*, 2004.
- [96] N. Gibbins, S. Harris, and N. Shadbolt. Agent-based Semantic Web Services. *Web Semantics: Science, Services and Agents on the World Wide Web*, 1(2):141–154, February 2004.
- [97] A. Dey. Understanding and Using Context. *Personal Ubiquitous Computing*, 5(1):4–7, 2001. ISSN 1617-4909.
- [98] K. Edwards, Victoria Bellotti, A. Dey, and M. Newman. Stuck in the Middle: The Challenges of User-Centered Design and Evaluation for Middleware. In *Proceedings of the 2003 Conference on Human Factors in Computing Systems (CHI 2003)*, pages 5–10, April 2003.
- [99] A. Dey, D. Salber, and G. Abowd. A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications. *Human-Computer Interaction (HCI) Journal*, 16(2–4):97–166, 2001.
- [100] T. Kindberg and J. Barton. A Web-Based Nomadic Computing System. *Computer Networks, Elsevier*, 35(4):443–456, March 2001.
- [101] O. Lassila and D. Khushraj. Contextualizing Applications via Semantic Middleware. In *Proceedings of the The Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (MOBIQUITOUS'05)*, pages 183–191, Washington, DC, USA, 2005. IEEE Computer Society. ISBN 0-7695-2375-7.
- [102] A. Toninelli, R. Montanari, L. Kagal, and O. Lassila. *The Semantic Web - ISWC 2006*, volume 4273 of *Lecture Notes in Computer Science*, chapter A Semantic Context-Aware Access Control Framework for Secure Collaborations in Pervasive Computing Environments, pages 473–486. Springer Berlin / Heidelberg, November 2006.
- [103] I. Pandis, J. Soldatos, A. Paar, J. Reuter, M. Carras, and L. Polymenakos. An Ontology-based Framework for Dynamic Resource Management in Ubiquitous Computing Environments. In *Proceedings of the second International Conference on Embedded Software and Systems (ICCESS'05)*, 2005.
- [104] A. Uszok, J.M. Bradshaw, M. Johnson, R. Jeffers, A. Tate, J. Dalton, and S. Aitken. KAoS Policy Management for Semantic Web Services. *IEEE Intelligent Systems*, 19(4):32–41, 2004.

- [105] L. Kagal, T. Finin, and A. Johshi. A Policy Language for Pervasive Computing Environment. In *Proceedings of IEEE fourth International Workshop on Policies for Distributed Systems and Networks (POLICY'03)*, 2003.
- [106] O. Kaykova, O. Khriyenko, D. Kovtun, A. Naumenko, V. Terziyan, and A. Zharko A. General Adaption Framework: Enabling Interoperability for Industrial Web Resources. *International Journal on Semantic Web and Information Systems*, 1(3):31–63, 2005.
- [107] A. Zafeiropoulos, N. Konstantinou, S. Arkoulis, D.-E. Spanos, and N. Mitrou. A Semantic-Based Architecture for Sensor Data Fusion. In *Second International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM'08)*, Valencia, Spain, October 2008.
- [108] N. Fukuta, T. Osawa, T. Iijima, and T. Yamaguchi. Semantic Service Integration Support for Web Portal. In *Proceedings of the 2005 IEEE / WIC / ACM International Conference on Web Intelligence (WI'05)*, pages 161–164, September 2005.
- [109] A. Ankolekar, M. Burstein, J. Hobbs, O. Lassila, D. Martin, S. McIlraith, S. Narayanan, M. Paolucci, T. Payne, K. Sycara, and H. Zeng. DAML-S: Web Service Description for the Semantic Web. In *International Semantic Web Conference (ISWC'02)*, Sardinia, June 2002.
- [110] S. Handschuh, S. Staab, and R. Volz. On deep Annotation. In *Proceedings of the WWW2003*, Budapest, Hungary, May 2003.
- [111] J. Hein and J. Hendler. Searching the Web with SHOE. In *Proceedings of AAAI-2000 Workshop on AI for Web Search*, 2000.
- [112] V. Lopez Garcia, E. Motta, and V. Uren. AquaLog: An Ontology-driven Question Answering System to Interface the Semantic Web. In *Proceedings of the 2006 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 269–272, Morristown, NJ, USA, 2006. Association for Computational Linguistics.
- [113] Ph. Cimiano, P. Haase, J. Heizmann, M. Mantel, and R. Studer. Towards portable natural language interfaces to knowledge bases - The case of the ORAKEL system. *Data and Knowledge Engineering*, 65(2):325–354, 2008. ISSN 0169-023X.
- [114] R. Guha, R. McCool, and E. Miller. Semantic Search. In *Proceedings of the 12th International Conference on World Wide Web (WWW'03)*, pages 700–709, New York, NY, USA, 2003. ACM. ISBN 1-58113-680-3.
- [115] O. Corby, R. Dieng-Kuntz, and C. Faron-Zucker. Querying the Semantic Web with the Corese Search Engine. In *Proceedings of the 15th European Conference on Artificial Intelligence (ECAI 2004), Prestigious Applications of Intelligent Systems (PAIS 2004)*, pages 705–709, 2004.
- [116] Y. Lei, V. Uren, and E. Motta. SemSearch A Search Engine for the Semantic Web. In *15th International Conference on Knowledge Engineering and Knowledge Management Managing Knowledge in a World of Networks (EKAW 2006)*, Podebrady, Czech Republic, 2006.
- [117] Th. Athanasiadis, V. Tzouvaras, K. Petridis, F. Precioso, Y. Avrithis, and Y. Kompatsiaris. Using a multimedia ontology infrastructure for semantic annotation of multimedia content. In *Proceedings of the 5th International Workshop on Knowledge Markup and Semantic Annotation (SemAnnot'05) at the 4th International Semantic Web Conference (ISWC'05)*, Galway, Ireland, November 2005.

- [118] S. Bloehdorn, K. Petridis, C. Saathoff, N. Simou, V. Tzouvaras, Y. Avrithis, S. Handschuh, Y. Kompatsiaris, S. Staab, and M. Strintzis. Semantic Annotation of Images and Videos for Multimedia Analysis. In *Proceedings of the 2nd European Semantic Web Conference (ESWC'05)*, Heraklion, Greece, May 2005.
- [119] N. Simou, V. Tzouvaras, Y. Avrithis, G. Stamou, and S. Kollias. A Visual Descriptor Ontology for Multimedia Reasoning. In *Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS'05)*, Montreux, Switzerland, April 2005.
- [120] K. Petridis, D. Anastasopoulos, C. Saathoff, N. Timmermann, I. Kompatsiaris, and S. Staab. M-OntoMat-Annotizer: Image Annotation. Linking Ontologies and Multimedia Low-Level Features. In *Engineered Applications of Semantic Web Session (SWEA) at the 10th International Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES'06)*, Bournemouth, U.K., October 2006.
- [121] K. Petridis, D. Anastasopoulos, C. Saathoff, N. Timmermann, I. Kompatsiaris, and S. Staab. M-OntoMat-Annotizer: Image Annotation. Linking Ontologies and Multimedia Low-Level Features. In *Engineered Applications of Semantic Web Session (SWEA) at the 10th International Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES'06)*, Bournemouth, U.K., October 2006.
- [122] S. Handschuh, S. Staab, and R. Studer. Leveraging Metadata Creation for the Semantic Web with Cream. In *Proceedings of the Annual German Conference on AI*, volume 2821, Berlin, 2003. Springer.
- [123] R. Jakkilinki, N. Sharda, and I. Ahmed. MUDPY Ontology: A Tool for Multimedia Project Planning, Design and Development. *Enterprise Information Management Journal and Business Process Management Journal*, 2005.
- [124] S. Castano, S. Espinosa, A. Ferrara, V. Karkaletsis, A. Kaya, S. Melzer, R. Moller, S. Montanelli, and G. Petasis. Ontology Dynamics with Multimedia Information: The BOEMIE Evolution Methodology. In *Proceedings of International Workshop on Ontology Dynamics (IWOD) ESWC 2007 Workshop*, Innsbruck, Austria, June 2007.
- [125] A. Chakravarthy, F. Ciravegna, and V. Lanfranchi. Cross-media document annotation and enrichment. In *Proc. 1st Semantic Web Authoring and Annotation Workshop (SAAW2006)*, 2006.
- [126] R. Schroeter, J. Hunter, J. Guerin, I. Khan, and M. Henderson. A Synchronous Multimedia Annotation System for Secure Collaboratories. In *Proceedings of the Second IEEE International Conference on e-Science and Grid Computing (E-SCIENCE'06)*, page 41, Washington, DC, USA, December 2006. IEEE Computer Society. ISBN 0-7695-2734-5.
- [127] M. Daconta, L. Obrst, and K. Smith. *The Semantic Web - A Guide to the Future of XML, Web Services, and Knowledge Management*, chapter The Business Case for the Semantic Web, pages 17–26. Wiley, 2003.
- [128] A. Ankolekar, M. Krötzsch, Th. Tran, and D. Vrandečić. The Two Cultures: Mashing up Web 2.0 and the Semantic Web. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(1):70–75, February 2008.
- [129] E. Goer (Yahoo! SearchMonkey Team). SearchMonkey Support for RDFa Enabled (Yahoo! Developer Network Blog). Available online at: developer.yahoo.net/blog/archives/2008/09/searchmonkey_su.html, September 2008.
- [130] J. Zhang and A. Dimitroff. The Impact of Webpage Content Characteristics on Webpage Visibility in Search Engine Results (Part I). *International Journal of Information Processing*

- and Management*, 41(3):665–690, 2005. ISSN 0306-4573.
- [131] I. Horrocks, L. Li, D. Turi, and S. Bechhofer. The Instance Store: Description Logic Reasoning with Large Numbers of Individuals. In *17th International Workshop on Description Logics (DL 2004)*, pages 31–40, January 2004.
 - [132] W. Hodak and S. Kumar. Oracle Database 11g Performance and Scalability. White paper, Oracle Corporation, November 2007. Available online at www.oracle.com/technology/ deploy/performance/pdf/db_perfscale_11gr1_twp.pdf.
 - [133] C.P. de Laborda and S. Conrad. Bringing relational data into the Semantic Web using SPARQL and Relational.OWL. In *Proceedings of the International Conference on Data Engineering Workshops (ICDEW 2006)*, page 55, 2006.
 - [134] Object Management Group. MOF 2.0 Query/Views/Transformations RFP, 2002.
 - [135] Maurizio Lenzerini. Data integration: a theoretical perspective. In *Proceedings of the 21st ACM SIGMOD-SIGACT-SIGART symposium on Principles Of Database Systems (PODS'02)*, pages 233–246, New York, NY, USA, 2002. ACM. ISBN 1-58113-507-6.
 - [136] I. Kurtev, J. Bézivin, and M. Aksit. Technological Spaces: an Initial Appraisal. In *10th International Conference on Cooperative Information Systems (CoopIS 2002)*, *International Symposium on Distributed Objects and Applications (DOA 2002)*, *Federated Conferences.*, 2002.
 - [137] G. Kappel, E. Kapsammer, H. Kargl, G. Kramler, T. Reiter, W. Retschitzegger, W. Schwinger, and M. Wimmer. Lifting Metamodels to Ontologies: A Step to the Semantic Integration of Modeling Languages. In *Proceedings of Model Driven Engineering Languages and Systems (MoDELS 2006)*, volume 4199 of *Lecture Notes in Computer Science*, pages 528–542. Springer, 2006.
 - [138] M. Denny. Ontology Tools Survey, Revisited. Technical report, XML.com, 2004.
 - [139] N. Konstantinou, D.E. Spanos, M. Chalas, E. Solidakis, and N. Mitrou. VisAVis: An Approach to an Intermediate Layer between Ontologies and Relational Database Contents. In *WISM'06, on the 18th Conference on Advanced Information Systems Engineering (CAiSE'06)*, pages 1050–1061, Luxembourg, June 2006.
 - [140] S. Harris and N. Gibbins. 3store: Efficient Bulk RDF Storage. In *Proceedings of the 1st International Workshop on Practical and Scalable Semantic Systems (PSSS'03)*, pages 1–15, Sanibel Island, Florida, 2003.
 - [141] D. Wood, P. Gearon, and T. Adams. Kowari: A Platform for Semantic Web Storage and Analysis. In *Proceedings of the XTech Conference*, 2005.
 - [142] D. Wood. *Web Information Systems Engineering - WISE 2005 Workshops*, volume 3807 of *Lecture Notes in Computer Science*, chapter Scaling the Kowari Metastore, pages 193–198. Springer Berlin / Heidelberg, October 2005.
 - [143] D. Beneventano, S. Bergamaschi, F. Guerra, and M. Vincini. Building a Tourism Information Provider with the MOMIS System. *Information Technology and Tourism*, 7(3–4): 221–238, 2005.
 - [144] S. Bergamaschi, S. Castano, M. Vincini, and D. Beneventano. Semantic Integration of Heterogeneous Information Sources. *Data and Knowledge Engineering, Elsevier*, 36:215–249, 2001.
 - [145] L. Haas, M. Hernandez, H. Ho, L. Popa, and M. Roth. Clio Grows Up: From Research

- Prototype to Industrial Tool. In *ACM SIGMOD International Conference on Management of Data*, pages 805–810, 2005.
- [146] L. Yan, R. Miller, L. Haas, and R. Fagin. Data-Driven Understanding and Refinement of Schema Mappings. In *International Conference on the Management of Data, ACM SIGMOD*, volume 30, pages 485–496, 2001.
- [147] D. Maier, A. Mendelzon, and Y. Sagiv. Testing Implications of Data Dependencies. *ACM Transactions on Database Systems (TODS)*, 4(4):455–469, 1979.
- [148] L. Stojanovic, N. Stojanovic, and R. Volz. Migrating Data-intensive Web Sites into the Semantic Web. In *ACM Symposium on Applied Computing (SAC 2002)*, pages 1100–1107, 2002.
- [149] C. Bizer. D2R MAP: A Database to RDF Mapping Language. In *The Twelfth International World Wide Web Conference*, Budapest, Hungary, May 2003.
- [150] C. Bizer and A. Seaborne. D2RQ – Treating Non-RDF Databases as Virtual RDF Graphs. In *3rd International Semantic Web Conference (ISWC 2004)*, 2004.
- [151] J. Barrasa, O. Corcho, and A. Gómez-Pérez. Fund Finder Wrapper: A Case Study of Database-to-Ontology Mapping. In *ISWC2003 Workshop on Semantic Integration*, Sanibel Island, Florida, October 2003.
- [152] J. Barrasa, O. Corcho, and A. Gómez-Pérez. R2O, an Extensible and Semantically Based Database-to-Ontology Mapping Language. In *Second Workshop on Semantic Web and Databases (SWDB'04)*, 2004.
- [153] D. Dou and P. LePendu. Ontology-Based Integration for Relational Databases. In *Proceedings of the 2006 ACM Symposium on Applied Computing (SAC'06)*, pages 461–466, New York, NY, USA, 2006. ISBN 1-59593-108-2.
- [154] D. Dou, J. Pan, H. Qin, and P. LePendu. Towards Populating and Querying the Semantic Web. In *Proceedings of 2nd International workshop on Scalable Semantic Web Knowledge Base Systems (SSWS'06)*, pages 129–142, November 2006. co-located with ISWC 2006.
- [155] Y. An, A. Borgida, and J. Mylopoulos. Building Semantic Mappings from Databases to Ontologies. In *21st National Conference on Artificial Intelligence (AAAI'06)*, 2006.
- [156] Y. Velegrakis, R. Miller, and L. Popa. Mapping Adaptation Under Evolving Schemas. In *Proceedings of the 29th International Conference on Very Large Data Bases (VLDB'03)*, pages 584–595. VLDB Endowment, 2003. ISBN 0-12-722442-4.
- [157] P. Bouquet, M. Ehrig, J. Euzenat, E. Franconi, P. Hitzler, M. Krötzsch, L. Serafini, G. Stamou, Y. Sure, and S. Tessaris. Specification of a Common Framework for Characterizing Alignment. Technical report, Institut AIFB, Universität Karlsruhe, 2004. KnowledgeWeb deliverable d2.2.1v2. Technical report.
- [158] Y. Kalfoglou and M. Schorlemmer. Ontology Mapping: The State of the Art. *The Knowledge Engineering Review*, 2003.
- [159] Y. Kalfoglou, B. Hu, and D. Reynolds. On Interoperability of Ontologies for Web-based Educational Systems. In *International Workshop on Interoperability of Web-based Educational Systems*, 2005.
- [160] R. Cyganiak. Benchmarking D2RQ v0.2. Available online at www4.wiwiiss.fu-berlin.de/bizer/d2rq/benchmarks/, 2006.
- [161] T. Weithöner, T. Liebig, M. Luther, S. Böhm, F. von Henke, and O. Noppens. *The Semantic Web: Research and Applications*, volume 4519 of *Lecture Notes in Computer Science*,

- chapter Real World Reasoning with OWL, pages 296–310. Springer Berlin / Heidelberg, June 2007.
- [162] Y. Guo, Z. Pan, and J. Heflin. An Evaluation of Knowledge Base Systems for Large OWL Datasets. In *Proceedings of the 3rd International Semantic Web Conference (ISWC 2004)*, pages 274–288, 2004.
- [163] T. Weithoner, T. Liebig, M. Luther, S. Bohm, F. von Henke, and O. Noppens. Real World Reasoning with OWL. In *Proceedings of 4th European Semantic Web Conference*, volume 4519, pages 296–310. Springer, 2007.
- [164] E. Dougherty and Ph. Laplante. *Introduction to Real-Time Imaging*, chapter What is Real-Time Processing?, pages 1–9. Wiley-IEEE Press, October 1995.
- [165] G. Stamou, J. van Ossenbruggen, J.Z. Pan, G. Schreiber, and J.R. Smith. Multimedia Annotations on the Semantic Web. *Multimedia, IEEE*, 13(1):86–90, 2006.
- [166] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of cognitive neuroscience*, 3(1):71–86, March 1991.
- [167] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. *IEEE Transactions on pattern analysis and machine intelligence*, 19(7):711–720, July 1997.
- [168] M. S. Bartlett, J. R. Movellan, and T. J. Sejnowski. Face Recognition by Independent Component Analysis. *IEEE Transactions on neural networks*, 13(6):1450–1464, November 2002.
- [169] D. Comaniciu and P. Meer. Mean shift analysis and applications. In *Proceedings of the Seventh IEEE International Conference on Computer Vision (ICCV'99)*, volume 2, pages 1197–1203, Kerkyra, Greece, September 1999.
- [170] G. R. Bradski. Computer Vision Face Tracking for Use in a Perceptual User Interface. Intel Technology Journal, 2nd Quarter 1998.
- [171] B. D. Allen, G. Bishop, and G. Welch. Tracking: Beyond 15 minutes of thought. *SIG-GRAPH Course Pack*, 2001.
- [172] J. Sohn, N. S. Kim, and W. Sung. A Statistical Model-based Voice Activity Detection. *IEEE Signal Processing Letters*, 6(1):1–3, 1999.
- [173] S. Liu, M. Xu, H. Yi, L. T. Chia, and D. Rajan. Multimodal Semantic Analysis and Annotation for Basketball Video. *EURASIP Journal on Applied Signal Processing*, 2006:1–13, 2006.
- [174] C. C. Lien, C. L. Chiang, and C. H. Lee. Scene-based Event Detection for Baseball Videos. *Journal of Visual Communication and Image Representation*, 18(1):1–14, 2007.
- [175] D. Zhang and S. F. Chang. Event Detection in Baseball Video Using Superimposed Caption Recognition. In *Proceedings of the tenth ACM international conference on Multimedia*, pages 315–318. ACM New York, NY, USA, 2002.
- [176] J. Hunter. An overview of the MPEG-7 description definition language (DDL). *IEEE Transactions on Circuits and Systems for Video Technology*, 11(6):765–772, June 2001.
- [177] A. R. J. François, R. Nevatia, J. Hobbs, , and R. C. Bolles. VERL: An ontology framework for representing and annotating video events. *IEEE MultiMedia*, 12(4):76–86, 2005.
- [178] P. Viola and M. Jones. Rapid Object Detection using a Boosted Cascade of Simple Features. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 511–518, Hawaii,

- 2001.
- [179] Andreas Stergiou, Aristodemos Pnevmatikakis, and Lazaros Polymenakos. *The AIT Multimodal Person Identification System for CLEAR 2007*, pages 221–232. Springer Berlin / Heidelberg, 2007.
 - [180] G. Karame, A. Stergiou, N. Katsarakis, P. Papageorgiou, and A. Pnevmatikakis. 2D and 3D Face Localization for Complex Scenes. In *Advanced Video and Signal Based Surveillance, 2007. AVSS 2007. IEEE Conference on*, pages 371–376, 2007.
 - [181] Ph. Levis, N. Lee, M. Welsh, and D. Culler. TOSSIM: Accurate and Scalable Simulation of Entire TinyOS Applications. In *Proceedings of the First ACM Conference on Embedded Networked Sensor Systems (SenSys'03)*, pages 126–137, Los Angeles, California, USA, 2003.
 - [182] F. Franco. Uma Ferramenta de Transformação de Esquemas Objeto-Relacionais para Ontologias. Master of science thesis, Universidade Federal de Pernambuco, Centro de Informática, July 2009.
 - [183] A. Stanimirović, M. Bogdanović, and L. Stoimenov. Data Access Layer Generation for Interoperable GIS Environments. In *12th AGILE International Conference on Geographic Information Science 2009*, Leibniz Universität Hannover, Germany, June 2009.
 - [184] CEN/ISSS WS/eTOUR. Harmonization of data interchange in tourism, June 2009. Final CEN Workshop Agreement (CWA).
 - [185] J. Ruiz-Martínez, D. Castellanos-Nieves, R. Valencia-García, J. Fernández-Breis, F. García-Sánchez, P. Vivancos-Vicente, J. Castejón-Garrido, J. Camón, and R. Martínez-Béjar. *Knowledge Acquisition: Approaches, Algorithms and Applications*, chapter Accessing Touristic Knowledge Bases through a Natural Language Interface, pages 147–160. Lecture Notes in Computer Science, Volume 5465. Springer-Verlag, May 2009.
 - [186] R. Ghawi and N. Cullot. Building Ontologies from Multiple Information Sources. In *Proceedings of the 15th International Conference on Information and Software Technologies (IT'09)*, Kaunas, Lithuania, April 2009.
 - [187] S. Sane and A. Shirke. Generating OWL ontologies from Relational Databases for the Semantic Web. In *Proceedings of the International Conference on Advances in Computing, Communication and Control (ICAC3'09)*, pages 157–162, Mumbai, India, January 2009. ACM.
 - [188] M. Wanigasooriya, V. Atapattu, T. Jayatunga, V. Gamage, S. Mohideen, and S. Rajapaksha. Generating Ontology from Relational Database Schema. In Faiz MMT Marikar, editor, *Proceedings of the Sri Lanka Institute of Information Technology (PSLIIT)*, volume 2 of *PSLIIT*, pages 50–54, Sri Lanka Institute of Information Technology, Sri Lanka, December 2008.
 - [189] Chen He-ping, He Lu, and Chen Bin. Research and Implementation of Ontology Automatic Construction Based on Relational Database. In *International Conference on Computer Science and Software Engineering*, volume 5 of *CSSE*, pages 1078–1081, Wuhan, China, December 2008. IEEE.
 - [190] M. Martínez, J. M. Vázquez, J. Pereira, and A. Pazos. Annotation of Colorectal Cancer Data Using the UMLS Metathesaurus. In *KES (2)*, volume 5178 of *Lecture Notes in Computer Science*, pages 58–65. Springer, 2008.
 - [191] K. Munir, M. Odeh, and R. McClatchey. Ontology Assisted Query Reformulation Using the Semantic and Assertion Capabilities of OWL-DL Ontologies. In *IDEAS '08: Proceedings*

- of the 2008 international symposium on Database engineering & applications*, pages 81–90, New York, NY, USA, September 2008. ACM. ISBN 978-1-60558-188-0.
- [192] K. Stewart Hornsby and K. King. Modeling Motion Relations for Moving Objects on Road Networks. *GeoInformatica*, 12(4):477–495, December 2008.
- [193] K. Stewart Hornsby and K. King. *Geosensor Networks*, chapter Linking Geosensor Network Data and Ontologies to Support Transportation Modeling, pages 191–209. Number 4540 in Lecture Notes in Computer Science. Springer-Verlag, Berlin Heidelberg, 2008.
- [194] K. Munir, M. Odeh, P. Bloodsworth, and R. McClatchey. Using Assertion Capabilities of an OWL-Based Ontology for Query Formulation. In *3rd International Conference on Information and Communication Technologies: From Theory to Applications (ICTTA 2008)*, pages 1–6, April 2008.
- [195] I. Astrova, N. Korda, and A. Kalja. Storing OWL Ontologies in SQL Relational Databases. *International Journal of Electrical, Computer, and Systems Engineering*, 1(4):242–247, 2007.
- [196] I. Astrova, N. Korda, and A. Kalja. Rule-based transformation of SQL relational databases to OWL ontologies. In *Proceedings of the 2nd International Conference on Metadata and Semantics Research (MTSR 2007)*, pages 1–16, Ionian Academy, Corfu, Greece, October 2007.
- [197] I. Astrova, A. Kalja, and N. Korda. Automatic Transformation of OWL Ontologies to SQL Relational Databases. In *IADIS European Conference on Data Mining (ECDM 2007)*, pages 145–149, Lisbon, Portugal, July 2007.
- [198] N. Cullot, R. Ghawi, and K. Yétongnon. DB2OWL: A Tool for Automatic Database-to-Ontology Mapping. In *Proceedings of the 15th Italian Symposium on Advanced Database Systems (SEBD 2007)*, pages 491–494, Torre Canne di Fasano (BR), Italy, June 2007.
- [199] R. Ghawi and N. Cullot. Database-to-Ontology Mapping Generation for Semantic Interoperability. In *VLDB 07*, Vienna, Austria, September 2007. ACM.
- [200] K. King. Linking Moving Object Databases with Ontologies. Master of science thesis, The University of Maine, December 2007.
- [201] M. Santorinaios. *Une Approche sur la Recherche et l’ Art à l’ Époque du Numérique à travers l’ Étude des Systèmes Artificiels d’ Organisation de la Mémoire : Proposition d’ un Outil d’ Enregistrement et de Synthèse des Données Susceptible de Contribuer à la Mise en Place d’ une Nouvelle Forme de Doctorat (“Doctorat_machine”)*. Phd thesis, Université Paris 8 – Vincennes Saint-Denis, October 2006.

Πίνακας ακρωνυμίων

ΒΔ	Βάση Δεδομένων
ΣΔΒΔ	Σύστημα Διαχείρισης Βάσης Δεδομένων
AJAX	Asynchronous Javascript And Xml
API	Application Programming Interface
CMS	Content Management System
CORBA	Common Object Request Broker Architecture
CVS	Concurrent Versions System
DAI	Distributed Artificial Intelligence
DAML	DARPA Agent Markup Language
DAML-S	DAML for Services
DARPA	Defense Advanced Research Project Agency
DC	Dublin Core
DDL	Description Definition Language
DIG	DL Implementation Group
DL	Description Logics
DOAP	Description Of A Project
DTD	Document Type Definition
FIPA	Foundation for Intelligent Physical Agents
FOAF	Friend Of A Friend
FOL	First Order Logic
GAV	Global As View ή global-schema-centric
GLAV	Global Local As View
GNU	GNU's Not Unix (αναδρομική δήλωση ακρωνυμίου)
GRDDL	Gleaning Resource Descriptions from Dialects of Languages
GSS	Graph StyleSheets
HTML	HyperText Markup Language
IEEE	Institute of Electrical and Electronics Engineers
IRI	Internationalized Resource Identifier
J2EE	Java 2 platform, Enterprise Edition
JADE	Java Agent DEveloper Framework
JDBC	Java DataBase Connectivity
KIF	Knowledge Interchange Format
LAV	Local As View ή source-centric
MIME	Multipurpose Internet Mail Extensions
MDE	Model Driven Engineering
MDS	Multimedia Description Schema
MPEG	Moving Pictures Expert Group
OAI-PMH	Open Archives Initiative Protocol for Metadata Harvesting
ODMG	Object Database Management Group
OIL	Ontology Interchange Layer
OLAP	On-Line Analytical Processing

Πίνακας Ακρωνυμίων

OLCD	Object Language with Complements allowing Descriptive cycles
OMG	Object Management Group
OWL	Web Ontology Language
OWL-S	OWL for Services
PHP	PHP: Hypertext Preprocessor (αναδρομική δήλωση ακρωνυμίου) Αρχικά είχε σχεδιαστεί να σημαίνει Personal Home Page
PNG	Portable Network Graphics
RDF	Resource Description Framework
RDFS	RDF Schema
RDQL	Resource Description Query Language
RFC	Request for Comments
RFID	Radio-Frequency IDentification
RMI	Remote Method Invocation
RPC	Remote Procedure Call
RQL	Rdf Query Language
RSS	Really Simple Syndication (RSS 2.0) ή RDF Site Summary (RSS 1.0 και RSS 0.90) ή Rich Site Summary (RSS 0.91)
SeRQL	Sesame Rdf Query Language
SHA	Secure Hash Algorithm
SIOC	Semantically-Interlinked Online Communities
SMS	Short Message Service
SOAP	Simple Object Access Protocol
SPARQL	Simple Protocol And Rdf Query Language
SQL	Structured Query Language
SVG	Scaleable Vector Graphic
SWRL	Semantic Web Rule Language
SWSF	Semantic Web Services Framework
SWSL	Semantic Web Services Language
TS	Technical Space
UDDI	Universal Description, Discovery and Integration
UMIACS	University of Maryland Institute for Advanced Computer Studies
URI	Universal Resource Identifier
W3C	World Wide Web Consortium
WSDL	Web Services Description Language
XHTML	eXtensible HyperText Markup Language
XMI	XML Metadata Interchange
XML	eXtensible Markup Language
XSD	XML Schema Definition
XSL	eXtensible Stylesheet Language

Παράρτημα

Στα παραρτήματα Α και Β παρατίθενται σε σύνταξη BNF⁷⁷ οι γλώσσες κανόνων που αναπτύχθηκαν για την επεξεργασία της πληροφορίας από το μεσισμικό που περιγράφεται στο κεφάλαιο 5.

Α. Γλώσσα Κανόνων αντιστοίχισης

Μέσω της γλώσσας κανόνων αντιστοίχισης μπορεί να αντιστοιχηθούν στοιχεία ενός XML εγγράφου σε έννοιες και άτομα μιας οντολογίας.

```

<if> ::= ( <xml element exists>
          | <xml element does not exist>
          | <xml element has value>
          | <xml element has siblings> )
<xml element exists> ::= XPathExpression ( <and> | <then> )
<xml element does not exist> ::= XPathExpression ( <and> | <then> )
<xml element has value> ::= XPathExpression
                          ( gt | lt | gte | lte | eq | neq )
                          ( Integer | String ) ( <and> | <then> )
<xml element has siblings> ::= XPathExpression
                              ( <and> | <then> )
<and> ::= ( <xml element exists>
            | <xml element does not exist>
            | <xml element has value>
            | <xml element has siblings> )
<then> ::= <insert individual in class>
          | <insert exactly one individual in class>
          | <execute SPARQL query>
<insert individual in class> ::= OntClass [ <named after> ]
                              [ <and set object property> |
                                ( <and set datatype property> )* ]
<named after> ::= XPathExpression
<insert exactly one individual in class> ::= OntClass
<and set object property> ::= OntProperty
                          ( <last inserted individual in class> |
                            <specific individual in class> )

```

⁷⁷Η Backus-Naur Form (BNF) είναι μια μετα-σύνταξη η οποία χρησιμοποιείται για τη δήλωση context-free γραμματικών.

```

<last inserted individual in class> ::= OntClass
<specific individual in class> ::= Individual
<and set datatype property> ::= OntProperty
    ( XPathExpression | String )
<execute SPARQL query> ::= String [ <and set ?variable value> ]
<and set ?variable value> ::= ( XPathExpression | String )

```

B. Γλώσσα Σημασιολογικών Κανόνων

Με τη χρήση σημασιολογικών κανόνων μπορούν να πυροδοτηθούν ενέργειες που να αφορούν είτε την οντολογία είτε το εξωτερικό περιβάλλον του μεσισμικού.

```

<if> ::= ( <class has individuals>
    | <class has subclasses>
    | <number of individuals in class>
    | <datatype property in class>
    | <SPARQL query has results>
    | <SPARQL query has number of results>
    | <SPARQL query does not have results> )
<class has individuals> ::= OntClass ( <and> | <then> )
<class has subclasses> ::= OntClass ( <and> | <then> )
<number of individuals in class> ::= OntClass
    ( gt | lt | gte | lte | eq | neq )
    ( Integer | String ) ( <and> | <then> )
<datatype property in class> ::= OntClass OntProperty
    ( gt | lt | gte | lte | eq | neq )
    ( Integer | String ) ( <and> | <then> )
<SPARQL query has results> ::= String ( <and> | <then> )
<SPARQL query has number of results> ::= String
    ( gt | lt | gte | lte | eq | neq )
    Integer ( <and> | <then> )
<SPARQL query does not have results> ::= String ( <and> | <then> )
<and> ::= ( <class has individuals>
    | <class has subclasses>
    | <number of individuals in class>
    | <datatype property in class>
    | <SPARQL query has results>
    | <SPARQL query has number of results>
    | <SPARQL query does not have results> )

<then> ::= ( <insert individual in class>

```

```
| <insert exactly one individual in class>
| <insert subclass of>
| <execute a system command>
| <execute a system command once>
| <send Web Service message>
| <execute SPARQL query> )
<insert individual in class> ::= OntClass
<insert exactly one individual in class> ::= OntClass
<insert subclass of> ::= OntClass String
<execute a system command> ::= PathToExecutable
<execute a system command once> ::= PathToExecutable
<send Web Service message> ::= String
<execute SPARQL query> ::= String [ <and set ?variable value> ]
<and set ?variable value> ::= String
```

Γ. Κανόνες υποστήριξης για το σενάριο ασφάλειας

1. Κανόνες αντιστοίχισης

```
1a. if,xml element has value,
/Event/Tracker/person/face/@dbpersonid,lt,10,
then,
insert individual in class,Student,
named after,/Event/Tracker/person/@id,
and set datatype property,hasXLocation,
/Event/Tracker/person/location2d/@x,
and set datatype property,hasYLocation,
/Event/Tracker/person/location2d/@y,
and set datatype property,hasCertainty,
/Event/Tracker/person/face/@certainty,
and set datatype property,hasTime,
/Event/Tracker/TimeStamp/@value,
and set datatype property,hasName,
/Event/Tracker/person/face/@name
```

```
1b. if,xml element has value,/Event/Tracker/person/face/@dbpersonid,gt,10,
then,insert individual in class,Professor,
named after,/Event/Tracker/person/@id,
and set datatype property,hasXLocation,
/Event/Tracker/person/location2d/@x,
```

```
and set datatype property,hasYLocation,  
/Event/Tracker/person/location2d/@y,  
and set datatype property,hasCertainty,  
/Event/Tracker/person/face/@certainty,  
and set datatype property,hasTime,  
/Event/Tracker/TimeStamp/@value,  
and set datatype property,hasName,  
/Event/Tracker/person/face/@name
```

2. Σημασιολογικοί κανόνες

```
2a. if,class has individuals,Student,  
and,  
SPARQL query does not have results,  
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>  
PREFIX a: <http://www.example.org#>  
SELECT ?x WHERE { ?x rdf:type a:Professor},  
and,  
SPARQL query does not have results,  
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>  
PREFIX a: <http://www.example.org#>  
SELECT ?x WHERE { ?x rdf:type a:Action },  
then,  
send Web Service message,"Intruder Detected!"
```

```
2b. if,class has individuals,Student,  
and,  
SPARQL query does not have results,  
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>  
PREFIX a: <http://www.example.org#>  
SELECT ?x WHERE { ?x rdf:type a:Professor },  
then,  
insert exactly one individual in class,Action
```

Δ. Κανόνες υποστήριξης για το σύστημα ευφυούς παρακολούθησης συνεδριακών χώρων

1. Κανόνες αντιστοίχισης

```
1a. if,xml element exists,/Event/Tracker/person/@id,then,
```



```
insert individual in class, Person,  
named after, /Event/Tracker/person/@id, AND  
set datatype property, hasYLocation,  
/Event/Tracker/person/location2d/@y,  
and set datatype property, hasXLocation,  
/Event/Tracker/person/location2d/@x,  
and set datatype property,  
hasTime, /Event/Tracker/TimeStamp/@value
```

2. Σημασιολογικοί κανόνες

```
2a. if, number of individuals in class,  
Person, gt, 3, and,  
datatype property in class, System, hasState, eq, Initial_State, then,  
send Web Service message, "Meeting Started!"  
2b. if, number of individuals in class,  
Person, gt, 3, and,  
datatype property in class, System, hasState, eq, Initial_State, then,  
execute SPARQL query,  
PREFIX a: <http://www.example.org/meeting.owl#>  
DELETE { ?x a:hasState ?z }  
WHERE { ?x a:hasState ?z }  
INSERT { a:System_State a:hasState 'Meeting_Started' }  
2c. if, datatype property in class,  
System, hasState, eq, Meeting_Started, and,  
datatype property in class, Person, hasXLocation, gt, 400, and,  
datatype property in class, Person, hasYLocation, gt, 300, then,  
send Web Service message, "Presentation Started!"  
2d. if, datatype property in class,  
System, hasState, eq, Meeting_Started, and,  
datatype property in class, Person, hasXLocation, gt, 400, AND,  
datatype property in class, Person, hasYLocation, gt, 300, then,  
execute SPARQL query,  
PREFIX a: <http://www.example.org/meeting.owl#>  
DELETE { ?x a:hasState ?z }  
WHERE { ?x a:hasState ?z }  
INSERT { a:System_State a:hasState 'Presentation_Started' }  
2e. if, datatype property in class,  
System, hasState, eq, Presentation_Started, and,  
SPARQL query does not have results,  
PREFIX a: <http://www.example.org/meeting.owl#>
```

```
SELECT ?x ?y
WHERE { ?x a:hasXLocation ?y FILTER ( ?y > '400' ) },and,
SPARQL query does not have results,
PREFIX a: <http://www.example.org/meeting.owl#>
SELECT ?x ?y
WHERE { ?x a:hasYLocation ?y FILTER ( ?y > '300' ) },then,
send Web Service message,"Presentation Ended!"
2f. if,datatype property in class,
System,hasState,eq,Presentation_Started,and,
SPARQL query does not have results,
PREFIX a: <http://www.example.org/meeting.owl#>
SELECT ?x ?y WHERE { ?x a:hasXLocation ?y FILTER ( ?y > '400' ) },and,
SPARQL query does not have results,
PREFIX a: <http://www.example.org/meeting.owl#>
SELECT ?x ?y
WHERE { ?x a:hasYLocation ?y FILTER ( ?y > '300' ) },then,
execute SPARQL query,
PREFIX a: <http://www.example.org/meeting.owl#>
DELETE { ?x a:hasState ?z }
WHERE { ?x a:hasState ?z }
INSERT { a:System_State a:hasState 'Presentation_Ended' }
2g. if,datatype property in class,
System,hasState,eq,Presentation_Ended,and,
SPARQL query does not have results,
PREFIX a: <http://www.example.org/meeting.owl#>
SELECT ?x ?y
WHERE { ?x a:hasXLocation ?y FILTER ( ?y > '10' ) },THEN,
send Web Service message,"Meeting Ended!"
2h. if,datatype property in class,
System,hasState,eq,Presentation_Ended,and,
SPARQL query does not have results,
PREFIX a: <http://www.example.org/meeting.owl#>
SELECT ?x ?y
WHERE { ?x a:hasXLocation ?y FILTER ( ?y > '10' ) },then,
execute SPARQL query,
PREFIX a: <http://example.org/meeting.owl#>
DELETE { ?x a:hasState ?z }
WHERE { ?x a:hasState ?z }
INSERT { a:System_State a:hasState 'Meeting_Ended' }
```

E. Οντολογία για το σενάριο παρακολούθησης (σε μορφοποίηση Turtle)

```
# Base: http://www.example.org/example.owl#
@prefix rdfs:    <http://www.w3.org/2000/01/rdf-schema#> .
@prefix default: <http://www.example.org#> .
@prefix xsd:    <http://www.w3.org/2001/XMLSchema#> .
@prefix owl:  <http://www.w3.org/2002/07/owl#> .
@prefix rdf:    <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

default:Professor
    rdf:type owl:Class ;
    rdfs:subClassOf default:Staff .
default:Location
    rdf:type owl:Class ;
    owl:disjointWith default:Event .
default:Staff
    rdf:type owl:Class ;
    rdfs:subClassOf default:Known .
default:Unknown
    rdf:type owl:Class ;
    rdfs:subClassOf default:Actor ;
    owl:disjointWith default:Known .
default:Classroom
    rdf:type owl:Class ;
    rdfs:subClassOf default:Indoors .
default:hasName
    rdf:type owl:DatatypeProperty ;
    rdfs:domain default:Human ;
    rdfs:range xsd:string .
default:Actor
    rdf:type owl:Class ;
    owl:disjointWith default:Event , default:Location .
default:Postgraduate
    rdf:type owl:Class ;
    rdfs:subClassOf default:Student .
default:Human
    rdf:type owl:Class ;
    rdfs:subClassOf default:Actor .
default:Event
    rdf:type owl:Class ;
```

```
    owl:disjointWith default:Location .
default:participates
    rdf:type owl:ObjectProperty ;
    rdfs:domain default:Actor ;
    rdfs:range default:Event .
default:Action
    rdf:type owl:Class ;
    rdfs:subClassOf default:Event .
default:Male
    rdf:type owl:Class ;
    rdfs:subClassOf default:Human .
default:Meeting_Room
    rdf:type owl:Class ;
    rdfs:subClassOf default:Indoors .
default:Visitor
    rdf:type owl:Class ;
    rdfs:subClassOf default:Known .
default:Female
    rdf:type owl:Class ;
    rdfs:subClassOf default:Human .
default:isLocated
    rdf:type owl:ObjectProperty ;
    rdfs:domain default:Event ;
    rdfs:range default:Location .
default:hasCertainty
    rdf:type owl:DatatypeProperty ;
    rdfs:domain default:Human ;
    rdfs:range xsd:string .
default:Enter
    rdf:type owl:Class ;
    rdfs:subClassOf default:Appear .
default:Disappear
    rdf:type owl:Class ;
    rdfs:subClassOf default:Action ;
    owl:disjointWith default:Appear .
default:Leave
    rdf:type owl:Class ;
    rdfs:subClassOf default:Disappear .
default:hasYLocation
    rdf:type owl:DatatypeProperty ;
    rdfs:domain default:Human ;
```

```
        rdfs:range xsd:string .
default:hasTime
    rdf:type owl:DatatypeProperty ;
    rdfs:domain default:Device ;
    rdfs:range xsd:string .
default:isDetectedFrom
    rdf:type owl:DatatypeProperty ;
    rdfs:domain default:Device ;
    rdfs:range xsd:string .
default:Appear
    rdf:type owl:Class ;
    rdfs:subClassOf default:Action ;
    owl:disjointWith default:Disappear .
default:Undergraduate
    rdf:type owl:Class ;
    rdfs:subClassOf default:Student .
default:hasId
    rdf:type owl:DatatypeProperty ;
    rdfs:domain default:Event ;
    rdfs:range xsd:string .
default:Indoors
    rdf:type owl:Class ;
    rdfs:subClassOf default:Location ;
    owl:disjointWith default:Outdoors .
default:Vehicle
    rdf:type owl:Class ;
    rdfs:subClassOf default:Device .
default:Human_Enters
    rdf:type default:Event .
default:Known
    rdf:type owl:Class ;
    rdfs:subClassOf default:Actor ;
    owl:disjointWith default:Unknown .
default:hasDescription
    rdf:type owl:ObjectProperty .
default:Student
    rdf:type owl:Class ;
    rdfs:subClassOf default:Known .
default:hasXLocation
    rdf:type owl:DatatypeProperty ;
    rdfs:domain default:Human ;
```

```
        rdfs:range xsd:string .
<http://www.example.org>
        rdf:type owl:Ontology .
default:Outdoors
        rdf:type owl:Class ;
        rdfs:subClassOf default:Location ;
        owl:disjointWith default:Indoors .
```

ΣΤ. Τμήμα κώδικα για επεξεργασία οντολογίας σε Java

Το τμήμα κώδικα που παρατίθεται στη συνέχεια αναλαμβάνει την ανάγνωση ενός μοντέλου οντολογίας OWL (ontModel) από τη Βάση Δεδομένων και στη συνέχεια αντιστοιχίζει έναν reasoner σε αυτό. Η ανάγνωση αυτή είναι persistent: ό,τι αλλαγές γίνονται στη μνήμη αποθηκεύονται ταυτόχρονα και στη Βάση Δεδομένων. Ο reasoner μπορεί να είναι ο εσωτερικός μηχανισμός κανόνων της Jena ή εξωτερικός, ο οποίος θα δέχεται μηνύματα στο strReasonerURI, για παράδειγμα στο <http://localhost:3490>.

```
// Πρώτα δημιουργείται ένα baseModel, το κενό μοντέλο.
Model baseModel;
ModelMaker maker = ModelFactory.createModelRDBMaker(getJenaConn());

// Αποθηκεύεται στη μνήμη το μοντέλο που βρίσκεται στη Βάση Δεδομένων.
baseModel = maker.createModel(strOntologyName, false);

OntModelSpec spec = null;

// Προσθήκη reasoner.
if (bInternalReasoner) {
    spec = new OntModelSpec(OntModelSpec.OWL_MEM_RULE_INF);
} else {
    spec = new OntModelSpec(OntModelSpec.OWL_MEM);
    Resource config = baseModel.createResource("reasoner");
    config.addProperty(ReasonerVocabulary.EXT_REASONER_URL,
        baseModel.createResource(strReasonerURI));

    DIGReasonerFactory drf = (DIGReasonerFactory) ReasonerRegistry
        .theRegistry().getFactory(DIGReasonerFactory.URI);
    DIGReasoner r = (DIGReasoner) drf.createWithOWLAxioms(config);
    spec.setReasoner(r);
}
```

```
OntModel ontModel = ModelFactory.createOntologyModel(spec, baseModel);
```

Z. Μεταφορά τριάδων από μια οντολογία σε μια άλλη με χρήση SPARQL

Για τη μεταφορά τριάδων από την προσωρινή (buffer) Βάση Γνώσεως στη μόνιμη χρησιμοποιείται το παρακάτω ερώτημα SPARQL (αντίστοιχα με το <http://jena.hpl.hp.com/~afs/SPARQL-Update.html>). Η δυνατότητα του ερωτήματος βασίζεται στη δυνατότητα εξαγωγής γράφων της SPARQL.

```
INSERT INTO <http://example.org/permanent>
  { ?s ?p ?o }
WHERE
  { GRAPH <http://example.org/buffer>
    {
      ?s ?p ?o
    }
  }
```

```
DELETE FROM <http://example.org/buffer>
  { ?s ?p ?o }
WHERE
  { GRAPH <http://example.org/buffer>
    {
      ?s ?p ?o
    }
  }
```

Αν όμως για τη μεταφορά τριάδων από την προσωρινή Βάση στη μόνιμη το επιθυμητό είναι να αντιγραφούν *μόνο* τα individuals της οντολογίας ενώ το σχήμα να διατηρείται το ίδιο, η πληροφορία που αφορά τα individuals μπορεί να ανακτηθεί με τη χρήση ενός SPARQL ερωτήματος όπως το ακόλουθο:

```
INSERT INTO <http://example.org/permanent>
  { ?member ?p ?o }
WHERE
  { GRAPH <http://example.org/buffer>
    {
      ?class a owl:Class .
    }
  }
```

```
?member a ?class .  
?member ?p ?o .  
}  
}
```

```
DELETE FROM <http://example.org/buffer>  
{ ?member ?p ?o }  
WHERE  
{ GRAPH <http://example.org/buffer>  
{  
  ?class a owl:Class .  
  ?member a ?class .  
  ?member ?p ?o .  
}  
}
```

Στο παραπάνω SPARQL ερώτημα, πρώτα εισάγονται στην μόνιμη Βάση Γνώσεως τα individuals της προσωρινής ενώ στη συνέχεια διαγράφονται από την προσωρινή. Σημειώνεται ότι το construct DELETE δεν έχει ενσωματωθεί ακόμη στον επίσημο ορισμό της SPARQL, αναμένεται σύντομα όμως να ενσωματωθεί καθώς ήδη είναι διαθέσιμο σε πολλές υλοποιήσεις της SPARQL (<http://esw.w3.org/topic/SPARQL/Extensions/Update>).

Κατάλογος δημοσιεύσεων του συγγραφέα

Διεθνή επιστημονικά περιοδικά με κριτές

1. N. Konstantinou, E. Solidakis, A. Zafeiropoulos, P. Stathopoulos, N. Mitrou: A Context-aware Middleware for Real-Time Semantic Enrichment of Distributed Multimedia Metadata, In *International Journal of Multimedia Tools and Applications*, Springer, special issue on Data Semantics for Multimedia Systems (doi:10.1007/s11042-009-0361-1)
2. A. Zafeiropoulos, I. Papaioannou, E. Solidakis, N. Konstantinou, P. Stathopoulos, N. Mitrou: Exploiting Bluetooth for deploying indoor LBS over a localization infrastructure independent architecture, In *International Journal of Computer Aided Engineering Technology (IJCAET)*, special issue on Smart Homes: Technologies and Applications, Vol. 2, No. 2 (2010, to appear)
3. N. Konstantinou, D.-E. Spanos, and N. Mitrou. Ontology and Database Mapping: A Survey of Current Implementations and Future Directions. *Journal of Web Engineering*, 7(1):1–24, March 2008.

Κεφάλαια σε βιβλία

1. A. Zafeiropoulos, D.E. Spanos, S. Arkoulis, N. Konstantinou, N. Mitrou: chapter Data Management in Sensor Networks Using Semantic Web Technologies, In *Data Management in Semantic Web* (Eds. Hai Jin and Zehua Lv), to be published by Nova Publishers

Διεθνή συνέδρια με κριτές

1. A. Zafeiropoulos, N. Konstantinou, S. Arkoulis, D.-E. Spanos, and N. Mitrou. A Semantic-Based Architecture for Sensor Data Fusion. In *Second International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM'08)*, Valencia, Spain, October 2008.
2. N. Konstantinou, E. Solidakis, S. Zoi, A. Zafeiropoulos, P. Stathopoulos, and N. Mitrou. Priamos: A Middleware Architecture for Real-Time Semantic Annotation of Context Features. In *IET International Conference on Intelligent Environments (IE'07)*, Ulm, Germany, September 2007.

3. A. Zafeiropoulos, E. Solidakis, S. Zoi, N. Konstantinou, P. Papageorgiou, P. Stathopoulos, and N. Mitrou. A lightweight approach for providing Location Based Content Retrieval. In *18th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'07)*, pages 1–5, Athens, Greece, September 2007.
4. A. Zafeiropoulos, E. Solidakis, S. Zoi, N. Konstantinou, P. Stathopoulos, and N. Mitrou. Location Based Guidance Services in a Museum Environment: Deployment Issues and a Proposed Architectural Approach. In *International Conference on Wireless Information Networks and Systems (WINSYS'07)*, Barcelona, Spain, September 2007.
5. N. Konstantinou, D.-E. Spanos, M. Chalas, E. Solidakis, and N. Mitrou. VisAVis: An Approach to an Intermediate Layer between Ontologies and Relational Database Contents. In *WISM'06, on the 18th Conference on Advanced Information Systems Engineering (CAiSE'06)*, pages 1050–1061, Luxembourg, June 2006.
6. P. Stathopoulos, S. Zoi, N. Konstantinou, E. Solidakis, C. Basios, A. Zafeiropoulos, P. Papageorgiou, and N. Mitrou. E-Museum - A Content Management System for Providing Museum Visitors with Personalized Audiovisual Information. In *3rd International Conference of Museology*, Mytilene, Greece, June 2006.
7. E. Solidakis, N. Konstantinou, E.S. Pashou, A. Papakonstantinou, and N. Mitrou. A Decentralized Multi-Agent Ontology-Based System for Information Retrieval. In *SWAMM'06, on the 15th World Wide Web Conference (WWW'06)*, Edinburgh, May 2006.