



«DARIAH-ΑΤΤΙΚΗ Ανάπτυξη της ελληνικής ερευνητικής υποδομής για τις ανθρωπιστικές επιστήμες ΔΥΑΣ»

ΑΚΑΔΗΜΙΑ ΑΘΗΝΩΝ

ΕΡΕΥΝΗΤΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΑΚΟ ΙΝΣΤΙΤΟΥΤΟ

ΕΚ ΑΘΗΝΑ

ΣΥΣΤΗΜΑΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

ΑΝΩΤΑΤΗ ΣΧΟΛΗ ΚΑΛΩΝ ΤΕΧΝΩΝ

ΙΔΡΥΜΑ ΤΕΧΝΟΛΟΓΙΑΣ ΚΑΙ ΕΡΕΥΝΑΣ

ΥΠΟΕΡΓΟ 4

Δράσεις ανάπτυξης της εθνικής υποδομής του
ΕΠΙΣΕΥ

Παραδοτέο

ΠΑ.1.4.ΕΠΙΣΕΥ.2 Τεχνολογίες και πλατφόρμες
σημασιολογικής αναπαράστασης της γνώσης
στην απάντηση των επερωτημάτων

Έκδοση 1.0-Τελική
6/3/2015

Έγγραφο:

Ταυτότητα Εγγράφου

ΥΠΟΕΡΓΟ	4. Δράσεις ανάπτυξης της εθνικής υποδομής του ΕΠΙΣΕΥ
Ενότητα εργασιών ΔΡΑΣΗ	1. ΚΟΙΝΩΝΙΑ ΨΗΦΙΑΚΩΝ ΠΟΡΩΝ
ΠΑΡΑΔΟΤΕΟ	1.4. Μητρώο λογισμικών υπηρεσιών
Εργασία	ΠΑ.1.4.ΕΠΙΣΕΥ.2 Τεχνολογίες και πλατφόρμες σημασιολογικής αναπαράστασης της γνώσης στην απάντηση των επερωτημάτων Συγγραφή εγγράφου
Ομάδα 'Εργου	Στέφανος Κόλλιας Γεώργιος Μαρανδιανός Φοίβος Μυλωνάς Αλέξανδρος Χορταράς
Χαρακτηρισμός	Εσωτερικό Ομάδας 'Εργου

Ιστορικό Αλλαγών

Έκδ.	Ημ/μηνία	Αιτιολογία	Σύνοψη Αλλαγών	Επισπεύδοντες
0.1	3/3/2015	Αρχική	Δημιουργία και Σύνταξη	Ομάδα 'Εργου
1.0	6/3/2015	Τελική		

Περιεχόμενα

1 Εισαγωγή	3
2 Σημασιολογική μοντελοποίηση	5
2.1 Οντολογίες	5
2.1.1 Δομή	6
2.1.2 Περιγραφικές Λογικές	8
2.2 Συλλογιστική	11
2.2.1 Απάντηση Ερωτημάτων	12
2.3 Πρότυπα	15
2.3.1 RDF	17
2.3.2 RDF Schema	18
2.3.3 OWL	19
2.3.4 Λεξιλόγια	25
2.3.5 Σειριοποίηση γράφων RDF	26
2.3.6 Γλώσσα ερωτημάτων SPARQL	28
3 Σημασιολογική ολοκλήρωση δεδομένων	31
3.1 Διασύνδεση σχεσιακών βάσεων με οντολογίες	32
3.1.1 Άμεση απεικόνιση	33
3.1.2 R2RML	34
4 Συστήματα και εργαλεία	39
4.1 Εργαλεία επεξεργασίας οντολογιών	40
4.1.1 Protégé	40
4.2 Εργαλεία συλλογιστικής	40
4.2.1 Hermit	40
4.2.2 Pellet	41
4.2.3 FaCT	42
4.2.4 Racer	43
4.3 Εργαλεία απάντησης σε συζευκτικά ερωτήματα	44
4.3.1 KAGON	44
4.3.2 CLIPPER	44
4.3.3 Mastro	44
4.3.4 Requiem	45
4.3.5 IQAROS	45
4.3.6 Rapid	46
4.3.7 HydrOwl	46
4.3.8 Nyaya	46
4.4 Εργαλεία απάντησης ερωτημάτων	47

4.4.1	Jena – Fuseki	47
4.4.2	Sesame	48
4.4.3	OWLim	48
4.4.4	AllegroGraph	49
4.4.5	Stardog	50
4.4.6	OpenLink Virtuoso	51
4.4.7	OWL-BGP	52
5	Συμπεράσματα	53

1 Εισαγωγή

Στο παρόν παραδοτέο εξετάζεται ένα σύνολο θεωρητικών ζητημάτων, τεχνολογιών και συστημάτων που σχετίζονται με το αντικείμενο της σημασιολογικής αναπαράστασης γνώσης και δεδομένων. Το αντικείμενο αυτό, που εν συντομίᾳ ασχολείται με το πώς οι μηχανές, και συγκεκριμένα τα υπολογιστικά συστήματα, μπορούν να διαχειριστούν και να επεξεργαστούν ένα σύνολο δεδομένων αναγνωρίζοντας αυτόμata κάποia πράγμata πou αφορούn tō nόyma tou (tēn σημασιολογίā tou), ώστe νa μporoύn p.χ. na eξagάgyouν suμperάsmata me báσi kápoia uпokeimene γnώση, ópouς mporéi me eukolía na kánēi énaç ánthroπoç, aσfałowç dēn eínaç kainoúrgio. Diáforoi trópoi anaparásτasηs γnώσηs me autou tou eídonz ta xaraktēriistikā e\xouν apascholj̄sei apó kaiρo tēn teχnηtή vnoymosúnη kai e\xouν odiȳhssei stēn epivón̄hs tēpōw anaparásτasηs γnώσηs ópouς ta sηmasioloyikā díktua, oī kanoñes kai η maθηmatikή loγikή.

Ωστόσο, το ζήτημα έχει αναζωπυρωθεί τα τελευταία χρόνια λόγω των ταχύτατων τεχνολογικών εξελίξεων και της εισβολής του διαδικτύου και του παγκόσμιου ιστού σε κάθε τομέα της ανθρώπινης δραστηριότητας. Λόγω του τεράστιου όγκου δεδομένων που είναι πλέον δημοσιευμένα στον παγκόσμιο ιστό και ο οποίος αυξάνεται διαρκώς, τίθεται το πρόβλημα μιας πιο ορθολογικής διαχείρισής tou, ώστe na apoteleí mia w̄s epí to plēistou ámiorph̄ kai aномoioygen̄ māza plēr̄ofořiōn kai na apokt̄ssei mia piø dom̄mēn̄ mofr̄ w̄s pr̄os to periech̄mēn̄ tou, η oπoia na diεukolūnei tis m̄χanēs na diaχeirízontai piø e\xupna tēn plēr̄ofořia, pr̄os ófrel̄os aσfałowç tw̄n tēlikōn χr̄hstōn. Pr̄áymat̄, to megalunt̄ero mēros tēs plēr̄ofořiās pou n̄párχei stōn paγkōsmio istó diatíθetai s̄hmera me t̄ mofr̄ istosel̄idw̄, eítē statikōn eítē kataskeuaz̄mewon dñnamikā apó t̄ periech̄mēn̄ miaç βáṣeis dēdōmēnōn, pou diaſunđéontai metax̄n tōn mēsow upeřsunđēsmōn. Paróti e\xouν anaptyx̄h̄eí pár̄a pollež̄ teχnoloyic̄s pou diεukolūnou tēn autómat̄ kataskeuñ̄ kai enηmérw̄s̄ istosel̄idw̄ apó uпokeimena montéla dēdōmēnōn, pr̄osférōntas parállyla e\xai retikēs dñnnatōt̄tecs all̄h̄lepídras̄s̄ me tōn χr̄hst̄, to baſikō montélo anaparásτas̄ tēs plēr̄ofořiās paraménei ouſiaſtikā analloliōt̄ apó tēn eisagawḡt̄ tou paγkōsmiou istoū. Pr̄ókeitaι ḡia sel̄idēs pou periech̄ouν adóm̄hto periech̄mēn̄ keīmēnou, me tēn oπoia dom̄ na aforā apokleistikā tōn tēpōto parouſiast̄s̄ tou kai óchi tō nōyma tou. Oi ánthrōpoi kai oī m̄χanēs mporoūn na diaþázouν autēs tis sel̄idēs kai na akolouθouν tōn upeřsunđēsmouς ḡia na metabaīnou apó tēn m̄ia sel̄ida stēn áll̄, all̄á p̄éra apó to na anaz̄htou lē\xeis-kleidiā se mia sel̄ida, oī m̄χanēs dñskolueñontai na e\xagáyouν kápoioi eidoñs nōyma apó tis plēr̄ofořiēs pou periech̄ontai se autēs tis sel̄idēs.

Έτσi, ta tēlēutaiā χr̄ónia e\xeit uпár̄xi e\xtoto eřeunh̄tikó evđiaφéron γia tēn epivón̄hs kai diádoſt̄ teχnoloyiōn s̄m̄asioloyikēs m̄ontelop̄oíhs̄ tōn dēdōmēnōn kai tēs uпokeimene γnώσηs, oī oπoies th̄ aalá\xouν stadiakā tēn katá-

σταση που περιγράφηκε παραπάνω, έτσι ώστε α) τα δεδομένα του παγκόσμιου ιστού να είναι πλέον προσπελάσιμα από μηχανές τεχνητής νοημοσύνης που να μπορούν να πραγματοποιούν αυτόματα εργασίες σημασιολογικής επεξεργασίας τους, β) οι διάφοροι φορείς που δημοσιεύουν περιεχόμενο στον ιστό να το κάνουν με βάση κάποια ανοιχτά πρότυπα που θα είναι συμβατά με την επίτευξη του πρώτου στόχου, και γ) οι διάφοροι φορείς να μπορούν να επαναχρησιμοποιούν εύκολα δεδομένα άλλων φορέων που έχουν δημοσιευτεί με βάση τα πρότυπα αυτά στον ιστό. Ουσιαστικά, αυτό ισοδυναμεί με την μετατροπή ολόκληρου του περιεχομένου των σελίδων του παγκόσμιου ιστού σε ένα κοινό μοντέλο δεδομένων το οποίο να επιτρέπει σε όποιον επιθυμεί να το προσπελάσει (π.χ. να πραγματοποιήσει αναζητήσεις) σαν να ήταν μια ενιαία βάση δεδομένων. Ο παγκόσμιος ιστός υπό αυτή τη μορφή καλείται σημασιολογικός ιστός [BLHL01]. Η ανάγκη για σημασιολογική μοντελοποίηση των δεδομένων ενισχύεται από τις σύγχρονες εφαρμογές που καλούνται να διαχειρίστούν ολοένα και μεγάλο όγκο πληροφορίας που ενδεχομένως προέρχεται από διαφορετικές πηγές. Η σημασιολογική περιγραφή των δεδομένων καθιστά δυνατή την αποδοτική οργάνωσή τους και την επεξεργασία τους από τα σύγχρονα συστήματα.

Η θεωρητική μελέτη του ζητήματος της σημασιολογικής αναπαράστασης των δεδομένων αφορά στις ιδιότητες των διάφορων φορμαλισμών αναπαράστασης γνώσης και δεδομένων και τη συμπεριφορά (μαθηματική και υπολογιστική) των συνακόλουθων αλγορίθμων διαχείρισης και επεξεργασίας της. Παράλληλα, έχουν προταθεί τεχνολογίες που επιτρέπουν την υλοποίηση πραγματικών συστημάτων τα οποία ενσωματώνουν τα αποτελέσματα της θεωρητικής έρευνας και κινούνται προς την κατεύθυνση της σταδιακής υλοποίησης του σημασιολογικού ιστού.

Στο παραδοτέο αντό η έμφαση δίνεται κυρίως στην περιγραφή των τεχνολογιών σημασιολογικής αναπαράστασης γνώσης, καθώς αυτές είναι πλέον αρκετά ώριμες και έχουν καθιερωθεί αρκετά πρότυπα που έχουν αρχίσει να υιοθετούνται από διάφορους φορείς. Παράλληλα, σε αρκετά σημεία θα υπάρξει και σύντομη αναφορά στα σχετιζόμενα θεωρητικά θέματα. Στην Ενότητα 2 θα αναφερθούμε στους φορμαλισμούς σημασιολογικής μοντελοποίησης γνώσης, παρουσιάζοντας κατ' αρχήν το θεωρητικό υπόβαθρο και κατόπιν τα διάφορα πρότυπα που έχουν προταθεί για το σκοπό αυτό στα πλαίσια του σημασιολογικού ιστού. Εν συνεχείᾳ, θα αναφερθούμε στην πρότυπη γλώσσα διατύπωσης ερωτημάτων πάνω σε δεδομένα που διαθέτουν σημασιολογικό περιεχόμενο, και στην Ενότητα 3 θα αναφερθούμε στον τρόπο με τον οποίο η πληροφορία που βρίσκεται αποθηκευμένη σε βάσεις δεδομένων συνδέεται με σημασιολογικές περιγραφές. Τέλος, στην ενότητα 4 του παραδοτέου παρουσιάζομε τα σύγχρονα συστήματα και εργαλεία σημασιολογικής αναπαράστασης της γνώσης και απάντησης ερωτημάτων περιγράφοντας τα βασικά χαρακτηριστικά και οι λειτουργίες τους.

2 Σημασιολογική μοντελοποίηση

Στον ενότητα αυτή θα μελετήσουμε το ζήτημα της σημασιολογικής μοντελοποίησης γνώσης και δεδομένων, δηλαδή των φορμαλισμών που επιτρέπουν την περιγραφή της δομής και των ιδιοτήτων ενός πεδίου γνώσης με αυστηρό τρόπο, και εν συνεχεία των προτύπων που έχουν αναπτυχθεί ώστε να υποστηρίξουν στην πράξη τους φορμαλισμούς αυτούς. Όπως θα δούμε αμέσως, ο βασικός φορμαλισμός σημασιολογικής μοντελοποίησης γνώσης που έχει προκριθεί είναι η οντολογία, λέξη που φέρει μακρύ φιλοσοφικό παρελθόν, αλλά στο πεδίο της επιστήμης των υπολογιστών έχει αποκτήσει μια πολύ συγκεκριμένη σημασία.

Πριν προχωρήσουμε θα πρέπει να κάνουμε μια διασαφήνιση ως προς την διαφορά μεταξύ γνώσης και δεδομένων. Γενικά, γνώση είναι μια αφαιρετική περιγραφή ενός πεδίου ενδιαφέροντος, ενώ δεδομένα είναι κάποια πραγματικά αντικείμενα τα οποία απαρτίζουν στην πράξη το πεδίο ενδιαφέροντος, και των οποίων οι ιδιότητες περιγράφονται συνεπώς αφαιρετικά μέσω της γνώσης. Μάλιστα η ίδια γνώση μπορεί να χρησιμοποιηθεί σε συνδυασμό με διαφορετικά δεδομένα, που μπορεί να είναι σύνολα αντικείμενων του κόσμου διαχωρισμένα μεταξύ τους, για πρακτικούς ή άλλους λόγους (π.χ. τα δεδομένα που αφορούν τους υπαλλήλους της Α εταιρεία και τα δεδομένα που αφορούν τους υπαλλήλους της Β εταιρείας). Παρά τη διάκριση αυτή, και επειδή οι φορμαλισμοί που θα μελετήσουμε επιτρέπουν συνήθως την ταυτόχρονη μοντελοποίηση γνώσης και δεδομένων ως ενιαίο μοντέλο, η διάκριση μπορεί να είναι πάντα ρητή.

2.1 Οντολογίες

Στο πεδίο της επιστήμης υπολογιστών μια οντολογία είναι ένα σύνολο περιγραφικών δηλώσεων που μοντελοποιούν με αυστηρό τρόπο ένα πεδίο ενδιαφέροντος. Οι δηλώσεις αυτές αφορούν στα είδη, τις ιδιότητες και τις συσχετίσεις μεταξύ των οντοτήτων που απαρτίζουν το πεδίο ενδιαφέροντος (δηλαδή τη γνώση). Οι οντότητες μπορεί να είναι είτε αφαιρετικές έννοιες, είτε πραγματικά αντικείμενα.

Με την αυστηρή περιγραφή ενός πεδίου αποφεύγονται πιθανές παρερμηνείες κατά την ανθρώπινη επικοινωνία και διασφαλίζεται επίσης ότι ένα υπολογιστικό σύστημα που βασίζει την λειτουργία του σε μια οντολογία θα συμπεριφέρεται με ομοιόμορφο και προβλέψιμο τρόπο και θα επικοινωνεί δίχως προβλήματα με άλλα παρόμοια συστήματα, καθώς η γλώσσα επικοινωνίας θα είναι κοινή και αυστηρώς ορισμένη. Επιπλέον, ενσωματώνοντας κατάλληλους αλγορίθμους επεξεργασίας της γνώσης, οι μηχανές μπορούν να πραγματοποιούν εργασίες συλλογιστικής ή αλλιώς εξαγωγής συμπερασμάτων, δηλαδή εξαγωγή υπονοούμενης γνώσης που προκύπτει από την ρητά δηλωμένη.

2.1.1 Δομή

Τα βασικά δομικά στοιχεία μιας οντολογίας, ανεξάρτητα από το σε ποιον ακριβώς φορμαλισμό βασίζεται ή σε ποια γλώσσα είναι εκπεφρασμένη, είναι τα *άτομα*, οι *έννοιες*, οι *σχέσεις*, και τα *αξιώματα*.

Τα *άτομα* είναι τα πραγματικά συστατικά μέρη μιας οντολογίας, τα δεδομένα. Τα *άτομα* είναι συνήθως συγκεκριμένα αντικείμενα του υπό περιγραφή κόσμου όπως π.χ. άνθρωποι, ζώα, αυτοκίνητα, εταιρίες, πλανήτες, ανάλογα με το πεδίο του ενδιαφέροντος. Μια οντολογία δεν είναι απαραίτητο να περιλαμβάνει *άτομα*, αφού μπορεί να περιορίζεται στην αφαιρετική περιγραφή του πεδίου ενδιαφέροντος (δηλαδή μόνο της γνώσης), ωστόσο ένας από τους βασικούς λόγους ύπαρξης μιας οντολογίας είναι, τελικά, η κατηγοριοποίηση κάποιων ατόμων στις έννοιες της οντολογίας, δηλαδή η κατάταξη των ατόμων στις έννοιες της οντολογίας ανάλογα με τις ιδιότητές τους, ανεξάρτητα από το αν τα *άτομα* περιλαμβάνονται ρητά στην οντολογία, ή δηλώνονται σε εξωτερικές πηγές.

Οι έννοιες (ή αλλιώς κατηγορίες ή κλάσεις) μιας οντολογίας αντιπροσωπεύουν σύνολα ατόμων του πεδίου ενδιαφέροντος που έχουν κάποια κοινά χαρακτηριστικά. Τα χαρακτηριστικά αυτά (ή οι τιμές κάποιων χαρακτηριστικών) αποτελούν τις συνθήκες που καθορίζουν την συμμετοχή ενός ατόμου σε κάποια έννοια. Ένα άτομο που ανήκει σε κάποια έννοια καλείται *στιγμιότυπο* της συγκεκριμένης έννοιας. Οι έννοιες διακρίνονται σε δύο βασικές κατηγορίες:

- Στις *πρωτογενείς* ή *ατομικές* έννοιες, για τις οποίες οι όποιες συνθήκες συμμετοχής είναι εκτός της οντολογίας (π.χ. οι έννοιες *Άνθρωπος*, *Άρρεν*).
- Στις *οριζόμενες* ή *σύνθετες* έννοιες των οποίων ο ορισμός περιλαμβάνεται υπό τη μορφή κάποιου αξιώματος στην οντολογία, και ο οποίος καθορίζει και τις συνθήκες συμμετοχής των ατόμων στις έννοιες αυτές (π.χ. η έννοια *Άντρας* μπορεί να οριστεί ως το σύνολο των ατόμων που ανήκουν σε αμφότερες τις έννοιες *Άνθρωπος* και *Άρρεν*).

Οι οριζόμενες έννοιες κατασκευάζονται βάσει άλλων πρωτογενών ή οριζόμενων έννοιών, με τη βοήθεια κάποιων κατασκευαστών που παρέχει η γλώσσα της οντολογίας (π.χ. ως ένωση δύο έννοιών). Επειδή οι παραπάνω έννοιες φέρουν πάντα κάποιο όνομα καλούνται *ονοματισμένες*. Όπως θα δούμε στη συνέχεια, στα αξιώματα μιας οντολογίας, πέραν των παραπάνω κατηγοριών έννοιών, είναι δυνατόν να συμμετέχουν και σύνθετες έννοιες που να ορίζονται τοπικά εντός του αξιώματος μέσω κάποιου κατασκευαστή, χωρίς έτσι να φέρουν κάποιο όνομα (π.χ. η σύνθετη έννοια *Άνθρωπος* και *Άρρεν* χωρίς κάποιον ορισμό που να τους αποδίδει ιδιαίτερο όνομα).

Οι σχέσεις μιας οντολογίας καθορίζουν τους τρόπους με τους οποίους σχετίζονται μεταξύ τους οι έννοιες και τα άτομα. Για παράδειγμα, η σχέση είναι παντρεμένος θα μπορούσε να χρησιμοποιηθεί για να περιγραφεί ότι το άτομο *Ανδρέας* (που είναι στιγμιότυπο της έννοιας *Άντρας*) είναι παντρεμένο με το άτομο *Ελένη* (που είναι στιγμιότυπο της έννοιας *Γυναίκα*). Οι σχέσεις διακρίνονται στις εξής δύο γενικές κατηγορίες.

- Στις *ταξονομικές σχέσεις*, οι οποίες είναι σχέσεις υπαγωγής, δηλαδή σχέσεις εξειδίκευσης ή εγκλεισμού: όταν μια έννοια υπάγεται σε κάποια άλλη, η πρώτη έννοια καλείται υπερέννοια και η δεύτερη υποέννοια της πρώτης. Για παράδειγμα, η έννοια *Άντρας* υπάγεται στην έννοια *Άνθρωπος*. Οι σχέσεις υπαγωγής οδηγούν στην δημιουργία *ιεραρχίων* εννοιών (*ταξονομιών*), από τις πιο γενικές σε ειδικότερες. Συνήθως, η πιο γενική έννοια της οντολογίας είναι μια έννοια που έχει ως στιγμιότυπα όλα τα άτομα του κόσμου. Μια σημαντική συνέπεια της σχέσης υπαγωγής είναι η κληρονομικότητα των ιδιοτήτων από μια έννοια στις υποεννοιές της, δηλαδή ότι ισχύει για την υπερέννοια πρέπει κατ' ανάγκη να ισχύει και για τις υποέννοιές της (π.χ. τα στιγμιότυπα μιας έννοιας είναι και στιγμιότυπα όλων των υπερεννοιών της). Οι περισσότερες οντολογίες επιτρέπουν σε μια έννοια να έχει περισσότερες της μίας υπερέννοιες. Ταξονομικές σχέσεις μπορούν να επιβληθούν και στις ίδιες τις σχέσεις εκτός από τις έννοιες, ώστε να οριστούν *ταξονομίες σχέσεων*.
- Στις *συσχετιστικές σχέσεις*, οι οποίες είναι σχέσεις που συνδέουν μεταξύ τους έννοιες που ανήκουν στις ίδια ή σε διαφορετικές *ταξονομίες*.

Οι σχέσεις μπορούν να έχουν και ορισμένες επιπλέον ιδιότητες που να καθορίζουν κάποιες ιδιαίτερες λεπτομέρειες για τον τρόπο της συσχέτισης που αντιπροσωπεύουν. Τέτοιες ιδιότητες είναι για παράδειγμα η μεταβατικότητα, η συμμετρικότητα μιας σχέσης, ή οι περιορισμοί που αφορούν τα δυνατά πεδία ορισμού και τιμών της. Κατ' αναλογία με την περίπτωση των εννοιών, ένα *ζεύγος* ατόμων τα οποία συνδέονται μεταξύ τους μέσω κάποιου συσχετιστικού ρόλου καλείται *στιγμιότυπο* του ρόλου. Μια γλώσσα οντολογίας μπορεί επίσης να υποστηρίζει την κατασκευή σύνθετων ρόλων.

Τέλος, τα αξιώματα είναι δηλώσεις που αποτυπώνουν την γνώση επί του πεδίου ενδιαφέροντος που περιγράφει η οντολογία. Τα αξιώματα είναι συνήθως αξιώματα ορισμού σχέσεων υπαγωγής ή σχέσεων ισοδυναμίας (που αντιστοιχούν σε αμφίδρομες σχέσεις υπαγωγής) μεταξύ απλών ή σύνθετων εννοιών και ρόλων, ή δήλωσης ιδιοτήτων κάποιων ρόλων. Οι ονοματισμένες έννοιες και σχέσεις αποτελούν το λεξιλόγιο της οντολογίας με βάση το οποίο διατυπώνονται τα αξιώματα. Το σύνολο των αξιωμάτων καλείται *ορολογία*. Οι δηλώσεις που ορίζουν κάποια

άτομα ή ζεύγη ατόμων ως στιγμιότυπα κάποιων εννοιών ή σχέσεων, αντίστοιχα, της οντολογίας καλούνται *ισχυρισμοί*. Τέλος, ένα σώμα ορολογίας μαζί με ένα σώμα ισχυρισμών λέμε ότι συνθέτουν μια βάση γνώσης.

2.1.2 Περιγραφικές Λογικές

Μεταξύ των διάφορων φορμαλισμών για την σημασιολογική μοντελοποίηση γνώσης ως οντολογιών εξέχουνσα θέση καταλαμβάνουν οι περιγραφικές λογικές [BCM⁺03]. Η πληροφορία και η γνώση που αφορά σε ένα πεδίο εφαρμογής μπορούν να αναπαρασταθούν τυπικά με γλώσσες περιγραφικής λογικής. Απότερος σκοπός είναι η αυτοματοποιημένη επεξεργασία της γνώσης με διεργασίες συλλογιστικής για την εξαγωγή συμπερασμάτων. Το όνομά τους προκύπτει από το γεγονός ότι ορίζουν τη γνώση ενός πεδίου ενδιαφέροντος από περιγραφές εννοιών, ενώ η σημασιολογία τους βασίζεται στη λογική. Πρόκειται για υποσύνολα της λογικής πρώτης τάξης επιλεγμένα έτσι ώστε να προσφέρουν τις δυνατότητες αναπαράστασης ενός πεδίου ενδιαφέροντος με βάση τα δομικά χαρακτηριστικά μιας οντολογίας που αναφέρθηκαν στην προηγούμενη ενότητα, έχοντας παράλληλα μια αυστηρά ορισμένη σημασιολογία που βασίζεται στη λογική θεωρία. Ένα επιπλέον σημαντικό χαρακτηριστικό των περιγραφικών λογικών είναι ότι έχουν κατασκευαστεί έτσι ώστε να διευκολύνουν την υλοποίηση πρακτικών αλγορίθμων συλλογιστικής. Συγκεκριμένα, οι γλώσσες περιγραφικής λογικής παρέχουν τη δυνατότητα επεξεργασίας από μηχανισμούς εξαγωγής συμπερασμάτων που έχουν σκοπό τον προσδιορισμό των κανόνων που πρέπει να εφαρμοστούν ώστε να προκύψει νέα γνώση από την αρχική η οποία έχει καταγραφεί σε κάποια γλώσσα περιγραφικής λογικής.

Οι περιγραφικές λογικές επιτρέπουν τον ορισμό μιας βάσης γνώσης και δεδομένων εν είδῃ οντολογίας, αποτελούμενης από δύο μέρη: το *σώμα ορολογίας* (TBox) και το *σώμα ισχυρισμών* (ABox). Το σώμα ορολογίας αποτελείται από τα αξιώματα τα οποία διατυπώνονται με βάση τις έννοιες και τις σχέσεις της οντολογίας (οι σχέσεις στα πλαίσια των περιγραφικών λογικών συνήθως καλούνται *ρόλοι*). Οι έννοιες και οι ρόλοι μπορεί να είναι είτε ατομικοί (δηλαδή πρωτογενείς) είτε σύνθετοι (δηλαδή οριζόμενοι). Η κατασκευή σύνθετων εννοιών στις περιγραφικές λογικές γίνεται μέσω συγκεκριμένων κατασκευαστών. Έτσι, αν C και D είναι δύο έννοιες, γενικά μπορούν να οριστούν οι σύνθετες έννοιες του Πίνακα 1. Για κάθε σύνθετη έννοια, στον πίνακα δίνεται το όνομά της, η έκφρασή της στη σύνταξη των περιγραφικών λογικών, και η περιγραφή της. Το σύνολο των κατασκευαστών που υποστηρίζει η περιγραφική λογική καθορίζει την εκφραστικότητά της. Στις περιγραφικές λογικές ορίζονται συνήθως επίσης η *καθολική* έννοια και η *κενή* έννοια, που συμβολίζονται με \top και \perp , και είναι οι έννοιες που έχουν εξ ορισμού ως στιγμιότυπα όλα τα άτομα του κόσμου και κανένα άτομο αντίστοιχα.

Όπως συμβαίνει με τις έννοιες, υπάρχουν και κατασκευαστές για την κατα-

Όνομα	Έκφραση	Περιγραφή
Τομή	$C \sqcup D$	Το σύνολο των ατόμων που είναι στιγμιότυπα αμφότερων των C και D .
Ένωση	$C \sqcap D$	Το σύνολο των ατόμων που είναι στιγμιότυπα της C ή της D .
Συμπλήρωμα	$\neg C$	Το σύνολο των ατόμων που δεν είναι στιγμιότυπα της C .
Περιορισμός τιμής	$\forall R.C$	Το σύνολο των ατόμων που συνδέονται μέσω του ρόλου R μόνο με στιγμιότυπα της C .
Υπαρξιακή ποσοδεικτοδότηση	$\exists R.C$	Το σύνολο των ατόμων που συνδέονται μέσω του ρόλου R με κάποιο στιγμιότυπα της C .
Γενικός περιορισμός πληθικότητας	$\geq nR$ $\leq nR$ $= nR$	Το σύνολο των ατόμων που συνδέονται μέσω του ρόλου R με τουλάχιστον, το πολύ, ή ακριβώς n άτομα.
Ειδικός περιορισμός πληθικότητας	$\geq nR.C$ $\leq nR.C$ $= nR.C$	Το σύνολο των ατόμων που συνδέονται μέσω του ρόλου R με τουλάχιστον, το πολύ, ή ακριβώς n στιγμιότυπα της έννοιας C .

Πίνακας 1: Κατασκευαστές εννοιών στις περιγραφικές λογικές.

σκευή σύνθετων ρόλων από ατομικούς, οι οποίοι παρουσιάζονται στον Πίνακα 2, όπου θεωρείται ότι R και S είναι δύο οποιοιδήποτε ρόλοι.

Στις περιγραφικές λογικές ένα αξίωμα υπαγωγής εννοιών ή ρόλων γράφεται ως $C \sqsubseteq D$ και $R \sqsubseteq S$, αντίστοιχα. Αμφίδρομες υπαγωγές ισοδυναμούν με ισοδυναμίες και γράφονται ως εξής $A \equiv B$ και $R \equiv S$.

Η σημασιολογία μιας γλώσσας περιγραφικής λογικής δίνεται από την ερμηνεία της \mathcal{I} , επί ενός πεδίου $\Delta^{\mathcal{I}}$. Μία ερμηνεία αντιστοιχίζει τα άτομα σε αντικείμενα του πεδίου $\Delta^{\mathcal{I}}$ (συνήθως μέσω της ταυτοικής απεικόνισης, οπότε το σύνολο των ατόμων και το $\Delta^{\mathcal{I}}$ ταυτίζονται), τις έννοιες σε υποσύνολά του, και τους ρόλους σε σύνολα από ζεύγη αντικειμένων του πεδίου. Για παράδειγμα εάν θεωρήσουμε ότι το πεδίο ενδιαφέροντος αφορά στους φοιτητές ενός πανεπιστημίου, η ερμηνεία μιας έννοιας Φοιτητής περιλαμβάνει το σύνολο των φοιτητών. Η καθολική έννοια \top αντιστοιχίζεται στο $\Delta^{\mathcal{I}}$ και η κενή έννοια \perp στο κενό σύνολο. Τα αξιώματα και οι ισχυρισμοί ερμηνεύονται με τον προφανή τρόπο και έτσι για παράδειγμα το αξίωμα Μεταπτυχιακός \sqsubseteq Φοιτητής δηλώνει ότι οι μεταπτυχιακοί φοιτητές είναι υποσύνολο του συνόλου των φοιτητών, $(\text{Μεταπτυχιακός} \sqsubseteq \text{Φοιτητής})^{\mathcal{I}} = (\text{Μεταπτυχιακός})^{\mathcal{I}} \sqsubseteq (\text{Φοιτητής})^{\mathcal{I}}$. Ομοίως, σύμφωνα με ερμηνεία ενός ισχυρι-

Όνομα	Έκφραση	Περιγραφή
Τομή	$R \sqcup S$	Το σύνολο των ζευγών ατόμων που είναι στιγμιότυπα αμφότερων των R και S .
Ένωση	$R \sqcap S$	Το σύνολο των ζευγών ατόμων που είναι στιγμιότυπα του R ή του S .
Συμπλήρωμα	$\neg R$	Το σύνολο των ζευγών ατόμων που δεν είναι στιγμιότυπα του R .
Αντιστροφή	R^-	Το σύνολο των αντεστραμμένων ζευγών ατόμων που είναι στιγμιότυπα του R .
Σύνθεση	$R \circ S$	Το σύνολο των ζευγών ατόμων (a, b) για τα οποία υπάρχει κάποια άτομο c για το οποίο το (a, c) είναι στιγμιότυπο του R και το (c, b) στιγμιότυπο του S .

Πίνακας 2: Κατασκευαστές ρόλων στις περιγραφικές λογικές.

σμού Φοιτητής(Άννα) το αντικείμενο Άννα^T ανήκει στο σύνολο Φοιτητής^T. Μια ερμηνεία που ικανοποιεί όλα τα αξιώματα του TBox καθώς και όλους τους ισχυρισμούς του ABox λέγεται μοντέλο της βάσης γνώσης.

Ανάλογα με τα είδη των κατασκευαστών και αξιωμάτων που επιτρέπεται να διατυπωθούν, ορίζονται διάφορες διάλεκτοι των περιγραφικών λογικών, από τις λιγότερο εκφραστικές που επιτρέπουν περιορισμένο σύνολο κατασκευαστών, έχουν όμως καλές ιδιότητες κλιμάκωσης στα ζητήματα συλλογιστικής, ως τις πλέον εκφραστικές στις οποίες η πραγματοποίηση της συλλογιστικής μπορεί να απαιτεί εκθετικό χρόνο. Παρακάτω εστιάζουμε στις γλώσσες περιγραφικών λογικών που, όπως θα δούμε στην συνέχεια, αποτελούν το θεωρητικό υπόβαθρο για τα διαφορετικά προφίλ της γλώσσας οντολογιών OWL (Web Ontology Language) [BvHH⁺04, PSHH04] που χρησιμοποιείται από τον σημασιολογικό ιστό.

- *Η οικογένεια DL-Lite.* Η οικογένεια DL-Lite [ACKZ09] ορίζει ένα σύνολο αρκετά περιορισμένων εκφραστικά περιγραφικών λογικών με παρόμοια χαρακτηριστικά χαμηλής υπολογιστικής πολυπλοκότητας. Μεταξύ αυτών ιδιαίτερο ενδιαφέρον παρουσιάζει η $DL\text{-}Lite_R$, στην οποία επιτρέπονται μόνο τα εξής αξιώματα: $C \sqsubseteq D$ όπου στη θέση της έννοιας C μπορεί να είναι μια ατομική έννοια A ή η σύνθετη έννοια $\exists R.T$, ενώ στη θέση της έννοιας D μπορεί να είναι μια ατομική έννοια A , ή μια σύνθετη έννοια της μορφής $A \sqcup B$, $\neg B$ ή $\exists R.A$ (όπου οι A και B είναι ατομικές έννοιες). Επίσης επιτρέπεται η υπαγωγή ρόλων μέσω την αξιωμάτων της μορφής $R \sqsubseteq S$. Σε όλα τα παραπάνω στη θέση των ρόλων R και S μπορεί να υπάρχουν και οι αντίστροφοι ρόλου R^- και S^- , όπου οι R και S είναι ατομικοί.

- *H oikoyéneia* \mathcal{EL} . [BBL05] Πρόκειται για μία οικογένεια εκφραστικών περιγραφικών λογικών. Ιδιαίτερο ενδιαφέρον παρουσιάζει η \mathcal{ELHI} που περιλαμβάνει αξιώματα της μορφής $A_1 \sqsubseteq A_2$, $A_1 \sqcap A_2 \sqsubseteq A$, $A_1 \sqsubseteq \exists P.A_2$ ή $\exists P.A_2 \sqsubseteq A_1$ όπου το $A_{(i)}$ είναι ατομική έννοια ή η καθολική (\top) και το P είναι ατομικός ή ανάστροφος ρόλος. Σημειώνουμε ότι η γλώσσα DL-Lite_R προκύπτει από την \mathcal{ELHI} αφαιρώντας αξιώματα της μορφής $\exists P.A_1 \sqsubseteq A_2$, όπου η A_1 δεν είναι η καθολική έννοια.
- *Horn-SHIQ*. Πρόκειται για την εκφραστικότερη από τις δύο προηγούμενες γλώσσες. Υποστηρίζει την καθολική και κενή έννοια, \top , \perp , και διαθέτει εκφραστικά χαρακτηριστικά όπως είναι οι μεταβατικοί ρόλοι, καθώς και οι ειδικοί κατασκευαστές πληθυκότητας. Επιτρέπει αξιώματα της μορφής: $B \sqsubseteq A$, $B_1 \sqcap B_2 \sqsubseteq A$, $A \sqsubseteq \exists R.B$, $A \sqsubseteq \forall R.B$, $A \sqsubseteq \leq 1 S.B$, $P \sqsubseteq R$ όπου R είναι ένας ατομικός ή ανάστροφος ρόλος, P ατομικός ρόλος, και A , B ατομικές έννοιες.

2.2 Συλλογιστική

Όπως ήδη αναφέρθηκε, ένα σημαντικό ζητούμενο από έναν φορμαλισμό σημασιολογικής μοντελοποίησης γνώσης είναι να επιτρέπει την υλοποίηση αλγορίθμων για την πραγματοποίηση συλλογιστικής, δηλαδή την εξαγωγή έμμεσης γνώσης από την γνώση που περιγράφεται ρητά από την οντολογία. Τα συστήματα που επιλύουν τέτοιου είδους προβλήματα καλούνται συλλογιστές.

Τα πλέον συνήθη προβλήματα συλλογιστικής που καλείται να επιλύσει ένας συλλογιστής σε πρακτικές εφαρμογές είναι τα εξής:

- Η κατηγοριοποίηση εννοιών, δηλαδή ο υπολογισμός όλων των σχέσεων υπαγωγής μεταξύ των εννοιών που συμμετέχουν στην ορολογία μιας οντολογίας, δηλαδή ο υπολογισμός της δομής της γνώσης που περιγράφει η ορολογία υπό τη μορφή ιεραρχίας. Μέσω της κατηγοριοποίησης εννοιών μπορούν να απαντηθούν ερωτήματα για παράδειγμα εάν η έννοια *Άνδρας* είναι υποέννοια της *Άνθρωπος* όπου αρνητική απάντηση καταδεικνύει προφανώς λάθη μοντελοποίησης.
- Η κατηγοριοποίηση ατόμων. Στην περίπτωση αυτή ο συλλογιστής καλείται να αποφανθεί αν ένα συγκεκριμένο άτομο αποτελεί στιγμιότυπο μιας συγκεκριμένης έννοιας (π.χ. είναι το άτομο *Άνδρεας* στιγμιότυπο της έννοιας *Πατέρας*;).
- Η ανάκτηση στιγμιοτύπων, δηλαδή η εύρεση όλων των ατόμων που είναι στιγμιότυπα κάποιας συγκεκριμένης έννοιας (π.χ. εύρεση όλων των πατέρων, δηλαδή όλων των στιγμιοτύπων της έννοιας *Πατέρας*).

- Η απάντηση ερωτημάτων, δηλαδή η εύρεση όλων των ατόμων που ικανοποιούν κάποια σύνθετη συνήθηκη συμμετοχής σε ένα σύνολο εννοιών (π.χ. η εύρεση όλων των ατόμων που είναι άνδρες και έχουν κάποιο παιδί που είναι φοιτητής). Τα ερωτήματα διατυπώνονται συνήθως υπό την μορφή συζευκτικών ερωτημάτων (βλ. παρακάτω) και αποτελούν εκφράσεις που περιέχουν μεταβλητές οι οποίες εμφανίζονται ως ορίσματα κάποιων εννοιών ή ρόλων. Π.χ. το παραπάνω παράδειγμα ερωτήματος, θεωρώντας κατάλληλη ορολογία, θα μπορούσε να διατυπωθεί ως έξης:

$$Q(x) \leftarrow \text{Άντρας}(x), \text{έχειΠαιδί}(x, y), \text{Φοιτητής}(y).$$

Οι απαντήσεις στο παραπάνω ερώτημα πρέπει να περιλαμβάνουν όλα τα άτομα (ως τιμές της μεταβλητής x) που είναι στιγμιότυπα της έννοιας Άντρας και για τα οποία υπάρχει κάποιο άτομο (ως τιμή της μεταβλητής y) το οποίο είναι στιγμιότυπο της έννοιας Φοιτητής και το ζεύγος των δύο ατόμων είναι στιγμιότυπο του ρόλου έχειΠαιδί.

Δεδομένου ότι οι συλλογιστές καλούνται να επιλύουν τα παραπάνω προβλήματα σε πραγματικές εφαρμογές, ενδιαφέρον έχουν οι αλγόριθμοι συλλογιστικής που τερματίζουν και αποφαίνονται για τα παραπάνω προβλήματα σε πραγματικό χρόνο. Καθώς όμως γενικά δεν υπάρχει καμία εγγύηση για το ότι ο τερματισμός μιας συλλογιστική διαδικασίας θα γίνει σε λογικά χρονικά πλαίσια, για κάθε γλώσσα οντολογίας είναι σημαντικό να έχει γίνει μελέτη της υπολογιστικής πολυπλοκότητάς της. Η αποφανσιμότητα και η υπολογιστική πολυπλοκότητα των αλγορίθμων συλλογιστικής για τα παραπάνω προβλήματα εξαρτώνται από την εκφραστική δύναμη της υποκείμενης γλώσσας οντολογίας. Πολύ εκφραστικές γλώσσες έχουν συνήθως υψηλή υπολογιστική πολυπλοκότητα και πιθανόν να είναι μη αποφάνσιμες, ενώ οι περιορισμένης εκφραστικότητας γλώσσες μπορεί να μην είναι αρκετές για την μοντελοποίηση ενός πεδίου ενδιαφέροντος με ικανοποιητικό τρόπο. Οι γλώσσες DL-Lite, \mathcal{EL} και Horn- \mathcal{SHIQ} που αναφέρθηκαν στην προηγούμενη ενότητα έχουν κατά σειρά αυξανόμενη εκφραστικότητα και πολυπλοκότητα.

2.2.1 Απάντηση Ερωτημάτων

Η απάντηση ερωτημάτων με βάση οντολογίες είναι ένα από τα βασικά προβλήματα που καλούνται να επιλύσουν ολοένα και περισσότερες σύγχρονες εφαρμογές. Στην παρούσα ενότητα θα μας απασχολήσει η απάντηση συζευκτικών ερωτημάτων με βάση οντολογίες εκφρασμένες σε γλώσσες περιγραφικής λογικής. Τα συζευκτικά ερωτήματα είναι εκφραστικά σύνθετα ερωτήματα ιδιαίτερα δημοφιλή στον κλάδο των βάσεων δεδομένων [AHV95]. Σαν παράδειγμα του προβλήματος που μας απασχολεί ας θεωρήσουμε την οντολογία που περιλαμβάνει το

αξιώμα Μεταπτυχιακός \sqsubseteq Φοιτητής \sqcap ΞέχειΠτυχίο και το συζευκτικό ερώτημα $Q(x) \leftarrow \Phi\text{οιτητής}(x) \wedge \text{έχειΠτυχίο}(x)$. Θεωρούμε επίσης το σύνολο ισχυρισμών {Μεταπτυχιακός(Άννα), έχειΠτυχίο(Τάσος), Φοιτητής(Ελένη), έχειΠτυχίο(Ελένη)}. Η απάντηση στο ερώτημα θα πρέπει να περιλαμβάνει τους φοιτητές που έχουν πτυχίο ή λαμβάνοντας υπόψιν την οντολογία, τους μεταπτυχιακούς φοιτητές. Τυπικά, η απάντηση του ερωτήματος αποτελείται από όλα τα άτομα της βάσης γνώσης με τα οποία μπορούμε να αντικαταστήσουμε τη μεταβλητή x του ερωτήματος ώστε η πρόταση που προκύπτει να είναι αληθής στη βάση γνώσης. Στο παράδειγμά μας οι απάντησεις είναι τα άτομα Άννα, Ελένη.

Οι ανάγκες των σύγχρονων εφαρμογών για σημασιολογική αναπαράσταση των δεδομένων αντιμετωπίζονται με την υιοθέτηση των οντολογιών για την περιγραφή τους. Ταυτόχρονα, η χρήση των οντολογιών σε εφαρμογές που συγκεντρώνουν δεδομένα από διαφορετικές πηγές εξασφαλίζει τη συνέπεια των δεδομένων και την ακεραιότητα της βάσης. Οι αυξανόμενες απαιτήσεις για την αποτελεσματική διαχείριση μεγάλου όγκου πληροφορίας οδήγησαν στην εκτεταμένη μελέτη του προβλήματος της απάντησης ερωτημάτων με βάση τις οντολογίες από την οποία έχει προκύψει πληθώρα πρακτικών υλοποιήσεων. Βασικό ζήτημα για τα διάφορα συστήματα είναι η υψηλή υπολογιστική πολυπλοκότητα του προβλήματος λόγω της οποίας δεν ανταποκρίνονται ικανοποιητικά σε περιπτώσεις μεγάλου όγκου δεδομένων.

Απάντηση Ερωτημάτων με Επαναγραφή

Επιφανής προσέγγιση του προβλήματος της απάντησης ερωτημάτων είναι η μέθοδος της επαναγραφής (rewriting). Σύμφωνα με τη μέθοδο αυτή το δεδομένο ερώτημα καθώς και η οντολογία επαναγράφονται ώστε να αναπαρασταθούν από φορμαλισμούς που είναι εύκολα διαχειρίσιμοι από συστήματα ανάκτησης δεδομένων. Πιο τυπικά, ένα ερώτημα Q και μία οντολογία μετατρέπονται σε ένα σύνολο από προτάσεις \mathcal{R} (Datalog πρόγραμμα) που ονομάζεται επαναγραφή, ώστε για κάθε σύνολο δεδομένων \mathcal{D} , οι απαντήσεις του Q με βάση την οντολογία και τα δεδομένα \mathcal{D} να συμπίπτουν με τις απαντήσεις του Q με βάση τα \mathcal{R} και \mathcal{D} χωρίς να λαμβάνεται δηλαδή υπόψιν η οντολογία [HMS08, CDGL⁺07, PUMH10].

Σαν παράδειγμα απάντησης ερωτήματος με επαναγραφή ας θεωρήσουμε μια οντολογία που χρησιμοποιείται για να περιγράψει τα δεδομένα ενός πανεπιστημίου και περιλαμβάνει τα αξιώματα: Μεταπτυχιακός \sqsubseteq Φοιτητής, Προπτυχιακός \sqsubseteq Φοιτητής. Εστω ότι διατυπώνουμε το ερώτημα $Q(x) \leftarrow \Phi\text{οιτητής} \wedge \text{Φοιτητής}(x)$ προκειμένου να αναζητήσουμε τους καταχωρημένους φοιτητές στη βάση του πανεπιστημίου. Σύμφωνα με την τεχνική της επαναγραφής το ερώτημα διατυπώνεται εκ νέου λαμβάνοντας υπόψιν τα αξιώματα του σώματος ορολογίας. Έτσι, προκύπτουν τα ερωτήματα $Q(x) \leftarrow \text{Προπτυχιακός} \wedge Q(x) \leftarrow \text{Μεταπτυχιακός}$. Κατόπιν, εφόσον η γνώση που περιγράφεται στο σώμα ορολογίας έχει αξιοποιηθεί εξ' ολοκλήρου για

την επαναγραφή του ερωτήματος, η ανάκτηση των δεδομένων μπορεί να ανατεθεί σε κάποια από τις σύγχρονες τεχνολογίες βάσεων δεδομένων.

Η τεχνική της επαναγραφής έχει μελετηθεί για διαφορετικές εκφραστικότητες γλωσσών περιγραφικής λογικής. Μία από τις πρώτες μελέτες έγινε για τη γλώσσα \mathcal{SHIQ} [HMS08] και οδήγησε στην ανάπτυξη του συστήματος KAON2 [MSS05]. Πλέον η εν λόγω μέθοδος αποτελεί τη συνηθέστερη ίσως πρακτική απάντησης ερωτημάτων με βάση κάποια οντολογία για γλώσσες περιγραφικής λογικής χαμηλότερης εκφραστικότητας όπως είναι η DL-Lite [CDGL⁺07, RA10], η \mathcal{ELH} [PUMH10], και η Horn- \mathcal{SHIQ} [EOS⁺12]. Πέρα από το KAON2 έχουν αναπτυχθεί και πολλά άλλα πρακτικά συστήματα που υλοποιούν τη μέθοδο επαναγραφής και υποστηρίζουν τις DL-Lite και \mathcal{ELH} (για παράδειγμα, Requiem [PUMH10], CLIPPER [EOS⁺12]) τα οποία θα περιγράψουμε σε επόμενη ενότητα.

Επαναγραφή με τη Μέθοδο Ανάλυσης

Σημαντική προσέγγιση της τεχνικής επαναγραφής είναι η μέθοδος της ανάλυσης (resolution) [BG01]. Σύμφωνα με την προσέγγιση αυτή η οντολογία και το ερώτημα μετατρέπονται σε ένα σύνολο προτάσεων (λογικής πρώτης τάξης) στις οποίες εφαρμόζεται εξαντλητικά ο κανόνας της ανάλυσης για την εξαγωγή νέας γνώσης. Η τεχνική της ανάλυσης κατά την επαναγραφή των ερωτημάτων μπορεί να είναι ιδιαίτερα χρήσιμη καθώς επιτρέπει πληθώρα βελτιστοποιήσεων που έχουν προταθεί στη θεωρία των αποδείξεων. Επιπλέον, ένας αλγόριθμος επαναγραφής ερωτημάτων που εφαρμόζει την τεχνική αυτή έχει το πλεονέκτημα ότι μπορεί εύκολα να επεκταθεί για να υποστηρίζει γλώσσες περιγραφικής λογικής υψηλότερης εκφραστικότητας.

Προκειμένου να γίνει περισσότερο κατανοητή η διαδικασία της ανάλυσης ας θεωρήσουμε δύο αξιώματα υπαγωγής, έστω $B \sqsubseteq A, C \sqsubseteq B$. Όταν ο κανόνας την ανάλυσης εφαρμοστεί σε αυτά, αφού προηγουμένως τα αξιώματα έχουν εκφραστεί σε προτάσεις λογικής πρώτης τάξης, προκύπτει η πρόταση που αντιστοιχεί στο $C \sqsubseteq A$.

Παρά τα πλεονεκτήματά της η τεχνική της ανάλυσης διαθέτει χαρακτηριστικά που καθιστούν την αποτελεσματικότητά της αμφισβητήσιμη [SKH11]. Το βασικό της μειονέκτημα είναι ότι είναι πιθανό να παράγει περιττές προτάσεις που δε θα συμπεριληφθούν στο τελικό αποτέλεσμα. ενώ η παραγωγή τους δυσχεραίνει τη διαδικασία επαναγραφής. Το αποτέλεσμα σε αυτές τις περιπτώσεις είναι μεγάλες αλυσίδες παραγωγής προτάσεων οι οποίες κοστίζουν σε χώρο και σε χρόνο εκτέλεσης του ερωτήματος.

2.3 Πρότυπα

Το μοντέλο των οντολογιών που περιγράφηκε στις προηγούμενες ενότητες αποτέλεσε το θεωρητικό υπόβαθρο πάνω στο οποίο βασίστηκε η κατασκευή του βασικού προτύπου σημασιολογικής μοντελοποίησης προς χρήση στα πλαίσια του σημασιολογικού ιστού. Είναι προφανές όμως, ότι για την υλοποίηση του σημασιολογικού ιστού, απαιτείται ένα σύνολο αλληλεπιδρουσών τεχνολογιών και προτύπων που να υποστηρίζουν την μοντελοποίηση των δεδομένων σε διάφορα επίπεδα, από το βασικό, συντακτικό επίπεδο μέχρι το σημασιολογικό επίπεδο, όπου ανήκουν οι οντολογίες. Σήμερα, οι τεχνολογίες του σημασιολογικού ιστού έχουν ωριμάσει αρκετά, και χρησιμοποιούνται διάφορα πρότυπα, τα οποία αν τοποθετηθούν από το κατώτερο ως το ψηλότερο επίπεδο αναπαράστασης, δημιουργείται η λεγόμενη στοίβα του σημασιολογικού ιστού.

Στην παρούσα ενότητα θα παρουσιάσουμε αναλυτικά τα πρότυπα του σημασιολογικού ιστού που επιτρέπουν την σημασιολογική μοντελοποίηση γνώσης και δεδομένων. Θα ξεκινήσουμε όμως με μια αναφορά στο σύνολο των τεχνολογιών που απαρτίζουν την στοίβα του σημασιολογικού ιστού. Στα κατώτερα επίπεδα βρίσκονται τεχνολογίες που χρησιμοποιούνται ήδη από τον παγκόσμιο ιστό και είναι οι εξής:

- Το πρότυπο IRI (International Resource Identifier), που επιτρέπει την αναφορά σε διάφορους πόρους με ενιαίο και μονοσήμαντο τρόπο. Τα IRI αποτελούν επέκταση των συνήθων URL που χρησιμοποιούνται ως διευθύνσεις στον παγκόσμιο ιστό, ωστόσο ο ορισμός του είναι πολύ γενικότερος. Συγκεκριμένα, πέραν του ότι επιτρέπουν την χρήση άλλων χαρακτήρων εκτός του λατινικού αλφαριθμητικού σύνολου (επεκτείνοντας έτσι το πρότυπο URI), επιτρέπουν την αναφορά σε κάποιον πόρο χωρίς να περιγράφουν απαραίτητα και τον τρόπο εντοπισμού του πόρου (παραθέτοντας π.χ. την δικτυακή του διεύθυνση).
- Το πρότυπο Unicode, που επιτρέπει την αναπαράσταση συμβολοσειρών και κειμένων στις περισσότερες γλώσσες του κόσμου.
- Την γλώσσα XML (EXtensible Markup Language), που επιτρέπει την δημιουργία εγγράφων που αποτελούνται από δομημένη πληροφορία. Ο σημασιολογικός ιστός προσδίδει νόημα (σημασιολογία) στον δομημένη πληροφορία. Η XML είναι μία γλώσσα σήμανσης, και σε ένα έγγραφο XML τα διάφορα τμήματα της πληροφορίας τοποθετούνται εντός ετικετών που φέρουν συγκεκριμένο όνομα, έτσι ώστε να επιτυχγάνεται η δομική διαφοροποίηση μεταξύ των διαφορετικών τμημάτων της πληροφορίας.
- Τους ονοματοχώρους XML, που επιτρέπουν την ταυτόχρονη αναφορά σε πόρους που ανήκουν σε διαφορετικές πηγές, με αποφυγή πιθανής σύγχυσης.

Ουσιαστικά, ένας ονοματοχώρος XML είναι ένα IRI που χρησιμοποιείται ως πρόθεμα στις ετικέτες ενός κειμένου XML. Π.χ. ορίζοντας τον ονοματοχώρο `rex` ως το IRI <http://ex.org/geo/> μπορούμε να αναφερόμαστε στον πόρο <http://ex.org/geo/Man> απλώς ως `geo:Man`. Η χρήση των ονοματοχώρων εκτός των ετικετών XML γίνεται και σε άλλα πρότυπα του σημασιολογικού ιστού για την αναπαράσταση πόρων οπουδήποτε χρειάζεται να γίνει αναφορά σε αυτούς.

Στα μεσαία επίπεδα της στοίβας του σημασιολογικού ιστού βρίσκονται τα κυρίως πρότυπα που σχετίζονται με την σημασιολογική μοντελοποίηση και είναι τα εξής:

- Το πρότυπο RDF (Resource Description Framework), που είναι ένα πλαίσιο αναπαράστασης πληροφορίας σχετικά με διάφορους πόρους υπό τη μορφή γράφου.
- Το RDF Schema, που παρέχει ένα βασικό λεξιλόγιο για το RDF ώστε να επιτρέπεται μια βασική σημασιολογική μοντελοποίηση ενός πεδίου γνώσης.
- Η γλώσσα OWL (Web Ontology Language), η οποία παρέχει ένα εξαιρετικά πλούσιο λεξιλόγιο και δομές για την διατύπωση σύνθετων οντολογικών αξιωμάτων, με βάση τις περιγραφικές λογικές.
- Η γλώσσα SPARQL (SPARQL Protocol and RDF Query Language), η οποία επιτρέπει την υποβολή ερωτημάτων επί δεδομένων που έχουν αναπαρασθεί με βάση το πρότυπο RDF.
- Η γλώσσα RIF (Rule Interchange Format), ένα πρότυπο που επιτρέπει την επέκταση μιας οντολογίας με την συμπεριληψη κανόνων σε αυτήν.

Τα ανώτερα επίπεδα της στοίβας του σημασιολογικού ιστού περιλαμβάνουν μη προτυποποιημένες ακόμη τεχνολογίες, που αφορούν διάφορα ζητήματα όπως την κρυπτογράφηση και την διεπαφή των χρηστών με την σημασιολογική πληροφορία.

Στη συνέχεια αυτής της ενότητας, θα αναφερθούμε αναλυτικότερα στα πρότυπα των μεσαίων επιπέδων που αφορούν την σημασιολογική μοντελοποίηση της πληροφορίας (με την εξαίρεση του προτύπου RIF, καθώς οι γλώσσες κανόνων αποτελούν μια από τις δυνατές επεκτάσεις της σημασιολογικής μοντελοποίησης και ξεφεύγουν από το πλαίσια του παρόντος παραδοτέου.)

2.3.1 RDF

Το πρότυπο RDF (Resource Description Framework) [RDF14a] είναι ένα γενικό πλαίσιο για την μοντελοποίηση πληροφοριών που αφορούν κάποιους πόρους. Οι πόροι μπορούν να είναι οτιδήποτε: έγγραφα, πρόσωπα, φυσικά αντικείμενα, άτομα και αφηρημένες έννοιες (όπως π.χ. τα άτομα και οι έννοιες μιας οντολογίας). Το RDF έχει σχεδιαστεί για την μοντελοποίηση της πληροφορίας για τις περιπτώσεις όπου η πληροφορία που υπάρχει στον παγκόσμιο ιστό απαιτείται να υποστεί κάποιου είδους επεξεργασία από συστήματα λογισμικού και όχι απλώς να παρουσιαστεί σε κάποιον χρήστη (όπως ισχύει π.χ. με τη γλώσσα HTML). Το RDF αποτελεί ένα κοινό πλαίσιο για την μοντελοποίηση της πληροφορίας, έτσι ώστε να μπορεί να ανταλλάσσεται μεταξύ των εφαρμογών χωρίς απώλεια νοήματος. Πιο συγκεκριμένα, το RDF επιτρέπει την δημοσίευση και διασύνδεση των δεδομένων στον παγκόσμιο ιστό. Για παράδειγμα, προσπελάζοντας την διεύθυνση <http://ex.org/res/John> (ή `res:John` για συντομία, ορίζοντας τον κατάλληλο ονοματοχώρο) αποκτούμε πρόσβαση στον μοναδικά ορισμένο πόρο `res:John`. Αυτό μπορεί να μας προσφέρει πρόσβαση σε διάφορα επιπλέον δεδομένα σχετικά με τον πόρο `res:John`, π.χ. ότι είναι κάποιος άνθρωπος και έχει ως τέκνο τον πόρο `res:Mary` (ο οποίος αναγνωρίζεται ομοίως μέσω της IRI διεύθυνσή του). Προσπελάζοντας στη συνέχεια τον πόρο `res:Mary` μπορούμε να πάρουμε περισσότερες πληροφορίες για τον άνθρωπο που αντιπροσωπεύει, π.χ. συνδέσμους προς πόρους που αντιπροσωπεύουν τους φύλους της, τα ενδιαφέροντά της κλπ.

Ο τρόπος μοντελοποίησης της πληροφορίας στο πρότυπο RDF γίνεται μέσω απλών δηλώσεων. Κάθε δήλωση έχει πάντα την εξής απλή μορφή

<υποκείμενο> <κατηγόρημα> <αντικείμενο>.

Κάθε τέτοια δήλωση εκφράζει μια σχέση μεταξύ δύο πόρων: του υποκειμένου και του αντικειμένου. Το είδος της σχέσης δηλώνεται μέσω του κατηγορήματος. Λόγω της μορφής τους, οι δηλώσεις RDF καλούνται τριάδες. Το γεγονός ότι ο ίδιος πόρος μπορεί να συμμετέχει σε πολλές τριάδες, π.χ. ως αντικείμενο σε κάποια τριάδα και ως αντικείμενο σε κάποια άλλη, καθιστά δυνατή την εύρεση συνδέσεων μεταξύ των τριάδων, όπως αναφέρθηκε στην προηγούμενη παράγραφο.

Ένα σύνολο τριάδων της παραπάνω μορφής μπορεί να θεωρηθεί ότι αποτελεί έναν γράφο, αποτελούμενο από κόμβους και ακμές. Τα υποκείμενα και τα αντικείμενα είναι οι κόμβοι του γράφου, και τα κατηγορήματα οι ακμές του. Η υποβολή ερωτημάτων επί ενός γράφου, ώστε να εξαχθεί πληροφορία που αποτελεί κάποιον υπογράφο του συνολικού γράφου ο οποίος πληροί κάποιες επιθυμητές ιδιότητες, μπορεί να γίνει μέσω της γλώσσας SPARQL.

Στο πρότυπο RDF, η αναφορά σε έναν πόρο γίνεται μέσω ενός IRI. Έτσι, για παράδειγμα η τριάδα ότι ο John έχει ως τέκνο την Mary που αναφέρθηκε προηγουμένως μπορεί να δηλωθεί ως εξής: `res:John reo:hasChild res:Mary`, με την

υπόθεση ότι τα `res:John` και `res:Mary` είναι πόροι που αντιπροσωπεύουν τα εν λόγω πρόσωπα. Εκτός από IRI, το αντικείμενο μιας τριάδας μπορεί να είναι κάποιο λεκτικό. Τα λεκτικά μπορεί να είναι κάποια συμβολοσειρά όπως «John Smith», κάποια ημερομηνία όπως «12/01/1988» ή ένας αριθμός όπως «34». Το κάθε λεκτικό έχει έναν τύπο δεδομένων που ορίζει το είδος των ορθών τιμών που μπορεί να λάβει (δηλαδή αν είναι συμβολοσειρά, ακέραιος αριθμός, αριθμός κινητής υποδιαστολής, ημερομηνία, κλπ). Οι συμβολοσειρές μπορούν να συνοδεύονται και από μια επιγραφή που να μαρτυρεί την γλώσσα. Π.χ. η «John Smith» μπορεί να συνοδεύεται από την επιγραφή `en`. Με τον τρόπο αυτό το όνομα και η ηλικία του πόρου `res:John` μπορούν να δηλωθούν ως εξής:

```
res:John reo:name "John Smith"@en
res:John reo:age "34"^^xsd:integer.
```

Το `xsd:integer` είναι ένας πόρος που αντιπροσωπεύει κατά σύμβαση τον τύπο δεδομένων των ακεραίων. Όπως θα δούμε στη συνέχεια ο τρόπος αυτός δήλωσης των τριάδων καλείται σύνταξη Turtle.

Το μοντέλο δεδομένων RDF δεν κάνει καμία υπόθεση και δεν διαθέτει καμία γνώση σχετικά με το τι αντιπροσωπεύει το IRI ενός πόρου. Ωστόσο, στα διάφορα IRI μπορεί να αποδοθεί κάποια προσυμφωνημένη σημασία βάσει κάποιας σύμβασης. Με τον τρόπο αυτό μπορούν να δημιουργηθούν λεξιλόγια αποτελούμενα από τα IRI που έχουν συμφωνηθεί ότι αντιπροσωπεύουν συγκεκριμένα αντικείμενα ή έννοιες του κόσμου. Π.χ. το IRI `http://ex.org/neo/hasChild` στο παραπάνω παράδειγμα χρησιμοποιείται για να περιγράψει την σχέση ενός ατόμου ως παιδί κάποιου άλλου.

2.3.2 RDF Schema

Ένα βασικό λεξιλόγιο που κατασκευάστηκε ώστε να υποστηριχτεί η χρήση του προτύπου RDF είναι το RDF Schema [RDF14b], το οποίο επιτρέπει τον ορισμό ορισμένων βασικών σημασιολογικών ιδιοτήτων επί των δεδομένων RDF. Π.χ. με χρήση του RDF Schema μπορεί να οριστεί ότι το υποκείμενο και το αντικείμενο του κατηγορήματος `res:hasChild` πρέπει να είναι πόροι που ανήκουν στην κλάση `res:Person`. Η κλάση είναι η βασική έννοια που χρησιμοποιεί το RDF Schema για να ορίσει κατηγορίες πόρων (αντίστοιχα με την έννοιες των οντολογιών), ενώ οι σχέσεις μεταξύ πόρων δηλώνονται μέσω κατηγορημάτων που καλούνται *ιδιότητες*. Με χρήση του RDF Schema μπορούν να οριστούν ιεραρχίες κλάσεων και ιδιοτήτων, όπως σε μια οντολογία, καθώς και πεδία ορισμού και τιμών των ιδιοτήτων.

Στο RDF Schema, όλοι οι περιγραφόμενοι πόροι θεωρούνται στιγμιότυπα της γενικής κλάσης `rdfs:Resource`. Η κλάση των πόρων που αντιπροσωπεύουν κλάσεις, λεκτικά, τύπους δεδομένων, ιδιότητες είναι οι `rdfs:Class`, `rdfs:Literal`,

Τριάδα	Περιγραφή
C rdf:type rdfs:Class	Ο πόρος C είναι μια κλάση RDF.
P rdf:type rdf:Property	Ο πόρος P είναι ιδιότητα RDF.
I rdf:type C	Ο πόρος I είναι στιγμιότυπο της κλάσης C.
C rdfs:subClassOf D	Η κλάση C είναι υποκλάση της D.
P rdfs:subPropertyOf R	Η ιδιότητα P είναι υποϊδιότητα της R.
P rdfs:domain C	Η κλάση C είναι πεδίο ορισμού της ιδιότητας P.
P rdfs:range C	Η κλάση C είναι πεδίο τιμών της ιδιότητας P.

Πίνακας 3: Βασικές δυνατότητες μοντελοποίησης του RDF Schema

rdfs:Datatype και rdf:Property αντίστοιχα. Υπάρχει επίσης η βασική ιδιότητα rdf:type που χρησιμοποιείται ως κατηγόρημα σε μια τριάδα για να δηλώσει ότι το υποκείμενο είναι στιγμιότυπο της κλάσης που αντιπροσωπεύει το αντικείμενο της τριάδας. rdf είναι ο ονοματοχώρος <http://www.w3.org/1999/02/22-rdf-syntax-ns#>, και rdfs ο ονοματοχώρος <http://www.w3.org/2000/01/rdf-schema#> αντίστοιχα. Με βάση αυτά, οι βασικές δυνατότητες μοντελοποίησης ενός πεδίου ενδιαφέροντος που παρέχει το RDF Schema παρουσιάζονται στον Πίνακα 3.

2.3.3 OWL

Όπως αναφέρθηκε παραπάνω, το λεξιλόγιο του RDF Schema προσφέρει κάποιες βασικές δυνατότητες ορισμού ταξονομικών σχέσεων, σχέσεων μεταξύ αντικειμένων και αξιωμάτων που αφορούν τα πεδία ορισμού και τιμών κάποιων ιδιότητων. Ωστόσο, οι δυνατότητες αυτές αφενός δεν συμβαδίζουν πάντα με την σημασιολογία που επιτρέπει η φορμαλιστική αναπαράσταση γνώσης ως οντολογία (π.χ. στις περιγραφικές λογικές δεν μπορεί μια έννοια να είναι στιγμιότυπο μιας άλλης έννοιας, κάτι το οποίο επιτρέπεται από το RDF Schema), και αφετέρου είναι εκφραστικά αρκετά περιορισμένες (δεν επιτρέπουν για παράδειγμα τον ορισμό μιας σύνθετης έννοιας ως ένωσης δύο έννοιών). Έτσι, για να καταστεί δυνατή η περιγραφή πεδίων γνώσης ως οντολογιών με βάση τις αρχές των περιγραφικών λογικών, αναπτύχθηκε η γλώσσα περιγραφής οντολογιών OWL (Web Ontology Language).

Καθώς η OWL αποτελεί πρότυπο του σημασιολογικού ιστού, όλοι οι πόροι τους οποίους περιγράφει ένα έγγραφο OWL είναι IRI. Πρέπει να σημειωθεί, η OWL είναι μια γλώσσα μοντελοποίησης οντολογιών σε αφαιρετικό επίπεδο, και όχι μια συγκεκριμένη σύνταξη. Μάλιστα, υπάρχουν διάφοροι ισοδύναμοι τρόποι να συνταχθεί μια οντολογία OWL. Μεταξύ αυτών είναι η συναρτησιακού τύπου σύνταξη, την οποία θα χρησιμοποιήσουμε στη συνέχεια καθώς είναι αρκετά ευανάγνωστη για τον άνθρωπο.

Δεδομένου ότι η OWL βασίζεται στις περιγραφικές λογικές, υποστηρίζει κα-

Είδος	Έκφραση	Παράδειγμα και περιγραφή
Υπαγωγή κλάσεων	$A \sqsubseteq B$	SubClassOf (reo:Woman reo:Person): Η κλάση res:Woman είναι υποκλάση της reo:Person.
Ισοδυναμία κλάσεων	$A \equiv B$	EquivalentClasses (reo:Person reo:Human): Οι κλάσεις reo:Person και res:Human είναι ισοδύναμες.
Ξενικότητα κλάσεων		DisjointClasses (reo:Woman reo:Man): Κανένα άτομο δεν μπορεί να είναι ταυτόχρονα στιγμιότυπο των κλάσεων reo:Woman και reo:Man.
Ξενικότητα ένωσης κλάσεων		DisjointClasses (reo:Child reo:Boy reo:Girl): Κάθε στιγμιότυπο της κλάσης reo:Child πρέπει να είναι και στιγμιότυπο είτε της κλάσης reo:Boy είτε της reo:Girl, αλλά όχι και των δύο.

Πίνακας 4: Αξιώματα κλάσεων της OWL.

τασκευαστές εννοιών και ρόλων, καθώς και διάφορα είδη αξιωμάτων ορολογίας και ισχυρισμών. Θα ξεκινήσουμε την έκθεση των εκφραστικών δυνατοτήτων της OWL παραθέτοντας τα είδη αξιωμάτων που μπορεί να εκφράσει. Τα υποστηριζόμενα αξιώματα που αφορούν την υπαγωγή, ισοδυναμία και ξενικότητα κλάσεων παρουσιάζονται στον Πίνακα 4. Για κάθε αξιώμα δίνεται το είδος του, η έκφραση του σε περιγραφική λογική (αν υπάρχει) και τέλος ένα παράδειγμα διατύπωσής του στην συναρτησιακού τύπου σύνταξη, μαζί με σύντομη επεξήγηση. Τα αξιώματα στα οποία εμπλέκεται η χρήση ιδιοτήτων, αφορούν είτε την δήλωση υπαγωγής, ισοδυναμίας και ξενικότητας ιδιοτήτων, είτε τον ορισμό των πεδίων τιμών ή ορισμού μιας ιδιότητας. Συγκεκριμένα υποστηρίζονται τα αξιώματα του Πίνακα 5.

Υποστηρίζονται επίσης αξιώματα με τα οποία δηλώνονται ορισμένα μαθηματικά χαρακτηριστικά των ιδιοτήτων, τα οποία παρουσιάζονται στον Πίνακα 6. Από τον πίνακα απουσιάζει η περιγραφή, όπου είναι προφανής.

Τα αξιώματα που αφορούν την περιγραφή ατόμων και την δήλωσή τους ως στιγμιοτύπων κλάσεων ή ιδιοτήτων παρουσιάζονται στον Πίνακα 7.

Στα παραδείγματα αξιωμάτων που παρουσιάστηκαν στους παραπάνω πίνακες χρησιμοποιήθηκαν απλές κλάσεις. Ωστόσο, εν γένει, στα διάφορα αξιώματα τη θέση μιας απλής κλάσης μπορεί να έχει μια σύνθετη κλάση που να ορίζεται π.χ. ως η τομή, ή η ένωση δύο άλλων κλάσεων. Οι κατασκευαστές κλάσεων που υποστηρίζει η OWL παρουσιάζονται στον Πίνακα 8.

Μια σύνθετη κλάση, εκτός των παραπάνω τρόπων, μπορεί να οριστεί και μέσω

Είδος	Έκφραση	Παράδειγμα και περιγραφή
Υπαγωγή ιδιοτήτων	$R \sqsubseteq S$	SubObjectPropertyOf (reo:hasWife reo:hasSpouse): Η reo:hasWife είναι υποϊδιότητα της reo:hasSpouse.
Ισοδυναμία ιδιοτήτων	$R \equiv S$	EquivalentObjectProperties (reo:hasChild ont:child): Οι ιδιότητες reo:hasChild και ont2:child είναι ισοδύναμες.
Ξενικότητα ιδιοτήτων		DisjointObjectProperties (reo:hasParent reo:hasSpouse): Οι reo:hasParent και reo:hasSpouse είναι ξένες μεταξύ τους ιδιότητες.
Αντίστροφη ιδιότητα	$R \equiv S^-$	InverseObjectProperties (reo:hasParent reo:hasChild): Η reo:hasParent είναι αντίστροφη ιδιότητα της reo:hasChild.
Περιορισμός πεδίου ορισμού	$\exists R \sqsubseteq A$	ObjectPropertyDomain (reo:hasWife reo:Man): Πεδίο ορισμού της ιδιότητας reo:hasWife είναι η κλάση reo:Man.
Περιορισμός πεδίου τιμών	$T \sqsubseteq \forall R.A$	ObjectPropertyRange (reo:hasWife reo:Woman): Πεδίο τιμών της ιδιότητας reo:hasWife είναι η κλάση reo:Woman.

Πίνακας 5: Αξιώματα ιδιοτήτων.

μιας ιδιότητας, δηλαδή ως το σύνολο των αντικειμένων που σχετίζονται με συγκεκριμένο τρόπο μέσω μιας ιδιότητας με κάποια άλλα αντικείμενα. Οι κατασκευαστές αυτοί κλάσεων παρουσιάζονται στον Πίνακα 9. Τέλος υπάρχουν και οι κατασκευαστές ιδιοτήτων του Πίνακα 10.

Πέραν των αξιωμάτων και των κατασκευαστών που αναφέρθηκαν παραπάνω, η OWL υποστηρίζει και την χρήση επισημειώσεων, οι οποίες μπορεί να αναφέρονται στην ίδια την οντολογία, σε οντότητες ή σε αξιώματά της. Οι επισημειώσεις δεν φέρουν σημασιολογικό χαρακτήρα, προσφέρουν όμως την δυνατότητα συμπερίληψης στην οντολογία επιπλέον χρήσιμων στοιχείων όπως είναι π.χ. κάποια σχόλια πάνω σε κάποια οντότητα, μια λεκτική απόδοση του τι αντιπροσωπεύει μια κλάση, κλπ. Τα είδη των επισημειώσεων ορίζονται από τον χρήστη. Ένα παράδειγμα επισημείωσης είναι το AnnotationAssertion (rdfs:label res:Person "Το σύνολο όλων των ανθρώπων.").

Τέλος, αξίζει να αναφερθεί ότι καθώς μια οντολογία συντάσσεται συνήθως για να χρησιμοποιηθεί από διάφορες εφαρμογές, πρέπει να μπορεί να γίνει αναφορά σε αυτήν. Για τον λόγο αυτό, μια οντολογία είναι και αυτή ένας πόρος, που πρέπει να έχει ένα συγκεκριμένο IRI όπως π.χ. το <http://ex.org/neo/> που ανα-

Είδος	Παράδειγμα και περιγραφή
Συμμετρικότητα	SymmetricObjectProperty (reo:hasSpouse)
Ασυμμετρικότητα	AsymmetricObjectProperty (reo:hasChild)
Ανακλαστικότητα	ReflexiveObjectProperty (reo:hasRelative)
Μη ανακλαστικότητα	IrreflexiveObjectProperty (reo:parentOf)
Αμφιμονοσημανότητα	FunctionalObjectProperty (reo:hasHusband)
Μεταβατικότητα	TransitiveObjectProperty (reo:hasAncestor)
Αμφιμονοσημαντότητα αντιστρόφου	InverseFunctionalObjectProperty (reo:hasHusband): Η αντίστροφη ιδιότητα της reo:hasHusband είναι αμφιμονοσήμαντη.

Πίνακας 6: Αξιώματα χαρακτηριστικών ιδιοτήτων.

Είδος	Έκφραση	Παράδειγμα και περιγραφή
Ταυτότητα ατόμων	$a = b$	SameIndividual(res:James res:Jim): Τα άτομα res:James, res:Jim είναι ίδια.
Διαφορετικότητα ατόμων	$a \neq b$	DifferentIndividuals(res:John res:Bill): Το άτομο res:John είναι διαφορετικό από το res:Bill.
Δήλωση στιγμιοτύπου κλάσης	$C(a)$	ClassAssertion(reo:Person reo:Mary): Το άτομο res:Mary είναι στιγμότυπο της κλάσης reo:Person.
Δήλωση στιγμιοτύπου ιδιότητας	$R(a, b)$	ObjectPropertyAssertion(reo:hasWife res:John res:Mary): Το ζεύγος ατόμων (res:John, res:Mary) είναι στιγμότυπο της ιδιότητας reo:hasWife.
Αρνητική δήλωση στιγμιοτύπου ιδιότητας		NegativeObjectPropertyAssertion(reo:hasWife res:Bill res:Mary): Το ζεύγος ατόμων (res:John, res:Mary) δεν είναι στιγμότυπο της ιδιότητας reo:hasWife.

Πίνακας 7: Αξιώματα δήλωσης ισχυρισμών.

φέρθηκε παραπάνω. Με τη χρήση κατάλληλων IRI για μια οντολογία είναι επίσης δυνατή και η διαχείριση διαφορετικών εκδόσεων της ίδιας οντολογίας (π.χ. <http://ex.org/reo/v1> και <http://ex.org/reo/v2/>). Επίσης ο ορισμός μιας οντολογίας μπορεί να χρησιμοποιεί ονοματολογία ήδη ορισμένη σε άλλες οντολογίες· αυτό επιτρέπεται μέσω την δηλώσεων εισαγωγής τρίτων οντολογιών σε μια νέα οντολογία, με τις οποίες οι τρίτες οντολογίες καθίστανται ουσιαστικά τμήμα της νέας οντολογίας.

Είδος	Έκφραση	Παράδειγμα και περιγραφή
Τομή κλάσεων	$A \sqcap B$	ObjectIntersectionOf (reo:Woman reo:Parent): Η ένωση των κλάσεων reo:Woman και reo:Parent.
Ένωση κλάσεων	$A \sqcup B$	ObjectUnionOf (reo:Mother reo:Father): Η τομή των κλάσεων reo:Mother και reo:Father.
Συμπλήρωμα κλάσης	$\neg A \sqcup B$	ObjectComplementOf (reo:Parent): Το συμπλήρωμα της κλάσης reo:Parent
Απαρίθμηση		ObjectOneOf (res:Bill res:John res:Mary): Η κλάση που αποτελείται από τα res:Bill res:John και res:Mary.

Πίνακας 8: Κατασκευαστές κλάσεων.

Προφίλ της OWL

Λόγω της πληθώρας κατασκευαστών κλάσεων και ιδιοτήτων που υποστηρίζει, η OWL είναι ιδιαίτερα εκφραστική (τόσο υπολογιστικά όσο και όσον αφορά τον τελικό χρήστη) με αποτέλεσμα να υπάρχουν διάφορα προβλήματα κατά την υλοποίηση πρακτικών συστημάτων συλλογιστικής που να την υποστηρίζουν. Για τον λόγο αυτό, η OWL ορίζει τρία προφίλ, τα οποία είναι υποσύνολα της συνολικής γλώσσας με μειωμένη εκφραστικότητα η οποία επιτρέπει την πρακτική χρήση της OWL από διάφορες εφαρμογές. Ο βασικός παράγοντας που καθόρισε τον ορισμό των προφίλ αυτών, είναι η υπολογιστική τους πολυπλοκότητα, σε συνάρτηση με τις ήδη υπάρχουσες πρακτικές ανάγκες πολλών συστημάτων.

Προκειμένου να εξασφαλιστεί η δυνατότητα κλιμάκωσης της συλλογιστικής με την OWL, τα διάφορα προφίλ θέτουν ορισμένους κοινούς περιορισμούς ως προς την εκφραστικότητά τους. Γενικά, απαγορεύουν την χρήση άρνησης και διάζευξης, καθώς αυτές είναι βασικές αιτίες αύξησης της πολυπλοκότητας της συλλογιστικής σε μη διαχειρίσιμα επίπεδα, ενώ αντιθέτως σπανίως απαιτούνται σε πραγματικά προβλήματα μοντελοποίησης. Έτσι, ένα παράδειγμα αξιώματος που δεν επιτρέπεται σε κανένα προφίλ της OWL είναι η δήλωση ξενικότητας ένωσης κλάσεων. Τα τρία προφίλ της OWL είναι το OWL 2 EL, OWL 2 QL και OWL 2 RL.

Η OWL 2 EL είναι χρήσιμη σε εφαρμογές που διαχειρίζονται οντολογίες που περιέχουν πολύ μεγάλο αριθμό κλάσεων και ιδιοτήτων. Η OWL 2 EL καλύπτει τις εκφραστικές δυνατότητες που απαιτούνται από τις περισσότερες τέτοιου είδους εφαρμογές και τα βασικά προβλήματα συλλογιστικής μπορούν να αντιμετωπιστούν σε χρόνο πολυωνυμικό ως προς το μέγεθος της οντολογίας. Για την OWL 2 EL έχουν κατασκευαστεί ειδικοί αλγόριθμοι συλλογιστικής με καλές ιδιότητες

Είδος	Έκφραση	Παράδειγμα και περιγραφή
Υπαρξιακή ποσοδεικτοδότηση	$\exists R.A$	<code>ObjectSomeValuesFrom(:hasChild :Person)</code> : Η κλάση των ατόμων που συνδέονται μέσω της ιδιότητας <code>:hasChild</code> με κάποιο στιγμιότυπο της κλάσης <code>:Person</code> .
Καθολική ποσοδεικτοδότηση	$\forall R.A$	<code>ObjectAllValuesFrom(:hasChild :HappyPerson)</code> : Η κλάση των ατόμων που σχετίζονται μέσω της ιδιότητας <code>:hasChild</code> μόνο με στιγμιότυπα της κλάσης <code>:HappyPerson</code> .
Περιορισμός τιμής ατόμου		<code>ObjectHasValue(reo:hasParent res:John)</code> : Η κλάση των ατόμων που συνδέονται μέσω της ιδιότητας <code>reo:hasParent</code> με το άτομο <code>res:John</code> .
Αυτοπεριορισμός		<code>ObjectHasSelf(reo:loves)</code> : Η κλάση όλων των ατόμων που σχετίζονται με τον εαυτό τους μέσω της ιδιότητας <code>reo:loves</code> .
Μέγιστη πληθικότητα	$\leq nR.C$	<code>ObjectMaxCardinality(2 reo:hasChild res:Parent)</code> : Η κλάση των ατόμων που σχετίζονται μέσω της ιδιότητας <code>res:hasChild</code> με το πολύ 2 στιγμιότυπα της κλάσης <code>res:Parent</code> . Ο προσδιορισμός της τελευταίας κλάσης είναι προαιρετικός.
Ελάχιστη πληθικότητα	$\geq nR.C$	<code>ObjectMinCardinality(2 reo:hasChild res:Parent)</code> : Η κλάση των ατόμων που σχετίζονται μέσω της κλάσης <code>res:hasChild</code> με τουλάχιστον 2 στιγμιότυπα της κλάσης <code>res:Parent</code> .
Ακριβής πληθικότητα	$= nR.C$	<code>ObjectExactCardinality(2 reo:hasChild res:Parent)</code> : Η κλάση των ατόμων που σχετίζονται μέσω της κλάσης <code>res:hasChild</code> με το ακριβώς 2 στιγμιότυπα της κλάσης <code>res:Parent</code> .

Πίνακας 9: Κατασκευαστές κλάσεων βάση ιδιοτήτων.

κλιμάκωσης. Η OWL 2 EL βασίζεται στην οικογένεια περιγραφικών λογικών \mathcal{EL} . Η OWL 2 QL έχει κατασκευαστεί για εφαρμογές που διαχειρίζονται πολύ μεγάλους όγκους δεδομένων και όπου η απάντηση ερωτημάτων είναι η πιο συχνή

Είδος	Έκφραση	Παράδειγμα και περιγραφή
Αντίστροφη ιδιότητα	P^-	<code>ObjectInverseOf(res:hasParent)</code> : Η αντίστροφη της ιδιότητας <code>res:hasParent</code> .
Αλυσίδα ιδιοτήτων	$P \circ R$	<code>ObjectPropertyChain(res:hasParent res:hasParent)</code> . Η ιδιότητα μεταξύ αντικειμένων που σχετίζονται δύο φορές διαδοχικά μεταξύ τους μέσω της <code>res:hasParent</code> .

Πίνακας 10: Κατασκευαστές ιδιοτήτων.

εργασία που καλείται να πραγματοποιήσει ένας αλγόριθμος συλλογιστικής. Για να επιτευχθεί αυτό η εκφραστικότητά της είναι αρκετά περιορισμένη. Στην OWL 2 QL η απάντηση συζευκτικών ερωτημάτων μπορεί να γίνει με τη χρήση βάσεων δεδομένων ως των υποκείμενων αποθηκών δεδομένων, πράγμα που σημαίνει ότι ένα ερώτημα εκπεφρασμένο βάσει της οντολογίας μπορεί να μεταφραστεί σε ένα σύνολο ερωτημάτων SQL που μπορούν να εκτελεστούν επί της υποκείμενης βάσης δεδομένων. Αυτό σημαίνει ότι δεν απαιτείται καμία επέμβαση στα υπάρχοντα δεδομένα. Η απάντησή των ερωτημάτων μπορεί να γίνει ορθά και πλήρως σε λογαριθμικό χώρο ως προς τον όγκο των δεδομένων (των ισχυρισμών). Όπως και στην OWL 2 EL, τα προβλήματα ελέγχου συνέπειας και υπαγωγής κλάσεων μπορούν να επιλυθούν σε πολυωνυμικό χρόνο. Η εκφραστικότητα της OWL 2 QL βασίζεται οικογένεια περιγραφικών λογικών DL-Lite.

Η OWL 2 RL είναι χρήσιμη σε εφαρμογές που απαιτούν η συλλογιστική να έχει καλές ιδιότητες κλιμάκωσης χωρίς όμως να θυσιάζεται υπερβολικά πολύ η εκφραστική δύναμη. Στην OWL 2 RL τα προβλήματα του ελέγχου συνέπειας, της ικανοποιησιμότητας μιας κλάσης, του ελέγχου υπαγωγής κλάσεων, του ελέγχου στιγμιοτύπων και της απάντησης σε συζευκτικά ερωτήματα μπορούν να επιλυθούν σε πολυωνυμικό χρόνο ως προς το μέγεθος της οντολογίας.

2.3.4 Λεξιλόγια

Όπως προαναφέρθηκε, συλλογές από IRI που έχουν συμφωνηθεί πώς αντιπροσωπεύουν συγκεκριμένες κλάσεις ή ιδιότητες του κόσμου μπορούν να χρησιμοποιηθούν ως λεξιλόγια. Τα IRI αυτά μπορούν να χρησιμοποιηθούν ως κατηγορήματα ή αντικείμενα σε βάσεις γνώσης αποτελούμενες από τριάδες RDF, έτσι ώστε τα περιεχόμενα της βάση γνώσης να είναι άμεσα ερμηνεύσιμα από τρίτους με βάση την κατά σύμβαση ερμηνεία των όρων των χρησιμοποιούμενων λεξιλογίων. Έτσι πέραν του RDF Schema που αναφέρθηκε προηγουμένως, έχει κατασκευαστεί πληθώρα λεξιλογίων RDF, για την κάλυψη διαφόρων πεδίων ενδιαφέροντος. Θα απαριθμήσουμε μερικά από τα πιο συχνά χρησιμοποιούμενα

- Dublin Core. Το Dublin Core παρέχει ένα βασικό λεξιλόγιο για την περιγραφή πόρων του παγκόσμιου ιστού (π.χ. βίντεο, εικόνων, ιστοσελίδων, κλπ.) καθώς και για φυσικούς πόρους όπως βιβλία, CD και διάφορα έργα τέχνης. Για παράδειγμα ορίζει ιδιότητες όπως dc:title, dc:creator, dc:subject, dc:description, dc:publisher, dc:language, κλπ. Ο ονοματοχώρος dc είναι το IRI <http://purl.org/dc/terms/>.
- Simple Knowledge Organization System (SKOS). Το SKOS είναι ένα λεξιλόγια για την αναπαράσταση θησαυρών, ταξονομιών και άλλων σχημάτων κατηγοριοποίησης. Έτσι ορίζει για παράδειγμα ιδιότητες όπως skos:broader, skos:narrower, skos:related για την περιγραφή των σχέσεων κατηγοριοποίησης. Ο ονοματοχώρος skos είναι το IRI <http://www.w3.org/2004/02/skos/core>.
- schema.org Το schema.org ορίζει ένα βασικό λεξιλόγιο αλλά και μια οντολογία για τον σημασιολογικό χαρακτηρισμό του περιεχομένου των ιστοσελίδων. Έτσι ορίζει πληθώρα κλάσεων για διάφορα πεδία ενδιαφέροντος όπως schema:Event, schema:Person, schema:Product, schema:Place. Ο ονοματοχώρος schema είναι το IRI <http://schema.org/>.
- Friend of a Friend (FOAF) Το FOAF είναι ένα λεξιλόγιο για την περιγραφή ανθρώπων, των δραστηριοτήτων και ενδιαφερόντων τους, καθώς και των σχέσεων τους με άλλους ανθρώπους και αντικείμενα· είναι κατάλληλο για την περιγραφή κοινωνικών δικτύων. Περιλαμβάνει π.χ. ιδιότητες όπως foaf:name, foaf:homepage, foaf:interest, foaf:knows. Ο ονοματοχώρος foaf είναι το IRI <http://xmlns.com/foaf/0.1/>.

2.3.5 Σειριοποίηση γράφων RDF

Όπως αναφέρθηκε προηγουμένως, ένας γράφος RDF ορίζεται αφαιρετικά, βάσει των τριάδων που τον αποτελούν, χωρίς να προϋποτίθεται κάποια συγκεκριμένη σύνταξη. Για την σειριοποίηση του προτύπου RDF, έχουν αναπτυχθεί διάφοροι τρόποι σύνταξης εγγράφων που περιέχουν γράφους RDF, οι οποίοι είναι όλοι μεταξύ τους ισοδύναμοι. Η βασικότεροι τρόποι σύνταξης είναι οι εξής:

- *N-Triples* Είναι η πιο απλή σύνταξη. Ο γράφος γράφεται ως ένα σύνολο τριάδων, μία σε κάθε γραμμή, η οποία τερματίζεται με μια τελεία. Δεν επιτρέπεται η χρήση ονοματοχώρων και τα IRI γράφονται εντός αγκυλών. Ακολουθεί ένα απλό παράδειγμα (λόγω περιορισμένου χώρου η κάθε τριάδα δεν γράφεται στην ίδια γραμμή):

```
<http://ex.org/res/John>
<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
```

```

<http://xmlns.com/foaf/0.1/Person> .
<http://ex.org/res/John>
<http://schema.org/birthDate>
"1990-07-04"^^<http://www.w3.org/2001/XMLSchema#date> .

```

- *Turtle*. Η σύνταξη Turtle βελτιώνει την σύνταξη N-Triples εισάγοντας διάφορες ευκολίες, όπως την δυνατότητα ορισμού ονοματοχώρων, την χρήση καταλόγων τριάδων με κοινό υποκείμενο καθώς και συντμήσεις για τους τύπους δεδομένων. Ακολουθεί το ίδιο παράδειγμα με προηγουμένως:

```

BASE   <http://ex.org/res/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX schema: <http://schema.org/>

<John>
  a foaf:Person ;
  schema:birthDate "1990-07-04"^^xsd:date .

```

Η συντομογραφία a αντιστοιχεί στην ιδιότητα `rdf:type`.

- *RDF/XML* Με βάση αυτή τη σύνταξη ο γράφος RDF αναπαρίσταται ως ένα έγγραφο XML. Η σύνταξη αυτή ήταν η μόνη δυνατή την εποχή που προτάθηκε το πρότυπο RDF. Ακολουθεί το ίδιο παράδειγμα με προηγουμένως:

```

<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF xmlns:foaf="http://xmlns.com/foaf/0.1/"
           xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
           xmlns:schema="http://schema.org"/>
  <rdf:Description rdf:about="http://ex.org/res/John">
    <rdf:type rdf:resource="http://xmlns.com/foaf/0.1/Person"/>
    <schema:birthDate rdf:datatype="http://www.w3.org/2001/XMLSchema#date">
      1990-07-04</schema:birthDate>
  </rdf:Description>
</rdf:RDF>

```

Πρέπει να σημειωθεί ότι στα πλαίσια των προτύπων του σημασιολογικού ιστού πολλές γλώσσες ορίζονται αφαιρετικά χωρίς κάποια συγκεκριμένη σύνταξη, και τελικά σειριοποιούνται ως γράφοι RDF με χρήση μιας εκ των παραπάνω συντάξεων. Αντό συμβαίνει όπως προαναφέραμε και για την γλώσσα OWL, για την οποία υπάρχουν διάφοροι τρόποι σύνταξής και μπορεί να σειριοποιηθεί ως γράφος RDF μέσω της σύνταξης RDF/XML, ή της σύνταξης Turtle. Άλλες δυνατοί τρόποι σύνταξης της OWL, πέραν την συναρτησιακού τύπου σύνταξης που χρησιμοποιήσαμε, είναι η σύνταξη Manchester και η σύνταξη OWL/XML.

2.3.6 Γλώσσα ερωτημάτων SPARQL

Η γλώσσα SPARQL [SPA08] είναι μία γλώσσα για τη διατύπωση ερωτημάτων σε ένα γράφο RDF δεδομένων. Προτυποποιήθηκε το 2008 από το World Wide Web Consortium (W3C) και αποτελεί μία από τις βασικές τεχνολογίες του Σημασιολογικού Ιστού (Semantic Web). Επιτρέπει τη σημασιολογική διατύπωση ερωτημάτων σε βάσεις δεδομένων όπου τα δεδομένα ακολουθούν το RDF πρότυπο. Τα RDF δεδομένα μπορούν να θεωρηθούν από τη σκοπιά των σχεσιακών βάσεων δεδομένων ως πίνακες με τρεις στήλες που αντιστοιχίζονται στο υποκείμενο, το κατηγορούμενο και το αντικείμενο. Τα ερωτήματα που διατυπώνονται στη γλώσσα SPARQL αποτελούνται από τριάδες (triple patterns), συζεύξεις, διαζεύξεις και προαιρετικά τμήματα. Οι τριάδες έχουν τη μορφή των RDF τριάδων όπου στη θέση του υποκειμένου, του κατηγορήματος ή του αντικειμένου μπορεί να υπάρχουν μεταβλητές. Παρακάτω δίνουμε ένα παράδειγμα σύνταξης SPARQL ερωτήματος:

```
SELECT ?x FROM <IRI οντολογίας> WHERE ?x :isAuthor ?y AND ?y :Book
```

Στο συγκεκριμένο ερώτημα αυτό ζητάμε όλα τα αντικείμενα μιας οντολογίας που είναι συγγραφείς κάποιου βιβλίου. Η τιμή που ακολουθεί το `FROM` είναι το IRI μιας οντολογίας ή ενός γράφου RDF δεδομένων από τον οποίο θα ανακτηθούν οι απαντήσεις του ερωτήματος. Η τιμή που ακολουθεί το `WHERE` είναι ένα σύνολο από RDF τριάδες όπου κάποιοι κόμβοι ή ακμές έχουν αντικατασταθεί από μεταβλητές. Οι μεταβλητές που ακολουθούν το `SELECT` είναι εκείνες από τις οποίες θα προκύψουν οι απαντήσεις του ερωτήματος. Συγκεκριμένα, οι μεταβλητές του `WHERE` αντιστοιχίζονται σε ένα σύνολο κόμβων και ακμών του RDF γράφου από το οποίο αυτό επιλέγονται όσοι αντιστοιχούν στις μεταβλητές της `SELECT` ώστε να επιστραφεί η απάντηση του ερωτήματος.

Όπως γίνεται κατανοητό η SPARQL επιτρέπει τη διατυπώση ερωτημάτων σε δεδομένα όπου το σχήμα που ακολουθούν καθορίζεται από την ίδια τους τη δομή (δεδομένα σε RDF), ενώ την ίδια στιγμή είναι δυνατό ένα εξωτερικό σχήμα (οντολογία) να καθορίζει τον τρόπο που μπορούν σύνολα δεδομένων από διαφορετικές πηγές να συνδυαστούν μεταξύ τους.

Η αποτίμηση των SPARQL ερωτημάτων πραγματοποιείται αντιστοιχίζοντας τις μεταβλητές του ερωτήματος με κόμβους του RDF γράφου. Ένα σύνολο από τριάδες ενός SPARQL ερωτήματος ονομάζεται *βασικό πρότυπο γράφου* (basic graph pattern, BGP). Ετσι, κάθε απόδοση τιμών στις μεταβλητές του ερωτήματος που σχετίζεται με κάποιο υπογράφο του RDF γράφου αποτελεί απάντηση του ερωτήματος. Η τεχνική αυτή είναι γνωστή ως *τεχνική αντιστοίχησης υπογράφου*. Επίσης η αποτίμηση των SPARQL ερωτημάτων μπορεί να θεωρηθεί και ως μια διαδικασία απλής συνεπαγωγής καθώς αντιμετωπίζεται και ως σχέση συνεπαγωγής μεταξύ RDF γράφων. Επιστρέφοντας στο παράδειγμά μας, η διαδικασία

αντιστοίχισης των υπογράφων θα επιστρέψει όλους τους κόμβους του γράφου δεδομένων <IRI οντολογίας> που αντιστοιχούν στο συγγραφέα ενός βιβλίου. Εάν υποθέσουμε ότι κάποιος συγγραφέας δεν είναι σαφώς δηλωμένος ως συγγραφέας κάποιου συγκεκριμένου βιβλίου τότε δε θα συμπεριληφθεί στο σύνολο των απαντήσεων του ερωτήματος. Κάτι τέτοιο μπορεί να συμβεί παρά τη σχετική γνώση της οντολογίας ότι κάθε άτομο που είναι δηλωμένο ως συγγραφέας έχει γράψει και κάποιο βιβλίο. Όπως γίνεται κατανοητό προκειμένου να ανακτηθούν αυτής της φύσεως οι απαντήσεις πρέπει να διεξαχθεί κάποια διαδικασία συνεπαγωγής. Η βασική σημιολογία της SPARQL δεν υποστηρίζει διαδικασίες συνεπαγωγής. Προκειμένου να αντιμετωπιστούν τέτοιες περιπτώσεις δεδομένων και ερωτημάτων χρειαζόμαστε τη συνεπαγωγή της OWL. Η SPARQL 1.1 [SPA] επιτρέπει την αποτίμηση των ερωτημάτων με βάση τη σημασιολογία της OWL [SPA13]. Η αποτίμηση ερωτημάτων σε αυτές τις περιπτώσεις είναι περισσότερο σύνθετη από ότι στην απλή αντιστοίχιση υπογράφων καθώς σχετίζεται με την ανάκτηση απαντήσεων που προκύπτουν έμεσα από τον γράφο δεδομένων. Στην επόμενη ενότητα δίνουμε μία σύντομη περιγραφή της βασικής λογικής κατά την απάντηση των SPARQL ερωτημάτων.

Απάντηση SPARQL ερωτημάτων

Το πρόβλημα της απάντησης ενός SPARQL ερωτήματος με βάση τη σημασιολογία του RDFs σχήματος αντιμετωπίζεται από τα διάφορα συστήματα με την επέκταση του γράφου δεδομένων, επί του οποίου διατυπώνεται το ερώτημα, με όλη την έμεση πληροφορία που μπορεί να εξαχθεί. Η τεχνική αυτή είναι γνωστή ως υλοποίηση του γράφου. Στη συνέχεια, το SPARQL ερώτημα μπορεί να αποτιμηθεί με τον προφανή τρόπο, προβάλοντας δηλαδή τις `SELECT` μεταβλητές στις τιμές που έχουν αντιστοιχιστεί οι `WHERE` μεταβλητές. Η απάντηση ερωτήματος με βάση τη σημασιολογία της OWL πραγματοποιείται με την αντιστοίχηση του RDF γράφου δεδομένων πάνω στον οποίο διατυπώνεται το ερώτημα σε μία OWL οντολογία. Ο μετασχηματισμός ενός RDF γράφου σε OWL οντολογία ορίζεται για ορθά ορισμένους RDF γράφους που ανταποκρίνονται σε οντολογίες σε OWL 2. Για παράδειγμα, θεωρούμε τα παρακάτω RDF δεδομένα:

```
:Orwell :isAuthor :1984
:1984 rdf:type :Book
```

Ο μετασχηματισμός τους σε OWL περιγράφεται από τους παρακάτω ισχυρισμούς:

```
ObjectPropertyAssertion(:isAuthor :Orwell :1984)
ClassAssertion(:Book :1984)
ObjectPropertyRange(:isAuthor :Book)
```

Η SPARQL διαθέτει δύο τύπους μεταβλητών αντικειμένων: τις διακεκριμένες μεταβλητές και τα αντικείμενα χωρίς όνομα, γνωστές και ως κενοί κόμβοι. Οι κενοί κόμβοι του ερωτήματος αντιμετωπίζονται σαν διακεκριμένες μεταβλητές με τη διαφορά ότι δεν μπορούν να ακολουθούν το `SELECT` και άρα δεν εμφανίζονται στην απάντηση του ερωτήματος. Αυτή είναι και η βασική διαφορά με τα συζευκτικά ερωτήματα όπου τα ανώνυμα αντικείμενα αντιμετωπίζονται ως υπαρξιακές μεταβλητές. Παρόλα αυτά η SPARQL διαχειρίζεται τους κενούς κόμβους σαν σταθερές που μπορεί να σχετίζονται με τις διακεκριμένες μεταβλητές στην οντολογία. Σημειώνουμε ότι οι βοηθητικοί κενοί κόμβοι που μπορεί να περιλαμβάνουν τα RDF δεδομένα δεν εμφανίζονται στις OWL δηλώσεις. Για παράδειγμα, έστω το OWL αξίωμα `ClassAssertion(ObjectSomeValuesFrom(:isAuthor :Book) : John)`, σύμφωνα με το οποίο ο `John` είναι συγγραφέας κάποιου βιβλίου το οποίο όμως δεν γνωρίζουμε. Η αντίστοιχη RDF πληροφορία διαθέτει κενό κόμβο, `_:b` που συνδέει τις τριάδες μεταξύ τους:

```
:John rdf:type _:b
_:b rdf:type owl:Restriction
_:b owl:onProperty :isAuthor
_:b owl:someValuesFrom :Book
```

Στην περίπτωση απάντησης των ερωτημάτων με βάση την σημασιολογία της SPARQL ο κενός κόμβος αντιμετωπίζεται ως σταθερά και έτσι για παράδειγμα το ερώτημα `SELECT ?x WHERE ?x :r ?y` επί της τριάδας `:a :r _:b` επιστρέφει σαν απάντηση το `a`. Συγκεκριμένα, η υλοποίηση του γράφου περιλαμβάνει τον κόμβο `_:b`, που είναι σταθερά και έτσι, γίνεται η αντιστοίχηση των μεταβλητών του ερωτήματος `?x ↦ a, ?y ↦ _:b`, από όπου προκύπτει σαν απάντηση το `a`. Εαν τώρα θεωρήσουμε την αποτίμηση του ίδιου ερωτήματος με βάση τη σημασιολογία της OWL τότε ο κόμβος `_:b` μεταφράζεται σαν υπαρξιακή μεταβλητή, η αντίστοιχη σύνταξη σε περιγραφικές λογικές είναι $\exists r.T(a)$. Σαν πρώτο βήμα, ο γράφος μεταφράζεται σε μια OWL οντολογία. Σε αυτή την περίπτωση η τριάδα `:a :r _:b` μεταφράζεται στο αξίωμα $r(a, _:b)$ όπου η μεταβλητή `_:b` συνοδεύει υπαρξιακό τελεστή. Επομένως, η απάντηση στο ερώτημα θα είναι το κενό σύνολο καθώς η μεταβλητή `?y` δεν αντιστοιχίζεται σε κάποια συγκεκριμένη τιμή (κόμβο). Όπως γίνεται κατανοητό εαν επιθυμούμε να διατυπώσουμε SPARQL ερώτημα με τη σημασιολογία της OWL τότε η τεχνική της υλοποίησης δεν επιστρέφει ένα πλήρες σύνολο απαντήσεων. Σημειώνουμε ότι έχουν αναπτυχθεί αρκετά εργαλεία που υποστηρίζουν SPARQL ερωτήματα με βάση την RDFs σημασιολογία (για παράδειγμα Jena, Stardog) καθώς και ορισμένα που υποστηρίζουν την OWL σημασιολογία τα οποία θα περιγράψουμε σε επόμενη ενότητα.

3 Σημασιολογική ολοκλήρωση δεδομένων

Στις προηγούμενες ενότητες περιγράφηκαν οι διάφορες τεχνολογίες και πρότυπα που επιτρέπουν την σημασιολογική μοντελοποίηση γνώσης και δεδομένων, και συγκεκριμένα οι δυνατότητες αναπαράστασης ενός πεδίου ενδιαφέροντος ως μιας ή περισσότερων οντολογιών και των δεδομένων ως ενός συνόλου δεδομένων RDF με βάση τα λεξιλόγια των εμπλεκόμενων οντολογιών, το οποίο αποθηκεύεται σε μια αποθήκη RDF. Συζητήθηκαν επίσης ορισμένα ζητήματα συλλογιστικής που αφορούν την εξαγωγή έμμεσης γνώσης από τα αξιώματα της οντολογίας, και την ανάκτηση δεδομένων από μια αποθήκη RDF μέσω της γλώσσας SPARQL.

Οπως ήδη έχουμε αναφέρει στο παραπάνω σχήμα μοντελοποίησης υπάρχει μια διάκριση μεταξύ γνώσης (οντολογιών) και δεδομένων. Παρότι η διάκριση αυτή μπορεί να μην τόσο σαφής, όταν π.χ. μια οντολογία εκπεφρασμένη σε OWL περιέχει τόσο αξιώματα ορολογίας όσο και αξιώματα ισχυρισμών, στις πρακτικές εφαρμογές ο διαχωρισμός της γνώσης από τα δεδομένα είναι σαφής: Τα δεδομένα βρίσκονται αποθηκευμένα σε κάποια αποθήκη δεδομένων, ενώ η οντολογική γνώση αποτελεί ένα μέσο σημασιολογικής πρόσβασης στα υπάρχοντα δεδομένα. Η σημασιολογική πρόσβαση αποτελεί στην ουσία μια νέα όψη των δεδομένων, χωρίς να παρεμβαίνει στα ίδια τα δεδομένα, τα οποία μπορεί μάλιστα να προϋπάρχουν και να έχουν δομηθεί κατά διαφορετικό τρόπο από αυτόν που θα επέβαλε η εξ αρχής σημασιολογική μοντελοποίησή τους. Μια χαρακτηριστική τέτοια περίπτωση είναι δεδομένα που βρίσκονται αποθηκευμένα σε προϋπάρχουσες σχεσιακές βάσεις δεδομένων.

Ετσι, κατά την σημασιολογική πρόσβαση σε προϋπάρχοντα δεδομένα, συνήθως υπάρχουν μια ή περισσότερες οντολογίες που περιγράφουν το πεδίο ενδιαφέροντος στο απαιτούμενο επίπεδο αφαίρεσης, και μια αποθήκη δεδομένων, σχεδιασμένη πιθανότατα με ανεξάρτητο τρόπο και σε διαφορετικό χρόνο, με το ζητούμενο να είναι να υπάρξει μια διασύνδεση των δύο, ώστε να μπορούν να ανακτώνται από την αποθήκη δεδομένων τα δεδομένα που απαντούν σε ερωτήματα που διατυπώνονται όχι στο σχήμα των δεδομένων, αλλά στο σχήμα της οντολογίας. Μεταξύ των προβλήματα που ανακύπτουν κατά την υλοποίηση της διασύνδεσης αυτής είναι τα εξής [PLC⁺08]:

- Μια οντολογία μπορεί να έχει ιδιαίτερα υψηλή εκφραστικότητα, με αποτέλεσμα η συλλογιστική που απαιτείται να πραγματοποιηθεί προτού πραγματοποιηθεί η πρόσβαση στα δεδομένα, να έχει ιδιαίτερα υψηλή υπολογιστική πολυπλοκότητα. Και αυτό διότι η υποκείμενη αποθήκη δεδομένων είναι ανεξάρτητη και δεν γνωρίζει τίποτα για την συλλογιστική, επομένως τα τελικά ερωτήματα που θα υποβληθούν στην αποθήκη δεδομένων, με βάση το σχήμα των δεδομένων, πρέπει να ενσωματώνουν πλήρως τα αποτελέσματα της συλλογιστικής που θα πρέπει να έχει προηγηθεί.

- Η υποκείμενη αποθήκη δεδομένων μπορεί να περιέχει τεράστιο όγκο δεδομένων, οπότε η πρόσβαση σε αυτά πρέπει να μπορεί να γίνεται με αποδοτικό τρόπο.

Οσον αφορά στο δεύτερο ζήτημα, οι αποθήκες δεδομένων που επιτρέπουν αποδοτική πρόσβαση στα δεδομένα που περιέχουν, είναι συνήθως βασισμένες στο σχεσιακό πρότυπο. Επιπλέον το σχεσιακό πρότυπο είναι ευρέως διαδεδομένο και οι περισσότερες υπάρχουσες αποθήκες δεδομένων, βασίζονται σε αυτό. Επομένως στην πράξη, το ζήτημα της σημασιολογικής ολοκλήρωσης δεδομένων αφορά κυρίως την ολοκλήρωση σχεσιακών δεδομένων με κάποια οντολογία. Η ολοκλήρωση αυτή μπορεί να επιτευχθεί με δύο βασικού τρόπους, είτε μέσω της μετατροπής της υπάρχουσας βάσης δεδομένων σε κάποια βάση τριάδων RDF που να συμμορφώνεται με το λεξιλόγιο της οντολογίας, είτε μέσω του ορισμού κάποιων απεικονίσεων μεταξύ του λεξιλογίου της οντολογίας και του σχεσιακού σχήματος των δεδομένων, ώστε να δεδομένα να μπορούν να αντιμετωπίζονται (με τη διαμεσολάβηση των απεικονίσεων) ως στιγμιότυπα των εννοιών και των σχέσεων της οντολογίας.

3.1 Διασύνδεση σχεσιακών βάσεων με οντολογίες

Για το πρόβλημα της σημασιολογικής πρόσβασης σε μιας σχεσιακής βάσης δεδομένων, έχουν προταθεί δύο πρότυπα από την W3C.

Το πρώτο, το πρότυπο της άμεσης απεικόνισης, δεν βασίζεται στην διατήρηση των δεδομένων στη σχεσιακή βάση και στην παροχή σημασιολογικής πρόσβασης σε αυτή μέσω κάποιας απεικόνισης μεταξύ του λεξιλογίου της οντολογίας και του σχήματος της βάσης· αντί αυτού υιοθετεί την πλέον δραστική προσέγγιση της ρητής μετατροπής ολόκληρης της βάσης σε ένα σύνολο τριάδων RDF, τα οποία μπορούν να αποτελέσουν το στρώμα δεδομένων ενός συστήματος σχεδιασμένου αποκλειστικά με τα πρότυπα του σημασιολογικού ιστού. Ωστόσο το πρότυπο αυτό είναι αρκετά δύσκαμπτο καθώς δεν παρέχει καμία ευελιξία ούτε ως προς την ονοματολογία: τα δεδομένα RDF παράγονται με αυτόματο τρόπο από την βάση δεδομένων και το λεξιλόγιο τους καθορίζεται εξ ολοκλήρου από το σχήμα της προϋπάρχουσας βάσης.

Στον αντίποδα βρίσκεται το δεύτερο πρότυπο, η γλώσσα ορισμού απεικονίσεων R2RML, η οποία επιτρέπει την ορισμό ιδιαίτερων απεικονίσεων μεταξύ του λεξιλογίου της οντολογίας και του σχήματος της σχεσιακής βάσης, προσφέροντας πλήρη ευελιξία όσον αφορά στον σχεδιασμό τους και στις επιμέρους παραμέτρους τους.

Στη συνέχεια θα περιγράψουμε συνοπτικά τα δύο αυτά πρότυπα ώστε να καταδειχθούν οι δυνατότητες και οι περιορισμοί τους.

3.1.1 Άμεση απεικόνιση

Το πρότυπο της άμεσης απεικόνισης [Dir12] της W3C ορίζει έναν τρόπο αναπάραστασης των περιεχομένων μιας σχεσιακής βάσης δεδομένων ως γράφου RDF. Η άμεση απεικόνιση δέχεται στην είσοδό της το σχήμα και τα περιεχόμενα μιας σχεσιακής βάσης δεδομένων και παράγει έναν γράφο RDF, ο οποίος καλείται άμεσος γράφος.

Το πρότυπο της άμεσης απεικόνισης ορίζει αναλυτικά τους διάφορους κανόνες με τους οποίους μια βάση δεδομένων μετατρέπεται σε έναν γράφο RDF. Οι κανόνες αφορούν κατ' αρχήν την μετατροπή των ονομάτων που συμμετέχουν στο σχήμα της βάσης (όπως ονόματα πινάκων και ονόματα χαρακτηριστικών) σε έγκυρα IRI, ώστε να μπορούν να χρησιμοποιηθούν ως υποκείμενα, αντικείμενα και κατηγορήματα του άμεσου γράφου, και κατά δεύτερο την μετατροπή κάθε γραμμής κάθε πίνακα της σχεσιακής βάσης σε ένα σύνολο τριάδων RDF, έτσι ώστε να διατηρείται το σύνολο των ιδιοτήτων του αρχικού σχήματος, όπως για παράδειγμα οι περιορισμοί ξένου κλειδιού.

Ακολουθεί ένα ενδεικτικό παράδειγμα του τρόπου μετατροπής. Έστω ότι το σχήμα της βάσης δεδομένων έχει οριστεί με το SQL έγγραφο του Σχήματος 1, και έστω ότι τα περιεχόμενα του πίνακα `People` είναι οι πλειάδες (7, Bob, 18, accounting, Cambridge) και (8, Sue, NULL, NULL, NULL), ότι τα περιεχόμενα του πίνακα `Addresses` είναι η πλειάδα (18, Cambridge, MA) και ότι τα περιεχόμενα του πίνακα `Department` είναι η πλειάδα (23, accounting, Cambridge, 18).

Για την κατασκευή του αντιστοιχούντος άμεσου γράφου πρέπει κατ' αρχήν να οριστεί κάποιο IRI βάσης με βάση το οποίο θα κατασκευαστούν όλα τα υπόλοιπα IRI. Έτσι, θεωρώντας ως IRI βάσης το <http://foo.example/DB/>, η σχεσιακή βάση δεδομένων του Σχήματος 1 μετατρέπεται στον άμεσο γράφο του Σχήματος 2.

Όπως είναι φανερό, στον άμεσο γράφο, κάθε γραμμή κάθε πίνακα, π.χ. η πλειάδα (7, Bob, 18, accounting, Cambridge) έχει μετατραπεί σε ένα σύνολο τριάδων με κοινό υποκείμενο. Το υποκείμενο είναι ένα IRI που κατασκευάζεται από την συναρμογή του IRI βάσης, του ονόματος του πίνακα (`People`), του ονόματος της στήλης του πρωτεύοντος κλειδιού (`ID`) και της τιμής του πρωτεύοντος κλειδιού (7). Το κατηγόρημα για κάθε στήλη είναι ένα IRI που κατασκευάζεται με την συναρμογή του IRI βάσης, του ονόματος του πίνακα και του ονόματος της στήλης. Τα αντικείμενα είναι λεκτικά RDF που σχηματίζονται βάσει της τιμής της κάθε στήλης. Κάθε ξένο κλειδί παράγει μια τριάδα με ένα κατηγόρημα που συντίθεται από τα ονόματα των στηλών του ξένου κλειδιού, τον πίνακα αναφοράς και τα ονόματα των αναφερόμενων στηλών. Το αντικείμενο αυτών των τριάδων είναι το αναγνωριστικό γραμμής (`<Addresses/ID=18>`) της αναφερόμενης τριάδας. Αυτές οι αναφορές πρέπει να συμπίπτουν με το υποκείμενο που χρησιμοποιήθηκε για την κατασκευή των τριάδων από την αναφερόμενη γραμμή. Η άμεση απεικόνιση δεν παράγει τριάδες για τιμές NULL, και αυτό γιατί δεν υπάρχει αντιστοιχία με-

```

CREATE TABLE "Addresses" (
    "ID" INT,
    "city" CHAR(10),
    "state" CHAR(2),
    PRIMARY KEY("ID")
)

CREATE TABLE "Department" (
    "ID" INT,
    "name" CHAR(10),
    "city" CHAR(10),
    "manager" INT,
    PRIMARY KEY("ID"),
    UNIQUE ("name", "city")
)

CREATE TABLE "People" (
    "ID" INT,
    "fname" CHAR(10),
    "addr" INT,
    "deptName" CHAR(10),
    "deptCity" CHAR(10),
    PRIMARY KEY("ID"),
    FOREIGN KEY("addr") REFERENCES "Addresses"("ID"),
    FOREIGN KEY("deptName", "deptCity") REFERENCES "Department"("name", "city")
)

ALTER TABLE "Department" ADD FOREIGN KEY("manager") REFERENCES "People"("ID")

```

Σχήμα 1: Παράδειγμα σχήματος σχεσιακής βάσης δεδομένων.

ταξύ της συμπεριφοράς ενός RDF και της σημασιολογίας των τιμών NULL μιας σχεσιακής βάσης δεδομένων σύμφωνα με το πρότυπο της SQL.

3.1.2 R2RML

Το πρότυπο R2RML [R2R12] της W3C ορίζει μια γλώσσα για την περιγραφή απεικονίσεων μεταξύ μιας σχεσιακής βάσης δεδομένων και συνόλων δεδομένων τύπου RDF. Μέσω των απεικονίσεων αυτών τα υπάρχοντα σχεσιακά δεδομένων μπορούν να ιδωθούν διαμέσου ενός μοντέλου RDF, κατά τον τρόπο που επιθυμεί κάθε φορά ο χρήστης. Στην άμεση απεικόνιση που περιγράφηκε στην προηγούμενη ενότητα, ο κατασκευαζόμενος γράφος RDF απεικονίζει επακριβώς την δομή της σχεσιακής βάσης δεδομένων, ενώ το λεξιλόγιο που χρησιμοποιείται στον γράφο ορίζεται με μοναδικό τρόπο βάσει των ονομάτων των σχεσιακού σχήματος, χωρίς να υπάρχει δυνατότητα επεμβάσεων και αλλαγών εκ μέρους του χρήστη. Σε

```

@base <http://foo.example/DB/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<People/ID=7> rdf:type <People> .
<People/ID=7> <People#ID> 7 .
<People/ID=7> <People#fname> "Bob" .
<People/ID=7> <People#addr> 18 .
<People/ID=7> <People#ref-addr> <Addresses/ID=18> .
<People/ID=7> <People#deptName> "accounting" .
<People/ID=7> <People#deptCity> "Cambridge" .
<People/ID=7> <People#ref-deptName;deptCity> <Department/ID=23> .
<People/ID=8> rdf:type <People> .
<People/ID=8> <People#ID> 8 .
<People/ID=8> <People#fname> "Sue" .

<Addresses/ID=18> rdf:type <Addresses> .
<Addresses/ID=18> <Addresses#ID> 18 .
<Addresses/ID=18> <Addresses#city> "Cambridge" .
<Addresses/ID=18> <Addresses#state> "MA" .

<Department/ID=23> rdf:type <Department> .
<Department/ID=23> <Department#ID> 23 .
<Department/ID=23> <Department#name> "accounting" .
<Department/ID=23> <Department#city> "Cambridge" .
<Department/ID=23> <Department#manager> 8 .
<Department/ID=23> <Department#ref-manager> <People#ID=8> .

```

Σχήμα 2: Άμεσος γραφος

αντίθεση με αυτό, χρησιμοποιώντας την R2RML ο χρήστης μπορεί να ορίσει τις δικές του εξατομικευμένες απεικονίσεις, ανάλογα με τις ανάγκες του.

Κάθε απεικόνιση R2ML σχεδιάζεται με βάση το σχήμα μιας συγκεκριμένης βάσης δεδομένων και χρησιμοποιεί ένα συγκεκριμένο λεξιλόγιο προορισμού. Έτσι, η είσοδος σε μια τέτοια απεικόνιση πρέπει να είναι μια βάση δεδομένων που υπακούει στο συγκεκριμένο σχήμα. Η έξοδος είναι ένα σύνολο δεδομένων RDF, που χρησιμοποιεί το λεξιλόγιο προορισμού και στο οποίο μπορούν να υποβληθούν ερωτήματα με χρήση της γλώσσας SPARQL. Ο ορισμός, ωστόσο, των απεικονίσεων γίνεται σε αφαιρετικό επίπεδο και μπορούν να υλοποιηθούν συστήματα που είτε εκτελούν ρητά τη μετατροπή των δεδομένων της σχεσιακής βάσης, παράγοντας ένα πραγματικό σύνολο δεδομένων RDF στο οποίο να μπορούν να υποβληθούν τα ερωτήματα SPARQL, είτε προσφέρουν εικονική πρόσβαση στο σύνολο δεδομένων RDF, μέσω μιας διεπαφής που να μεταφράζει τα ερωτήματα SPARQL σε κατάλληλα ερωτήματα προς την υποκείμενη βάση λαμβάνοντας υπόψη τις απεικονίσεις.

Μια απεικόνιση R2ML εκφράζεται η ίδια ως ένας γράφος RDF, δηλωμένος σε ένα έγγραφο συντεταγμένο κατά την σύνταξη Turtle. Μια απεικόνιση R2ML αναφέρεται σε λογικούς πίνακες οι οποίοι προέρχονται από την υποκείμενη βάση δεδομένων. Ένας λογικός πίνακας μπορεί να είναι είτε ένας βασικός πίνακας της βάσης δεδομένων, είτε μια όψη, είτε ένα έγκυρο ερώτημα SQL. Κάθε λογικός πίνακας απεικονίζεται σε ένα σύνολο τριάδων RDF μέσω μιας απεικόνισης τριάδων. Κάθε τέτοια απεικόνιση είναι ένας κανόνας που απεικονίζει κάθε γραμμή του λογικού πίνακα σε ένα σύνολο τριάδων RDF. Ο κανόνας αυτός έχει δύο βασικά μέρη: α) μια απεικόνιση υποκειμένου που παράγει το υποκείμενο όλων των τριάδων RDF που θα παραχθούν από την γραμμή ενός λογικού πίνακα και συνήθως είναι IRI που κατασκευάζονται από τις στήλες πρωτεύοντος κλειδιού του πίνακα, και β) πολλαπλές απεικονίσεις κατηγορήματος-αντικειμένου οι οποίες με τη σειρά τους αποτελούνται από απεικονίσεις κατηγορήματος και από απεικονίσεις αντικειμένου. Οι τριάδες παράγονται μέσω του συνδυασμού μιας απεικόνισης υποκειμένου με μια απεικόνιση κατηγορήματος και μια απεικόνιση αντικειμένου και της ταυτόχρονης εφαρμογής τους σε μια γραμμή ενός λογικού πίνακα.

Στη συνέχεια θα περιγράψουμε ένα σύντομο παράδειγμα όπου καταδεικνύονται ο τρόπος ορισμού των απεικονίσεων R2RML. Θεωρούμε κατ' αρχάς τη σχεσιακή βάση του Σχήματος 3.

```

CREATE TABLE "Employee" (
    "empno" INT,
    "ename" VARCHAR(100),
    "job" VARCHAR(20),
    "deptno" INT,
    PRIMARY KEY("ID"),
    FOREIGN KEY("deptno") REFERENCES "Department"("deptno"),
)
CREATE TABLE "Department" (
    "deptno" INT,
    "dname" VARCHAR(30),
    "city" VARCHAR(100),
    PRIMARY KEY("deptno")
)

```

Σχήμα 3: Παράδειγμα σχήματος σχεσιακής βάσης δεδομένων.

Έστω επιπλέον ότι ο πίνακας Employee περιέχει την πλειάδα (7369, Smith, Clerk, 10) και ότι ο πίνακας Department περιέχει την πλειάδα (10, Appserver, New York). Ζητούμενο είναι να οριστεί μια R2RML απεικόνιση που να απεικονίζει τα παραπάνω δεδομένα στο σύνολο τριάδων του Σχήματος 4.

Ο ορισμός της απεικόνισης αυτής παρουσιάζεται στο Σχήμα 5. Όπως προα-

```

<http://data.example.com/employee/7369> rdf:type ex:Employee.
<http://data.example.com/employee/7369> ex:name "Smith".
<http://data.example.com/employee/7369> ex:department
<http://data.example.com/department/10>.

<http://data.example.com/department/10> rdf:type ex:Department.
<http://data.example.com/department/10> ex:name "AppServer".
<http://data.example.com/department/10> ex:location "New York".
<http://data.example.com/department/10> ex:staff 1.

```

Σχήμα 4: RDF τριάδες της απεικόνισης

ναφέραμε ο ορισμός γίνεται με βάση το πρότυπο RDF σε σύνταξη Turtle. Ορίζονται δύο επιμέρους απεικονίσεις, η #TriplesMap1 και η #TriplesMap2. Ξεκινώντας από την #TriplesMap1, παρατηρούμε ότι ορίζεται κατ' αρχάς ο λογικός πίνακας επί του οποίου θα εφαρμοστεί (μέσω του κατηγορήματος rr:logicalTable) και στη συνέχεια οι απεικονίσεις κατηγορήματος-αντικειμένου μέσω των κατηγορημάτων rr:subjectMap και rr:predicateObjectMap αντίστοιχα. Στην απεικόνιση υποκειμένου ορίζεται ο τρόπος κατασκευής του IRI του υποκειμένου από τα δεδομένα του πίνακα (μέσω του κατηγορήματος rr:template) και η κλάση στην οποία θα δηλωθεί ως στιγμιότυπο το υποκείμενο (μέσω του κατηγορήματος rr:class). Με ανάλογο τρόπο στην απεικόνιση κατηγορήματος-αντικειμένου για κάθε είδος τριάδων RDF που θα παραχθούν ορίζεται το κατηγόρημά τους (μέσω του κατηγορήματος rr:predicate) και το αντικείμενο (μέσω του κατηγορήματος rr:objectMap). Η #TriplesMap2 διαφέρει από την την #TriplesMap1 πρώτον κατά το ότι ορίζεται επί ενός λογικού πίνακα που ορίζεται ως ένα ερώτημα SQL προς την σχεσιακή βάση, και δεύτερον κατά το ότι περιλαμβάνει πολλαπλές απεικονίσεις κατηγορήματος-αντικειμένου για την ίδια απεικόνιση υποκειμένου.

```

PREFIX rr: <http://www.w3.org/ns/r2rml#>.
PREFIX ex: <http://example.com/ns#>.

<#TriplesMap1>
    rr:logicalTable [ rr:tableName "Employee" ];
    rr:subjectMap [
        rr:template "http://data.example.com/employee/{empno}";
        rr:class ex:Employee;
    ];
    rr:predicateObjectMap [
        rr:predicate ex:name;
        rr:objectMap [ rr:column "ename" ];
    ].

<#DeptTableView> rr:sqlQuery """
    SELECT deptno, dname, city,
        (SELECT COUNT(*) FROM Employee WHERE
        employee.deptno=Department.deptno) AS staff
    FROM Department;
""".

<#TriplesMap2>
    rr:logicalTable <#DeptTableView>;
    rr:subjectMap [
        rr:template "http://data.example.com/department/{deptno}";
        rr:class ex:Department;
    ];
    rr:predicateObjectMap [
        rr:predicate ex:name;
        rr:objectMap [ rr:column "dname" ];
    ];
    rr:predicateObjectMap [
        rr:predicate ex:location;
        rr:objectMap [ rr:column "city" ];
    ];
    rr:predicateObjectMap [
        rr:predicate ex:staff;
        rr:objectMap [ rr:column "staff" ];
    ].

```

Σχήμα 5: Απεικόνιση R2RML

4 Συστήματα και εργαλεία

Η απάντηση ερωτημάτων σε δεδομένα που διαθέτουν σημασιολογικό περιεχόμενο αποτελεί ένα από τα βασικά προβλήματα που καλούνται να αντιμετωπίσουν οι σύγχρονες εφαρμογές. Συγκεκριμένα, σε περιπτώσεις όπου τα δεδομένα περιγράφονται από το σχήμα κάποιας οντολογίας, τα συστήματα προκειμένου να απαντήσουν στα ερωτήματα του χρήστη δεν παρέχουν υπηρεσίες απλής ανάκτησης δεδομένων όπως συμβαίνει με τις απλές βάσεις δεδομένων, αλλά χρησιμοποιούν τα αξιώματα της οντολογίας προκειμένου τα εξάγουν όλα τα συμπεράσματα που είναι δυνατό να αξιοποιηθούν κατά την ανάκτηση των απαντήσεων. Σε τέτοιου είδους σενάρια οι απαντήσεις στα ερωτήματα χρηστών ανακλούν τα δεδομένα καθώς και όλη τη γνώση του πεδίου εφαρμογής που κωδικοποιείται στην οντολογία. Κατά κύριο λόγο οι σύγχρονες εφαρμογές υιοθετούν το W3C πρότυπο σύμφωνα με το οποίο οι οντολογίες εκφράζονται σε OWL 2 για να περιγράψουν δεδομένα που προέρχονται από διάφορες πηγές και αναπαρίστανται με χρήση του σχήματος RDF, ενώ τα ερωτήματα διατυπώνονται σε SPARQL.

Διάφορες προσεγγίσεις έχουν μελετηθεί για την αντιμετώπιση του προβλήματος της απάντησης των ερωτημάτων με σημασιολογικό περιεχόμενο, για παράδειγμα η τεχνική της επαναγραφής που μελετήσαμε σε προηγούμενη ενότητα. Η έρευνα προς αυτή την κατεύθυνση έχει οδηγήσει στην ανάπτυξη πολλών πρακτικών συστημάτων. Τα διάφορα συστήματα ανάλογα με το πεδίο εφαρμογής υποστηρίζουν οντολογίες διαφορετικής εκφραστικότητας. Τα συστήματα που υποστηρίζουν την πλήρη εκφραστικότητα της OWL όπου το πρόβλημα απάντησης ερωτημάτων παρουσιάζει υψηλή πολυπλοκότητα δεν αποκρίνονται ικανοποιητικά, ιδιαίτερα σε περιπτώσεις μεγάλου όγκου δεδομένων. Συχνά, η πληρότητα στις απαντήσεις θυσιάζεται ώστε οι χρόνοι απόκρισης να είναι ταχύτεροι και τα συστήματα πιο αποτελεσματικά. Επιπλέον, ορισμένα συστήματα δίνοντας έμφαση στην αποτελεσματικότητα και τους ταχύτερους χρόνους απόκρισης υποστηρίζουν τη χαμηλότερης εκφραστικότητας γλώσσα OWL-QL. Παραδείγματα πρακτικών συστημάτων είναι οι συλλογιστές Pellet, HermiT, που επιστρέφουν όλες τις απαντήσεις, Apache-Jena, OWLIM, που επιστρέφουν υποσύνολα των απαντήσεων. Στο υπόλοιπο της ενότητας θα περιγράψουμε τις εφαρμογές και τα συστήματα που παρέχουν υπηρεσίες απάντησης ερωτημάτων για την ανάκτηση δεδομένων με σημασιολογικό περιεχόμενο.

4.1 Εργαλεία επεξεργασίας οντολογιών

4.1.1 Protégé

Το Protégé¹ είναι ένα ολοκληρωμένο πακέτο επεξεργασίας οντολογιών με πλήρη υποστήριξη για την OWL 2 από το πανεπιστήμιο του Stanford. Πρόκειται για την περισσότερο διαδεδομένη εφαρμογή στο είδος της. Η εφαρμογή έχει ένα εύχρηστο και πλούσιο γραφικό περιβάλλον που επιδέχεται διαμόρφωση από το χρήστη, υποστηρίζει διάφορους τρόπους παρουσίασης οντολογιών και λειτουργίες για τη διευκόλυνση της ταυτόχρονης ανάπτυξης και επεξεργασίας πολλών οντολογιών. Ο χρήστης μπορεί να περιηγηθεί ταυτόχρονα σε πολλαπλές οντολογίες και να δει την εικόνα του γράφου τους. Προσφέρει υποστήριξη διεπαφών με τους συλλογιστές Pellet, Fact++ και HermiT με απευθείας παρουσίαση των αποτελεσμάτων της χρήσης τους στο γραφικό περιβάλλον. Επίσης η αρχιτεκτονική του προσφέρει υποστήριξη για ποικίλα plug-ins που αφορούν σε διαφορετικά μενού μέχρι και επιπλέον εργαλεία συλλογιστικής.

Το εργαλείο διατίθεται δωρεάν για λήψη από το διαδίκτυο καθώς και ως διαδικτυακή εφαρμογή. Πρόκειται για μία εφαρμογή πλήρως τεκμηριωμένη, με εγχειρίδια χρήσης, tutorials και περιοδικά.

4.2 Εργαλεία συλλογιστικής

4.2.1 HermiT

Ο HermiT² [SMH08] [GHM⁺14] είναι ένας συλλογιστής ελεύθερου λογισμικού για την OWL 2 που έχει υλοποιηθεί σε Java. Αναπτύχθηκε από την ομάδα Πληροφοριακών Συστημάτων του Τμήματος Επιστήμης των Υπολογιστών του Πανεπιστημίου της Οξφόρδης και διατίθεται δωρεάν υπό την άδεια LGPL. Είναι πλήρως συμβατός με την άμεση σημασιολογία της OWL 2 (αντίστοιχα της περιγραφικής λογικής *SROIQ*) σύμφωνα με τα πρότυπα της W3C και υποστηρίζει όλη την εκφραστικότητα της γλώσσας συμπεριλαμβανομένων όλων των τύπων δεδομένων.

Ο αλγόριθμος που υλοποιείται στον HermiT βασίζεται σε τεχνικές συλλογιστικής hypertableau. Οι εν λόγω τεχνικές που αποφεύγουν τις μη ντετερμινιστικές συμπεριφορές των λογισμών tableau. Επιπλέον, ο αλγόριθμος υποστηρίζει ένα ευρύ φάσμα βελτιστοποιήσεων που αφορούν στην απόδοση της διαδικασίας συλλογιστικής σε σενάρια χρήσης πραγματικών οντολογιών. Εκτός από τις υπηρεσίες συλλογιστικής για τον ελέγχου υπαγωγής, προσφέρει υπηρεσίες κατηγοριοποίησης κλάσεων και ιδιοτήτων καθώς και μια σειρά από χαρακτηριστικά που υπερβαίνουν το πρότυπο της OWL 2, όπως οι ασφαλείς DL SWRL κανόνες.

¹<http://protege.stanford.edu>

²<http://hermit-reasoner.com/>

Οι χρήστες μπορούν να αλληλεπιδρούν με τον συλλογιστή μέσω τριών διαφορετικών διεπαφών: μιας εγγενούς Java διεπαφής, του OWL API και μιας διεπαφής γραμμής εντολών. Η εγγενής διεπαφή ανάγει τις τυπικές διεργασίες συλλογιστικής σε έλεγχο συνέπειας της οντολογίας. Επίσης υλοποιεί την διεπαφή OWLReasoner του OWL API, που επιτρέπει στον HermiT να χρησιμοποιηθεί από οποιαδήποτε εφαρμογή βασίζεται στο OWL API. Η διεπαφή γραμμής εντολών επιτρέπει στους χρήστες να καλέσουν κάποιες βασικές εργασίες συλλογιστικής από τη γραμμή εντολών. Έτσι για παράδειγμα, είναι δυνατή η χρήση του HermiT χωρίς κάποια προηγούμενη ρύθμιση (π.χ. χωρίς να χρειάζεται να γραφεί κάποιο πρόγραμμα σε Java που να καλεί μια διεπαφή του συλλογιστή). Επίσης ο HermiT ύπαρχει σαν εγκατεστημένη διεπαφή στο Protégé 4.3 ενώ μπορεί να χρησιμοποιηθεί και με το Protégé 4.1 beta και τις μετέπειτα εκδόσεις που είναι συμβατές με το OWL API 3.0.

4.2.2 Pellet

Ο Pellet³ [SPG⁺07] είναι ένας δημοφιλής και ευρέως διαδεδομένος συλλογιστής που υποστηρίζει την OWL 2. Πρόκειται για ένα συλλογιστή ελεύθερου λογισμικού που έχει υλοποιηθεί σε Java. Υλοποιήθηκε από τους Clark & Parsia και διατίθεται δωρεάν υπό διπλή άδεια, την AGPL⁴ για χρήση σε εφαρμογές ελεύθερου λογισμικού και μια ειδική άδεια για εφαρμογές κλειστού λογισμικού. Ο Pellet είναι ο πρώτος συλλογιστής που υποστήριξε την εκφραστικότητα της της περιγραφικής λογικής $\mathcal{SHOIN}(D)$, και έχει επεκταθεί ώστε να υποστηρίζει τις προδιαγραφές της OWL 2, δηλαδή της περιγραφικής λογικής $\mathcal{SROIQ}(D)$.

Ο Pellet είναι βασισμένος στους tableau αλγορίθμους ενώ διαφορετικοί αλγόριθμοι χρησιμοποιούνται ανάλογα με την εκάστοτε περίπτωση χρήσης. Παρέχει όλες τις υπηρεσίες συλλογιστικής που προσφέρονται από τους συλλογιστές για περιγραφικές λογικές, δηλαδή υποστηρίζει έλεγχο συνέπειας, έλεγχο ικανοποιησιμότητας μιας έννοιας, την κατηγοριοποίηση εννοιών και την εύρεση ειδικών κλάσεων που έχουν σα στιγμότυπο κάποιο άτομο. Πέραν όμως από τις συνήθεις υπηρεσίες συλλογιστικής ο Pellet προσφέρει υπηρεσίες όπως την απάντηση συζευκτικών επερωτημάτων με ή χωρίς διακριτές μεταβλητές. Διάφορες τεχνικές και βελτιστοποιήσεις που έχουν αναπτυχθεί στο συλλογιστή προκειμένου να αντιμετωπίζει τις περιπτώσεις επερωτημάτων με κύκλους. Τα επερωτήματα διατυπώνονται σε γλώσσα SPARQL. Επιπρόσθετα, αναφέρουμε ότι ο συλλογιστής παρέχει τη δυνατότητα συλλογιστικής με χρήση τύπων δεδομένων που ορίζονται από το χρήστη (πέρα των βασικών τύπων που ορίζονται από το XML Schema) καθώς και τη δυνατότητα συλλογιστικής με ασφαλείς DL κανόνες κωδικοποιημένους σε

³<http://clarkparsia.com/pellet>

⁴<http://www.gnu.org/licenses/agpl-3.0.html>

SWRL σε συνδυασμό με τα αξιώματα της οντολογίας σε OWL. Διαθέτει ένα σύνολο ευρετικών μεθόδων για την αναγνώριση και επιδιόρθωση οντολογιών και συμβάλει στη διαδικασία της εκσφαλμάτωσης του κώδικα καταδεικνύοντας τα αξιώματα που προκαλούν την ασυνέπεια στην οντολογία και τις συγκεκριμένες κλάσεις στην ιεραρχία που δεν είναι ικανοποιήσιμες. Τέλος, στην κατεύθυνση της αυξητικής συλλογιστικής, παρέχει τη δυνατότητα αυξητικής κατηγοριοποίησης και αυξητικού ελέγχου συνέπειας για τις περιπτώσεις που οι οντολογίες μεταβάλλονται.

Οι υπηρεσίες συλλογιστικής που προσφέρει είναι προσβάσιμες με διάφορους τρόπους. Η λήψη της τελευταίας σταθερής έκδοσης του Pellet (pellet-2.3.1) μαζί με τον πηγαίο κώδικα μπορεί να γίνει από την επίσημη ιστοσελίδα του συλλογιστή. Μαζί με το πακέτο διανομής παρέχεται μια διεπαφή γραμμής εντολών για πρόσβαση στο συλλογιστή. Ο Pellet μπορεί ακόμα να ενσωματωθεί στο περιβάλλον σύνταξης οντολογιών του Protégé 4 σα διεπαφή, μέσω του χαρακτηριστικού αυτόματης ενημέρωσης του Protégé 4, είτε μέσω του διακομιστή DIG και να χρησιμοποιηθεί από διαφορετικούς πελάτες συμπεριλαμβανομένων των προηγούμενων εκδόσεων του Protégé. Επιπλέον παρέχεται πρόσβαση είτε μέσω προγραμματιστικού API που μπορεί να χρησιμοποιηθεί σαν αυτόνομη εφαρμογή είτε μέσω των διεπαφών για συλλογιστές των OWL API και Jena.

4.2.3 FaCT

Ο FaCT++⁵ [TH06] είναι ένας συλλογιστής ελεύθερου λογισμικού που έχει υλοποιηθεί στο πανεπιστήμιο του Manchester σε C++ υπό την άδεια LGPL.

Είναι ένας OWL 2 συλλογιστής που βασίζεται στους αλγορίθμους tableaux και καλύπτει την γλώσσα περιγραφικής λογικής \mathcal{SHOIQ} , με πρόσθετη υποστήριξη για τους τύπους δεδομένων, συμπεριλαμβανομένων των γραμματοσειρών και των ακεραίων. Το σύστημα χρησιμοποιεί μια σειρά βελτιστοποιήσεων για την αύξηση της απόδοσης που περιλαμβάνουν τις περισσότερες κλασικές τεχνικές (όπως η απορρόφηση και η συγχώνευση μοντέλων) και μεταγενέστερες (όπως ευρετικές μεθόδους διάταξης και ταξινομική κατηγοριοποίηση).

Η συλλογιστική διαδικασία που εκτελεί ο FaCT++ είναι η εξής: στο στάδιο της προεπεξεργασίας η βάση γνώσης που φορτώνεται στον συλλογιστή κανονικοποιείται και μετασχηματίζεται σε μια εσωτερική αναπαράσταση, ενώ υφίσταται διάφορες βελτιστοποιήσεις συντακτικών επαναγραφών. Στη συνέχεια ο συλλογιστής εκτελεί την κατηγοριοποίηση υπολογίζοντας τις σχέσης υπαγωγής. Σε αυτό το στάδιο εφαρμόζονται βελτιστοποιήσεις που αφορούν κυρίως στην σειρά με την οποία επιλέγονται οι έννοιες προς επεξεργασία ώστε να μειωθεί ο αριθμός των δοκιμών υπαγωγής που εκτελούνται. Ο ταξινομητής χρησιμοποιεί έναν ελεγ-

⁵<http://owl.cs.man.ac.uk/tools/fact/>

κτή ικανοποιησιμότητας της βάσης δεδομένων προκειμένου να αποφανθεί για την υπαγωγή μεταξύ των εννοιών.

Ο FaCT++ μπορεί να χρησιμοποιηθεί ως αυτόνομος συλλογιστής μέσω της διεπαφής DIG ή ως back-end συλλογιστής για το OWL API. Επιπλέον είναι ένας από τους συλλογιστές που συμπεριλαμβάνονται στο Protégé 4. Ο πηγαίος κώδικας και ήδη μεταγλωττισμένα εκτελέσιμα αρχεία (για Windows, Linux, Mac OS X) είναι διαθέσιμα στο Google code⁶.

4.2.4 Racer

Ο Racer⁷ [HHMW12] είναι ένας συλλογιστής που χρησιμοποιεί μια βελτιωμένη έκδοση του αλγορίθμου tableau για την περιγραφική λογική $\mathcal{SRIQ(D)}$. Διατίθεται ελεύθερα υπό την άδεια χρήσης BSD 3-clause license. Μια έκδοση ελεύθερου λογισμικού του Racer είναι διαθέσιμη στο Github⁸. Ο Racer 2.0 διατίθεται από την επίσημη ιστοσελίδα του ως έκδοση διακομιστή (για Linux, Mac OS X και Windows) ενώ διαθέτει APIs για Common Lisp και Java. Υποστηρίζει την διεπαφή επικοινωνίας OWLlink (η οποία είναι διάδοχος της DIG). Υλοποιεί το OWL API και υπάρχει διαθέσιμη διεπαφή για το Protégé 4. Επίσης μπορεί να εγκατασταθεί εύκολα μέσω του διαχειριστή βιβλιοθηκών Common Lisp, Quicklisp⁹.

Ο Racer μπορεί να επεξεργαστεί οντολογίες σε OWL 2 ενώ δεν υποστηρίζει XML τύπους δεδομένων που ορίζονται από το χρήστη. Μια υλοποίηση της γλώσσας κανόνων SWRL παρέχεται με τον RacerPro 2.0 έτσι ώστε οι κανόνες να μπορούν να καθοριστούν ως τμήμα ενός εγγράφου σε OWL σε σύνταξη SWRL με κάποιους περιορισμούς. Οι οντολογίες μπορούν να διαβαστούν από αρχεία, να ανακτηθούν από το διαδίκτυο ή να φορτωθούν σαν τριάδες RDF από δευτερεύουσα μνήμη (για παράδειγμα, από την αποθήκη τριάδων AllegroGraph).

Ο Racer υποστηρίζει τη γλώσσα επερωτημάτων SPARQL, για ένα σημαντικό υποσύνολο της οποίας προσφέρεται η δυνατότητα συλλογιστικής με βάση την οντολογία. Υποστηρίζονται επίσης πολλές επεκτάσεις της SPARQL (για παράδειγμα, η γλώσσα συζευτικών επερωτημάτων nRQL, new Racer Query Language).

Πέρα από τα τυπικά προβλήματα συλλογιστικής (έλεγχος συνέπειας, κατηγοριοποίηση εννοιών, κατηγοριοποίηση ατόμων, ανάκτηση στιγμιοτύπων και απάντηση ερωτημάτων) προσφέρει επιπλέον δυνατότητες όπως είναι ο εντοπισμός εννοιών που παρουσιάζουν ασυνέπεια. Επιπλέον, μπορεί να προσδιορίσει την ελάχιστη κοινή υπερέννοια, δηλαδή την πιο ειδική έννοια σε ένα σύνολο εννοιών. Επιπλέον, υποστηρίζει τη λειτουργία του αλγεβρικού συμπερασμού με χρήση συγκεκριμένων πεδίων (concrete domains) [BH11].

⁶<http://code.google.com/p/factplusplus/>

⁷<https://www.ifis.uni-luebeck.de/index.php?id=385>

⁸<https://github.com/ha-mo-we/Racer>

⁹<http://quicklisp.org>

4.3 Εργαλεία απάντησης σε συζευκτικά ερωτήματα

4.3.1 KAON

Το KAON2 είναι ένα εργαλείο επεξεργασίας οντολογιών που ενσωματώνει συστήματα πρόσβασης, επεξεργασίας, συλλογιστικής και απάντησης σε συζευκτικά ερωτήματα για οντολογίες στη γλώσσα OWL (καθώς και στις SWRL, F-LOGIC). Έχει υλοποιηθεί αρχικά από το πανεπιστήμιο της Καρλσρούης και στη συνέχεια του Manchester.

Το KAON2 υποστηρίζει τη περιγραφική λογική $\mathcal{SHIQ}(D)$. Ο αλγόριθμος συλλογιστικής που υλοποιεί εφαρμόζει τη μέθοδο της επαναγραφής χρησιμοποιώντας την τεχνική της ανάλυσης. Επιπλέον, υποστηρίζει την απάντηση ερωτημάτων διατυπωμένα σε SPARQL, με κάποιους περιορισμούς. Τα συζευκτικά ερωτήματα που υποστηρίζει δεν περιλαμβάνουν μεταβλητές αλλά σταθερές της βάσης γνώσης.

Η πρόσβαση στις οντολογίες επιτρέπεται ακόμη και αν είναι κατανεμημένες μέσω ενός εξυπηρετητή (με χρήση ενός RMI, Remote Method Invocation). Ακόμη παρέχονται διεπαφές για πρόσβαση από εργαλεία όπως το Protégé, καθώς και για εξαγωγή στιγμιοτύπων από σχεσιακές βάσεις δεδομένων.

Η εφαρμογή του συστήματος διατίθεται δωρεάν για μη κερδοσκοπική ακαδημαϊκή χρήση, ενώ ο κώδικας της εφαρμογής δεν είναι ανοικτός. Σημειώνουμε ότι το σύστημα έχει βρει εφαρμογή σε διάφορα έργα.

4.3.2 CLIPPER

Το CLIPPER είναι ένα εργαλείο απάντησης ερωτημάτων για οντολογίες Horn- \mathcal{SHIQ} που υλοποιεί την τεχνική της επαναγραφής. Η εφαρμογή, που πρόκειται για ενα αρχείο .jar και εκτελείται από την γραμμή εντολών με παραμέτρους μια οντολογία και ένα ερώτημα SPARQL, ενώ απαιτεί την εξωτερική επαγωγική βάση δεδομένων DLV, πρόκειται για μια μηχανή εξαγωγής συμπερασμάτων Datalog.

Το εργαλείο παράχθηκε στο Ινστιτούτο συστημάτων πληροφορικής του τεχνολογικού ιδρύματος της Βιέννης. Πρόκειται για εφαρμογή ανοιχτού κώδικα.

4.3.3 Mastro

Το Mastro είναι ένα σύστημα πρόσβασης δεδομένων με χρήση οντολογιών, που είναι διατυπωμένες στη γλώσσα περιγραφικής λογικής DL-Lite. Πραγματοποιεί σύνδεση οντολογιών με εξωτερικά συστήματα διαχείρισης βάσεων δεδομένων και αντιστοιχίζει τα SPARQL ερωτημάτων επί της οντολογίας ας ερωτήματα SQL στις υποκείμενες βάσεις δεδομένων. Ακόμη παρέχει υπηρεσίες συλλογιστικής όπως υπηρεσίες ταξινόμησης οντολογιών και ελέγχου συνέπειας. Τα συζευκτικά ερωτήματα απαντώνται με μια διαδικασία δύο βημάτων. Αρχικά, ο αλγό-

ριθμος Presto υλοποιεί μία διαδικασία επαναγραφής, και έπειτα η ανάκτηση των απαντήσεων των ερωτημάτων γίνεται με τα συστήματα διαχείρισης δεδομένων.

Το σύστημα εξελίχθηκε από την ομάδα DASI-lab του Πανεπιστημίου Sapienza της Ρώμης πάνω σε μια αρχική υλοποίηση του IBM Center for Advanced Studies. Η εφαρμογή διατίθεται ελεύθερα.

4.3.4 Requiem

Η ομάδα πληροφοριακών συστημάτων του πανεπιστημίου της Οξφόρδης παρέχει την εφαρμογή REQUIEM¹⁰ που υλοποιεί αλγόριθμο επαναγραφής με βάση την ανάλυση. Παρέχει διαφορετικούς τρόπους εκτελέσης με βελτιστοποιήσεις για την αποφυγή περιττών ερωτημάτων και υποστηρίζει την εκφραστικότητα της περιγραφικής λογικής *ΕΛΗΣ*.

Η εφαρμογή χρησιμοποιείται μέσω του προγραμματιστικού περιβάλλοντος Eclipse, χρησιμοποιεί το συλλογιστή Hermit. Σημειώνουμε ότι η εφαρμογή δεν αποτιμά τα ερωτήματα αλλά υπολογίζει την επαναγραφή η οποία είναι δυνατό εν συνεχείᾳ να απαντηθεί από κάποια επαγωγική βάση δεδομένων. Η εφαρμογή είναι ανοιχτού κώδικα.

4.3.5 IQAROS

Το σύστημα IQAROS¹¹ υπολογίζει την επαναγραφή συζευκτικών ερωτημάτων σε οντολογίες που είναι εκφρασμένες σε DL-Lite. Ο υπολογισμός της επαναγραφής αφορά στα ερωτήματα που έχουν υποστεί κάποια τροποποίηση η οποία σχετίζεται με την προσθήκη ή την αφαίρεση μεταβλητών ή ατόμων του ερωτήματος.

Ο αλγόριθμος συλλογιστικής που υλοποιεί το σύστημα υπολογίζει την επαναγραφή του αρχικού ερωτήματος που θέτει ο χρήστης. Έστω για παράδειγμα ότι ο χρήστης διατυπώνει ένα ερώτημα για την εύρεση όλων των συγγραφέων μίας βάσης. Εν συνεχείᾳ το ερώτημα μεταβάλεται ώστε να αναζητά τους συγγραφείς βιβλίων επιστημονικής φαντασίας. Το σύστημα κατασκευάζει τη νέα επαναγραφή για το τροποποιημένο ερώτημα αξιοποιώντας την ήδη υπάρχουσα. Η τεχνική που ακολουθείται και οι βελτιστοποιήσεις που εισάγει έχουν ως αποτέλεσμα την ταχύτατη απόκριση του συστήματος. Πρόκειται για ένα ανταγωνιστικό σύστημα ανοιχτού κώδικα.

¹⁰<http://www.cs.ox.ac.uk/isg/tools/Requiem/>

¹¹<https://code.google.com/p/iqaros/>

4.3.6 Rapid

Το Rapid¹² [TSCS13] είναι ένα σύστημα επαναγραφής συζευκτικών ερωτημάτων που διατυπώνονται σε οντολογίες σε \mathcal{ELHI} . Βασίζεται σε ένα σύνολο από βελτιστοποιήσεις που αφορούν στην τεχνική της ανάλυσης. Οι βελτιστοποιήσεις που εισάγει ο αλγόριθμος του Rapid στοχεύουν στην επιλεκτική εφαρμογή των κανόνων της ανάλυσης. Εκμεταλλευόμενος τη δομή του ερωτήματος και τα αξιώματα της οντολογίας εκτελεί μία ακολουθία στοχευμένων κανόνων ώστε να αποφύγει την εξαγωγή περιττών συμπερασμάτων-προτάσεων που ανεβάζουν πολύ το υπολογιστικό κόστος της συλλογιστικής διαδικασίας. Σημειώνουμε ότι το σύστημα επιστρέφει ένα σύνολο από προτάσεις (datalog κανόνες) που συνθέτουν την επαναγραφή του αρχικού ερωτήματος με βάση την οντολογία. Από την πειραματική αξιολόγηση του συστήματος προκύπτει ότι σε περιπτώσεις σύνθετων οντολογιών μεγάλης κλίμακας το σύστημα ξερνά κατά πολύ τα υπόλοιπα συστήματα επαναγραφής.

Το Rapid είναι σύστημα ανοιχτού κώδικα υπό την άδεια LGPL. Τέλος, αποτελεί μέρος του εργαλείου HydrOwl που παρουσιάζουμε στη συνέχεια.

4.3.7 HydrOwl

Το Hydrowl¹³ [Sto14] είναι ένα υβριδικό εργαλείο απάντησης συζευκτικών ερωτημάτων. Συνδυάζει υπάρχουσες τεχνολογίες για να αντιμετωπίσει το πρόβλημα απάντησης ερωτημάτων για πολύ εκφραστικά υποσύνολα της OWL 2.

Συγκεκριμένα, για οποιοδήποτε συνδυασμό των OWL 2 RL, OWL 2 EL και OWL 2 DL, το Hydrowl πραγματοποιεί κάποια προεπεξεργασία επί της οντολογίας και εν συνεχείᾳ, χρησιμοποιεί τον αλγόριθμο συλλογιστικής OWLIM για την απάντηση των ερωτημάτων σε πραγματικό χρόνο. Συνδυάζοντας τα αποτελέσματα συστημάτων όπως Hermit και Rapid καταφέρνει να αξιοποιήσει της τεχνολογίες του OWLIM ώστε να επιστρέψει ένα πλήρες σύνολο απαντήσεων.

Η εφαρμογή αναπτύχθηκε από τον Γιώργο Στοίλο και είναι ανοιχτού κώδικα με την άδεια χρήσης GNU Affero General Public License.

4.3.8 Nyaya

Το Nyaya [VOTT12] είναι ένα σύστημα για τη διαχείριση σημασιολογικών δεδομένων μεγάλης κλίμακας το οποίο επιτρέπει την αποθήκευση δεδομένων και την απάντηση συζευκτικών ερωτημάτων επί οντολογίων. Πρόκειται για ένα σύστημα υλοποιημένο σε Java. Παρέχει μια διεπαφή διαδικτύου για το σχηματισμό

¹²<http://www.image.ece.ntua.gr/~achort/rapid/>

¹³<http://www.image.ece.ntua.gr/~gstoil/hydrowl/>

και την υποβολή των επεερωτήματων, ενώ ο αλγόριθμος απάντησης ερωτημάτων που εφαρμόζει υλοποιεί την τεχνική της επαναγραφής. Παρέχει υπηρεσίες συλλογιστικής σε σύνολα σημασιολογικών δεδομένων που έχουν εκφραστεί με διαφορετικούς φορμαλισμούς. Πέρα από τις υπηρεσίες συλλογιστικής που προσφέρει, λειτουργεί επίσης σαν μια γενική και επεκτάσιμη αποθήκη δεδομένων για δεδομένα του σημασιολογικού ιστού, η οποία μπορεί να προσαρμοστεί σε διαφορετικές μηχανές απάντησης ερωτημάτων.

Το NYAYA έχει τη δυνατότητα να ελέγχει συντακτικά αν τα μεταδεδομένα σε μια αποθήκη μπορούν να ερωτηθούν αποτελεσματικά και να χρησιμοποιήσει το υποσύνολο της γλώσσας που εξασφαλίζει την επιθυμητή πολυπλοκότητα. Έτσι αντί να θέτονται περιορισμοί στην γλώσσα που χρησιμοποιείται ελέγχονται τα σύνολα των μεταδεδομένων σε σχέση με την αποφασισμότητα και την πολυπλοκότητα τους.

4.4 Εργαλεία απάντησης ερωτημάτων

4.4.1 Jena – Fuseki

Το Apache Jena¹⁴ είναι ένα προγραμματιστικό πλαίσιο στη γλώσσα Java για το σημασιολογικό ιστό. Αποτελείται από μια συλλογή εργαλείων που συνοδεύουν μία αποθήκη τριάδων RDF, με επιπλέον λειτουργίες που υποστηρίζουν το πρότυπο της OWL και της SPARQL.

Προσφέρει ένα κεντρικό API για τη δημιουργία, επεξεργασία και πρόσβαση γράφων RDF σε όλες τις γνωστές συντάξεις όπως RDF/XML, Turtle. Οι γράφοι RDF αναπαρίστανται σε ένα εσωτερικό μοντέλο με το οποίο συνεργάζονται όλα τα τμήματα της εφαρμογής. Ταυτόχρονα διατίθεται μία διεπαφή για την OWL, που υποστηρίζει από την απλή RDFS έως την πλήρη εκφραστικότητα της OWL. Το Jena μπορεί να συνδεθεί αλγόριθμους συλλογιστικής όπως o Pellet, o Racer, o Fact++.

Η εφαρμογή διατίθεται δωρεάν από το ίδρυμα Apache, με την ομώνυμη άδεια χρήσης που επιτρέπει και την χρήση για εμπορικούς σκοπούς.

Τέλος, σημειώνουμε ότι το υποσύστημα αποθήκευσης τριάδων RDF στο δίσκο και απάντησης ερωτημάτων που χρησιμοποιείται είναι το TDB. Ο σκοπός του είναι κυρίως να χρησιμοποιείται μέσω του API της Jena σε ένα μοναδικό μηχάνημα. Για την απάντηση ερωτημάτων εκτελεί διάφορες στατικές και δυναμικές βελτιστοποιήσεις, δηλαδή κάνει μετατροπές στο ερώτημα πριν αυτό εκτελεστεί, αλλά και κατά την διάρκεια της απάντησης ανάλογα με τα δεδομένα που έχει λάβει ως τότε. Για εκτέλεση ερωτημάτων SPARQL μέσω διαδικτύου χρησιμοποιείται ο εξυπηρετητής Fuseki (με εσωτερική έκδοση του TDB, που επιτρέπει λειτουργίες στο πρωτόκολλο SPARQL μέσω HTTP).

¹⁴<https://jena.apache.org>

4.4.2 Sesame

Το Sesame¹⁵ αποτελεί με τη σειρά του μία αποθήκη τριάδων RDF. Όπως και το Jena, προφέρει ένα πακέτο λειτουργιών για αποθήκευση και αναζήτηση δεδομένων RDF με τη γλώσσα SPARQL, χωρίς όμως την υποστήριξη στα υποσύνολα της γλώσσας OWL όπως το Jena, η οποία μπορεί να επιτευχθεί με τη σύνδεση σε εξωτερικές αποθήκες τριάδων.

Στη βάση της αρχιτεκτονικής του Sesame βρίσκονται δύο διεπαφές επικοινωνίας. Το χαμηλού επιπέδου Sail (Storage And Inference Layer) API, υποστηρίζει την αποθήκευση και συμπερασματολογία των δεδομένων RDF. Η δομή του επιτρέπει την προσαρμογή του Sesame σε διαφορετικά πακέτα αποθήκευσης. Το Repository API είναι υψηλότερου επιπέδου και προσφέρει μεθόδους φόρτωσης, εξαγωγής, επεξεργασίας δεδομένων και απάντησης ερωτημάτων. Πρόκειται για τη βασική διεπαφή για την ανάπτυξη εφαρμογών και μπορεί να λειτουργήσει με τοπικές και απομακρυσμένες ρυθμίσεις πελάτη - εξυπηρετητή. Το Sesame προσφέρει και ένα Server REST API σύμφωνα με το πρωτόκολλο SPARQL για RDF της W3C, για την επικοινωνία με τον εξυπηρετητή που τρέχει το Sesame μέσω HTTP. Ακόμη μία διεπαφή χρήστη που προσφέρεται και μπορεί να χρησιμοποιηθεί και ξεχωριστά από το υπόλοιπο πακέτο είναι το Rio, δηλαδή RDF I/O, για εγγραφή και ανάλυση δεδομένων σε αρχεία RDF. Με αυτό μπορεί κανείς να γράψει και να διαβάσει αρχεία, να τα μετατρέψει από μια σύνταξη σε άλλη κτλ. Το Sesame υποστηρίζει όλες τις γνωστές μορφές αρχείων RDF. Τέλος υπάρχει το OpenRDF workbench, μια διαδικτυακή εφαρμογή που προσφέρει πρόσβαση στις λειτουργίες του Sesame και να διαχειριστεί τα αποθετήρια (repositories) στον εξυπηρετητή του Sesame. Το Sesame διαθέτει δύο εσωτερικές βάσεις δεδομένων RD: την in-memory που κρατά τα αρχεία στη μνήμη, και την native που χρησιμοποιεί αποθήκευση στο δίσκο με χρήση ευρετηρίων. Μέσω του Repository API του όμως, μπορεί να συνδεθεί με διαφορετικές αποθήκες τριάδων όλων των τύπων, δηλαδή σχεσιακές βάσεις δεδομένων (δεν υποστηρίζεται πλέον), απομακρυσμένων τερματικών (remote endpoints), τερματικών SPARQL μέσω HTTP ή ομοσπονδιακές αποθήκες δεδομένων.

Τέλος, το Sesame είναι ανοιχτού κώδικα και πολλές εταιρίες το επιλέγουν ως βασικό διεπαφή στα προϊόντα τους. Η εφαρμογή διατίθεται δωρεάν.

4.4.3 OWLIM

Το GraphDB (παλαιότερα γνωστό ως OWLIM)¹⁶ της OntoText είναι μία αποθήκη τριάδων, ή εναλλακτικά, μία αποθήκη σημασιολογικού περιεχομένου σύμφωνα με την ορολογία που υιοθετείται από την OntoText. Διατίθεται σε τρεις εκ-

¹⁵<http://rdf4j.org>

¹⁶<http://www.ontotext.com/products/ontotext-graphdb-owlim/>

δόσεις, την Lite που προορίζεται για χρήση σε προσωπικό υπολογιστή και είναι η μόνη από τις τρεις που διατίθεται δωρεάν, την Standard που επιτρέπει την απάντηση ερωτημάτων από πολλούς χρήστες ταυτόχρονα και την Enterprise που επιπλέον προορίζεται για χρήση σε συστάδα. Βασική διαφορά της Lite από τις άλλες δύο είναι ότι εκτελεί τις λειτουργίες συλλογιστικής και απάντησης ερωτημάτων στη μνήμη, ενώ οι άλλες δύο τις εκτελούν με βάση ευρετήρια βασισμένα σε αρχεία. Αυτή η διαφορά δίνει στην Lite ένα χαμηλότερο κατώφλι στο μέγιστο αριθμό τριάδων που μπορεί να διαχειριστεί, δηλαδή στα 100 εκατομμύρια, ενώ οι άλλες μπορούν να επεξεργαστούν μέχρι δεκάδες δισεκατομμύρια.

Η OntoText έχει συμβάλλει στην ανάπτυξη του Sesame και η αρχιτεκτονική του GraphDB είναι πλήρως συμβατή με αυτό. Μάλιστα, είναι σχεδόν απαραίτητο για τις περισσότερες λειτουργίες που προσφέρει το API του. Όλες οι εκδόσεις χρησιμοποιούνται ως επίπεδα SAIL στο Sesame, μεταξύ άλλων για την υποστήριξη στις γλώσσες ερωτημάτων SPARQL και των διαφόρων τύπων συντακτικού αρχείων RDF.

Οι εκδόσεις είναι κλειστού κώδικα σε μεταγλωττισμένα αρχεία .jar και η δωρεάν έκδοση GraphDB Lite βρίσκεται στη σελίδα της OntoText, μαζί με τις σχετικές οδηγίες για τις εφαρμογές. Για πλήρη λειτουργικότητα πρέπει να υπάρχει σωστά εγκατεστημένο και το Sesame, ενώ για τις εμπορικές εκδόσεις υπάρχει και η διεπαφή για το Jena.

4.4.4 AllegroGraph

Το AllegroGraph¹⁷ είναι μία αποθήκη τριάδων. Οπως και οι προηγούμενες, υποστηρίζει λειτουργίες αποθήκευσης και επεξεργασίας αρχείων δεδομένων RDF και απάντησης ερωτημάτων με τη γλώσσα SPARQL. Ξεχωρίζει όμως σε διάφορα σημεία, κυρίως με την αρχιτεκτονική του που βασίζεται στο πρωτόκολλο REST (ουσιαστικά ένα υπερσύνολο του HTTP Client του Sesame) και μέσω της κοινότητας του διαθέτει προσαρμογές (Client Programming Interfaces) για πολλές γλώσσες προγραμματισμού.

Η αρχιτεκτονική που ακολουθεί είναι μιας βάσης δεδομένων γράφου, οπότε δεν περιορίζεται σε τριάδες RDF παρέχοντας κάποιες επιπλέον λειτουργίες, πέραν της RDF. Η διαχείριση των δεδομένων γίνεται με ένα ευλίκτο διανεμημένο σύστημα. Έτσι το σύστημα μπορεί να τρέξει με πολλά στιγμιότυπα ταυτόχρονα, στο ίδιο μηχάνημα ή σε συστάδα και με διαφορετικές πηγές δεδομένων που μπορούν να ενοποιούνται. Οι επιδόσεις του είναι ανταγωνιστικές, με την εταιρεία να ισχυρίζεται ότι έχει φορτώσει ένα τρισεκατομμύριο τριάδες σε συστάδα. Τέλος, είναι συμβατό με αρκετές εξωτερικές τεχνολογίες όπως ο περιηγητής RDF Gruff, για την οπτικοποίηση του γράφου δεδομένων, ο αλγόριθμος συλλογιστικής RacerPro,

¹⁷<http://franz.com/agraph/allegrograph/>

ενώ είναι δυνατή και η χρήση του πάνω από το MongoDB, παρόλο που αυτό δεν υποστηρίζει RDF και SPARQL από μόνο του.

Η εφαρμογή είναι κλειστού κώδικα και αποτελείται από τον εξυπηρετητή, που τρέχει σε linux ή μέσω εικονικού μηχανήματος σε άλλα λειτουργικά συστήματα, και τους διάφορους πελάτες που προαναφέραμε, ενώ υπάρχει και επιλογή πελάτη για διαδικτυακή διεπαφή με τον εξυπηρετητή, μέσω σε πλοιηγητή ιστοσελίδων (web browser). Παρέχεται πλήρη τεκμηρίωση της στη σελίδα της, και διατίθεται σε τρεις εκδόσεις, μια δωρεάν με όριο τα πέντε εκατομμύρια τριάδες και δύο εμπορικές την Developer με πενήντα εκατομμύρια και την Enterprise χωρίς όριο στον αριθμό.

4.4.5 Stardog

Το Stardog¹⁸ είναι μια εμπορική σημασιολογική βάση δεδομένων RDF υλοποιημένη σε Java, που κατασκευάστηκε από τους Clark & Parsia. Υπάρχουν τρεις διαθέσιμες εκδόσεις (Community, Developer και Enterprise). Η πρώτη είναι εκπαιδευτική με αρκετούς περιορισμούς και είναι ελεύθερη, η δεύτερη απευθύνεται σε προγραμματιστές με ελεύθερη δοκιμή και η τρίτη απευθύνεται σε επιχειρήσεις για τη δημιουργία εμπορικών εφαρμογών. Στο εγχειρίδιο που βρίσκεται στην επίσημη ιστοσελίδα του υπάρχουν οδηγίες για την εγκατάσταση της τρέχουσας έκδοσης (Stardog 2.2.4) η οποία υποστηρίζει το μοντέλο RDF, όλα τα προφίλ της OWL 2 και τους κανόνες για την εξαγωγή συμπερασμάτων και την ανάλυση δεδομένων. Επίσης προσφέρει δυνατότητες συλλογιστικής και απάντησης ερωτημάτων και χρησιμοποιεί την SPARQL1.1 σαν γλώσσα ερωτημάτων. Υποστηρίζει τις διεπαφές Sesame API και Jena (μέσω μιας γέφυρας Sesame-Jena), το SNARL, που είναι το εγγενές API του Stardog και διεπαφή γραμμής εντολών η οποία χρησιμοποιεί τα πρωτόκολλα HTTP και SNARL.

Η συλλογιστική του βασίζεται κυρίως σε μια τεχνική επαναγραφής του ερωτήματος, όπου το ερώτημα του χρήστη επανεγγράφεται σε έναν αριθμό ν ερωτημάτων τα οποία στη συνέχεια εκτελούνται αυτόματα. Η σημασιολογία της συλλογιστικής του Stardog βασίζεται στην OWL 2. Ωστόσο, για λόγους ευελιξίας, το σύστημα συλλογιστικής του βασίζεται σε μια οκνηρή τεχνική όπου δεν γίνεται η υλοποίηση της έμμεσης πληροφορίας (όπως περιγράφηκε στην Ενότητα 2.3.6), αλλά η συλλογιστική πραγματοποιείται τη στιγμή του ερωτήματος, σύμφωνα με ένα καθορισμένο από τον χρήστη επίπεδο συλλογισμού. Τα επίπεδα συλλογιστικής που προσφέρει είναι:

- NONE – Δε εκτελείται συλλογιστική
- RDFS – Για OWL 2 αξιώματα που επιτρέπονται στο RDF schema

¹⁸<http://stardog.com/>

- QL – Για OWL 2 QL αξιώματα
- RL – Για OWL 2 RL αξιώματα
- EL - Για OWL 2 EL αξιώματα
- DL - Για OWL 2 DL αξιώματα
- SL – Για συνδιασμό RDFS, QL, RL, EL αξιωμάτων και κανόνων SWRL

To Stardog υποστηρίζει την συλλογιστική με κανόνες σε συνδυασμό με αξιώματα OWL. Επιτρέπει δύο διαφορετικές συντάξεις για τον καθορισμό των κανόνων. Αυτή των εγγενών κανόνων Stardog, η οποία βασίζεται στη SPARQL, και τη σύνταξη των κανόνων που ορίζονται από το χρήστη, η οποία είναι η τυπική RDF/XML σύνταξη για SWRL.

Υποστηρίζει την επικύρωση δεδομένων RDF που είναι αποθηκευμένα σε κάποια βάση δεδομένων σύμφωνα με τους κανόνες των δεδομένων, περιορισμούς ακεραιότητας, που περιγράφονται από τους χρήστες (οι περιορισμοί αυτοί μπορούν να γραφτούν σε SPARQL, OWL ή SWRL με σημασιολογία κλειστού κόσμου). Η χρήση γλωσσών υψηλού επιπέδου σαν γλώσσες σχήματος η περιορισμών ακεραιότητας για RDF και διασυνδεδεμένα δεδομένα επιτρέπει την χρήση αυτοματοποιημένης συλλογιστικής για τα μοντέλα, τους κανόνες και τους περιορισμούς ταυτόχρονα καθώς και την εκτέλεση ερωτημάτων σε SPARQL.

To Stardog δίνει τη δυνατότητα εκτέλεσης ερωτημάτων μόνο σε RDF δεδομένα που έχει υπό την διαχείρισή του. Δεν ανακτά δεδομένα από το διαδίκτυο ή άλλα δίκτυα μέσω HTTP URLs επομένως για την εκτέλεση ερωτημάτων τα δεδομένα απαιτείται να εισαχθούν σε μια δική του βάση δεδομένων.

4.4.6 OpenLink Virtuoso

O Virtuoso Universal Server¹⁹ είναι μια ένας υβριδικός διακομιστής που προσφέρει μεταξύ άλλων τη δυνατότητα διαχείρισης δεδομένων που ακολουθούν το RDF σχήμα. Παρέχει μία αποθήκη RDF δεδομένων η πρόσβαση στην οποία γίνεται μέσω διεπαφής που επιτρέπει την εκτέλεση ερωτημάτων SPARQL 1.1 Διατίθεται σε εμπορική έκδοση αλλά και σε έκδοση ανοικτού κώδικα. O Virtuoso είναι επίσης ένας συλλογιστής για τη γλώσσα OWL. Υποστηρίζει τους τελεστές sameAs, equivalentClass, equivalentProperty της OWL, subClassOf, subPropertyOf του rdfs. Επιπλέον, υποστηρίζει τη μεταβατική, συμμετρική και αντίστροφη ιδιότητα της OWL. Ο συλλογιστής που υλοποιείται στο Virtuoso εφαρμόζει τη συλλογιστική “προς τα πίσω” για την εξαγωγή των συμπερασμάτων, ενώ ο χρήστης έχει

¹⁹<http://virtuoso.openlinksw.com/>

τη δυνατότητα να επιλέξει την “προς τα εμπρός” συλλογιστική που πραγματοποιεί υλοποίηση.

Τέλος, σημειώνουμε ότι σύνολα δεδομένων μεγάλης κλίμακας στο διαδίκτυο όπως η DBpedia, το Bio2RDF, το LOD Cloud Cache, φιλοξενούνται από το Virtuoso. Επιπλέον, η διεπαφή SPARQL του Virtuso είναι η προεπιλεγμένη διεπαφή για τη DBpedia και άλλες βάσεις.

4.4.7 OWL-BGP

Το OWL BGP²⁰ [KG14] αποτελεί ένα μηχανισμό απάντησης SPARQL ερωτημάτων για OWL συλλογιστές που βασίζονται (που κάνουν χρήση του OWL API). Παράχει υπηρεσίες απάντησης SPARQL ερωτημάτων με βάση τη σημασιολογία της OWL. Βασίζεται εσωτερικά στη μηχανή ερωτημάτων ARQ του Apache Jena²¹. Το OWL BGP παρέχει ειδική υποστήριξη για τον συλλογιστή HermiT, αφού χρησιμοποιεί τα στατιστικά στοιχεία που παρέχονται από το συλλογιστή για το σχεδιασμό του ερωτήματος.

²⁰<http://www.uni-ulm.de/en/in/ki/software/owl-bgp.html>

²¹<http://jena.apache.org/documentation/query/index.html>

5 Συμπεράσματα

Στο παρόν παραδοτέο παρουσιάσαμε τη θεωρία που αφορά στη σημασιολογική αναπαράσταση γνώσης καθώς και τα σχετικά προβλήματα που αποτελούν αντικείμενο έρευνας και μελέτης. Στη συνέχεια, περιγράψαμε τα πρότυπα έτσι όπως έχουν αναπτυχθεί για χρήση από συστήματα και πλατφόρμες που αξιοποιούν τεχνολογίες αναπαράστασης γνώσης. Τέλος, παρουσιάσαμε τα σύχρονα πρακτικά συστήματα που διαχειρίζονται δεδομένα με σημασιολογικό περιεχόμενο και παρέχουν υπηρεσίες απάντησης ερωτημάτων.

Αναφορές

- [ACKZ09] Alessandro Artale, Diego Calvanese, Roman Kontchakov, and Michael Zakharyashev. The dl-lite family and relations. *J. Artif. Int. Res.*, 36(1):1–69, September 2009.
- [AHV95] Serge Abiteboul, Richard Hull, and Victor Vianu. *Foundations of Databases*. Addison-Wesley, 1995.
- [BBL05] Franz Baader, Sebastian Brand, and Carsten Lutz. Pushing the el envelope. In *In Proc. of IJCAI 2005*, pages 364–369. Morgan-Kaufmann Publishers, 2005.
- [BCM⁺03] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, New York, NY, USA, 2003.
- [BG01] Leo Bachmair and Harald Ganzinger. Resolution theorem proving. In *Handbook of Automated Reasoning*, pages 19–99. Elsevier and MIT Press, 2001.
- [BH11] Franz Baader and Philipp Hanschke. A scheme for integrating concrete domains into concept languages. 2011.
- [BLHL01] Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web. *Scientific American*, 2001.
- [BvHH⁺04] Sean Bechhofer, Frank van Harmelen, James Hendler, Ian Horrocks, Deborah L. McGuinness, Peter F. Patel-Schneider, and Lynn Andrea Stein eds. Owl web ontology language reference. URL <http://www.w3.org/TR/owl-ref/>, Feb 2004.
- [CDGL⁺07] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Tractable reasoning and efficient query answering in description logics: The DL-Lite family. *J. of Automated Reasoning*, 39(3):385–429, 2007.
- [Dir12] A Direct Mapping of relational data to RDF. W3C Recommendation, 2012.
- [EOS⁺12] Thomas Eiter, Magdalena Ortiz, Mantas Simkus, Trung-Kien Tran, and Guohui Xiao. Query rewriting for Horn- \mathcal{SHIQ} plus rules. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI 2012)*, 2012.

- [GHM⁺14] Birte Glimm, Ian Horrocks, Boris Motik, Giorgos Stoilos, and Zhe Wang. Hermit: An owl 2 reasoner. *J. Autom. Reason.*, 53(3):245–269, October 2014.
- [HHMW12] Volker Haarslev, Kay Hidde, Ralf Möller, and Michael Wessel. The racerpro knowledge representation and reasoning system. *Semantic Web Journal*, 3(3):267–277, 2012.
- [HMS08] Ullrich Hustadt, Boris Motik, and Ulrike Sattler. Deciding Expressive Description Logics in the Framework of Resolution. *Information & Computation*, 206(5):579–601, 2008.
- [KG14] Ilianna Kollia and Birte Glimm. Optimizing sparql query answering over owl ontologies. *arXiv preprint arXiv:1402.0576*, 2014.
- [MSS05] Boris Motik, Ulrike Sattler, and Rudi Studer. Query answering for owl-dl with rules. *Journal of Web Semantics*, 3(1):41–60, 2005.
- [PLC⁺08] Antonella Poggi, Domenico Lembo, Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, and Riccardo Rosati. Journal on data semantics x. chapter Linking Data to Ontologies, pages 133–173. Springer-Verlag, Berlin, Heidelberg, 2008.
- [PSHH04] Peter F. Patel-Schneider, Patrick Hayes, and Ian Horrocks. OWL Web Ontology Language Semantics and Abstract Syntax. Technical report, W3C, Feb. 2004. W3C Recommendation, URL <http://www.w3.org/TR/2004/REC-owl-semantics-20040210/>.
- [PUMH10] Héctor Pérez-Urbina, Boris Motik, and Ian Horrocks. Tractable query answering and rewriting under description logic constraints. *Journal of Applied Logic*, 8(2):186–209, 2010.
- [R2R12] R2RML: RDB to RDF Mapping Language. W3C Recommendation, 2012.
- [RA10] Riccardo Rosati and Alessandro Almatelli. Improving query answering over DL-Lite ontologies. In *Proceedings of the 12th International Conference on the Principles of Knowledge Representation and Reasoning (KR-10)*, 2010.
- [RDF14a] RDF 1.1 concepts and abstract syntax. W3C Recommendation, 2014.

- [RDF14b] RDF Schema 1.1. W3C Recommendation, 2014.
- [SKH11] Frantisek Simancik, Yevgeny Kazakov, and Ian Horrocks. Consequence-based reasoning beyond horn ontologies. In *IJCAI*, pages 1093–1098, 2011.
- [SMH08] Rob Shearer, Boris Motik, and Ian Horrocks. HermiT: A Highly-Efficient OWL Reasoner. In Alan Ruttenberg, Ulrike Sattler, and Cathy Dolbear, editors, *Proc. of the 5th Int. Workshop on OWL: Experiences and Directions (OWLED 2008 EU)*, Karlsruhe, Germany, October 26–27 2008.
- [SPA] SPARQL 1.1 Query Language. W3C Recommendation.
- [SPA08] SPARQL Query Language for RDF. W3C Recommendation, 2008.
- [SPA13] SPARQL 1.1 Entailment Regimes. W3C Recommendation, 2013.
- [SPG⁺07] Evren Sirin, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur, and Yarden Katz. Pellet: A practical owl-dl reasoner. *Web Semant.*, 5(2):51–53, June 2007.
- [Sto14] Giorgos Stoilos. Hydrowl: A hybrid query answering system for owl 2 dl ontologies. In *Proceedings of the 8th International Conference on Web Reasoning and Rule Systems (RR 2014)*, 2014.
- [TH06] Dmitry Tsarkov and Ian Horrocks. Fact++ description logic reasoner: System description. In *Proceedings of the Third International Joint Conference on Automated Reasoning*, IJCAR’06, pages 292–297, Berlin, Heidelberg, 2006. Springer-Verlag.
- [TSCS13] Despoina Trivela, Giorgos Stoilos, Alexandros Chortaras, and Giorgos Stamou. Optimising resolution-based rewriting algorithms for dl ontologies. In *Proceedings of the 26th Workshop on Description Logics (DL 2013), Ulm, Germany*, 2013.
- [VOTT12] Roberto De Virgilio, Giorgio Orsi, Letizia Tanca, and Riccardo Torlone. NYAYA: A system supporting the uniform management of large sets of semantic data. In *IEEE 28th International Conference on Data Engineering (ICDE 2012), Washington, DC, USA (Arlington, Virginia), 1-5 April, 2012*, pages 1309–1312, 2012.