



Μέθοδοι και Τεχνικές Ανακάλυψης Γνώσης στο Σημαντικό Ιστό

Παραγωγική Απόκτηση Γνώσης από
Οντολογικά Έγγραφα και η Τεχνική
της Σημασιακής Προσαρμογής

Δημήτριος Α. Κουτσομητρόπουλος

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

Επιβλέπων: Θεόδωρος Σ. Παπαθεοδώρου, Καθηγητής

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ ΚΑΙ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΔΗΜΗΤΡΙΟΣ Α. ΚΟΥΤΣΟΜΗΤΡΟΠΟΥΛΟΣ

**ΜΕΘΟΔΟΙ ΚΑΙ ΤΕΧΝΙΚΕΣ ΑΝΑΚΑΛΥΨΗΣ ΓΝΩΣΗΣ ΣΤΟ
ΣΗΜΑΝΤΙΚΟ ΙΣΤΟ**
ΠΑΡΑΓΩΓΙΚΗ ΑΠΟΚΤΗΣΗ ΓΝΩΣΗΣ ΑΠΟ ΟΝΤΟΛΟΓΙΚΑ ΕΓΓΡΑΦΑ
ΚΑΙ Η ΤΕΧΝΙΚΗ ΤΗΣ ΣΗΜΑΣΙΑΚΗΣ ΠΡΟΣΑΡΜΟΓΗΣ

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

ΕΠΙΒΛΕΠΩΝ: ΘΕΟΔΩΡΟΣ Σ. ΠΑΠΑΘΕΟΔΩΡΟΥ, ΚΑΘΗΓΗΤΗΣ

ΤΡΙΜΕΛΗΣ ΣΥΜΒΟΥΛΕΥΤΙΚΗ ΕΠΙΤΡΟΠΗ:

ΛΥΚΟΘΑΝΑΣΗΣ ΣΠΥΡΙΔΩΝ, ΚΑΘΗΓΗΤΗΣ
ΜΠΟΥΡΑΣ ΧΡΗΣΤΟΣ, ΑΝ. ΚΑΘΗΓΗΤΗΣ
ΠΑΠΑΘΕΟΔΩΡΟΥ ΘΕΟΔΩΡΟΣ, ΚΑΘΗΓΗΤΗΣ

ΕΠΤΑΜΕΛΗΣ ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ:

ΚΑΠΙΔΑΚΗΣ ΣΑΡΑΝΤΟΣ, ΑΝ. ΚΑΘΗΓΗΤΗΣ, ΙΟΝΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ
ΚΟΛΛΙΑΣ ΣΤΕΦΑΝΟΣ, ΚΑΘΗΓΗΤΗΣ, ΕΜΠ
ΛΥΚΟΘΑΝΑΣΗΣ ΣΠΥΡΙΔΩΝ, ΚΑΘΗΓΗΤΗΣ
ΜΠΟΥΡΑΣ ΧΡΗΣΤΟΣ, ΑΝ. ΚΑΘΗΓΗΤΗΣ
ΠΑΠΑΘΕΟΔΩΡΟΥ ΘΕΟΔΩΡΟΣ, ΚΑΘΗΓΗΤΗΣ
ΧΑΤΖΗΛΥΓΓΕΡΟΥΔΗΣ ΙΩΑΝΝΗΣ, ΕΠ. ΚΑΘΗΓΗΤΗΣ
ΧΡΙΣΤΟΔΟΥΛΑΚΗΣ ΔΗΜΗΤΡΙΟΣ, ΚΑΘΗΓΗΤΗΣ

ΠΑΤΡΑ 2008

*στη μνήμη του πατέρα μου,
Αλέξανδρου*

«Μέθοδοι και Τεχνικές Ανακάλυψης Γνώσης στο Σημαντικό Ιστό: Παραγωγική Απόκτηση Γνώσης από Οντολογικά Έγγραφα και η Τεχνική της Σημασιακής Προσαρμογής»

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

ΠΕΡΙΛΗΨΗ

Ο Σημαντικός Ιστός (Semantic Web) είναι ένας συνδυασμός τεχνολογιών και προτύπων με σκοπό να προσδοθεί στη διαδικτυακή πληροφορία αυστηρά καθορισμένη σημασιακή δομή και ερμηνεία. Στόχος είναι να μπορούν οι χρήστες του Παγκόσμιου Ιστού καθώς και αυτοματοποιημένοι πράκτορες να επεξεργάζονται, να διαχειρίζονται και να αξιοποιούν την κατάλληλα χαρακτηρισμένη πληροφορία με τρόπο ευφυή και αποδοτικό.

Ωστόσο, παρά τις τεχνικές που έχουν κατά καιρούς προταθεί, δεν υπάρχει ξεκάθαρη μέθοδος ώστε, αξιοποιώντας το φάσμα του Σημαντικού Ιστού, η διαδικτυακή πληροφορία να ανακτάται με τρόπο παραγωγικό, δηλαδή με βάση τα ήδη εκπεφρασμένα γεγονότα να συνάγεται νέα, άρρητη πληροφορία.

Για την αντιμετώπιση αυτής της κατάστασης, αρχικά εισάγεται και προσδιορίζεται το πρόβλημα της *Ανακάλυψης Γνώσης στο Σημαντικό Ιστό* (Semantic Web Knowledge Discovery, SWKD). Η Ανακάλυψη Γνώσης στο Σημαντικό Ιστό εκμεταλλεύεται το σημασιακό υπόβαθρο και τις αντίστοιχες σημασιακές περιγραφές των πληροφοριών, όπως αυτές είναι θεμελιωμένες σε μια λογική θεωρία (οντολογίες εκφρασμένες σε γλώσσα OWL). Βάσει αυτών και με τη χρήση των κατάλληλων μηχανισμών αυτοματοποιημένου συλλογισμού μπορεί να συμπεραθεί νέα, άδηλη γνώση, η οποία, μέχρι τότε, μόνο υπονοούνταν στα ήδη υπάρχοντα δεδομένα.

Για να απαντηθεί το ερώτημα αν και σε πιο βαθμό οι τεχνολογίες και η λογική θεωρία του Σημαντικού Ιστού συνεισφέρουν *αποδοτικά* και *εκφραστικά* στο πρόβλημα της SWKD καταρτίζεται μια πρότυπη *Μεθοδολογία Ανακάλυψης Γνώσης στο Σημαντικό Ιστό*, η οποία θεμελιώνεται σε πρόσφατα θεωρητικά αποτελέσματα, αλλά και στην ποιοτική και πειραματική συγκριτική αξιολόγηση διαδεδομένων μηχανισμών συμπερασμού (inference engines) που βασίζονται σε Λογικές Περιγραφές (Description Logics). Η αποδοτικότητα και η εκφραστικότητα της μεθόδου αυτής δείχνεται ότι εξαρτώνται από συγκεκριμένους θεωρητικούς, οργανωτικούς και τεχνικούς περιορισμούς.

Η πειραματική επαλήθευση της μεθοδολογίας επιτυγχάνεται με την κατασκευή και επίδειξη της *Διεπαφής Ανακάλυψης Γνώσης* (Knowledge Discovery Interface) μιας κατανεμημένης δηλαδή δικτυακής υπηρεσίας, η οποία έχει εφαρμοστεί με επιτυχία σε πειραματικά δεδομένα. Τα αποτελέσματα που προκύπτουν με τη χρήση της διεπαφής επαληθεύουν, μέχρι ορισμένο βαθμό, τις υποθέσεις που έχουν γίνει σχετικά κυρίως με την παράμετρο της *εκφραστικότητας* και δίνουν το έναυσμα για την αναζήτηση και εξέταση της υποστήριξης των νέων προτεινόμενων επεκτάσεων της λογικής θεωρίας του Σημαντικού Ιστού, δηλαδή της γλώσσας OWL 1.1.

Για την ενίσχυση της εκφραστικότητας της ανακάλυψης γνώσης στην περίπτωση συγκεκριμένων πεδίων γνώσης (knowledge domains) εισάγεται μια νέα τεχνική, αποκαλούμενη *Σημασιακή Προσαρμογή*. Η τεχνική αυτή εξελίσσει την *Προσαρμογή Μεταδεδομένων Εφαρμογής* (Metadata Application Profiling) από μια επίπεδη συρραφή και

συγχώνευση σχημάτων και πεδίων μεταδεδομένων, σε μία ουσιαστική επέκταση και σημασιακή αναγωγή και εμπλουτισμό του αντίστοιχου μοντέλου στο οποίο εφαρμόζεται. Έτσι, η σημασιακή προσαρμογή εξειδικεύει ένα οντολογικό μοντέλο ως προς μια συγκεκριμένη εφαρμογή, όχι απλά με την προσθήκη λεξιλογίου από ετερογενή σχήματα, αλλά μέσω της σημασιακής εμβάθυνσης (*semantic intension*) και εκλέπτυνσης (*semantic refinement*) του αρχικού μοντέλου. Η τεχνική αυτή και τα αποτελέσματά της επαληθεύονται πειραματικά με την εφαρμογή στο μοντέλο πληροφοριών πολιτιστικής κληρονομιάς CIDOC-CRM και δείχνεται ότι, με τη χρήση κατάλληλων μεθόδων, η γενική εφαρμοσιμότητα του μοντέλου μπορεί να διαφυλαχθεί.

Για να μπορεί όμως η Ανακάλυψη Γνώσης στο Σημαντικό Ιστό να δώσει ικανοποιητικά αποτελέσματα, απαιτούνται όσο το δυνατόν πληρέστερες και αυξημένες περιγραφές των δικτυακών πόρων. Παρόλο που πληροφορίες άμεσα συμβατές με τη λογική θεωρία του Σημαντικού Ιστού δεν είναι ευχερείς, υπάρχει πληθώρα δεδομένων οργανωμένων σε επίπεδα σχήματα μεταδεδομένων (*flat metadata schemata*). Διερευνάται επομένως αν η SWKD μπορεί να εφαρμοστεί *αποδοτικά* και *εκφραστικά* στην περίπτωση τέτοιων *ημιδομημένων μοντέλων γνώσης*, όπως για παράδειγμα στην περίπτωση του σχήματος μεταδεδομένων Dublin Core. Δείχνεται ότι το πρόβλημα αυτό ανάγεται μερικώς στην εφαρμογή της σημασιακής προσαρμογής στην περίπτωση τέτοιων μοντέλων, ενώ για τη διαφύλαξη της διαλειτουργικότητας και την επίλυση αμφισημιών που προκύπτουν εφαρμόζονται ανάλογες μέθοδοι και επιπλέον εξετάζεται η τεχνική της *παρονομασίας* (*running*) που εισάγει η OWL 1.1, βάσει της οποίας ο ορισμός ενός ονόματος μπορεί να έχει κυμαινόμενη σημασιακή ερμηνεία ανάλογα με τα συμφραζόμενα.

Συμπερασματικά, οι νέες μέθοδοι που προτείνονται μπορούν να βελτιώσουν το πρόβλημα της Ανακάλυψης Γνώσης στο Σημαντικό Ιστό ως προς την εκφραστικότητα, ενώ ταυτόχρονα η πολυπλοκότητα παραμένει η μικρότερη δυνατή. Επιτυγχάνουν επίσης την παραγωγή πιο εκφραστικών περιγραφών από υπάρχοντα μεταδεδομένα, προτείνοντας έτσι μια λύση στο πρόβλημα της *εκκίνησης* (*bootstrapping*) για το Σημαντικό Ιστό. Παράλληλα, μπορούν να χρησιμοποιηθούν ως βάση για την υλοποίηση πιο αποδοτικών τεχνικών κατανεμημένου και αυξητικού συλλογισμού.

DIMITRIOS A. KOUTSOMITROPOULOS

«Methods and Techniques for Semantic Web Knowledge Discovery: Deductive Knowledge Acquisition from Ontology Documents and the Semantic Profiling Technique»

DOCTORATE THESIS

ABSTRACT

Semantic Web is a combination of technologies and standards in order to give Web information strictly defined semantic structure and meaning. Its aim is to enable Web users and automated agents to process, manage and utilize properly described information in intelligent and efficient ways.

Nevertheless, despite the various techniques that have been proposed, there is no clear method such that, by taking advantage of Semantic Web technologies, to be able to retrieve information deductively, i.e. to infer new and implicit information based on explicitly expressed facts.

In order to address this situation, the problem of *Semantic Web Knowledge Discovery* (SWKD) is first specified and introduced. SWKD takes advantage of the semantic underpinnings and semantic descriptions of information, organized in a logic theory (i.e. ontologies expressed in OWL). Through the use of appropriate automated reasoning mechanisms, SWKD makes then possible to deduce new and unexpressed information that is only implied among explicit facts.

The question as to whether and to what extent do Semantic Web technologies and logic theory contribute *efficiently* and *expressively* enough to the SWKD problem is evaluated through the establishment of a *SWKD methodology*, which builds upon recent theoretical results, as well as on the qualitative and experimental comparison of some popular inference engines, based on Description Logics. It is shown that the efficiency and expressivity of this method depends on specific theoretical, organizational and technical limitations.

The experimental verification of this methodology is achieved through the development and demonstration of the *Knowledge Discovery Interface* (KDI), a web-distributed service that has been successfully applied on experimental data. The results taken through the KDI confirm, to a certain extent, the assumptions made mostly about expressivity and motivate the examination and investigation of the newly proposed extensions to the Semantic Web logic theory, namely the OWL 1.1 language.

In order to strengthen the expressivity of knowledge discovery in the case of particular knowledge domains a new technique is introduced, known as *Semantic Profiling*. This technique evolves traditional *Metadata Application Profiling* from a flat aggregation and mixing of schemata and metadata elements to the substantial extension and semantic enhancement and enrichment of the model on which it is applied. Thus, semantic profiling actually profiles an ontological model for a particular application, not only by bringing together vocabularies from disparate schemata, but also through the semantic intension and semantic refinement of the initial model. This technique and its results are experimentally verified through its application on the CIDOC-CRM cultural heritage information model and it

is shown that, through appropriate methods, the general applicability of the model can be preserved.

However, for SWKD to be of much value, it requires the availability of rich and detailed resource descriptions. Even though information compatible with the Semantic Web logic theory are not always readily available, there are plenty of data organized in flat metadata schemata. To this end, it is investigated whether SWKD can be *efficiently* and *expressively* applied on such *semi-structured knowledge models*, as is the case for example with the Dublin Core metadata schema. It is shown that this problem can be partially reduced to applying semantic profiling on such models and, in order to retain interoperability and resolve potential ambiguities, the OWL 1.1 *punning* feature is investigated, based on which a name definition may have variable semantic interpretation depending on the ontological context.

In conclusion, these newly proposed methods can improve the SWKD problem in terms of expressive strength, while keeping complexity as low as possible. They also contribute to the creation of expressive descriptions from existing metadata, suggesting a solution to the Semantic Web *bootstrapping* problem. Finally, they can be utilized as the basis for implementing more efficient techniques that involve distributed and incremental reasoning.

Ευχαριστίες

Πολλή ευγνωμοσύνη οφείλω και πάντα θα χρωστώ στον καθηγητή κ. Θεόδωρο Παπαθεοδώρου, επιβλέποντα της διατριβής, που (και όχι μόνο εγώ) θεωρώ σπουδαίο Καθηγητή και Δάσκαλο, στον άνθρωπο αυτό που μπορεί και εμπνέει αμεσότητα μα και ακαδημαϊκή πληρότητα, και που με το παράδειγμά του, όχι μόνο με τα λόγια, μου έκανε την τιμή όλα αυτά τα χρόνια να είμαι μαθητής του.

Ευχαριστώ απ' την καρδιά μου τους καθηγητές κκ. Σπύρο Λυκοθανάση και Χρήστο Μπούρα, μέλη της συμβουλευτικής επιτροπής, που έχτισαν πάνω στην ατέλεια των προσπαθειών μου με τη διδασκαλία, τις συμβουλές και την ολόπλευρη στήριξή τους.

Θερμά επίσης ευχαριστώ τους καθηγητές κκ. Σαράντο Καπιδάκη, Στέφανο Κόλλια, Ιωάννη Χατζηλυγερούδη και Δημήτρη Χριστοδουλάκη, μέλη της εξεταστικής επιτροπής, που πρόθυμα δέχτηκαν να αξιολογήσουν το περιεχόμενο της διατριβής. Οι ιδέες και η έρευνά τους ίσως δεν είναι τυχαίο που διαπνέουν σε πολλά σημεία τη δουλειά αυτή.

Ευχαριστώ ακόμα το κοινωφελές ίδρυμα «Αλέξανδρος Σ. Ωνάσης» για την υποστήριξη που μου προσέφερε κατά τη διάρκεια εκπόνησης της διατριβής.

Το Εργαστήριο Πληροφοριακών Συστημάτων Υψηλών Επιδόσεων (HPCLab), αποτέλεσε για μένα, στη δεκαετή πορεία μου μέσα σ' αυτό, μόνιμο ερέθισμα για προσπάθεια, συνέχεια και πρόοδο, αλλά και διαρκές περιβάλλον φιλίας κι άμιλλας. Όλους τους ανθρώπους του λοιπόν ευχαριστώ, γιατί αυτοί είναι το Εργαστήριο. Ξεχωρίζω όμως ιδιαίτερα τη συνεργάτιδά μου Γεωργία Σολωμού που βοήθησε και εμπράκτως στην ευόδωση των στόχων της διατριβής.

Ευχαριστώ τους φίλους μου που με υπέμειναν και με υπομένουν·

τις αδερφές μου, Δέσποινα και Μίτση, που δεν έπαψαν να με αγαπούν και να με στηρίζουν·

την μητέρα μου, Έφη, ευχαριστώ για όλα.

Αντί Προλόγου

«Γιατί Σημαντικός και όχι Σημασιολογικός;»

Η ερώτηση αυτή, απευθυνόμενη σε μένα και στους περισσότερους με τους οποίους έχουμε συνεργαστεί στο αντικείμενο, έχει πια αρχίσει να αποκτά κλασσικό χαρακτήρα. Νομίζω πώς εδώ, αντί για πρόλογο αυτής της διατριβής που θεραπεύει το *Σημαντικό* Ιστό, είναι μια καλή ευκαιρία να ξεδιαλύνω το θέμα αυτό.

Στα περισσότερα λεξικά (Τεγόπουλου-Φυτράκη, ηλεκτρονικό Τριανταφυλλίδη, Μπαμπινιώτη...) το επίθετο *σημαντικός* έχει πρωτεύουσα (δηλ. πρώτη, κύρια) ερμηνεία *αυτός που έχει ή δηλώνει ορισμένη σημασία*. Μετά, δεύτερη έρχεται η ερμηνεία *αξιόλογος, που έχει μεγάλη σημασία*. Επομένως, η χρήση του όρου *σημαντικός* γίνεται με την έννοια του φέροντος σημασία και όχι του σπουδαίου.

Ας δούμε όμως και τον όρο *σημασιολογικός*, πάλι στα λεξικά: *ο αναφερόμενος στη σημασιολογία*. Ακριβώς από πάνω είναι το λήμμα *σημασιολογία: κλάδος της γλωσσολογίας που μελετά τη σημασία των λέξεων*. Άρα λοιπόν *σημασιολογικός* είναι ο Ιστός που σχετίζεται με «*κλάδο της γλωσσολογίας*»; Μα ο ίδιος ο Tim Berners Lee, ο πρωτεργάτης (και του *Σημαντικού*) Ιστού, ήδη από το 2000, στην εναρκτήρια ομιλία που έδωσε στο συνέδριο XML, απέκλεισε ρητά κάθε ερμηνεία που σχετίζεται με τη φυσική γλώσσα: “not natural language, human inference”¹.

Ας δούμε άλλωστε τον ακόλουθο ορισμό:

Semantic comes from the Greek words for sign, signify, and significant, and today means of or relating to meaning, even in language (μτφ: Η λέξη σημαντικός προέρχεται από τις ελληνικές λέξεις σήμα, σημαίνω και σημαντικός και σήμερα σημαίνει αυτόν που ανήκει ή έχει σημασία, ακόμα και στη γλώσσα).

Ο ορισμός αυτός ήταν στην αρχική σελίδα για το Semantic Web στην πύλη του W3C, τώρα μπορεί να βρεθεί στο <http://www.w3.org/2003/03/semantic-tour.html>). Εντυπωσιακό είναι επίσης και το παρακάτω, από την IBM αυτή τη φορά:

*The word "semantic" comes from the Greek semantikos, or "significant meaning," derived from sema, or "sign."*²

Προέρχεται δηλαδή “*from the Greek semantikos*” (*sic*) και όχι “*from the Greek semasiologikos*”! Εξάλλου, στην καθημερινή μας ομιλία λέμε π.χ. *ηλεκτρικό ρεύμα, κοινωνικός ιστός, ψυχικός κόσμος κλπ*, αντί για *ηλεκτρολογικό ρεύμα, κοινωνιολογικός ιστός, ψυχολογικός κόσμος!* Σκόπιμο είναι επίσης να αναφερθεί ότι η *σημαντική*, ως απόδοση του *semantics* έχει πιο ευθεία ιστορικά αντιστοιχία προς την ερμηνεία της, από ότι η *σημασιολογία*, που αποτελεί αντιδάνειο από τα γαλλικά (όπως η οντολογία αποτελεί αντιδάνειο από τα λατινικά, βλ. και λεξικό Μπαμπινιώτη).

¹ Tim Berners-Lee (2000). “Semantic Web”. XML 2000 Conference. <http://www.w3.org/2000/Talks/1206-xml2k-tbl/Overview.html>

² Naveen Balani. (2005). “The future of the Web is Semantic”. <http://www.ibm.com/developerworks/web/library/wa-semweb/>

Επιπλέον, όταν πρωτοχρησιμοποίησα τον όρο σε επίσημο κείμενο, στην πρόταση της μεταπτυχιακής μου εργασίας στο Τμήμα, εν έτει 2001, έγραφα *σημαντικός*, για τους ίδιους λόγους. Τότε, δεν υπήρχε η μετάφραση του βιβλίου του Berners Lee «Weaving the Web» (Tim Berners Lee, Mark Fischetti. «Υφαίνοντας τον Παγκόσμιο Ιστό», εκδόσεις Γκοβόστη, 2002). Τότε επίσης δεν είχε γίνει η περιοδεία για το Σημαντικό Ιστό, με ομιλίες και στην Αθήνα, το καλοκαίρι του 2003, οπότε και το ελληνικό γραφείο του W3C ξεκίνησε να μεταφράζει το Semantic Web ως Σημασιολογικός Ιστός. Το Google φυσικά, το 2001, δεν έβγαζε τίποτα («σελίδες από Ελλάδα»).

Είναι σαφές λοιπόν ότι ο μόνος λόγος για τον οποίο θα ήταν επιθυμητή η μετάφραση *σημασιολογικός*, είναι η προσπάθεια διαφύλαξης της *επιστημονικότητας* του νεολογισμού Semantic Web, ώστε να μην συγχέεται με την έννοια του σπουδαίου. Δυστυχώς, η προσπάθεια αυτή φαίνεται ότι αποκλίνει από τη λεξική ορθότητα και, εκτός του ότι αποδίδει το όρο με ξεκάθαρα λανθασμένη ερμηνεία, δεν καταφέρνει να πετύχει το στόχο της: Στην πραγματικότητα εκλαϊκεύει τον όρο, εισάγοντας, στην καλύτερη περίπτωση, άλλον ένα γλωσσικό σολοικισμό.

Υπάρχει ακόμη η περίπτωση των *semantic networks*, όπου, λόγω της πολυετούς καθιέρωσης του όρου, έχει επικρατήσει η απόδοση *σημασιολογικά* δίκτυα. Μήπως όμως αυτά σχετίζονται με «κλάδο της γλωσσολογίας»; Ο Ross Quillian, στον οποίο πολλοί αποδίδουν την επινόηση των σημασιολογικών δικτύων, είναι (επίτιμος) καθηγητής *πολιτικών επιστημών*. Όχι φυσικών, ούτε θετικών. Στο επάγγελμα είναι *ψυχολόγος*. Τα σημασιολογικά δίκτυα, όπως τα εισήγαγε στη διατριβή του,³ είναι ένας τρόπος για την αναπαράσταση της ανθρώπινης μνήμης, και συγκεκριμένα του μέρους εκείνου που σχετίζεται «με τη σημασία των λέξεων». Δεν είναι λοιπόν δίκτυα που απλώς φέρουν ορισμένη και ρητή σημασία, αλλά εργαλείο που μπορεί και να εμπίπτει στον αντίστοιχο κλάδο της γλωσσολογίας. Είναι επομένως η συγκεκριμένη περίπτωση τέτοια, που η απόδοση *σημασιολογικός* μπορεί να θεωρηθεί ανεκτή. Υπάρχουν βέβαια και περιπτώσεις, στην ακαδημαϊκή και ερευνητική κοινότητα, που τα δίκτυα αυτά αποδίνονται (ορθώς) και ως *σημαντικά*⁴.

Ο Σημαντικός Ιστός είναι λοιπόν ο Ιστός που φέρει καλώς ορισμένη *σημασία* και για αυτό, μπορεί να (βρεθεί ενδεχομένως ότι) *διέπεται* από μια σημασιολογία, όχι της φυσικής γλώσσας, αλλά τη σημασιολογία του Σημαντικού Ιστού. Με αυτόν τον τρόπο και για αυτούς τους λόγους υιοθετείται η παραπάνω απόδοση του όρου στη διατριβή. Για την αναφορά σε τεχνικές που εκμεταλλεύονται, βασίζονται ή κατά οιονδήποτε τρόπο σχετίζονται με αυτή τη σημασία που ο Σημαντικός Ιστός φέρει, όπως είναι οι τεχνικές που αναφέρονται στη διατριβή, μπορεί κάλλιστα να χρησιμοποιηθεί το επίθετο *σημασιακός* που, σύμφωνα με τα λεξικά είναι αυτός: *που αναφέρεται στη σημασία*.

Πάτρα, Μάιος 2008

³ Quillian, M. R. (1968). "Semantic Memory". In Minsky, M. (Ed.) (1968). *Semantic Information Processing*. Cambridge, Mass.: MIT Press.

⁴ Βλ. για παράδειγμα Ι. Χατζηλυγερούδης. «Τεχνητή Νοημοσύνη - Ύλη Μαθήματος». Διατμηματικό ΜΠΣ "Μαθηματικά των Υπολογιστών & των Αποφάσεων"
<http://mmlab.ceid.upatras.gr/aigroup/postgrad/ai/index.html>

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

Ευχαριστίες.....	xi
Αντί Προλόγου.....	xiii
Κατάλογος Εικόνων.....	xix
Κατάλογος Πινάκων.....	xxi
1. Εισαγωγή.....	1
1.1. Αντικείμενο της Διατριβής.....	1
1.1.1. Το πρόβλημα της Ανακάλυψης Γνώσης.....	1
1.1.2. Συνεισφορά και αποτελέσματα.....	2
1.2. Δομή της Διατριβής.....	5
1.3. Συναφείς δημοσιεύσεις.....	7
2. Αναπαράσταση Γνώσης με Οντολογίες.....	9
2.1. Εισαγωγή.....	9
2.2. Ορισμοί της Οντολογίας.....	10
2.2.1. Οντολογία κατά Gruber.....	10
2.2.2. Οντολογία κατά Guarino.....	11
2.3. Γλώσσες Αναπαράστασης Οντολογιών.....	12
2.3.1. KIF.....	13
2.3.2. OKBC.....	13
2.3.3. XOL.....	13
2.3.4. SHOE.....	14
2.3.5. OML.....	14
2.3.6. RDF(S).....	15
2.3.7. OIL.....	15
2.3.8. DAML+OIL.....	16
2.4. OWL.....	16
2.4.1. Γενική περιγραφή.....	17
2.4.2. Από το RDF(S) στην OWL.....	18
2.4.3. OWL και Λογικές Περιγραφές.....	19
2.5. OWL 1.1.....	19
2.5.1. Σημασιολογία.....	21
2.5.2. Νέα χαρακτηριστικά-Παραδείγματα.....	25
2.6. Βατά Υποσύνολα.....	29
2.6.1. EL++.....	31
2.6.2. DL-Lite.....	32
2.6.3. OWL-R.....	34
2.6.4. Υπολογιστικές Ιδιότητες.....	35

3.	Συλλογισμός Στο Σημαντικό Ιστό.....	37
3.1.	Εισαγωγή.....	37
3.2.	Φορμαλισμοί και Τεχνικές.....	38
3.2.1.	Ορισμοί.....	38
3.2.2.	Προγράμματα Λογικών Περιγραφής.....	40
3.2.3.	SWRL και DL-safe κανόνες.....	41
3.2.4.	Μηχανισμοί απόδειξης θεωρημάτων.....	42
3.3.	Λογικές Περιγραφές.....	43
3.3.1.	Συντακτικό και σημασιολογία.....	44
3.3.2.	Εκφραστική ισχύς και ονοματολογία.....	46
3.4.	Συλλογισμός με Λογικές Περιγραφές.....	47
3.4.1.	Υπηρεσίες συλλογισμού.....	48
3.4.2.	Συλλογισμός στο ABox.....	48
3.4.3.	Ξεδίπλωμα.....	49
3.4.4.	Εσωτερίκευση.....	50
3.5.	Αλγόριθμοι Ελέγχου Υπαγωγής.....	51
3.5.1.	Δομικοί αλγόριθμοι υπαγωγής.....	51
3.5.2.	Λογικοί αλγόριθμοι.....	52
3.5.3.	Αλγόριθμοι ταμπλό.....	53
3.5.4.	Συλλογισμός με απτά πεδία.....	55
3.6.	Υλοποίηση και Βελτιστοποίηση Αλγορίθμων Ταμπλό.....	58
3.6.1.	Λεξική κανονικοποίηση και απλοποίηση.....	59
3.6.2.	Υπαναχώρηση κατευθυνόμενη από εξάρτηση.....	59
3.6.3.	Απαλοιφή ABox.....	61
3.7.	Η Λογική Περιγραφής <i>SROIQ</i>	62
3.7.1.	Η δυσκολία του συλλογισμού.....	62
3.7.2.	Αλγόριθμοι ταμπλό για <i>SHOIQ</i> και <i>SROIQ</i>	64
3.7.3.	Υποθέσεις και πολυπλοκότητα.....	67
4.	Μεθοδολογία Ανακάλυψης γνώσης στο Σημαντικό Ιστό.....	70
4.1.	Εισαγωγή.....	70
4.2.	Προηγούμενη Εργασία.....	72
4.3.	Καταλληλότητα για τον Ιστό.....	73
4.3.1.	Η μέθοδος των συμπερασμών.....	73
4.3.2.	Συζευγμένα ερωτήματα.....	75
4.3.3.	Η SPARQL-DL.....	76
4.4.	Αξιολόγηση Μηχανισμών Συμπερασμού.....	78
4.4.1.	Cerebra.....	79
4.4.2.	FaCT.....	80
4.4.3.	FaCT++.....	82
4.4.4.	RACER.....	82
4.4.5.	Pellet.....	84
4.4.6.	KAON2.....	85
4.5.	Διαχείριση και Επερώτηση Οντολογικών Εγγράφων.....	86
4.5.1.	Sesame.....	87

4.5.2.	Jena.....	88
4.5.3.	OWLim	88
4.5.4.	Υποστήριξη Διαχρονικότητας και Περιορισμοί.....	90
4.6.	Αποδοτικότητα και Εκφραστικότητα	90
4.6.1.	Θεωρητικοί περιορισμοί-Πηγές πολυπλοκότητας.....	91
4.6.2.	Οργανωτικοί περιορισμοί	93
4.6.3.	Τεχνικοί περιορισμοί.....	96
4.7.	Πειραματική Επαλήθευση - KDI	97
4.7.1.	Γενική περιγραφή.....	98
4.7.2.	Αρχιτεκτονική	100
5.	Σημασιακή Προσαρμογή	102
5.1.	Εισαγωγή.....	102
5.2.	Προσαρμογή μεταδεδομένων εφαρμογής.....	104
5.3.	Το Εννοιολογικό Μοντέλο CIDOC.....	105
5.3.1.	Εννοιολογική δομή	106
5.3.2.	Υλοποιήσεις και εκφραστικότητα	107
5.3.3.	Σχετικές εφαρμογές και προσεγγίσεις	108
5.4.	Η προσαρμογή του CIDOC-CRM.....	109
5.4.1.	Συντακτικός μετασχηματισμός.....	109
5.4.2.	Σημασιακή εμβάθυνση και εκλέπτυνση	110
5.4.3.	Η τεχνική της σημασιακής προσαρμογής	113
5.5.	Αποτελέσματα	114
5.5.1.	Θετικοί συμπερασμοί.....	115
5.5.2.	Αρνητικοί συμπερασμοί.....	117
5.5.3.	Ένα σενάριο χρήσης	118
5.6.	Στοχευμένη προσαρμογή.....	120
6.	Σημασιακή Προσαρμογή και Ημιδομημένα Πεδία Γνώσης.....	123
6.1.	Εισαγωγή.....	123
6.2.	RDF υλοποιήσεις και σημασιολογία του Dublin Core.....	125
6.2.1.	DC και RDF	125
6.2.2.	Υλοποιήσεις σε OWL	127
6.2.3.	Άλλες προσεγγίσεις	127
6.3.	Δημιουργία οντολογίας OWL για το Dublin Core	128
6.3.1.	Ο αγώνας για αποφασιστικότητα	128
6.3.2.	Βήματα υλοποίησης	130
6.4.	Αποτελέσματα	133
6.4.1.	Τεχνική διαμόρφωση	134
6.4.2.	Διεξαγωγή επερωτήσεων.....	134
6.5.	Αναγωγή προφίλ μεταδεδομένων Dublin Core σε Λογικές Περιγραφές	136
6.5.1.	Σκοπός και αναγκαιότητα	137
6.5.2.	Η διαδικασία της αναγωγής.....	137

7. Συμπεράσματα-Μελλοντικές Κατευθύνσεις.....	140
7.1. Επισκόπηση και συνεισφορά.....	140
7.1.1. Αποδοτικότητα και εκφραστικότητα.....	140
7.1.2. Μοντελοποίηση πεδίου και περιγραφές.....	142
7.2. Μελλοντικές γραμμές έρευνας και εφαρμογές.....	143
7.2.1. Βελτίωση της επίδοσης.....	144
7.2.2. Εξελικτικότητα σημασιακών προφίλ.....	146
Βιβλιογραφία	148
Ευρετήριο.....	155
Γλωσσάρι.....	157

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 2.1: Σχηματική απεικόνιση του κατά Guarino ορισμού της οντολογίας.	11
Εικόνα 2.2: Η «πυραμίδα» των γλωσσών αναπαράστασης οντολογιών για το Σημαντικό Ιστό.	12
Εικόνα 3.1: Αλληλεπικάλυψη Λογικών Περιγραφής και LP.	40
Εικόνα 3.2: Λυγισμός κατά την αναζήτηση υπαναχώρησης.	60
Εικόνα 3.3: Κλάδεμα της αναζήτησης με χρήση άλματος προς τα πίσω.	60
Εικόνα 4.1: Φόρτωση και διαχείριση οντολογιών στο σύστημα Cerebra.	79
Εικόνα 4.2: Φόρτωση οντολογίας και υποβολή ερωτήματος στο σύστημα FaCT χρησιμοποιώντας την εφαρμογή πελάτη της διανομής.	81
Εικόνα 4.3: Υποβολή ερωτήματος στο σύστημα RACER μέσω της γραφικής διεπαφής RICE.	83
Εικόνα 4.4: Εμφάνιση πληροφοριών και εξήγηση ασυνέπειας από το Pellet, μέσα από την εφαρμογή OWLSight.	84
Εικόνα 4.5: Χρόνοι απόκρισης ερωτημάτων για KAON2, RACER και Pellet.	85
Εικόνα 4.6: Η αρχιτεκτονική του KAON2.	86
Εικόνα 4.7: Η αρχιτεκτονική του Sesame 2.0.	87
Εικόνα 4.8: Κατάταξη της εκφραστικότητας του OWLim.	89
Εικόνα 4.9: Η χρονική ιεραρχία των κλάσεων πολυπλοκότητας.	92
Εικόνα 4.10: Ο σχεδιασμός τριών επιπέδων της KDI.	99
Εικόνα 4.11: Η αρχιτεκτονική της KDI.	101
Εικόνα 5.1: Ένα ποιοτικό μετα-σχήμα του CIDOC-CRM [Doerr, 2003].	106
Εικόνα 5.2: Η ταξινόμια του CIDOC-CRM, όπως φαίνεται μέσα από την KDI.	107
Εικόνα 5.3: Η διαδικασία μετασχηματισμού του CIDOC-CRM.	109
Εικόνα 5.4: Η διαδικασία για την ανάπτυξη σημασιακών προφίλ εφαρμογής.	113
Εικόνα 5.5: Παράδειγμα συμπερασμού χρησιμοποιώντας περιορισμό τιμής.	116
Εικόνα 5.6: Παράδειγμα συμπερασμού με χρήση υπαρξιακής ποσοτικοποίησης και ονοματικών. ..	117
Εικόνα 5.7: Εντοπισμός μη ικανοποιήσιμων εννοιών.	118
Εικόνα 5.8: Ανάκτηση δικτυακού πόρου με χρήση ιδιότητας τύπων δεδομένων.	120
Εικόνα 5.9: Αλληλεπίδραση ονοματικών, περιορισμών αριθμού και αντίστροφων ρόλων.	121
Εικόνα 6.1: Λεπτομερής προβολή στοιχείου στο DSpace.	123
Εικόνα 6.2: Βήματα υλοποίησης της οντολογίας DC.	130
Εικόνα 6.3: Ερώτημα συντακτικού ταιριάσματος και αποτέλεσμα μέσω του Protégé 4.0.	135
Εικόνα 6.4: Ερώτημα Λογικών Περιγραφής και αποτέλεσμα, με χρήση αλυσίδων ρόλων.	135
Εικόνα 6.5: Ερώτημα Λογικών Περιγραφής και αποτέλεσμα, με χρήση περιορισμών πληθικότητας.	136
Εικόνα 7.1: Σχηματική απεικόνιση της σημασιακής σωλήνωσης.	145

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 2.1: Διερμήνευση των εκφράσεων ιδιότητας αντικειμένου.....	22
Πίνακας 2.2: Διερμήνευση των πεδίων τιμών δεδομένων.....	23
Πίνακας 2.3: Διερμήνευση περιγραφών.....	23
Πίνακας 2.4: Ικανοποίηση των αξιωμάτων σε μια διερμήνευση.....	24
Πίνακας 2.5: Χαρακτηριστικά της OWL επιθυμητά για την ταξινόμηση χημικών ενώσεων.....	25
Πίνακας 2.6: Συντακτικός περιορισμός αξιωμάτων SubClassOf στην OWL-R DL.....	35
Πίνακας 2.7: Η πολυπλοκότητα των προφίλ της OWL 2.....	36
Πίνακας 3.1: Εκφραστική δύναμη των Λογικών Περιγραφής.....	47
Πίνακας 3.2: Κανόνες επέκτασης ταμπλό για την ALC.....	54
Πίνακας 3.3: Επέκταση των κανόνων ταμπλό για υποστήριξη απτών πεδίων στην περίπτωση της SHOQ(D).....	57
Πίνακας 3.4: Κανόνες επέκτασης ταμπλό για τη SROIQ.....	66
Πίνακας 4.1: Παραδείγματα ερωτημάτων SPARQL-DL.....	77
Πίνακας 4.2: Συγκριτική αξιολόγηση μηχανισμών συμπερασμού βασισμένων σε Λογικές Περιγραφής.....	78
Πίνακας 4.3: Σύγκριση Jena, Sesame και OWLim.....	87
Πίνακας 4.4: Οι μη ντετερμινιστικοί κανόνες στο ταμπλό της SHOIQ.....	92
Πίνακας 4.5: Η πολυπλοκότητα του συλλογισμού στην OWL.....	93
Πίνακας 5.1: Εκφραστικά κέρδη της σημασιακής προσαρμογής του CIDOC-CRM ως προς την OWL.....	112
Πίνακας 5.2: «Τι είναι το De Stijl;».....	119
Πίνακας 5.3: «Ποια είναι τα μέλη του De Stijl;».....	119
Πίνακας 5.4: «Ποιο είναι το επάγγελμα του Mondrian;».....	119
Πίνακας 6.1: Αντιστοιχία μεταξύ DCAM και RDF(S).....	126

1. ΕΙΣΑΓΩΓΗ

1.1. Αντικείμενο της Διατριβής

Βασικός στόχος της διατριβής είναι να εξετάσει και να βελτιώσει τις δυνατότητες ανακάλυψης γνώσης που παρέχει ο Σημαντικός Ιστός, να αναγνωρίσει τους παράγοντες από τους οποίους οι δυνατότητες αυτές εξαρτώνται και να καταδείξει (με συγκεκριμένες νέες εφαρμογές) πώς μπορούν να αξιοποιηθούν, με αποτελεσματικό τρόπο, ακόμα και σε ρεαλιστικά σενάρια.

Μεθοδολογικά, η διατριβή ακολουθεί την πρακτική των *ερευνητικών υποθέσεων*: τίθενται δηλαδή ορισμένες υποθέσεις εργασίας και στη συνέχεια επιχειρείται ο έλεγχός της ορθότητάς τους, διερευνάται δηλαδή κατά πόσο επιβεβαιώνονται ή όχι. Οι υποθέσεις αυτές εκφράζουν τον πυρήνα του προβλήματος της ανακάλυψης γνώσης στο Σημαντικό Ιστό, όπως προσδιορίζεται στη συνέχεια.

Στην κατεύθυνση αυτή, κύρια εργαλεία αποτέλεσαν η συστηματική μελέτη και αξιολόγηση της σχετικής βιβλιογραφίας, η κατηγοριοποίηση και αποτίμηση, από θεωρητική και πρακτική σκοπιά, των τεχνικών που έχουν κατά καιρούς προταθεί, καθώς και ο εντοπισμός των επιμέρους προβλημάτων στα οποία η ανακάλυψη γνώσης στο Σημαντικό Ιστό μπορεί να αναλυθεί. Βάσει αυτών, στοιχειοθετούνται και εφαρμόζονται ορισμένες πρωτότυπες μέθοδοι, των οποίων η αποτελεσματικότητα επίσης ελέγχεται θεωρητικά και πειραματικά.

Στη συνέχεια, αρχικά προσδιορίζεται και τοποθετείται το πρόβλημα της ανακάλυψης γνώσης στο Σημαντικό Ιστό. Κατόπιν, παρατίθεται μια σύνοψη των αποτελεσμάτων που αποτελούν τη συνεισφορά τη διατριβής και φαίνεται πώς αυτά προκύπτουν και είναι οργανωμένα γύρω από τις βασικές ερευνητικές υποθέσεις. Ακολουθεί η περιγραφή της δομής της διατριβής, δίνοντας μια σύντομη περίληψη κάθε κεφαλαίου. Τέλος, δίνεται μια λίστα με τις κυριότερες και πιο συναφείς με τη διατριβή δημοσιεύσεις και ανακοινώσεις, στις οποίες ο γράφων υπήρξε ο κύριος ερευνητής.

1.1.1. Το πρόβλημα της Ανακάλυψης Γνώσης

Ο Σημαντικός Ιστός και το πλέγμα των τεχνολογιών του μπορεί να μην είναι ακριβώς η επανάκαμψη της τεχνητής νοημοσύνης στο χώρο του Διαδικτύου. Είναι όμως τουλάχιστον αξιοσημείωτο ότι συχνά παραγνωρίζεται η πιο αξιόλογη ίσως δυνατότητα που ο Σημαντικός Ιστός φέρει, η απόκτηση δηλαδή νέας πληροφορίας, μέσω της λογικής συνέπειας, από τον συνδυασμό ήδη δηλωμένων γεγονότων. Μάλιστα, η δυνατότητα αυτή δεν θα παρέχεται μόνο σε ένα στατικό περιβάλλον, όπως στα παραδοσιακά συστήματα τεχνητής νοημοσύνης, όπου οι βάσεις γνώσης είναι κεντρικές, αλλά η ανακάλυψη γνώσης θα είναι δυναμική και καταναμημένη, αξιοποιώντας πόρους που είναι διασκορπισμένοι στο Διαδίκτυο. Αυτός είναι και ο βασικός λόγος για τον οποίο η αύξηση γνώσης στο Σημαντικό Ιστό γίνεται *μονοτονικά*: Καινούριες δηλώσεις και γεγονότα μπορούν μόνο να προστεθούν, αυξάνοντας μονοτονικά την οντολογία. Τα νέα αυτά γεγονότα μπορούν να έρχονται σε

αντίθεση με προηγούμενα (καθιστώντας ενδεχομένως την οντολογία μη ικανοποιήσιμη), αλλά σε καμιά περίπτωση δεν μπορούν να τα αναιρέσουν ή να τα ακυρώσουν (βλ. για παράδειγμα, ενότητα 3.2).

Στη διατριβή αυτή εισάγεται επομένως η έννοια της **Ανακάλυψης Γνώσης στο Σημαντικό Ιστό (Semantic Web Knowledge Discovery, SWKD)**.

Η SWKD είναι προϊόν συλλογισμού που αξιοποιεί την υπάρχουσα πληροφορία (δηλαδή τις δηλώσεις και τα γεγονότα), η οποία (μπορεί να) βρίσκεται διασκορπισμένη σε οντολογικά έγγραφα και, με τρόπο παραγωγικό, οδηγεί στην εξαγωγή λογικών συμπερασμάτων τα οποία δεν είναι ρητά εκφρασμένα.

Στον παραπάνω ορισμό χρησιμοποιούμε τον όρο *ανακάλυψη*, εφόσον η γνώση υπονοείται, αλλά δεν είναι ορατή, χρησιμοποιούμε τον όρο *γνώση*, εφόσον το συμπέρασμα είναι προϊόν συνδυασμού πληροφοριών και κάνουμε λόγο για *παραγωγικό* τρόπο, σε αντιδιαστολή με τον *επαγωγικό*, αφού ο συμπερασμός ακολουθεί πορεία από το γενικό στο μερικό, εξάγοντας ασφαλή συμπεράσματα. Ο ορισμός είναι επίσης σε συμφωνία και με τους στόχους των απαρχών της καθιέρωσης οντολογιών Ιστού [Luke, Spector, & Rager, 1996] και των εφαρμογών τους [Shah, Finin, Joshi, Cost, & Mayfield, 2002]. Επιπλέον αντιδιαστέλλεται προς την πιο παραδοσιακή ιδέα της Ανακάλυψης Γνώσης σε Βάσεις Δεδομένων (KDD, Knowledge Discovery in Databases).

Η SWKD είναι λοιπόν διαφορετική από την *Ανάκτηση Πληροφορίας* (Information Retrieval) η οποία στοχεύει να βρει πληροφορίες που, ενώ είναι ρητά εκφρασμένες, είναι δύσκολο να ανακτηθούν από τη μηχανή. Επίσης διαφέρει από την *Εξόρυξη Δεδομένων* (Data Mining) που κυρίως ασχολείται με την ανακάλυψη επαναλαμβανόμενων μοτίβων (patterns) και έμμεσων συσχετίσεων σε μεγάλες βάσεις δεδομένων. Η SWKD εκμεταλλεύεται το σημασιακό υπόβαθρο και τις αντίστοιχες σημασιακές περιγραφές των πληροφοριών του Σημαντικού Ιστού, όπως αυτές είναι θεμελιωμένες στη λογική του θεωρία, και με τη χρήση των κατάλληλων μηχανισμών Τεχνητής Νοημοσύνης, δηλαδή των μηχανισμών συμπερασμού, μπορεί να συμπεράνει νέα, άδηλη γνώση, η οποία, μέχρι τότε, μόνο υπονοούνταν στα ήδη υπάρχοντα δεδομένα. Αν κάτι συνεισφέρει ο Σημαντικός Ιστός είναι τουλάχιστον αυτό, δηλαδή η ευκαιρία για ευφυή περιγραφή και, συνακόλουθα, ευφυή ανάκτηση των διαδικτυακών πόρων, βασιζόμενη όμως στις αντίστοιχες περιγραφές και τους αντίστοιχους μηχανισμούς της Τεχνητής Νοημοσύνης.

Το παραπάνω ισχύει κατ' αντιστοιχία και στις παραδοσιακές βάσεις δεδομένων: Η αναπαράσταση των πληροφοριών σε ένα σχεσιακό σχήμα μπορεί να έχει ένα σχετικό βαθμό δυσκολίας, που συνίσταται κυρίως στην αποτύπωση της πληροφορίας αυτής σε πίνακες, εγγραφές και στήλες. Αξίζει όμως, γιατί είναι αναγκαία και αποδεδειγμένα βοηθά όχι τόσο στη δόμηση, αλλά κυρίως στην αποτελεσματική ανάκτηση της πληροφορίας αυτής, μέσω της σχεσιακής άλγεβρας.

1.1.2. Συνεισφορά και αποτελέσματα

Στην ενότητα αυτή συνοψίζονται τα αποτελέσματα που απαρτίζουν την συνεισφορά της διατριβής. Για την κατανόησή τους δίνεται, στις δύο επόμενες παραγράφους, μια πολύ σύντομη περίληψη των εννοιών που χρησιμοποιούνται στην τεκμηρίωσή τους.

Οι οντολογίες μπορούν να ειπωθούν και ως σχήματα μεταδεδομένων τα οποία επιπλέον διέπονται από συσχετίσεις και είναι οργανωμένα σε μια ταξινόμια (κλάσεις και υποκλάσεις). Η υποκείμενη αυτή Λογική Θεωρία του Σημαντικού Ιστού αποτυπώνεται στην Γλώσσα Οντολογιών Ιστού OWL (Ontology Web Language). Η OWL πρόσφατα προτάθηκε να επεκταθεί στην έκδοση 1.1, ενσωματώνοντας μερικώς την εκφραστικότητα των λογικών κανόνων (Horn Clauses). Η σημασιολογία και εκφραστική δύναμη της OWL βασίζεται στη Λογική Πρώτης Τάξης και συγκεκριμένα σε ένα υποσύνολό της που καλείται Λογικές Περιγραφές. Από τον χώρο της Τεχνητής Νοημοσύνης υπάρχουν διάφορα συστήματα που μπορούν να επεξεργαστούν εκφράσεις Λογικών Περιγραφής και βάσει αυτών να βγάλουν συμπεράσματα. Τα λογισμικά αυτά είναι γνωστά ως Μηχανισμοί Συλλογισμού ή Συμπερασμού.

Μια εννοιολογική σύλληψη (conceptualization) αποτυπώνει ιδανικά μια ορισμένη εικόνα που έχουμε για τον κόσμο. Μια οντολογία μπορεί μόνο να προσεγγίσει αυτή την εννοιολογική σύλληψη, και μάλιστα οσοδήποτε κοντά [Guarino, 1998].

Η συνεισφορά της διατριβής συνοψίζεται στα παρακάτω σημεία (οι αριθμημένες αναφορές παραπέμπουν στην εργογραφία που παρατίθεται στην ενότητα 1.3) :

1. **Θεμελιώνεται η δυνατότητα συλλογισμού που παρέχει ο Σημαντικός Ιστός στις Λογικές Περιγραφές.** Από τον θεωρητικό υπαινιγμό στο [Horrocks & Patel-Schneider, 2003]¹ προχωρούμε στην μελέτη, δοκιμή και συγκριτική αξιολόγηση μηχανισμών συμπερασμού βασισμένων σε Λογικές Περιγραφές και εξετάζουμε εάν και πώς αυτοί μπορούν να χρησιμοποιηθούν και στην πράξη για την παροχή υπηρεσιών συλλογισμού στο Σημαντικό Ιστό. [4], [5]
2. Προτείνεται, δοκιμάζεται και τεκμηριώνεται μια πρότυπη **Μεθοδολογία Ανακάλυψης Γνώσης στο Σημαντικό Ιστό**. Η μεθοδολογία αυτή αποτελείται από τα βήματα: α) την επιλογή του κατάλληλου φορμαλισμού που θα επιτρέψει τη χρήση εργαλείων Τεχνητής Νοημοσύνης για διεξαγωγή συλλογισμού σε έγγραφα του Σημαντικού Ιστού, β) την ανάδειξη βασικών κριτηρίων που πρέπει να ληφθούν υπόψη, ώστε ο συλλογισμός να είναι κατάλληλος για το περιβάλλον του Ιστού και γ) την επιλογή ενός συγκεκριμένου μηχανισμού συμπερασμού που να ανταποκρίνεται στα κριτήρια που τέθηκαν στο προηγούμενο βήμα.

Η μεθοδολογία αυτή καταρτίζεται έχοντας ως απώτερο στόχο να επαληθεύσει (ή όχι) την παρακάτω υπόθεση εργασίας:

Υπόθεση 1: Οι τεχνολογίες και η λογική θεωρία του Σημαντικού Ιστού μπορούν να συνεισφέρουν αποδοτικά και εκφραστικά στο πρόβλημα της SWKD.

Πράγματι, η μεθοδολογία αυτή και τα κριτήρια που τίθενται σε κάθε στάδιο της, δείχνουν επακριβώς μέχρι πιο σημείο επαληθεύεται η παραπάνω υπόθεση. Συνδυάζοντας επομένως και τα αποτελέσματα από τα σημεία 1, 4, 5 μπορεί συμπερασματικά να αναφερθεί ότι οι παράγοντες που καθορίζουν το βαθμό επαλήθευσης της παραπάνω διττής υπόθεσης χωρίζονται σε τρεις κατηγορίες:

¹ Ότι δηλαδή μηχανισμοί συμπερασμού Λογικών Περιγραφής **θα μπορούσαν** να χρησιμοποιηθούν για συλλογισμό στην OWL DL.

- α. Θεωρητικοί (θεωρητικοί περιορισμοί, μη αποφασισιμότητα, νέοι αλγόριθμοι)
 - β. Οργανωτικοί (σχεδιασμός, πολιτικές και πρότυπα)
 - γ. Τεχνικοί (τεχνικοί περιορισμοί μηχανισμών, διεπαφές, επίδοση in-memory, APIs)
- [1], [2], [3]

3. Εισάγεται για πρώτη φορά η έννοια της **Σημασιακής Προσαρμογής**, η οποία εξελίσσει την *Προσαρμογή Μεταδεδομένων Εφαρμογής* (Metadata Application Profiling) από μια επίπεδη συρραφή και συγχώνευση σχημάτων και πεδίων μεταδεδομένων σε μία ουσιαστική επέκταση και σημασιακό εμπλουτισμό/αναγωγή του αντίστοιχου μοντέλου. Η υπόθεση που πρέπει να επαληθευτεί εδώ έχει ως εξής:

Υπόθεση 2: Η τεχνική της σημασιακής προσαρμογής επιτυγχάνει την εντασιακή επέκταση (εμβάθυνση) ενός μοντέλου, χωρίς όμως να βλάπτεται η γενική του εφαρμοσιμότητα.

Για την επαλήθευση της υπόθεσης αυτής, η ως άνω τεχνική εξελίσσεται χρησιμοποιώντας τις σημασιακές δομές του αρχικού μοντέλου ως βάση και παρέχοντας κατάλληλες εξειδικεύσεις, μέσω συγκεκριμένων κατασκευαστών της OWL, που επεκτείνουν και εξειδικεύουν σε βάθος το αντίστοιχο μοντέλο. Το ζητούμενο είναι:

- α. χρήση των πιο εκφραστικών δομών της OWL, ώστε να προσεγγίζεται καλύτερα η εννοιολογική σύλληψη,
- β. διαχωρισμένη περαιτέρω εξειδίκευση των υπαρχουσών και νέων δομών, ώστε να αποτυπωθεί η συγκεκριμένη εννοιολογική σύλληψη της συγκεκριμένης εφαρμογής.

Η πειραματική επαλήθευση της υπόθεσης έρχεται μέσω της εφαρμογής της τεχνικής στο μοντέλο CIDOC-CRM. [7], [8], [9]

4. Επαληθεύεται και πειραματικά η Μεθοδολογία Ανακάλυψης Γνώσης μέσω της υλοποίησης και ανάπτυξης μιας πρότυπης **Διεπαφής Ανακάλυψης Γνώσης (KDI)**. Μια τέτοια πραγματικά δικτυακά κατανεμημένη αρχιτεκτονική, κατ' αναλογία του παραδοσιακού 3-tier μοντέλου, δεν είναι ακόμα, όπως φαίνεται και από το σημείο 1, εφικτή με την πλειοψηφία των πλέον πρόσφατων και εκφραστικών μηχανισμών Λογικών Περιγραφής (λόγω περιορισμών στις δυνατότητες διεπαφής τους). Με τη χρήση της διεπαφής αυτής παρέχονται μια σειρά από πειραματικά αποτελέσματα πάνω στο προσαρμοσμένο μοντέλο CIDOC-CRM τα οποία επαληθεύουν, μέχρι ορισμένο βαθμό, τις υποθέσεις που έχουν γίνει σχετικά κυρίως με την παράμετρο της *εκφραστικότητας* και δίνουν το έναυσμα για την αναζήτηση και εξέταση της υποστήριξης των νέων προτεινόμενων επεκτάσεων της λογικής θεωρίας του Σημαντικού Ιστού, δηλαδή της γλώσσας OWL 1.1. [1], [2], [3], [6]
5. Η μεθοδολογία ανακάλυψης γνώσης, καθώς και η τεχνική της σημασιακής προσαρμογής που έχει εισαχθεί στο σημείο 3, συνδυάζονται και εφαρμόζονται σε ένα υπαρκτό παράδειγμα παραγωγικής εφαρμογής: Στο σύστημα του Εκπαιδευτικού Αποθέματος του Πανεπιστημίου Πατρών. Το πρόβλημα εντοπίζεται στο γεγονός ότι, στην συγκεκριμένη περίπτωση, υπάρχουν ήδη διαθέσιμα μεταδεδομένα, διαχειριζόμενα από ένα ΣΔΒΔ, τα οποία όμως έχουν ελάχιστη σημασιακή αξία, καθώς είναι οργανωμένα καθόλου ή ελάχιστα σύμφωνα με μια λογική θεωρία (έχουν επίπεδη δόμηση). Ωστόσο το υποκείμενο σχήμα μεταδεδομένων Dublin Core υπονοεί μια

ενοιολογική σύλληψη, ασχέτως αν αυτή δεν είναι επαρκώς εκφρασμένη τεχνικά. Άρα λοιπόν το ζητούμενο είναι να επαληθευθούν οι παρακάτω υποθέσεις:

Υπόθεση 1^α: *Η SWKD μπορεί να εφαρμοστεί αποδοτικά και εκφραστικά στην περίπτωση σημασιακά ημιδομημένων μοντέλων γνώσης,*

που εν μέρει ανάγεται στην

Υπόθεση 2^α: *Η τεχνική της σημασιακής προσαρμογής επιτυγχάνει την εντασιακή αποτύπωση και επέκταση ενός σημασιακά ημιδομημένου μοντέλου, χωρίς να βλέπεται η γενική του εφαρμοσιμότητα.*

[10]

6. Παραπλεύρως προς τις βασικές γραμμές που περιγράφονται πιο πάνω, εισάγονται επίσης δύο σημαντικά υποπροϊόντα έρευνας:

Πρώτον, ότι η σημασιακή προσαρμογή, όπως εισάγεται στο σημείο 3 και εφαρμόζεται στο μοντέλο CIDOC-CRM είναι *στοχευμένη*: Αυτό σημαίνει ότι πετυχαίνουμε την αναγωγή της εκφραστικότητας του μοντέλου σε επίπεδο OWL DL με τέτοιο τρόπο, ώστε η πολυπλοκότητα του συλλογισμού να είναι η καλύτερη δυνατή. Πράγματι, αποφεύγεται ο συνδυασμός συγκεκριμένων εκφραστικών κατασκευαστών της γλώσσας, οι οποίοι οδηγούν σε αύξηση της πολυπλοκότητας. Παρόλο που αναφέρεται ότι τέτοιοι συνδυασμοί είναι σπάνιοι σε ρεαλιστικές οντολογίες [Tsarkov, Horrocks, & Patel-Schneider, 2007], δίνουμε συγκεκριμένο *αντιπαράδειγμα* όπου η αλληλεπίδραση αυτή είναι απαραίτητη για την αναπαράσταση εννοιών που εντάσσονται με φυσικό τρόπο στο μοντέλο αναφοράς.

Δεύτερον, δεδομένης της συνάφειας της εργασίας με τα προφίλ μεταδεδομένων εφαρμογής, διερευνούμε τη δυνατότητα αναπαράστασης και κλασσικών (δηλ. όχι σημασιακά προσαρμοσμένων) προφίλ μεταδεδομένων, όπως πολύ πρόσφατα επιχειρείται να τυποποιηθούν από το DCMI [Nilsson, Baker, & Johnston, 2008], με τη γλώσσα OWL. Δίνουμε την αναγωγή της πρότασης αυτής σε Λογικές Περιγραφής και αποδεικνύουμε παραδόξως ότι η OWL (1.1) με την τρέχουσα εκφραστικότητα, δεν επαρκεί για το σκοπό αυτό.

1.2. Δομή της Διατριβής

Το υπόλοιπο της διατριβής είναι οργανωμένο ως ακολούθως:

Αντικείμενο του κεφαλαίου 2 αποτελούν οι οντολογίες και η αναπαράσταση γνώσης που επιτυγχάνεται μέσα από αυτές. Ύστερα από μια σύντομη αναφορά στους ορισμούς της οντολογίας που υιοθετούνται στη διατριβή, γίνεται μια επισκόπηση των κυριότερων γλωσσών που έχουν προταθεί για την αναπαράσταση οντολογιών σε μηχανικά αναγνώσιμη μορφή. Η επισκόπηση αυτή εστιάζει τελικά στην OWL και πιο συγκεκριμένα στην πρόσφατη προδιαγραφή της OWL 1.1 (ή 2). Δίνεται η τυπική σημασιολογία της γλώσσας και αναδεικνύονται τα χαρακτηριστικά της εκείνα που τη διαφοροποιούν από τις προηγούμενες προτάσεις, για κάθε ένα από τα οποία παρατίθενται συγκεκριμένα

παραδείγματα. Ενδιαφέρον έχει επίσης η εξέταση των λεγόμενων *βατών τεμαχίων* της OWL 1.1 με την οποία ολοκληρώνεται το κεφάλαιο, αφού αυτά απολαμβάνουν συνήθως καλύτερες υπολογιστικές ιδιότητες.

Το *κεφάλαιο 3* επικεντρώνει στη σημασία, τις δυνατότητες και τις υποδομές για συλλογισμό με οντολογίες του Σημαντικού Ιστού. Παρόλο που η σχέση των οντολογιών OWL με τις Λογικές Περιγραφές είναι ξεκάθαρη, αρχικά γίνεται μια εισαγωγή στους εναλλακτικούς φορμαλισμούς, σχετικούς κυρίως με Λογική Πρώτης Τάξης και στους τρόπους με τους οποίους μπορούν να αξιοποιηθούν για την παροχή υπηρεσιών συλλογισμού σε υποσύνολα ή υπερσύνολα της OWL. Οι Λογικές Περιγραφές παραμένουν το πιο πρόσφορο θεωρητικό υπόβαθρο για συλλογισμό, ιδιαίτερα με πολύ εκφραστικές οντολογίες, και για αυτό περιγράφονται στη συνέχεια οι τεχνικές συλλογισμού σε αυτές τις λογικές. Εξηγούνται επομένως οι αλγόριθμοι ταμπλό και αναφέρονται ορισμένες συχνά υλοποιούμενες μέθοδοι για τη βελτιστοποίησή τους, κάτι που ήδη υπονοεί την υψηλή πολυπλοκότητα των αλγορίθμων αυτών. Η OWL 1.1 έχει το αντίστοιχό της σε Λογικές Περιγραφές στη λογική *SROIQ*, για την οποία εξηγείται γιατί είναι τόσο δύσκολος ο συλλογισμός, δίνονται οι πρόσφατοι αλγόριθμοι ταμπλό που έχουν προταθεί στη βιβλιογραφία και γίνεται προσπάθεια να αναδειχθούν οι απλοποιήσεις που δέχονται οι αλγόριθμοι και να εκτιμηθεί η πολυπλοκότητά τους.

Το *κεφάλαιο 4* είναι αφιερωμένο στην προσπάθεια ανάδειξης μιας μεθοδολογίας ανακάλυψης γνώσης στο Σημαντικό Ιστό. Τα στάδια της μεθοδολογίας είναι καταρτισμένα με στόχο να διερευνηθούν οι αναγκαίες εκείνες προϋποθέσεις και τα προβλήματα που πρέπει να απαντηθούν, προκειμένου για το σχεδιασμό αποδοτικών και εκφραστικών μεθόδων και εφαρμογών ανακάλυψης γνώσης στο περιβάλλον του (Σημαντικού) Ιστού. Προς το σκοπό αυτό επιχειρείται μία διεξοδική συγκριτική αξιολόγηση μιας σειράς υλοποιημένων μηχανισμών συμπερασμού, κυρίως βασισμένων σε Λογικές Περιγραφές, οι οποίοι μπορούν να χρησιμοποιηθούν ως βάση για την παροχή ευφύων υπηρεσιών. Παράλληλα, εξετάζονται και ορισμένα περιβάλλοντα τα οποία επιδεικνύουν χαρακτηριστικά επιθυμητά σε ένα παραγωγικό σύστημα, όπως το χαρακτηριστικό της διαχρονικότητας. Όπως είναι διαμορφωμένη η κατάσταση, οι παράμετροι της αποδοτικότητας και της εκφραστικότητας κάθε μεθόδου ανακάλυψης γνώσης στο Σημαντικό Ιστό φράσσονται από συγκεκριμένους περιορισμούς, σχετικούς με τα θεωρητικά όρια της πολυπλοκότητας του συλλογισμού, αλλά και με παράγοντες οργανωτικής φύσεως, καθώς και με τεχνικές αδυναμίες. Το κεφάλαιο ολοκληρώνεται με την περιγραφή της υλοποίησης της Διεπαφής Ανακάλυψης Γνώσης (KDI).

Η τεχνική της σημασιακής προσαρμογής αποτελεί το αντικείμενο του *κεφαλαίου 5*. Αρχικά, εισάγεται το μοντέλο CIDOC-CRM, η τρέχουσα σημασιολογία του και οι μηχανικά αναγνώσιμες υλοποιήσεις. Στη συνέχεια εφαρμόζεται η τεχνική πάνω στο μοντέλο αυτό, πρώτα ως προς τον συντακτικό του μετασχηματισμό και μετά ως προς την εντασιακή εμπάθυνση του μοντέλου και την εκλέπτυνσή του για μια συγκεκριμένη εφαρμογή, διατηρώντας ταυτόχρονα τη διαλειτουργικότητα. Η τεχνική γενικεύεται ως αυτόνομη και επαναλήψιμη προσέγγιση και παρατίθενται τα αποτελέσματα της εφαρμογής της στο CIDOC-CRM, μέσω μιας σειράς συμπερασμών, με χρήση της KDI και ενός ενδεικτικού σεναρίου χρήσης. Παρά την υψηλή πολυπλοκότητα που έχει κατά κανόνα ο συλλογισμός, η τεχνική επιτυγχάνει την αύξηση της εκφραστικότητας του μοντέλου με το μικρότερο δυνατό κόστος.

Το κεφάλαιο 6 πραγματεύεται την εφαρμογή της τεχνικής της σημασιακής προσαρμογής σε ημιδομημένα πεδία γνώσης, όπως είναι συνήθως οι επίπεδες υλοποιήσεις των μεταδεδομένων. Η βαρύτητα δίνεται στο σχήμα μεταδεδομένων Dublin Core (DC), όπως εφαρμόζεται στο εκπαιδευτικό απόθεμα του Πανεπιστημίου Πατρών, και, μέσω της τεχνικής, δημιουργείται μια οντολογία για το DC, επιπέδου OWL 1.1, εκλεπτύνοντας και επαυξάνοντας το αρχικό ημιδομημένο μοντέλο. Η σκοπιμότητα της προσέγγισης τεκμηριώνεται μέσω της διεξαγωγής μιας σειράς εκφραστικών συμπερασμών πάνω στη νέα οντολογία, η οποία ενημερώνεται από τα ήδη υπάρχοντα μεταδεδομένα χωρίς χειρωνακτική παρέμβαση. Παραπλεύρως, δείχνεται ότι τα κλασικά προφίλ εφαρμογής μεταδεδομένων, όπως επιχειρείται πρόσφατα να τυποποιηθούν, δεν είναι συμβατά εκφραστικώς ούτε με την OWL 1.1 και εξακριβώνονται τα χαρακτηριστικά εκείνα που απουσιάζουν, καθώς και η επίδρασή τους στην πολυπλοκότητα.

Τέλος, το κεφάλαιο 7 αποκρυσταλλώνει τα συμπεράσματα και σκιαγραφεί τις μελλοντικές ερευνητικές κατευθύνσεις που προκύπτουν από τη διατριβή.

1.3. Συναφείς δημοσιεύσεις

Στη συνέχεια παρατίθεται η πιο συναφής με τη διατριβή εργογραφία, όπου ο γράφων υπήρξε βασικός ερευνητής.

- [1] Dimitrios A. Koutsomitropoulos, Markos F. Fragakis and Theodore S. Papatheodorou. A Methodology for Conducting Knowledge Discovery on the Semantic Web. In S. Sirmakessis (Ed.), Adaptive and Personalized Semantic Web, Studies In Computational Intelligence (14), pp. 95-105. Springer, 2006.
- [2] Dimitrios A. Koutsomitropoulos, Markos F. Fragakis and Theodore S. Papatheodorou. A Methodology for Conducting Knowledge Discovery on the Semantic Web. In Proc. of 16th ACM Conference on Hypertext and Hypermedia (Hypertext 2005)-International Workshop on Adaptive and Personalized Semantic Web, 2005.
- [3] Dimitrios A. Koutsomitropoulos, Markos F. Fragakis and Theodore S. Papatheodorou. Discovering Knowledge in Web Ontologies: A Methodology and Prototype Implementation. In Proc. of SEMANTICS 2006 International Conference (SEMANTICS 06), pp.151-164, 2006.
- [4] Dimitrios A. Koutsomitropoulos, Dimitrios P. Meidanis, Anastasia N. Kandili and Theodore S. Papatheodorou. Establishing the Semantic Web Reasoning Infrastructure on Description Logic Inference Engines. In Y. Manolopoulos, J. Filipe, P. Constantopoulos and J. Cordeiro (Eds.), Enterprise Information Systems VIII, Lecture Notes in Business Information Processing 3 (5). Springer, 2008.
- [5] Dimitrios A. Koutsomitropoulos, Dimitrios P. Meidanis, Anastasia N. Kandili and Theodore S. Papatheodorou. OWL-based Knowledge Discovery Using Description Logics Reasoners. In Proc. of 8th International Conference on Enterprise Information Systems (ICEIS 2006). SAIC track, pp.43-50, 2006.
- [6] Dimitrios A. Koutsomitropoulos and Theodore S. Papatheodorou. Expressive Reasoning about Cultural Heritage Knowledge Using Web Ontologies. In Proc. of 3d International Conference on Web Information Systems and Technologies (WEBIST 07). WIA track, pp.276-281, 2007.
- [7] Dimitrios A. Koutsomitropoulos, George E. Paloukis and Theodore S. Papatheodorou. From Metadata Application Profiles to Semantic Profiling: Ontology Refinement and Profiling to Strengthen Inference-based Queries on the Semantic Web. Int. J. on Metadata, Semantics and Ontologies, to appear.
- [8] Dimitrios A. Koutsomitropoulos, George E. Paloukis and Theodore S. Papatheodorou. Ontology-based Knowledge Acquisition through Semantic Profiling. An Application to the Cultural Heritage Domain. In Proc. of the 2nd International Conference on Metadata and Semantics Research (MTSR 2007), CD-ROM.

- [9] Dimitrios A. Koutsomitropoulos, George E. Paloukis and Theodore S. Papatheodorou. Semantic Application Profiles: A Means to Enhance Knowledge Discovery in Domain Metadata Models. In M. A. Sicilia and M. Lytras (Eds.), *Metadata and Semantics, Lecture Notes in Computer Science*. Springer, to appear.
- [10] Dimitrios A. Koutsomitropoulos, Georgia D. Solomou and Theodore S. Papatheodorou. Towards a Practical Dublin Core Ontology for the Semantic Web. 2008 Semantic Web Int. Conference (ISWC 2008), submitted.

2. ΑΝΑΠΑΡΑΣΤΑΣΗ ΓΝΩΣΗΣ ΜΕ ΟΝΤΟΛΟΓΙΕΣ

2.1. Εισαγωγή

Η πληροφορία και κατ' επέκταση, η γνώση που αποκτούμε για το πραγματικό ή φανταστικό περιβάλλον είναι πραγματικά μακραίωνα ζητούμενο το πώς αποτυπώνεται. Η πληροφορία συγκεκριμένα περί του υπαρκτού, περί του όντος, ο «λόγος» δηλαδή «περί του όντος», όπως για πρώτη φορά στην ιστορία ορίζεται στα αριστοτέλεια κείμενα, συνιστά την οντολογία:

Οντολογία είναι η οριστική και εξαντλητική ταξινόμηση των οντοτήτων σε όλες τις σφαίρες της ύπαρξης, συμπεριλαμβανομένων των ειδών των σχέσεων με τις οποίες συνδέονται [Smith & Welty, 2001].

Η ανάγκη βέβαια για μια τέτοια αποτύπωση είναι εμφανής: α) διασώζει τη μέχρι τώρα κερδηθείσα γνώση β) είναι το πρώτο βήμα για το διαμοιρασμό και την κοινοποίηση της γνώσης αυτής και γ) αποτελεί τη βάση για τη δημιουργία νέας γνώσης. Για παράδειγμα, η εφεύρεση της τυπογραφίας εξυπηρετεί αυτούς τους σκοπούς. Αξιοσημείωτο είναι όμως πως, ακριβώς οι τρεις αυτοί λόγοι είναι που οδηγούν και επιβάλλουν την αναπαράσταση της πληροφορίας και σε έναν ηλεκτρονικό υπολογιστή.

Ένας πραγματικά πολύσημος και γενικός όρος όπως κυρίως η φιλοσοφική του πορεία έχει δείξει, η οντολογία είναι σήμερα μια συνήθης λέξι-κλειδί στα επηρεασμένα από το Σημαντικό Ιστό πληροφοριακά περιβάλλοντα. Χωρίς να παραγνωρίζουμε την πρωτοκαθεδρία του σχεσιακού μοντέλου για αναπαράσταση πληροφορίας σε υπολογιστικά συστήματα (το οποίο άλλωστε ως ένα βαθμό μπορεί να αναχθεί σε οντολογία Ιστού), καθώς και τις φιλοσοφικές προεκτάσεις του όρου, στο κεφάλαιο αυτό της διατριβής δεχόμαστε την υπόθεση ότι:

Το μέσο για την αποτύπωση της γνώσης στο περιβάλλον του Παγκόσμιου Ιστού είναι οι Οντολογίες Ιστού.

Η υπόθεση αυτή επιτυγχάνει να παρακάμψει το φιλοσοφικό πρόβλημα της οντολογίας και ταυτόχρονα εισάγει την ιδιαιτερότητα της αναπαράστασης γνώσης στον Παγκόσμιο Ιστό, παίρνοντας έτσι αποστάσεις από άλλες μεθόδους αναπαράστασης, δόκιμες σε πληροφοριακά συστήματα. Για μια διεξοδικότερη συζήτηση σχετικά με τις χρήσεις, τους ορισμούς και τις αρχές της οντολογίας ο αναγνώστης παραπέμπεται στο [Κουτσομητρόπουλος, 2004].

Το παρόν κεφάλαιο εστιάζει επομένως στις αναγνώσιμες από τον υπολογιστή γλώσσες αναπαράστασης οντολογιών για τον Παγκόσμιο Ιστό και συγκεκριμένα στην OWL (Γλώσσα Οντολογιών Ιστού, *Ontology Web Language*). Πρώτα γίνεται μια συνοπτική παρουσίαση των γλωσσών αναπαράστασης οντολογιών που έχουν προταθεί κατά καιρούς για μηχανική χρήση. Στη συνέχεια μελετάται η πιο πρόσφατη πρόταση του W3C (Κοινοπραξία για τον Παγκόσμιο Ιστό, *World Wide Web Consortium*), η OWL 1.1, η σημασιολογία της και οι

πρόσθετες ιδιότητες που φέρει, όπως η παρονομασία. Η γλώσσα αυτή συγκρίνεται με τις προκατόχους της, OWL (1.0) και RDF(S) και εντοπίζονται τα οφέλη που θα προκύψουν από την υιοθέτησή της σε ένα σενάριο ανακάλυψης γνώσης στον Ιστό. Τέλος εξετάζονται τα λεγόμενα *επιλύσιμα* (ή *βατά*) *τεμάχια* (tractable fragments) της γλώσσας και η σχέση τους με την OWL 1.1. Αρχικά όμως γίνεται μια πολύ σύντομη παρουσίαση των δύο συνηθέστερων ορισμών της (πληροφοριακής) οντολογίας, καθότι θα φανούν χρήσιμοι και στα επόμενα κεφάλαια και ιδιαίτερα στο κεφάλαιο 5, όπου εισάγεται η έννοια της *σημασιακής προσαρμογής*.

2.2. Ορισμοί της Οντολογίας

2.2.1. Οντολογία κατά Gruber

Ίσως ο πιο διαδεδομένος και αξιόπιστος ορισμός της οντολογίας, όσον αφορά τη χρήση της στην επιστήμη των υπολογιστών εν γένει, είναι αυτός που δόθηκε πρώτα από τον Gruber [Gruber, 1993a; Gruber, 1993b]:

Οντολογία είναι ο προσδιορισμός μιας εννοιολογικής σύλληψης.

Μια *εννοιολογική σύλληψη* (conceptualization) ορίζεται ως το σύνολο των αντικειμένων, των εννοιών και άλλων οντοτήτων που θεωρείται ότι υπάρχουν σε μια περιοχή ενδιαφέροντος και των μεταξύ τους σχέσεων. Ουσιαστικά μια εννοιολογική σύλληψη είναι μια αφηρημένη, απλοποιημένη εικόνα του κόσμου που επιθυμούμε να αναπαραστήσουμε για κάποιο συγκεκριμένο σκοπό. Κάθε βάση γνώσης, κάθε σύστημα βασισμένο σε γνώση και κάθε ευφυής πράκτορας *δεσμεύεται* προς μια τέτοια εννοιολογική σύλληψη, άμεσα ή έμμεσα.

Όταν η γνώση ενός πεδίου αναπαρίσταται με έναν δηλωτικό φορμαλισμό, το σύνολο των αντικειμένων που μπορούν να αναπαρασταθούν (το τι «υπάρχει» ως προς τα συστήματα Τεχνητής Νοημοσύνης) ονομάζεται *σύμπαν αναφοράς*. Το σύνολο αυτό των αντικειμένων και οι περιγραφόμενες μεταξύ τους σχέσεις αποτυπώνονται στο λεξιλόγιο αναπαράστασης με το οποίο ένα πρόγραμμα αναπαριστά τη γνώση. Έτσι, στο πεδίο της Τεχνητής Νοημοσύνης, η οντολογία ενός προγράμματος μπορεί να προσδιοριστεί ως το σύνολο των όρων αναπαράστασης. Σε μια τέτοια οντολογία υπάρχουν ορισμοί, που συσχετίζουν τα ονόματα των οντοτήτων του σύμπαντος αναφοράς (όπως κλάσεις, σχέσεις, συναρτήσεις ή άλλα αντικείμενα) με τις εκφράσεις που περιγράφουν το τι σημαίνουν αυτά τα ονόματα, καθώς και τυπικά αξιώματα, που περιορίζουν τη διερμηνευση και την ορθή χρήση αυτών των όρων.

Ένας πράκτορας *δεσμεύεται* προς μια οντολογία, όταν οι παρατηρούμενες ενέργειές του είναι συνεπείς προς τους ορισμούς της οντολογίας. Πρακτικά, μια οντολογία ορίζει το λεξιλόγιο με το οποίο οι πράκτορες ανταλλάσσουν δηλώσεις και ερωτήματα, ενώ *οντολογικές δεσμεύσεις* είναι συμφωνίες να χρησιμοποιηθεί το κοινό λεξιλόγιο με συνεκτικό και συνεπή τρόπο.

Ο κατά Gruber ορισμός επομένως ορίζει την οντολογία σε στενή σχέση με τη φιλοσοφική της σημασία, ως την απτή αποτύπωση μιας αντίληψης για ένα σύμπαν αναφοράς, αρκετά

συγκεκριμενοποιημένης και περιορισμένης από τυπικά αξιώματα, αλλά και αρκετά γενικής για να είναι χρήσιμη.

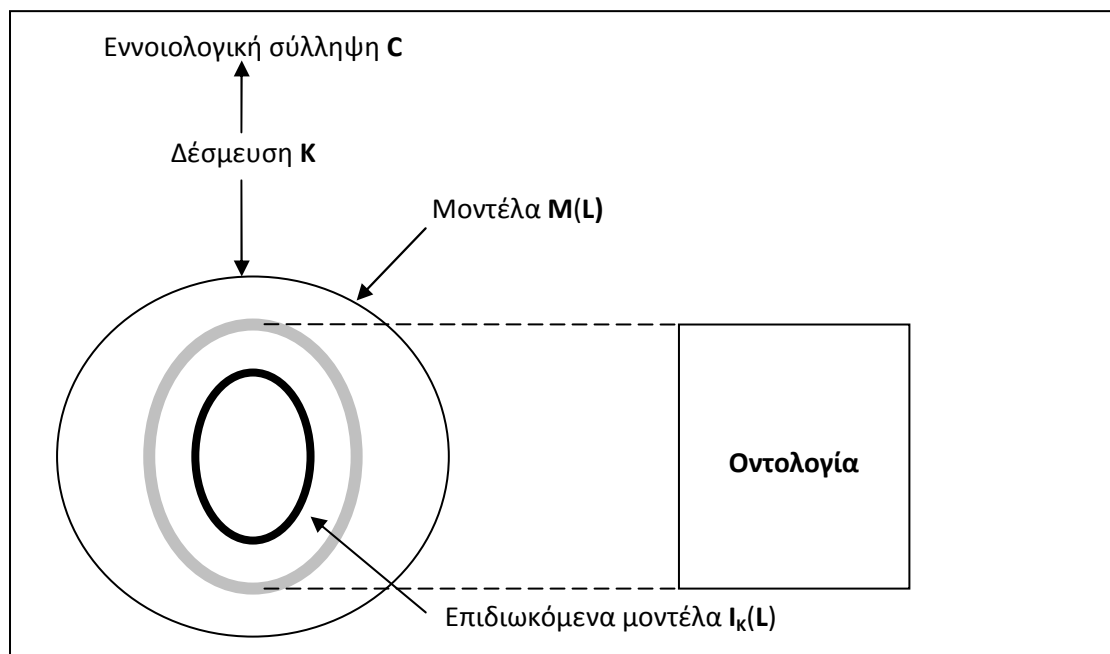
2.2.2. Οντολογία κατά Guarino

Όμως, όπως φάνηκε, το πρόβλημα με τον ορισμό του Gruber ήταν ότι άφηνε χώρο για πολλές πιθανές ερμηνείες, που οδηγούσαν πολλές φορές σε κατάχρηση του όρου, καθώς εκφραζόταν με ιδιαίτερα γενικό τρόπο. Στο [Guarino, 1998] ο Guarino επιχειρεί να εκλεπτύνει τον παραπάνω ορισμό, απομακρύνοντας περισσότερο την οντολογία από την εννοιολογική σύλληψη, διότι μόνο προσεγγιστικά μπορεί η πρώτη να εξηγήσει τη δεύτερη:

Μια οντολογία είναι μια λογική θεωρία που ερμηνεύει την επιδιωκόμενη σημασία ενός τυπικού λεξιλογίου, δηλαδή την οντολογική του δέσμευση προς μια συγκεκριμένη εννοιολογική σύλληψη του κόσμου. Τα επιδιωκόμενα μοντέλα μιας λογικής γλώσσας που χρησιμοποιούν ένα τέτοιο λεξιλόγιο περιορίζονται από αυτήν την οντολογική δέσμευση. Μια οντολογία αποτυπώνει έμμεσα αυτή τη δέσμευση (και την υποκείμενη εννοιολογική σύλληψη), με το να προσεγγίζει αυτά τα επιδιωκόμενα μοντέλα.

Ένα μοντέλο μιας λογικής γλώσσας είναι μια ερμηνεία του λεξιλογίου της γλώσσας, δηλαδή ένα σύνολο συσχετισμών των εννοιών και των αντικειμένων του κόσμου με τους όρους του λεξιλογίου. Τα επιδιωκόμενα μοντέλα είναι τα μοντέλα εκείνα που είναι συμβατά με την οντολογική δέσμευση, είναι δηλαδή ερμηνείες του λεξιλογίου που ανταποκρίνονται στη συγκεκριμένη εννοιολογική σύλληψη.

Τα επιδιωκόμενα λοιπόν μοντέλα μιας λογικής γλώσσας αποτυπώνουν άμεσα (επακριβώς) τη δέσμευσή της προς μια εννοιολογική σύλληψη. Μια οντολογία αποτυπώνει μόνο έμμεσα αυτή τη δέσμευση προσεγγίζοντας τα επιδιωκόμενα μοντέλα (βλ. εικόνα 2.1).



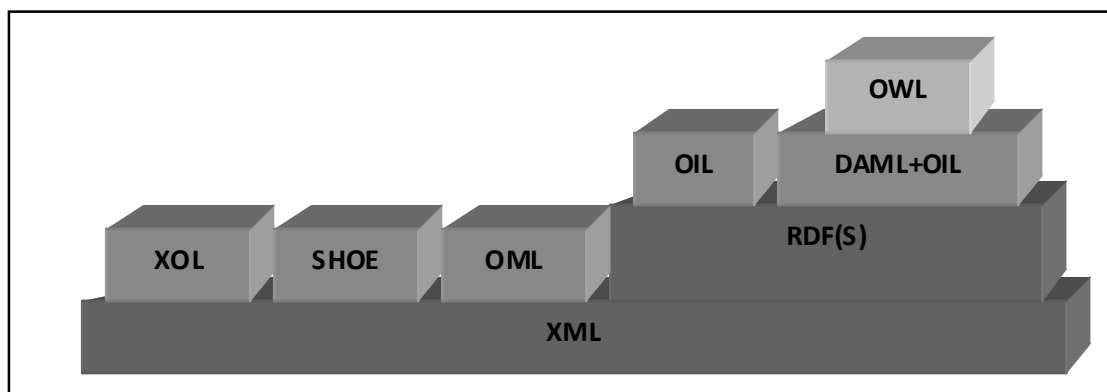
Εικόνα 2.1: Σχηματική απεικόνιση του κατά Guarino ορισμού της οντολογίας.

2.3. Γλώσσες Αναπαράστασης Οντολογιών

Στα πλαίσια του Σημαντικού Ιστού οι οντολογίες αναμένεται να παίξουν σημαντικό ρόλο στο να βοηθούν αυτοματοποιημένες διαδικασίες («ευφυείς πράκτορες») να προσπελάζουν πληροφορίες. Συγκεκριμένα οι οντολογίες αναμένεται να χρησιμοποιηθούν για την παροχή δομημένων λεξιλογίων που θα εξηγούν τις σχέσεις μεταξύ διαφορετικών όρων, επιτρέποντας έτσι στους ευφυείς πράκτορες (αλλά και στους ανθρώπους) να ερμηνεύουν τη σημασία τους ευέλικτα, αλλά και χωρίς αμφισημία.

Οι όροι των οποίων η σημασία ορίζεται σε οντολογίες, μπορούν να χρησιμοποιηθούν για τη *σημασιολογική σήμανση* (semantic markup) του περιεχομένου και της λειτουργικότητας πόρων που είναι διαθέσιμοι μέσω του Διαδικτύου. Οι οντολογίες και η βασισμένη σε αυτές σημασιολογική σήμανση μπορούν να αξιοποιηθούν σε τομείς όπως:

- το ηλεκτρονικό εμπόριο, όπου μπορούν να διευκολύνουν την επικοινωνία μεταξύ των πρακτόρων αγοράς και πώλησης παρέχοντας ένα κοινό λεξιλόγιο για την περιγραφή των αγαθών και των υπηρεσιών [Georgantis, Koutsomitropoulos, Zafiris, & Paratheodorou, 2002],
- οι μηχανές αναζήτησης, όπου μπορούν να βοηθήσουν στην ανεύρεση σελίδων που περιέχουν σημασιολογικά όμοιες, αλλά συντακτικά διαφορετικές λέξεις και φράσεις,
- η διαχείριση και ιδιαίτερα η ανακάλυψη γνώσης στο Διαδίκτυο, όπου, με τη χρήση συστημάτων συλλογισμού, μπορούν να βοηθήσουν στην εξαγωγή πληροφορίας που δεν είναι ρητά ορισμένη,
- οι υπηρεσίες Ιστού και πλέγματος (web και grid services), όπου μπορούν να παρέχουν πλούσιες περιγραφές που βοηθούν στον εντοπισμό των κατάλληλων υπηρεσιών (*matchmaking*).



Εικόνα 2.2: Η «πυραμίδα» των γλωσσών αναπαράστασης οντολογιών για το Σημαντικό Ιστό.

Κατά καιρούς έχουν προταθεί διάφορες αναγνώσιμες από τη μηχανή γλώσσες για την αναπαράσταση και ανταλλαγή γνώσης και οντολογιών. Στις πρώτες ιστορικά από αυτές τις γλώσσες και πρωτόκολλα συγκαταλέγονται η KIF και το OKBC. Τα τελευταία χρόνια, ο συνδυασμός των πλεονεκτημάτων που έφερε η XML και η επιθυμία για σημασιολογική σήμανση στο Διαδίκτυο, οδήγησαν σε μια νέα γενιά γλωσσών, κατάλληλων για αναπαράσταση οντολογιών στον Ιστό: Η SHOE, η XOL, η OML, η DAML+OIL και η OWL ανήκουν στις γλώσσες αυτές. Στα επόμενα πραγματοποιείται μια επισκόπηση των

κυριότερων από αυτές τις γλώσσες και τα πρότυπα, ιδιαίτερα αυτών που βασίζονται στην XML και στο RDF και επομένως μπορούν να αξιοποιηθούν στο περιβάλλον του (Σημαντικού) Ιστού. Η εικόνα 2.2 παρουσιάζει την ιεραρχία, από άποψη της χρονολογικής εξέλιξης και εκφραστικής ισχύος, των βασισμένων σε XML γλωσσών αναπαράστασης. Η OWL, λόγω της σπουδαιότητάς της, παρουσιάζεται στην επόμενη ενότητα.

2.3.1. KIF

Η KIF (Knowledge Interchange Format, Μορφότυπος Ανταλλαγής Γνώσης) [Fikes & Genesereth, 1994] υπήρξε μια από τις πρώτες γλώσσες αναπαράστασης γνώσης. Η KIF είναι μια γλώσσα για την ανταλλαγή γνώσης μεταξύ ετερογενών προγραμμάτων. Έχει δηλωτική σημασιολογία (δηλαδή η σημασία των εκφράσεων στην αναπαράσταση μπορεί να γίνει κατανοητή χωρίς να χρειάζεται κλήση του μεταγλωττιστή για την επεξεργασία των εκφράσεων αυτών), είναι λογικά διεξοδική (δηλαδή καθιστά δυνατή την έκφραση αυθαίρετων προτάσεων σε κατηγορηματικό λογισμό πρώτης τάξης), επιτρέπει την αναπαράσταση κανόνων μη μονοτονικού συλλογισμού και καθιστά δυνατό τον ορισμό αντικειμένων, συναρτήσεων και σχέσεων.

Η KIF δεν προορίζεται για αναπαράσταση γνώσης κατανοήσιμη από τον άνθρωπο (αν και το πετυχαίνει, αφού είναι δηλωτική), ούτε και για αναπαράσταση γνώσης εσωτερικά από ένα πρόγραμμα. Είναι μια γλώσσα αυξημένης εκφραστικής ισχύος και για το λόγο αυτό είναι ιδιαίτερα δύσκολο να σχεδιαστούν αποδοτικά συστήματα που να την υποστηρίζουν και δεν είναι κατάλληλη για τη διεξαγωγή συλλογισμού.

2.3.2. OKBC

Για την αναπαράσταση οντολογιών έχουν παραδοσιακά προταθεί διάφορες γλώσσες βασισμένες σε πλαίσια (frames), όπως η Ontolingua, η Ocelot, η LOOM κ.ά. που προέκυψαν από τα ομώνυμα συστήματα. Το πρωτόκολλο OKBC (Open Knowledge Base Connectivity, Ανοιχτή Συνδεσιμότητα Βάσεων Γνώσης) [Chaudhri, Farquhar, Fikes, Karp, & Rice, 1998] παρέχει ένα ενιαίο μοντέλο για Συστήματα Αναπαράστασης Γνώσης (Knowledge Representation Systems, KRSs) που βασίζεται σε μια κοινή εννοιολογική σύλληψη για τις κλάσεις, τα άτομα, τις σχισμές (slots), τις όψεις (facets) και την κληρονομικότητα. Το Μοντέλο Γνώσης GFP (Generic Frame Protocol, Γενικό Πρωτόκολλο Πλαισίων) είναι ο φορμαλισμός αναπαράστασης πάνω στον οποίο βασίζεται το OKBC. Το πρωτόκολλο είναι ορισμένο ανεξαρτήτως γλώσσας προγραμματισμού και επιτρέπει τη διαφανή δικτυακή ή και άμεση προσπέλαση σε KRSs και βάσεις γνώσης.

2.3.3. XOL

Η XOL (XML-based Ontology exchange Language, Γλώσσα ανταλλαγής Οντολογιών βασισμένη στην XML) [Karp, Chaudhri, & Thomere, 1999] σχεδιάστηκε από την κοινότητα της βιοπληροφορικής σε ορισμένα πανεπιστήμια και εταιρίες της Αμερικής για την ανταλλαγή ορισμών οντολογιών μεταξύ ετερογενών συνόλων πληροφοριακών συστημάτων στο πεδίο αυτό, αλλά μπορεί να επεκταθεί και σε άλλα πεδία. Ο ορισμός της γλώσσας προήλθε από δύο διαφορετικές ανάγκες:

- την ανάγκη για μια γλώσσα με τη σημασιολογία των αντικειμενοστραφών συστημάτων αναπαράστασης γνώσης, κάτι που οδήγησε στη χρήση ενός υποσυνόλου του OKBC που καλείται OKBC-Lite,
- την ανάγκη για το συντακτικό της XML, με σκοπό τη διαλειτουργικότητα και την αξιοποίηση των υπάρχοντων συντακτικών αναλυτών (parsers).

Η XOL δημιουργήθηκε ύστερα από αξιολόγηση των (υπαρχουσών τότε) γλωσσών αναπαράστασης οντολογιών, όπου η Ontolingua και η OML επιλέχθηκαν ως βάση ανάπτυξης, συνδυάζοντας το απλό και αναγνώσιμο συντακτικό της OML (που χρησιμοποιεί XML) με την υψηλή εκφραστικότητα της Ontolingua. Το υποσύνολο του OKBC που τελικά επιλέχθηκε ως μοντέλο γνώσης για τη γλώσσα, καλύπτει μόνο κλάσεις, σχισμές και όψεις. Επιπλέον η γλώσσα δεν επιτρέπει τον ορισμό αξιωμάτων σχηματισμού εννοιών (δηλαδή ορισμού εννοιών από άλλες έννοιες με τη χρήση των συνολοθεωρητικών τελεστών) και κανόνων [Ribière & Charlton, 2000].

2.3.4. SHOE

Η SHOE (Simple HTML Ontology Extension, Απλή Οντολογική Επέκταση της HTML) [Luke & Heflin, 2000; Heflin, Hendler, & Luke, 1999] αναπτύχθηκε στο πανεπιστήμιο του Maryland και χρησιμοποιήθηκε σαν βάση για την ανάπτυξη της OML. Η SHOE αρχικά δημιουργήθηκε ως μια επέκταση της HTML, με στόχο την ενσωμάτωση αναγνώσιμης από τη μηχανή σημασιολογικής γνώσης σε HTML ή άλλα έγγραφα του Ιστού. Πρόσφατα το συντακτικό της SHOE προσαρμόστηκε στην XML.

Η SHOE επιδιώκει να δώσει τη δυνατότητα στους πράκτορες να συλλέγουν χρήσιμες πληροφορίες για ιστοσελίδες και έγγραφα, βελτιώνοντας τους μηχανισμούς αναζήτησης και συλλογής γνώσης. Η διαδικασία αυτή αποτελείται από τρία στάδια: Ορισμό μιας οντολογίας, σχολιασμό των HTML σελίδων με οντολογική πληροφορία για την περιγραφή των ίδιων, καθώς και άλλων σελίδων και τέλος σημασιολογική ανάκτηση πληροφορίας από πράκτορες που ερευνούν όλες τις υπάρχουσες σελίδες και κρατούν ενημερωμένες πληροφορίες.

2.3.5. OML

Η OML (Ontology Markup Language, Οντολογική Γλώσσα Σήμανσης) (<http://www.ontologos.org/OML/OML%200.3.htm>) αναπτύχθηκε στο πανεπιστήμιο της Washington και βασίζεται μερικώς στη SHOE. Ουσιαστικά η OML υπήρξε στην αρχή μια μετεγγραφή της SHOE σε XML. Επομένως η SHOE και η OML έχουν πολλά κοινά χαρακτηριστικά.

Υπάρχουν τέσσερα διαφορετικά επίπεδα της OML. Αυτή η προσέγγιση των διαδοχικών επιπέδων υιοθετήθηκε από την OIL και μετέπειτα από την OWL. Τα επίπεδα αυτά είναι:

- OML Core, που σχετίζεται με τις λογικές παραμέτρους της γλώσσας και συμπεριλαμβάνεται στα άλλα επίπεδα,
- Simple OML, που αντιστοιχεί απευθείας στο RDF(S),
- Abbreviated OML, που περιλαμβάνει χαρακτηριστικά εννοιολογικών γραφημάτων,
- Standard OML, που αποτελεί την πιο εκφραστική έκδοση της γλώσσας.

2.3.6. RDF(S)

Το RDF (Resource Description Framework, Πλαίσιο Περιγραφής Πόρων) [Manola & Miller, 2004] σχεδιάστηκε για την αναπαράσταση μεταδεδομένων που περιγράφουν πόρους του Ιστού χρησιμοποιώντας τρεις τύπους αντικειμένων:

- *Πόρους* (resources), δηλαδή οντότητες που περιγράφονται στις εκφράσεις του RDF και πάντοτε αναγνωρίζονται με κάποιο URI ή και ένα επιπλέον id,
- *Ιδιότητες* (properties), που ορίζουν συγκεκριμένες παραμέτρους, χαρακτηριστικά, ιδιότητες ή σχέσεις τα οποία χρησιμοποιούνται για να περιγράψουν έναν πόρο,
- *Δηλώσεις* (statements), που αντιστοιχίζουν μια τιμή για μια ιδιότητα σε έναν πόρο (η τιμή αυτή μπορεί να είναι με τη σειρά της μια άλλη RDF δήλωση).

Το μοντέλο δεδομένων του RDF είναι πολύ κοντά στο φορμαλισμό των Σημαντικών Δικτύων. Το RDF ωστόσο δεν παρέχει μηχανισμούς για την περιγραφή των ιδιοτήτων, ούτε και μηχανισμούς για την περιγραφή των σχέσεων μεταξύ των ιδιοτήτων αυτών και άλλων πόρων. Αυτός είναι ο ρόλος της γλώσσας περιγραφής του λεξιλογίου του RDF, που καλείται RDF Schema (RDFS) [Brickley & Guha, 2004]. Η RDFS καθιστά δυνατή την αναπαράσταση κλάσεων, ρόλων και περιορισμών σε αυτούς τους ρόλους, ενώ το RDF επιτρέπει την αναπαράσταση στιγμιότυπων και γεγονότων (δηλώσεων). Συγκεκριμένα με την RDFS επιτρέπονται τα οντολογικά αξιώματα της κλάσης, της κληρονομικότητας κλάσεων, της κληρονομικότητας ιδιοτήτων και των περιορισμών πεδίου ορισμού και πεδίου τιμών.

Ο συνδυασμός RDF και RDFS καλείται RDF(S). Η γλώσσα δεν επιτρέπει την αναπαράσταση αξιωμάτων, ενώ ο μόνος δυνατός συμπερασμός είναι η υπαγωγή. Το πλεονέκτημα της γλώσσας είναι η χρήση του χαρακτηριστικού χώρων ονομάτων (namespaces) από την XML και η χρήση URIs για την αναγνώριση οντοτήτων που είναι ήδη ορισμένες στον Ιστό. Τούτο συνεπάγεται τις ακόλουθες δυνατότητες:

- Οι δηλώσεις μπορούν να αναφέρονται σε άλλες οντολογίες που είναι κατανεμημένες στο Διαδίκτυο.
- Μια οντολογία μπορεί να οριστεί σύμφωνα με μια άλλη οντολογία που είναι διαθέσιμη στον Ιστό.

2.3.7. OIL

Η OIL (Ontology Inference Layer, Επίπεδο Συμπερασμού Οντολογίας) [Fensel, Horrocks, Harmelen, Decker, Erdmann, & Klein, 2000] αναπτύχθηκε στα πλαίσια του IST προγράμματος OnToKnowledge (<http://www.ontoknowledge.org/oil/index.shtml>) με στόχο τη σημασιακή διαλειτουργικότητα μεταξύ διαδικτυακών πόρων. Το συντακτικό και η σημασιολογία της γλώσσας βασίζονται σε υπάρχουσες προτάσεις (OKBC, XOL και RDF(S)) και παρέχουν μηχανισμούς μοντελοποίησης που χρησιμοποιούνται ευρέως στις βασισμένες σε πλαίσια προσεγγίσεις και στο σχεδιασμό και ανάπτυξη οντολογιών (έννοιες, ταξινομίες εννοιών, σχέσεις κοκ). Η τυπική σημασιολογία της OIL και η δυνατότητα διεξαγωγής συλλογισμού με αυτήν προέρχονται από τις Λογικές Περιγραφής: Η OIL (στην πιο εκφραστική της μορφή) ισοδυναμεί με την Λογική Περιγραφής *SHOIQ(D)*.

Η OIL κατασκευάζεται πάνω στην RDF(S) και ακολουθεί την παρακάτω (γνωστή από την OML) προσέγγιση διαδοχικών επιπέδων:

- Η *Core OIL* περιλαμβάνει τους μηχανισμούς που αντιστοιχούν απευθείας στο RDF(S).
- Η *Standard OIL* θεωρείται το πλήρες μοντέλο της OIL και περιλαμβάνει περισσότερα εκφραστικά αξιώματα από ότι η RDF(S).
- Η *Instance OIL* προσθέτει στιγμιότυπα και ρόλους στο προηγούμενο μοντέλο.
- Η *Heavy OIL* είναι το επίπεδο που θα φιλοξενήσει της μελλοντικές επεκτάσεις της γλώσσας.

2.3.8. DAML+OIL

Η DAML+OIL [McGuinness, Fikes, Hendler, & Stein, 2002] υπήρξε, για αρκετό καιρό, η βασική γλώσσα αναφοράς για το Σημαντικό Ιστό. Ο υποσκελισμός της DAML+OIL από την πρόσφατη OWL δεν μειώνει την αξία του παραπάνω ισχυρισμού, καθώς η γλώσσα απολαμβάνει σταθερότητας και, πάνω από όλα, ενός συνόλου εργαλείων και πρακτικών εφαρμογών που η OWL δεν έχει ακόμα γνωρίσει.

Με στόχο τη χρηματοδότηση της έρευνας για γλώσσες, υποδομή και εργαλεία που θα κάνουν τον Ιστό πιο προσπελάσιμο και κατανοήσιμο, η Διεύθυνση Προηγμένων Ερευνητικών Προγραμμάτων για την Άμυνα των Η.Π.Α. (DARPA, Defense Advanced Research Programs Agency) ξεκίνησε το πρόγραμμα DAML (DARPA Agent Markup Language, Γλώσσα Σήμανσης Πρακτόρων). Από τα πρώτα αποτελέσματα ήταν ο ορισμός, τον Οκτώβριο του 2000, της γλώσσας DAML-ONT, η οποία παρουσίαζε μεγαλύτερη εκφραστική δύναμη από την RDF(S) και είχε σαφείς επιρροές από την Ontolingua, το OKBC, την SHOE και την XOL. Ύστερα από συζητήσεις σχετικά με τις διαφορές μεταξύ της DAML-ONT και της OIL αποφασίστηκε η συγχώνευση των δύο γλωσσών με αποτέλεσμα την DAML+OIL.

Η DAML+OIL είναι μια RDF(S) γλώσσα με πλουσιότερους μηχανισμούς μοντελοποίησης, όπως αυτοί που απαντώνται στα πλαίσια, καθώς και ξεκάθαρη και καλώς ορισμένη σημασιολογία που βασίζεται στις Λογικές Περιγραφές. Η πρώτη έκδοση της γλώσσας δεν περιλάμβανε τύπους δεδομένων. Η πιο πρόσφατη έκδοση (<http://www.daml.org/2001/03/daml+oil-index.html>) ενσωματώνει αυθαίρετους (ενσωματωμένους και παραγόμενους) τύπους δεδομένων με το σύστημα της XML Schema και επομένως, το εκφραστικό μοντέλο της γλώσσας μπορεί να θεωρηθεί ότι ισοδυναμεί με τη Λογική Περιγραφής SHOIQ(D).

2.4. OWL

Η ενότητα αυτή είναι αφιερωμένη στη Γλώσσα Οντολογιών Ιστού (OWL, Ontology Web Language). Θεωρούμε ότι η γλώσσα αυτή αποτελεί πλέον τη βάση για την αναπαράσταση γνώσης στο Σημαντικό Ιστό, λόγω της προώθησής της από το W3C, του αυξανόμενου βαθμού υιοθέτησής της στις σχετικές εφαρμογές και των επιθυμητών χαρακτηριστικών που φέρει, λόγω σχεδιασμού, από εκφραστικής και υπολογιστικής άποψης.

Στα επόμενα γίνεται μια εισαγωγή στη γλώσσα αυτή και στη διάκρισή της στις τρεις βασικές διαλέκτους που την απαρτίζουν. Κατόπιν εξετάζονται οι αδυναμίες του RDF(S) και πώς η OWL συμπληρώνει τα κενά αυτά. Τέλος εντοπίζονται οι διαφορές με το θεωρητικό λογικό της υπόβαθρο, τις Λογικές Περιγραφής.

2.4.1. Γενική περιγραφή

Η OWL [McGuinness & Harmelen, 2004] είναι η πιο πρόσφατη εξέλιξη στις γλώσσες οντολογιών για το (Σημαντικό) Ιστό. Είναι σημαντικό να αναφερθεί ότι η OWL (που αναπτύσσεται από το W3C) δεν έχει περάσει πολύς καιρός από τότε που οριστικοποιήθηκε, καθώς μόλις τον Φεβρουάριο του 2004 πέρασε σε κατάσταση Σύστασης (W3C Recommendation).

Η OWL έχει δεχτεί επιρροές από τις περισσότερες γλώσσες που έχουν μέχρι τώρα αναφερθεί, αλλά ιδιαίτερα η DAML+OIL είναι αυτή που αποτέλεσε το εναρκτήριο σημείο για την ανάπτυξη της γλώσσας. Η σημαντική ομοιότητα με την DAML+OIL αποτελεί ουσιαστικό πλεονέκτημα, καθώς τα υπάρχοντα εργαλεία μπορούν, σχετικά εύκολα, να τροποποιηθούν, ώστε να υποστηρίζουν την OWL.

Η OWL, όπως και οι προκάτοχοί της, ακολουθεί το συντακτικό της RDF(S), εμπλουτίζοντας όμως το λεξιλόγιό της για την περιγραφή των κλάσεων και των ιδιοτήτων, υποστηρίζοντας, μεταξύ άλλων, σχέσεις μεταξύ των κλάσεων (π.χ. ότι δύο κλάσεις είναι ξένες), πληθικότητα (περιορισμοί αριθμού), ισότητα, περισσότερα χαρακτηριστικά των ιδιοτήτων (π.χ. συμμετρία) και κλάσεις απαρίθμησης (χρήση ονοματικών). Η μεγαλύτερη εκφραστικότητα που προσδίδεται στη γλώσσα με τον τρόπο αυτό, είναι αναγκαία για το Σημαντικό Ιστό, καθώς επιτρέπει τη χρήση συστημάτων συλλογισμού για την διεξαγωγή αποτελεσματικών συμπερασμών πάνω στις οντολογίες που αναπαρίστανται στον Ιστό με τη γλώσσα αυτή.

Μεγαλύτερη όμως εκφραστική δύναμη συνεπάγεται τη διεξαγωγή συμπερασμών αυξανόμενης πολυπλοκότητας έως και μη αποφασισιμότητας. Για το λόγο αυτό, η OWL ακολουθεί τη γνωστή προσέγγιση των διαδοχικών επιπέδων:

- Η *OWL Lite* έχει σχεδιαστεί για την έκφραση ιεραρχιών ταξινόμησης και απλών περιορισμών ιδιοτήτων. Για παράδειγμα, ενώ η *OWL Lite* υποστηρίζει περιορισμούς πληθικότητας, οι μόνες τιμές που επιτρέπονται είναι 0 και 1. Είναι πιο εύκολο να σχεδιαστούν εργαλεία και να αντιστοιχιστούν θησαυροί όρων και ταξινομίες στην *OWL Lite* από ότι στα άλλα εκφραστικότερα επίπεδα. Η *OWL Lite* αντιστοιχεί στη Λογική Περιγραφής *SHIF(D)*.
- Η *OWL DL* υποστηρίζει τη μέγιστη δυνατή εκφραστικότητα, χωρίς απώλεια της αποφασισιμότητας. Η *OWL DL* ονομάζεται έτσι λόγω της αντιστοιχίας της με τις Λογικές Περιγραφής και συγκεκριμένα με τη Λογική Περιγραφής *SHOIN(D)*.
- Η *OWL Full* προορίζεται για χρήστες που επιθυμούν μέγιστη εκφραστικότητα και την πλήρη εκφραστική ελευθερία του RDF χωρίς όμως εγγυήσεις επιλυσιμότητας. Για παράδειγμα η *OWL Full* επιτρέπει σε μια κλάση να ορίζεται ως συλλογή ατόμων και συγχρόνως να θεωρείται άτομο η ίδια (πράγμα που οδηγεί σε κυκλικούς ορισμούς εννοιών και επομένως σε μη αποφασισιμότητα). Θεωρείται απίθανο ότι κάποιο σύστημα συλλογισμού θα μπορέσει ποτέ να υποστηρίξει όλα τα χαρακτηριστικά της *OWL Full*.

Η OWL DL και η OWL Full χρησιμοποιούν το ίδιο λεξιλόγιο, αλλά η OWL DL υπάγεται σε κάποιους περιορισμούς, κυριότερος από τους οποίους είναι η απαίτηση για διαχωρισμό των τύπων: Μια κλάση δεν μπορεί να είναι συγχρόνως άτομο ή ιδιότητα και μια ιδιότητα δεν μπορεί να είναι συγχρόνως άτομο ή κλάση. Όσον αφορά τη σχέση με το RDF, η OWL Full μπορεί να θεωρηθεί επέκταση του RDF, ενώ οι OWL Lite και OWL DL μπορούν να θεωρηθούν επεκτάσεις μιας περιορισμένης όψης του RDF: Κάθε OWL (Lite, DL, Full) έγγραφο είναι ένα RDF έγγραφο και κάθε RDF έγγραφο είναι ένα OWL Full έγγραφο.

Η OWL έχει αυξημένη εκφραστικότητα και σε σχέση με την DAML+OIL. Η OWL DL μπορεί να θεωρηθεί ως η περισσότερο ισοδύναμη με την DAML+OIL, με τη διαφορά ότι έχουν αφαιρεθεί οι προσδιορισμένοι περιορισμοί αριθμού από τη γλώσσα, δεδομένου ότι δεν παρουσιαζόταν ουσιαστική ανάγκη για τη διατήρησή τους και επιπλέον ήταν δύσκολο να υλοποιηθούν. Άλλες αλλαγές περιλαμβάνουν τυπογραφικές αλλαγές και μετονομασίες, ενώ στην OWL έχουν ενσωματωθεί και οι πρόσφατες ενημερώσεις στον ορισμό της RDF(S).

2.4.2. Από το RDF(S) στην OWL

Το RDF και το RDFS επιτρέπουν την αναπαράσταση ενός μόνο μέρους της οντολογικής γνώσης. Τα αξιώματα μοντελοποίησης του RDF(S) αφορούν την οργάνωση λεξιλογίων σε ιεραρχίες: σχέσεις υποκλάσης και υπο-ιδιότητες, περιορισμοί πεδίου ορισμού και πεδίου τιμών και στιγμιότυπα κλάσεων. Λείπουν όμως ορισμένα χαρακτηριστικά, που περιλαμβάνουν [Antoniou & Harmelen, 2004]:

- *Τοπική εμβέλεια των ιδιοτήτων.* Το `rdfs:range` ορίζει το πεδίο τιμών μιας ιδιότητας, π.χ. `τρώει`, για όλες τις κλάσεις. Έτσι στο RDFS δεν μπορούμε να δηλώσουμε τέτοιους περιορισμούς, που να εφαρμόζονται για ορισμένες κλάσεις μόνο. Για παράδειγμα, δεν μπορούμε να πούμε ότι οι αγελάδες τρώνε μόνο φυτά, ενώ κάποια άλλα ζώα μπορούν να τρώνε και κρέας.
- *Ξένες κλάσεις.* Ορισμένες φορές είναι χρήσιμο να πούμε ότι δυο κλάσεις δεν έχουν κοινά στοιχεία. Για παράδειγμα, οι κλάσεις `αρσενικό` και `θηλυκό` είναι ξένες. Στο RDFS όμως μπορούμε να δηλώσουμε μόνο σχέσεις υπαγωγής, π.χ. ότι το `θηλυκό` είναι υποκλάση του `άνθρωπος`.
- *Διαδικοί συνδυασμοί των κλάσεων.* Είναι συχνά επιθυμητή η κατασκευή νέων κλάσεων από το συνδυασμό άλλων κλάσεων, χρησιμοποιώντας ένωση, τομή και συμπλήρωμα. Για παράδειγμα, ο ορισμός της κλάσης `άνθρωπος` μπορεί να είναι η ένωση των `αρσενικό` και `θηλυκό`. Το RDFS δεν επιτρέπει την έκφραση τέτοιων ορισμών.
- *Περιορισμοί πληθικότητας.* Κάποιες φορές είναι επιθυμητό να τίθενται περιορισμοί στο πόσες διακριτές τιμές μπορεί ή πρέπει να έχει μια ιδιότητα. Για παράδειγμα, ότι ένας `άνθρωπος` έχει ακριβώς δύο γονείς ή ότι ένα μάθημα διδάσκεται από τουλάχιστον ένα καθηγητή.
- *Ειδικά χαρακτηριστικά των ιδιοτήτων.* Είναι σκόπιμο ορισμένες φορές να πούμε ότι μια ιδιότητα είναι μεταβατική (π.χ. «μεγαλύτεροΑπό»), μοναδιαία («έχειΜητέρα») ή η αντίστροφη μιας άλλης («τρώει» και «τρώγεταιΑπό»).

Χρειαζόμαστε επομένως μια οντολογική γλώσσα πιο πλούσια από το RDFS, που να προσφέρει τουλάχιστον αυτά τα χαρακτηριστικά. Κατά το σχεδιασμό μιας τέτοιας γλώσσας,

θα πρέπει να ληφθεί υπόψη η ανταγωνιστικότητα μεταξύ υψηλής εκφραστικής ισχύος και αποδοτικού συλλογισμού. Όσο πιο πλούσια είναι η γλώσσα, τόσο μειώνεται η αποδοτικότητα του συλλογισμού. Χρειάζεται επομένως ένας συμβιβασμός, μια γλώσσα δηλαδή που να είναι αρκούντως εκφραστική, ενώ ταυτόχρονα να υποστηρίζεται από σχετικά αποδοτικούς μηχανισμούς συλλογισμού.

2.4.3. OWL και Λογικές Περιγραφές

Η OWL Full, όντας πιο κοντά στη θεωρία μοντέλου του RDF και μη έχοντας το χαρακτηριστικό της αποφασισιμότητας [Horrocks, Patel-Schneider, & Harmelen, 2003] δεν αντιστοιχεί ούτε συντακτικά ούτε σημασιακά σε κάποια Λογική Περιγραφή. Ωστόσο, ακόμα και η OWL DL, που ισοδυναμεί εκφραστικά με την Λογική Περιγραφή *SHOIN(D)* [Horrocks & Patel-Schneider, 2003], έχει ορισμένες διαφορές από τις Λογικές Περιγραφές, κυρίως ως προς το συντακτικό, τις εκφράσεις σχολιασμού (annotations) και τις δυνατότητες εισαγωγής (import). Οι διαφορές αυτές παρέχουν τη γέφυρα μεταξύ των τυπικών Λογικών Περιγραφών και του Σημαντικού Ιστού:

- Η OWL χρησιμοποιεί αναφορές URI ως ονόματα και κατασκευάζει τις αναφορές αυτές με τον τρόπο που χρησιμοποιείται στο RDF. Είναι επομένως σύνηθες στην OWL να χρησιμοποιούνται προσδιορισμένα ονόματα σαν συντομογραφίες για τις αναφορές URI, όπως, για παράδειγμα, το προσδιορισμένο όνομα `owl:Thing` για την αναφορά URI <http://www.w3.org/2002/07/owl#Thing>.
- Η OWL συγκεντρώνει πληροφορίες σε οντολογίες, που είναι γενικά αποθηκευμένες ως έγγραφα Ιστού, γραμμένα σε RDF/XML. Οι οντολογίες μπορούν να εισάγουν άλλες οντολογίες, προσθέτοντας τις πληροφορίες από την εισαγόμενη οντολογία στην τρέχουσα.
- Η OWL επιτρέπει τη χρήση ιδιοτήτων σχολιασμού RDF για την επισύναψη πληροφοριών σε κλάσεις, ιδιότητες και οντολογίες, όπως π.χ. `owl:DeprecatedClass`. Οι σχολιασμοί αυτοί είναι τριπλέτες RDF και απαιτείται επομένως να φέρουν πλήρες σημασιακό βάρος στο RDF. Στην OWL DL όμως, οι σχολιασμοί αυτοί φέρουν μια ξεχωριστή, περιορισμένη σημασία.
- Η OWL χρησιμοποιεί τις υπηρεσίες τύπων δεδομένων του RDF και του XML Schema για την παροχή τύπων και τιμών δεδομένων (μια πολύ περιορισμένη μορφή απτών πεδίων).
- Η OWL DL και η OWL Lite έχουν αφηρημένη σύνταξη τύπου πλαισίων, ενώ η RDF/XML είναι το επίσημο συντακτικό ανταλλαγής και για τα τρία είδη της OWL.

2.5. OWL 1.1

Η πιο πρόσφατη εξέλιξη στις γλώσσες αναπαράστασης οντολογιών για τον Ιστό εντοπίζεται στην γλώσσα OWL 1.1 [Parsia & Patel-Schneider, 2008; Patel-Schneider & Horrocks, 2006]. Η γλώσσα αυτή αποτελεί επέκταση της κλασικής OWL που ενσωματώνει όμως περισσότερα, πιο εκφραστικά χαρακτηριστικά. Στην πραγματικότητα, η OWL 1.1 είναι η έκφραση σε RDF/XML σύνταξη της Λογικής Περιγραφής *SROIQ(D)* (βλ. ενότητα 3.7). Ως τέτοια,

περιλαμβάνει επιπλέον χαρακτηριστικά για τις ιδιότητες (ανακλαστικότητα, αρνητικές δηλώσεις, αλυσίδες ιδιοτήτων) τα οποία είχαν ήδη ζητηθεί από την εποχή της OWL (1.0), αλλά ήταν δύσκολο να υλοποιηθούν ή να προδιαγραφούν (βλ. ενότητα 4.6.2 και [Grau, Horrocks, Parsia, Patel-Schneider, & Sattler, 2006]). Ταυτόχρονα, είναι πλέον διαθέσιμοι αποφασίσιμοι αλγόριθμοι που επιτρέπουν το συλλογισμό στη γλώσσα αυτή, καθώς και αντίστοιχα υλοποιημένα συστήματα.

Άλλο πολύ ενδιαφέρον χαρακτηριστικό της OWL 1.1 είναι η *παρονομασία* (running) η οποία επιτρέπει τη διερμηνευση ενός ονόματος σύμφωνα με τη χρήση του και άρα καταργεί το διαχωρισμό μεταξύ των διαφορετικών λεξιλογίων μιας οντολογίας. Το χαρακτηριστικό αυτό φαίνεται ότι υποστηρίζεται, έστω και μερικώς, από ορισμένες υλοποιήσεις και αξιοποιείται στο κεφάλαιο 6, στην περίπτωση των ημιδομημένων πεδίων γνώσης. Ο διαχωρισμός των πεδίων (δηλαδή διαφορετικά ονόματα για τα στιγμιότυπα, τις κλάσεις και τις ιδιότητες) ήταν η κύρια αιτία μη αποφασισιμότητας στην OWL Full. Να σημειωθεί ότι στην παρονομασία το ίδιο όνομα μπορεί να σημαίνει διαφορετικά πράγματα *ανάλογα με το πώς χρησιμοποιείται*, ενώ στην OWL Full το ίδιο όνομα μπορεί να σημαίνει διαφορετικά πράγματα *ταυτόχρονα*.

Τον Απρίλιο του 2008 η νέα αυτή έκδοση της OWL άλλαξε όνομα σε OWL 2. Σαφέστατα, η αλλαγή αυτή εκφράζει την προσπάθεια για αποτελεσματικότερη προώθηση του προτύπου, δεν φαίνεται όμως να δικαιολογείται σε σχέση με την όχι και τόσο δραματική απόσταση από την προκάτοχό της. Παράλληλα, στα νέα αυτά κείμενα εργασίας, γίνεται λόγος για OWL 2 DL και OWL 2 Full, χωρίς περαιτέρω διευκρινίσεις, ενώ, μέχρι τότε (δηλαδή για την OWL 1.1) δεν γινόταν τέτοια διάκριση σε διαλέκτους και μάλιστα φαινόταν ότι θα αποφευχθεί, λόγω της παρονομασίας. Τα γεγονότα αυτά είναι δύο μόνο από τα πολλά που διαρκώς επιβεβαιώνουν ότι η γλώσσα αυτή δεν είναι ακόμα σταθερή, αλλά προϊόν διαρκούς εξέλιξης και ζυμώσεων. Άλλωστε, σύμφωνα με το καταστατικό της αντίστοιχης ομάδας εργασίας του W3C, η όλη διαδικασία τυποποίησης θα πρέπει να έχει ολοκληρωθεί μέχρι το καλοκαίρι του 2009.

Από πλευράς συντακτικού, η OWL 2 ακολουθεί την RDF/XML σύνταξη, ορίζοντας τους επιπρόσθετους όρους κάτω από τον νέο χώρο ονομάτων owl11 (δεν έχει ακόμα οριστικοποιηθεί). Ως βάση όμως, η δομή της γλώσσας προδιαγράφεται σύμφωνα με ένα συντακτικό λειτουργικής μορφής (functional-style syntax) [Motik, Patel-Schneider, & Horrocks, 2008]. Η σύνταξη αυτή στοχεύει να είναι πιο αναγνώσιμη από ανθρώπους και έχει άμεση σχέση με τη θεωρία μοντέλου της γλώσσας. Η λειτουργική σύνταξη έχει επίσης κωδικοποιηθεί σε ένα XML Schema, ώστε υποστηρικτικά εργαλεία να μπορούν να υλοποιήσουν την XML σύνταξη ανταλλαγής [Grau, Motik, & Patel-Schneider, 2008], ως πιο στενά συνδεδεμένη με τη λειτουργική σύνταξη και λιγότερο «φλύαρη» από την RDF/XML. Ακόμη, ως μίμηση της φυσικής γλώσσας, έχει προταθεί και το λεγόμενο «συντακτικό του Μάντσεστερ» (Manchester syntax) [Horridge, Drummond, Goodwin, Rector, Stevens, & Wang, 2006] το οποίο αντικαθιστά τα λογικά σύμβολα με εκφράσεις τις αγγλικής (π.χ. \exists με *some*, \forall με *only* κτλ) και χρησιμοποιείται στο Protégé 4.

Για να γίνει κατανοητή η εκφραστικότητα της νέας γλώσσας, στη συνέχεια παρουσιάζεται η σημαντική της OWL 2, όπως προκύπτει από την *SROIQ(D)* με τρόπο συμβατό με τη θεωρία μοντέλων [Hodges, 1993]. Η νέα γλώσσα είναι σχεδιασμένη ώστε να αποτελεί υπερσύνολο της OWL, άρα τα λεχθέντα στα επόμενα ισχύουν και για την OWL (1.0). Η ενότητα

ολοκληρώνεται με την παράθεση ορισμένων παραδειγμάτων, που είναι ενδεικτικά των δυνατοτήτων της νέας OWL και της παρονομασίας.

2.5.1. Σημασιολογία

Η σημασιολογία της OWL 2, όπως ορίζεται στο [Grau & Motik, 2008] είναι συμβατή με τη Λογική Περιγραφής *SROIQ* (βλ. ενότητα 3.7). Ωστόσο, η *SROIQ* δεν προβλέπει υποστήριξη για τους τύπους δεδομένων και την παρονομασία. Από την άλλη, η OWL 2 δεν περιλαμβάνει τον λεγόμενο *καθολικό ρόλο*. Συνεπώς, η σημασιολογία της OWL 2 ορίζεται με έναν απευθείας μοντελο-θεωρητικό τρόπο, διερμηνεύοντας τις δομές του συντακτικού λειτουργικής μορφής της γλώσσας. Για τις κατασκευές που προβλέπονται στην *SROIQ*, αυτές αντιστοιχούν τετριμμένα προς εκείνες που δίνονται παρακάτω.

Η OWL 2 δεν έχει σημασιολογία συμβατή με το RDF. Η απόδοση σημασιολογίας στις οντολογίες που εκφράζονται σε OWL/RDF σύνταξη γίνεται μετατρέποντάς τις σε συντακτικό λειτουργικής μορφής και διερμηνεύοντας τα αποτελέσματα με τον τρόπο που περιγράφεται εδώ.

Η OWL 2 επιτρέπει το σχολιασμό (annotation) των οντολογιών και των οντοτήτων μιας οντολογίας (κλάσεις, ιδιότητες και άτομα), καθώς και των αξιωμάτων της οντολογίας. Ωστόσο, οι σχολιασμοί δεν έχουν καμία σημασιολογική ερμηνεία στην OWL 2 και επομένως αγνοούνται. Παρομοίως, στην OWL 2 και οι ορισμοί δεν έχουν σημασιολογία. Κατά συνέπεια, δομές που χρησιμοποιούνται μόνο σε σχολιασμούς και ορισμούς, όπως η `ObjectProperty`, δεν αναφέρονται εδώ.

Δεδομένου ότι η OWL 2 αποτελεί επέκταση της OWL DL, η σημασιολογία που δίνεται εδώ παρέχει την επίσημη σημασιολογία και για τις OWL Lite και OWL DL και είναι ισοδύναμη με τον ορισμό που δίνεται στο [Patel-Schneider, Hayes, & Horrocks, 2004].

Ένα λεξιλόγιο (ή υπογραφή - signature) $V = (N_C, N_{P_o}, N_{P_d}, N_I, N_D, N_V)$ είναι ένα μια εξάδα όπου:

- N_C είναι ένα σύνολο κλάσεων OWL (OWL classes),
- N_{P_o} είναι ένα σύνολο ιδιοτήτων αντικειμένων (object properties),
- N_{P_d} είναι ένα σύνολο ιδιοτήτων δεδομένων (data properties),
- N_I είναι ένα σύνολο ατόμων (individuals) και
- N_D είναι ένα σύνολο τύπων δεδομένων (datatypes) καθένας από τους οποίους συνδέεται με μια ακέραια και θετική πολλαπλότητα τύπου δεδομένων (datatype arity).
- N_V είναι ένα σύνολο καλά ορισμένων σταθερών (constants).

Δεδομένου ότι η OWL 2 επιτρέπει την παρονομασία μέσα σε μια υπογραφή, τα σύνολα N_C , N_{P_o} , N_{P_d} , N_I , N_D και N_V δεν είναι απαραίτητο να είναι αμοιβαίως αποκλειόμενα. Κατά συνέπεια, μέσα σε μια οντολογία, το ίδιο όνομα μπορεί να χρησιμοποιηθεί προκειμένου να υποδηλώσει μια κλάση, ένα τύπο δεδομένων, μια ιδιότητα (αντικειμένου ή δεδομένων), ένα άτομο, όπως και μια σταθερά. Το σύνολο N_D ορίζεται ως έχει, αφού ένας τύπος δεδομένων προσδιορίζεται από το όνομα και την πολλαπλότητα του, γεγονός που επιτρέπει την επαναχρησιμοποίηση του ίδιου ονόματος, αλλά με διαφορετικές πολλαπλότητες.

Η σημασιολογία της OWL 2 ορίζεται σε σχέση με ένα *απτό πεδίο* (concrete domain), που είναι μια πλειάδα $D = (\Delta_D, \cdot^D)$ όπου

- Το Δ_D είναι ένα καθορισμένο (fixed) σύνολο που καλείται *πεδίο δεδομένων (data domain)*,
- $H.^D$ αναθέτει σε κάθε σταθερά $v \in N_V$ ένα στοιχείο v^D του Δ_D , και
- $H.^D$ αναθέτει σε κάθε τύπο δεδομένων $d \in N_D$ με πολλαπλότητα n , μια n -αδική σχέση d^D πάνω στο Δ_D .

Σε κάθε λεξιλόγιο της OWL 2, το σύνολο τύπων δεδομένων N_D πρέπει να περιλαμβάνει ένα μοναδιαίο (unary) τύπο δεδομένων `rdfs:Literal` που να διερμηνεύεται ως Δ_D . Επιπρόσθετα, πρέπει οπωσδήποτε να περιλαμβάνει τους μοναδιαίους (δηλ. πολλαπλότητας 1) τύπους δεδομένων που προβλέπονται από το XML Schema. Αυτοί οι τύποι δεδομένων, όπως και οι καλά ορισμένες σταθερές από το σύνολο N_V , διερμηνεύονται με τον τρόπο που ορίζεται στο [Biron & Malhotra, 2004] (να σημειωθεί ότι οι χώροι τιμών των αρχικών τύπων δεδομένων του XML Schema αποτελούν ξένα σύνολα, επομένως το `"2"^^xsd:decimalD` είναι διαφορετικό από το `"2"^^xsd:floatD`).

Το σύνολο Δ_D είναι ένα καθορισμένο σύνολο που πρέπει να είναι *αρκετά μεγάλο*: πρέπει να περιέχει την επέκταση κάθε τύπου δεδομένων που υπάρχει στο σύνολο N_D , όπως επίσης και ένα άπειρο πλήθος (infinite number) άλλων αντικειμένων. Έτσι, ο ορισμός αυτός είναι αμφίσημος, καθώς δεν προσδιορίζει με μοναδικό τρόπο ένα συγκεκριμένο σύνολο Δ_D . Εντούτοις, η επιλογή του συνόλου που τελικά θα χρησιμοποιηθεί, στην ουσία δε σχετίζεται με τον καθορισμό της σημασιολογίας, από τη στιγμή που περιέχει τις διερμηνεύσεις όλων των τύπων δεδομένων τις οποίες κάποιος «μπορεί λογικά να σκεφθεί» ("reasonably think of"). Το γεγονός αυτό επιτρέπει στις υλοποιήσεις την υποστήριξη και διαφορετικών τύπων δεδομένων, χωρίς να επηρεάζεται η σημασιολογία.

Δεδομένου ενός λεξιλογίου V και ενός απτού πεδίου D , μια διερμηνεύση $I = (\Delta_I, .^{Ic}, .^{Ipo}, .^{Ipd}, .^{Ii})$ είναι μια πεντάδα στοιχείων όπου:

- Το Δ_I είναι ένα μη κενό σύνολο, ξένο με το Δ_D , το οποίο καλείται *πεδίο αντικειμένων (object domain)*.
- $H.^{Ic}$ είναι η συνάρτηση *διερμηνεύσης κλάσης*, που αναθέτει σε κάθε κλάση $A \in N_C$ ένα υποσύνολο A^{Ic} του Δ_I .
- $.^{Ipo}$ είναι η συνάρτηση *διερμηνεύσης ιδιότητας αντικειμένου*, που αναθέτει σε κάθε ιδιότητα αντικειμένου $R \in N_{Po}$ ένα υποσύνολο R^{Ipo} του $\Delta_I \times \Delta_I$.
- $.^{Ipd}$ είναι η συνάρτηση *διερμηνεύσης ιδιότητας δεδομένων*, που αναθέτει σε κάθε ιδιότητα δεδομένων $U \in N_{Pd}$ ένα υποσύνολο U^{Ipd} του $\Delta_I \times \Delta_D$.
- $.^{Ii}$ είναι η συνάρτηση *διερμηνεύσης ατόμου* όπου αναθέτει σε κάθε άτομο $a \in N_I$ ένα στοιχείο (element) a^{Ii} από το Δ_I .

Εκφράσεις ιδιοτήτων αντικειμένου

Στον παρακάτω πίνακα φαίνεται πώς επεκτείνεται η συνάρτηση διερμηνεύσης ιδιότητας αντικειμένου $.^{Ipo}$ σε εκφράσεις ιδιοτήτων αντικειμένου.

Έκφραση Ιδιότητας Αντικειμένου	Διερμηνεύση
<code>InverseObjectProperty(R)</code>	$\{(x, y) \mid (y, x) \in R^{Ipo}\}$

Πίνακας 2.1: Διερμηνεύση των εκφράσεων ιδιότητας αντικειμένου.

Εκφράσεις πεδίου τιμών δεδομένων

Στον πίνακα 2.2 φαίνεται πώς επεκτείνεται η συνάρτηση διερμηνεύσης \cdot^D σε πεδία τιμών δεδομένων (data ranges).

Πεδίο Τιμών Δεδομένων	Διερμηνεύση
$\text{DataOneOf}(v_1 \dots v_n)$	$\{v_1^D, \dots, v_n^D\}$
$\text{DataComplementOf}(DR)$	$(\Delta_D)^n \setminus DR^D$ όπου n είναι η πολλαπλότητα του DR
$\text{DatatypeRestriction}(DR \text{ f } v)$	Η n -αδική σχέση πάνω στο Δ_D που αποκτάται εφαρμόζοντας την όψη (facet) f με τιμή v στο πεδίο τιμών δεδομένων DR όπως ορίζεται στο XML Schema

Πίνακας 2.2: Διερμηνεύση των πεδίων τιμών δεδομένων.

Εκφράσεις κλάσεων

Στον πίνακα που ακολουθεί φαίνεται πώς επεκτείνεται η συνάρτηση διερμηνεύσης κλάσης \cdot^{lc} σε περιγραφές. Με το $\#S$ υποδηλώνεται ο αριθμός των στοιχείων μέσα σε ένα σύνολο S .

Περιγραφή	Διερμηνεύση
owl:Thing	Δ_I
owl:Nothing	κενό σύνολο
$\text{ObjectComplementOf}(C)$	$\Delta_I \setminus C^{lc}$
$\text{ObjectIntersectionOf}(C_1 \dots C_n)$	$C_1^{lc} \cap \dots \cap C_n^{lc}$
$\text{ObjectUnionOf}(C_1 \dots C_n)$	$C_1^{lc} \cup \dots \cup C_n^{lc}$
$\text{ObjectOneOf}(a_1 \dots a_n)$	$\{a_1^{li}, \dots, a_n^{li}\}$
$\text{ObjectSomeValuesFrom}(R \ C)$	$\{x \mid \exists y : (x, y) \in R^{lpo} \text{ και } y \in C^{lc}\}$
$\text{ObjectAllValuesFrom}(R \ C)$	$\{x \mid \forall y : (x, y) \in R^{lpo} \text{ σημαίνει ότι } y \in C^{lc}\}$
$\text{ObjectHasValue}(R \ a)$	$\{x \mid (x, a^{li}) \in R^{lpo}\}$
$\text{ObjectExistsSelf}(R)$	$\{x \mid (x, x) \in R^{lpo}\}$
$\text{ObjectMinCardinality}(n \ R \ C)$	$\{x \mid \#\{y \mid (x, y) \in R^{lpo} \text{ και } y \in C^{lc}\} \geq n\}$
$\text{ObjectMaxCardinality}(n \ R \ C)$	$\{x \mid \#\{y \mid (x, y) \in R^{lpo} \text{ και } y \in C^{lc}\} \leq n\}$
$\text{ObjectExactCardinality}(n \ R \ C)$	$\{x \mid \#\{y \mid (x, y) \in R^{lpo} \text{ και } y \in C^{lc}\} = n\}$
$\text{DataSomeValuesFrom}(U_1 \dots U_n \ DR)$	$\{x \mid \exists y_1, \dots, y_n : (x, y_k) \in U_k^{lpd} \text{ για κάθε } 1 \leq k \leq n \text{ και } (y_1, \dots, y_n) \in DR^D\}$
$\text{DataAllValuesFrom}(U_1 \dots U_n \ DR)$	$\{x \mid \forall y_1, \dots, y_n : (x, y_k) \in U_k^{lpd} \text{ για κάθε } 1 \leq k \leq n \text{ σημαίνει ότι } (y_1, \dots, y_n) \in DR^D\}$
$\text{DataHasValue}(U \ v)$	$\{x \mid (x, v^D) \in U^{lpd}\}$
$\text{DataMinCardinality}(n \ U \ DR)$	$\{x \mid \#\{y \mid (x, y) \in U^{lpd} \text{ και } y \in DR^D\} \geq n\}$
$\text{DataMaxCardinality}(n \ U \ DR)$	$\{x \mid \#\{y \mid (x, y) \in U^{lpd} \text{ και } y \in DR^D\} \leq n\}$
$\text{DataExactCardinality}(n \ U \ DR)$	$\{x \mid \#\{y \mid (x, y) \in U^{lpd} \text{ και } y \in DR^D\} = n\}$

Πίνακας 2.3: Διερμηνεύση περιγραφών.

Αξιώματα

Στον παρακάτω πίνακα φαίνεται πώς ορίζεται η ικανοποίηση των αξιωμάτων της OWL 2 σε μια διερμηνεύση I . Με το \circ υποδηλώνεται η σύνθεση δυαδικών σχέσεων.

Αξίωμα	Συνθήκη
SubClassOf(C D)	$C^{lc} \subseteq D^{lc}$
EquivalentClasses($C_1 \dots C_n$)	$C_j^{lc} = C_k^{lc}$ για κάθε $1 \leq j, k \leq n$
DisjointClasses($C_1 \dots C_n$)	$C_j^{lc} \cap C_k^{lc}$ είναι κενό για κάθε $1 \leq j, k \leq n$ και $j \neq k$
DisjointUnion(A $C_1 \dots C_n$)	$A^{lc} = C_1^{lc} \cup \dots \cup C_n^{lc}$ και $C_j^{lc} \cap C_k^{lc}$ είναι κενό για κάθε $1 \leq j, k \leq n$ και $j \neq k$
SubObjectPropertyOf(R S)	$R^{lpo} \subseteq S^{lpo}$
SubObjectPropertyOf (SubObjectPropertyChain($R_1 \dots R_n$) S)	$R_1^{lpo} \circ \dots \circ R_n^{lpo} \subseteq S^{lpo}$
EquivalentObjectProperties($R_1 \dots R_n$)	$R_j^{lpo} = R_k^{lpo}$ για κάθε $1 \leq j, k \leq n$
DisjointObjectProperties($R_1 \dots R_n$)	$R_j^{lpo} \cap R_k^{lpo}$ είναι κενό για κάθε $1 \leq j, k \leq n$ και $j \neq k$
ObjectPropertyDomain(R C)	$\{x \mid \exists y : (x, y) \in R^{lpo}\} \subseteq C^{lc}$
ObjectPropertyRange(R C)	$\{y \mid \exists x : (x, y) \in R^{lpo}\} \subseteq C^{lc}$
InverseObjectProperties(R S)	$R^{lpo} = \{(x, y) \mid (y, x) \in S^{lpo}\}$
FunctionalObjectProperty(R)	$(x, y_1) \in R^{lpo}$ και $(x, y_2) \in R^{lpo}$ υποδηλώνουν ότι $y_1 = y_2$
InverseFunctionalObjectProperty(R)	$(x_1, y) \in R^{lpo}$ και $(x_2, y) \in R^{lpo}$ υποδηλώνουν ότι $x_1 = x_2$
ReflexiveObjectProperty(R)	$x \in \Delta_i$ υποδηλώνει ότι $(x, x) \in R^{lpo}$
IrreflexiveObjectProperty(R)	$x \in \Delta_i$ υποδηλώνει ότι (x, x) δεν είναι στο R^{lpo}
SymmetricObjectProperty(R)	$(x, y) \in R^{lpo}$ υποδηλώνει ότι $(y, x) \in R^{lpo}$
AsymmetricObjectProperty(R)	$(x, y) \in R^{lpo}$ υποδηλώνει ότι (y, x) δεν είναι στο R^{lpo}
TransitiveObjectProperty(R)	$R^{lpo} \circ R^{lpo} \subseteq R^{lpo}$
SubDataPropertyOf(U V)	$U^{lpd} \subseteq V^{lpd}$
EquivalentDataProperties($U_1 \dots U_n$)	$U_j^{lpd} = U_k^{lpd}$ για κάθε $1 \leq j, k \leq n$
DisjointDataProperties($U_1 \dots U_n$)	$U_j^{lpd} \cap U_k^{lpd}$ είναι κενό για κάθε $1 \leq j, k \leq n$ και $j \neq k$
DataPropertyDomain(U C)	$\{x \mid \exists y : (x, y) \in U^{lpd}\} \subseteq C^{lc}$
DataPropertyRange(U DR)	$\{y \mid \exists x : (x, y) \in U^{lpd}\} \subseteq DR^D$
FunctionalDataProperty(U)	$(x, y_1) \in U^{lpd}$ και $(x, y_2) \in U^{lpd}$ υποδηλώνουν ότι $y_1 = y_2$
SameIndividual($a_1 \dots a_n$)	$a_j^{li} = a_k^{li}$ για κάθε $1 \leq j, k \leq n$
DifferentIndividuals($a_1 \dots a_n$)	$a_j^{li} \neq a_k^{li}$ για κάθε $1 \leq j, k \leq n$ και $j \neq k$
ClassAssertion(a C)	$a^{li} \in C^{lc}$
ObjectPropertyAssertion(R a b)	$(a^{li}, b^{li}) \in R^{lpo}$
NegativeObjectPropertyAssertion(R a b)	(a^{li}, b^{li}) δεν είναι στο R^{lpo}
DataPropertyAssertion(U a v)	$(a^{li}, v^D) \in U^{lpd}$
NegativeDataPropertyAssertion(U a v)	(a^{li}, v^D) δεν είναι στο U^{lpd}

Πίνακας 2.4: Ικανοποίηση των αξιωμάτων σε μια διερμηνευση.

Οντολογίες

Έστω ότι O είναι μια οντολογία OWL 2 με λεξιλόγιο V . Ισχύουν οι ακόλουθοι ορισμοί:

$H O$ είναι συνεπής αν υπάρχει μια διερμηνευση I που ικανοποιεί όλα τα αξιώματα της O .

Μια τέτοια διερμηνευση I καλείται μοντέλο της O .

Μια περιγραφή C είναι ικανοποιήσιμη ως προς την O , αν υπάρχει ένα μοντέλο I της O τέτοιο ώστε το C^I να μην είναι κενό.

Η O συνεπάγεται (entails) μια οντολογία O' ($O \models O'$), αν κάθε μοντέλο της O είναι και μοντέλο της O' . Επιπλέον, η O και O' είναι ισοδύναμες ($O \equiv O'$) αν η O συνεπάγεται την O' και η O' συνεπάγεται την O .

2.5.2. Νέα χαρακτηριστικά-Παραδείγματα

Η φαρμακευτική κοινότητα και η βιοϊατρική τεχνολογία έχουν ισχυρές επιρροές στο σχεδιασμό των οντολογικών γλωσσών. Ο πίνακας 2.5 παρουσιάζει πώς πολλά από τα νέα χαρακτηριστικά της OWL 1.1 βοηθούν στην ταξινόμηση χημικών ενώσεων [Villanueva-Rosales & Dumontier, 2007].

Δομή	Χαρακτηριστικό	Λεπτομέρειες	OWL 1.0	OWL 1.1
Ομάδα Αμινών	Υπαρξιακός Περιορισμός	έχειΔεσμόΜε <i>some</i> Άτομο	Ναι	Ναι
Υδρογον-άνθρακας	Καθολικός Περιορισμός	έχειΜέρος <i>only</i> Άνθρακα ή Υδρογόνο	Ναι	Ναι
Πρωτοταγείς Αμίνες	Προσδιορισμένος Περιορισμός Αριθμού	έχειΔεσμόΜε <i>exactly</i> 2 ΆτομοΥδρογόνου	Όχι	Ναι
--	Ομάδες Ξένων Συνόλων	100+ τύποι ξένων Ατόμων	Όχι	Ναι
--	Συμμετρία	έχειΔεσμόΜε	Ναι	Ναι
Κυκλικότητα	Τοπική Ανακλαστικότητα	συνδέεταιΜε "Self"	Όχι	Ναι
--	Αλυσίδες	έχειΜέρος ο βρίσκεταιΣε → βρίσκεταιΣε	Όχι	Ναι

Πίνακας 2.5: Χαρακτηριστικά της OWL επιθυμητά για την ταξινόμηση χημικών ενώσεων.

Στα συνέχεια δίνονται παραδείγματα χρήσης των νέων δυνατοτήτων της OWL 1.1 στα οποία υπερτερεί εκφραστικά έναντι των προκατόχων της. Παράλληλα, αναφέρονται οι συμπερασμοί που καθίστανται δυνατοί, λόγω των νέων κατασκευαστών. Τα παραδείγματα είναι γραμμένα στην κλασική RDF/XML σύνταξη.

Χαρακτηριστικά Ιδιοτήτων

Στο παρακάτω απόσπασμα κώδικα, η ιδιότητα `hasSon` (έχειΓιο) ορίζεται ως ξένη προς την ιδιότητα `hasDaughter` (έχειΚόρη). Επίσης η ιδιότητα `hasWife` (έχειΓυναίκα) ορίζεται ταυτόχρονα ως μη συμμετρική και ως μη ανακλαστική.

```
<owl:ObjectProperty rdf:about="#hasSon">
  <owl2:disjointObjectProperties rdf:resource="#hasDaughter"/>
</owl:ObjectProperty>

<owl:DatatypeProperty rdf:about="#hasAge">
  <rdf:type rdf:resource="#owl:FunctionalProperty"/>
</owl:DatatypeProperty>

<owl:ObjectProperty rdf:about="#hasWife">
  <rdf:type rdf:resource="#owl:FunctionalProperty"/>
```

```

<rdf:type rdf:resource="&owl;InverseFunctionalProperty"/>
<rdf:type rdf:resource="&owl2;IrreflexiveProperty"/>
<rdf:type rdf:resource="&owl2;AsymmetricProperty"/>
</owl:ObjectProperty>

```

Ως συνέπεια των παραπάνω δηλώσεων, ένας μηχανισμός συλλογισμού μπορεί να συμπεράνει ότι ένας άνθρωπος δεν μπορεί να έχει σαν γιο και κόρη το ίδιο παιδί (ξένες ιδιότητες). Επίσης, δεν μπορεί να έχει ως σύζυγο τον εαυτό του (μη ανακλαστικότητα) ούτε ότι, αν έχει γυναίκα τη Μαρία, η Μαρία μπορεί να έχει γυναίκα τον ίδιο (μη συμμετρικότητα). Μπορούν όμως να οριστούν και ιδιότητες που είναι συμμετρικές ή ανακλαστικές.

Είναι επίσης δυνατή η έκφραση *αυτοπαθών* εννοιών (local reflexivity), όπως π.χ. οι νάρκισσοι, δηλαδή οι άνθρωποι εκείνοι που αγαπούν τον εαυτό τους:

```

<owl:Class rdf:about="#Narcissist">
  <owl:equivalentClass>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <rdf:Description>
          <rdf:type
            rdf:resource="&owl2;SelfRestriction"/>
          <owl:onProperty rdf:resource="#loves"/>
        </rdf:Description>
        <rdf:Description rdf:about="#Person"/>
      </owl:intersectionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>

```

Μπορούν επίσης να εκφραστούν *αρνητικά γεγονότα* (negative facts) για ιδιότητες αντικειμένου και ιδιότητες δεδομένων:

```

<owl2:NegativeDataPropertyAssertion>
  <rdf:subject rdf:resource="#Jack"/>
  <rdf:predicate rdf:resource="#hasAge"/>
  <rdf:object rdf:datatype="&xsd;integer">53</rdf:object>
</owl2:NegativeDataPropertyAssertion>

```

Άρα ο Jack μπορεί να έχει ηλικία οποιαδήποτε άλλη εκτός από 53.

Περιορισμοί αριθμού και εύρος δεδομένων

Στο επόμενο παράδειγμα ορίζεται η κλάση Parent (Γονέας) ως οι άνθρωποι εκείνοι που έχουν τουλάχιστον ένα παιδί άνθρωπο (προσδιορισμένοι περιορισμοί αριθμού ή πληθικότητας). Η διαφορά με την παλιότερη OWL είναι ότι τώρα μπορεί να προσδιοριστεί η κλάση από την οποία πρέπει να προέρχονται οι πληρωτές του ρόλου πάνω στον οποίο εφαρμόζεται ο περιορισμός.

```

<owl:Class rdf:about="#Parent">
  <owl:equivalentClass>
    <owl:Class>

```

```

    <owl:intersectionOf rdf:parseType="Collection">
      <owl:Restriction>
        <owl:onProperty rdf:resource="#hasChild"/>
        <owl2:onClass rdf:resource="#Person"/>
        <owl:minCardinalityQ rdf:datatype=
          "&xsd;nonNegativeInteger">1
        </owl:minCardinalityQ>
      </owl:Restriction>
    </owl:intersectionOf>
    <rdf:Description rdf:about="#Person"/>
  </owl:Class>
</owl:equivalentClass>
</owl:Class>

```

Με τον τρόπο αυτό ορίζονται επακριβώς οι συνθήκες που θα κατατάξουν ένα άτομο στην κλάση Parent.

Ενώ η OWL (1.0) επέτρεπε την αντιστοίχιση ιδιοτήτων δεδομένων προς μία μόνο τιμή, η OWL 1.1 επιτρέπει την αναπαράσταση ψευδο-κλάσεων που αντιστοιχούν σε πεδία τιμών δεδομένων:

```

<owl:Class rdf:about="#Teenager">
  <owl:equivalentClass>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <rdf:Description rdf:about="#Person"/>
        <owl:Restriction>
          <owl:onProperty rdf:resource="#hasAge"/>
          <owl:someValuesFrom>
            <rdf:Description>
              <rdf:type rdf:resource=
                "&owl;DataRange"/>
              <owl2:minInclusive rdf:datatype=
                "&xsd;integer">13
              </owl2:minInclusive>
              <owl2:maxExclusive rdf:datatype=
                "&xsd;integer">20
              </owl2:maxExclusive>
              <owl2:onDataRange rdf:resource=
                "&xsd;integer"/>
            </rdf:Description>
          </owl:someValuesFrom>
        </owl:Restriction>
      </owl:intersectionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>

<rdf:Description rdf:about="#Bill">
  <f:hasAge rdf:datatype="http://www.w3.org/2001/XMLSchema#
  integer">13</f:hasAge>
</rdf:Description>

```

Άρα ο Bill συμπεραίνεται ότι είναι έφηβος (Teenager).

Αλυσίδες ιδιοτήτων

Οι ιδιότητες αντικειμένου μπορούν να συνδυαστούν σε αλυσίδες (role chains), κάτι που ήταν αδύνατον να εκφραστεί στην OWL 1.0. Έτσι μπορεί να μοντελοποιηθεί το γεγονός ότι τα παιδιά του συζύγου ενός ατόμου είναι και δικά του παιδιά, καθώς και η ιδιότητα έχειΘείο (hasUncle).

```
<rdf:Description>
  <rdf:type rdf:resource="&rdf;List"/>
  <rdf:first rdf:resource="#hasSpouse"/>
  <rdf:rest rdf:parseType="Collection">
    <rdf:Description rdf:about="#hasChild"/>
  </rdf:rest>
  <rdfs:subPropertyOf rdf:resource="#hasChild"/>
</rdf:Description>

<rdf:Description>
  <rdf:type rdf:resource="&rdf;List"/>
  <rdf:first rdf:resource="#hasParent"/>
  <rdf:rest rdf:parseType="Collection">
    <rdf:Description rdf:about="#hasBrother"/>
  </rdf:rest>
  <rdfs:subPropertyOf rdf:about="#hasUncle"/>
</rdf:Description>
```

Παρονομασία

Η παρονομασία αποτελεί μια εξασθενημένη μορφή μετα-μοντελοποίησης. Με την παρονομασία, τα ονόματα μπορούν να χρησιμοποιηθούν για διαφορετικούς σκοπούς: για παράδειγμα το Person μπορεί να δηλώνει ταυτόχρονα το όνομα μιας κλάσης και το όνομα ενός ατόμου. Οι διαφορετικές χρήσεις όμως ενός ονόματος είναι εντελώς ανεξάρτητες και, από σημασιακή άποψη, μπορούν να ειπωθούν ως διαφορετικά ονόματα, δηλ. Person-η-Κλάση και Person-το-Άτομο.

Με τον τρόπο αυτό χαλαρώνεται ο περιορισμός του διαχωρισμού των λεξιλογίων που ίσχυε στην OWL DL, για τις περιπτώσεις εκείνες που μπορεί να αποσαφηνιστεί η ακριβής χρήση ενός ονόματος και άρα δεν θα υπάρξει σημασιακή επιβάρυνση. Για παράδειγμα, μπορεί πάντα να αποσαφηνιστεί αν ένα όνομα χρησιμοποιείται ως κλάση ή ως άτομο, αλλά αυτό δεν μπορεί να γίνει πάντα στην περίπτωση που μια ιδιότητα χρησιμοποιείται τόσο ως ιδιότητα αντικειμένου, όσο και ως ιδιότητα δεδομένων.

Για παράδειγμα η Ελλάδα μπορεί να οριστεί ως *απαρίθμηση* (enumeration) των γεωγραφικών της διαμερισμάτων:

```
<owl:Class rdf:about="#Ελλάδα">
  <owl:oneOf rdf:parseType="Collection">
    <rdf:Description rdf:about="#Μακεδονία"/>
    <rdf:Description rdf:about="#Θράκη"/>
    <rdf:Description rdf:about="#Ηπειρος"/>
    <rdf:Description rdf:about="#Στερεά"/>
```

```

<rdf:Description rdf:about="#Θεσσαλία"/>
<rdf:Description rdf:about="#Πελοπόννησος"/>
<rdf:Description rdf:about="#Κρήτη"/>
<rdf:Description rdf:about="#Ιόνιο"/>
<rdf:Description rdf:about="#Αιγαίο"/>
</owl:oneOf>
</owl:Class>

```

Ταυτόχρονα όμως η Ελλάδα είναι μέλος της Ευρωπαϊκής Ένωσης:

```

<owl:Class rdf:about="#EUMembers"/>
<ex:EUMembers rdf:resource="&ex;Ελλάδα"/>

```

Στην πρώτη περίπτωση η Ελλάδα χρησιμοποιείται ως κλάση (και μας επιτρέπει να συμπεράνουμε π.χ. ότι η Μακεδονία *είναι* Ελλάδα), όμως στη δεύτερη χρησιμοποιείται ως άτομο. Δεν θα ήταν ισοδύναμο (ούτε προκύπτει από την παρονομασία) να πούμε ότι η Ελλάδα *είναι υποκλάση* της EUMembers, γιατί τότε θα προέκυπτε π.χ. ότι η Μακεδονία είναι μέλος της Ευρωπαϊκής Ένωσης.

2.6. Βατά Υποσύνολα

Όπως φαίνεται στο επόμενο κεφάλαιο, ο συλλογισμός στην OWL 1.1 (ή 2) είναι πολύ δύσκολος, αφού τα αντίστοιχα προβλήματα και αλγόριθμοι έχουν πολύ υψηλές πολυπλοκότητες. Για το λόγο αυτό, στο [Grau, Motik, Wu, Fokoue, & Lutz, 2008] προτείνονται τα λεγόμενα *βατά τεμάχια* ή *προφίλ* της OWL 1.1, όπως μετονομάστηκαν πρόσφατα. Τα βατά τεμάχια είναι ουσιαστικά υποσύνολα ή *διάλεκτοι* της γλώσσας OWL, με περιορισμούς στην εκφραστικότητα, ώστε να έχουν βελτιωμένες υπολογιστικές ιδιότητες. Τα υποσύνολα αυτά, είτε επαρκούν για συγκεκριμένα πεδία εφαρμογής είτε είναι πιο κοντά προς άλλους φορμαλισμούς, συμπληρωματικούς προς τις Λογικές Περιγραφής (βλ. και Κεφάλαιο 3). Στην ενότητα αυτή παρουσιάζονται τα ακόλουθα προφίλ:

- Το προφίλ *EL++* επιτρέπει πολυωνυμικούς αλγόριθμους για τις παρακάτω υπηρεσίες συλλογισμού: συνέπεια, ταξινόμηση και έλεγχο στιγμιότυπου. Για αυτό το προφίλ είναι διαθέσιμοι εξειδικευμένοι αλγόριθμοι συλλογισμού και έχει δειχθεί ότι μπορούν να υλοποιηθούν με διαβαθμίσιμο τρόπο.
- Στο προφίλ *DL-Lite*, η απάντηση συζευγμένων ερωτημάτων μπορεί να υλοποιηθεί χρησιμοποιώντας παραδοσιακά συστήματα σχεσιακών βάσεων δεδομένων, εφόσον το ερώτημα επαναγραφεί. Συγκεκριμένα, αυτό συνεπάγεται ότι η απάντηση συζευγμένων ερωτημάτων έχει πολυπλοκότητα δεδομένων LOGSPACE. Όπως και στην *EL++*, έτσι κι εδώ υπάρχουν πολυωνυμικοί αλγόριθμοι για την συνέπεια, την ταξινόμηση και τον έλεγχο στιγμιότυπων.
- Το προφίλ *OWL-R* επιτρέπει την υλοποίηση αλγορίθμων συλλογισμού εφαρμόζοντας προς τα εμπρός ακολουθία εκτέλεσης κανόνων (forward chaining rules) στις τριπλέτες που προκύπτουν από την γραφή σε RDF. Συγκεκριμένα, μπορούν να χρησιμοποιηθούν τέτοιες προσεγγίσεις κανόνων προκειμένου να υπολογιστεί η συνέπεια, η ταξινόμηση και ο έλεγχος στιγμιότυπων σε πολυωνυμικό

χρόνο. Στην OWL-R η απάντηση συζευγμένων ερωτημάτων μπορεί να υλοποιηθεί επεκτείνοντας ένα τυπικό σύστημα σχεσιακής βάσης δεδομένων με κανόνες. Η OWL-R εμφανίζεται σε δύο διαφοροποιήσεις, την *OWL-R DL* και την *OWL-R Full*.

Τα προφίλ *EL++*, *DL-Lite* και *OWL-R DL* ορίζονται, θέτοντας περιορισμούς στη *σύνταξη* της OWL 2. Αντιθέτως, η *OWL-R Full* ορίζεται επιβάλλοντας περιορισμούς στην *σημασιολογία* της OWL 2.

Οι υπηρεσίες συλλογισμού που αναφέρονται στην περιγραφή των προφίλ ορίζονται ως ακολούθως:

- *Συνέπεια οντολογίας* (ontology consistency): δοσμένης μιας οντολογίας, αποφασίζεται αν αυτή είναι αντιφατική ή όχι (δηλαδή, αν έχει κάποιο μοντέλο). Η συνέπεια οντολογίας ανάγεται στον έλεγχο της ικανοποιησιμότητας των κλάσεων (εννοιών) που την αποτελούν.
- *Ικανοποιησιμότητα έννοιας* (concept satisfiability): δοσμένης μιας οντολογίας O και μιας κλάσης A , ελέγχεται αν υπάρχει (έστω ένα) μοντέλο της O στο οποίο η διερμηνευση της A δεν είναι κενή.
- *Ταξινόμηση οντολογίας* (ontology classification): δοσμένης μιας οντολογίας, υπολογίζονται οι σχέσεις υποκλάσης μεταξύ των κλάσεων που την αποτελούν. Η ταξινόμηση ανάγεται στο έλεγχο των σχέσεων υπαγωγής.
- *Υπαγωγή έννοιας* (concept subsumption): δοσμένης μιας οντολογίας O και δύο κλάσεων A, B ελέγχεται αν η διερμηνευση της A είναι υποσύνολο της διερμηνευσης της B σε κάθε μοντέλο της O .
- *Έλεγχος στιγμιότυπου* (instance checking): δοσμένης μιας οντολογίας, ενός ατόμου a και μιας κλάσης A , αποφασίζεται αν το a είναι στιγμιότυπο της A , για κάθε μοντέλο της οντολογίας.
- *Απάντηση συζευγμένων ερωτημάτων* (conjunctive query answering): δοσμένης μιας οντολογίας O και μιας συζευγμένης επερώτησης q , επιστρέφονται όλες τις πλειάδες των ατόμων της O που ταιριάζουν στην q .

Οι αντίστοιχες πολυπλοκότητες διακρίνονται ως εξής:

- *Πολυπλοκότητα δεδομένων* (data complexity): η πολυπλοκότητα που μετράται ως προς το πλήθος των γεγονότων μιας οντολογίας, δηλαδή των δηλώσεων που αφορούν στα άτομα (ABox).
- *Ταξινομική πολυπλοκότητα* (taxonomic complexity): η πολυπλοκότητα που μετράται ως προς το πλήθος των αξιωμάτων μιας οντολογίας, δηλαδή των αξιωμάτων που αφορούν στις σχέσεις μεταξύ των κλάσεων και των ιδιοτήτων (TBox και RBox).
- *Πολυπλοκότητα επερώτησης* (query complexity): η πολυπλοκότητα που μετράται ως προς το μέγεθος του ερωτήματος.
- *Συνδυασμένη πολυπλοκότητα* (combined complexity): η πολυπλοκότητα που μετράται ως προς το συνολικό πλήθος γεγονότων και αξιωμάτων. Στην περίπτωση απάντησης συζευγμένων ερωτημάτων, συμπεριλαμβάνεται και το μέγεθος του ερωτήματος.

Εκτός από αυτά που αναφέρονται εδώ, υπάρχουν και πολλά άλλα πιθανά προφίλ για την OWL 2. Για παράδειγμα, η έκδοση OWL Lite της OWL 1.0 μπορεί να θεωρηθεί ως ένα

προφίλ της OWL 2. Δεδομένου όμως ότι οι υπολογιστικές της ιδιότητες αποδείχθηκαν χειρότερες από εκείνες που είχαν πιθανολογηθεί όταν εγκρίθηκε η σύσταση για την OWL 1.0 (για παράδειγμα, η ικανοποιησιμότητα και η υπαγωγή είναι EXP-complete), δεν γίνεται άμεση αναφορά στην OWL Lite. Η OWL DL μπορεί επίσης να θεωρηθεί ως ένα προφίλ της OWL 2.

2.6.1. EL++

Το προφίλ *EL++* είναι σχεδιασμένο ως το μέγιστο δυνατό υποσύνολο της OWL 2:

- το οποίο συλλαμβάνει την εκφραστική ισχύ που χρησιμοποιείται από πολλές, μεγάλης κλίμακας οντολογίες και
- για το οποίο τα ακόλουθα προβλήματα συλλογισμού μπορούν να αποφασιστούν σε πολυωνυμικό χρόνο: ικανοποιησιμότητα, υπαγωγή, ταξινόμηση, και έλεγχος στιγμιότυπων.

Στην πραγματικότητα, το προφίλ αυτό βασίζεται στην Λογική Περιγραφής *EL++*, για την οποία υπάρχει βέλτιστος πολυωνυμικός αλγόριθμος ταμπλό και υλοποιείται στο σύστημα CEL (<http://lat.inf.tu-dresden.de/systems/cel/>) [Baader, Brandt, & Lutz, 2005]. Μια βασική σχεδιαστική αρχή του *EL++* είναι να επικεντρώνει στους κατασκευαστές κλάσεων *ObjectIntersectionOf* και *ObjectSomeValuesFrom*, αλλά να παρέχει την *ObjectAllValuesFrom* μόνο με τη μορφή περιορισμών πεδίου τιμών. Πολλές οντολογίες στον τομέα της βιοϊατρικής, όπως η SNOMED CT, εμπίπτουν σε αυτό το προφίλ.

Το προφίλ *EL++* έχει τα ακόλουθα γνωρίσματα:

- Υπαρξιακή ποσοτικοποίηση σε μια κλάση (*ObjectSomeValuesFrom*) ή πεδίο τιμών δεδομένων (*DataSomeValuesFrom*).
- Υπαρξιακή ποσοτικοποίηση σε ονοματικά (*ObjectHasValue*) ή σταθερές (*DataHasValue*).
- Αυτοπάθεια (*ObjectExistsSelf*).
- Απαριθμήσεις που εμπλέκουν ένα μόνο στοιχείο (*ObjectOneOf*) ή μία σταθερά (*DataOneOf*).
- Τομή κλάσεων (*ObjectIntersectionOf*).
- Συμπερίληψη κλάσεων (*SubClassOf*).
- Ισοδυναμία κλάσεων (*EquivalentClasses*).
- Ξένες κλάσεις (*DisjointClasses*).
- Συμπερίληψη ιδιοτήτων αντικειμένου (*SubObjectPropertyOf*) που μπορεί να περιλαμβάνει και αλυσίδες, καθώς και συμπερίληψη ιδιοτήτων δεδομένων (*SubDataPropertyOf*).
- Ισοδυναμία ιδιοτήτων (*EquivalentObjectProperties* και *EquivalentDataProperties*).
- Μεταβατικές ιδιότητες αντικειμένου (*TransitiveObjectProperty*).
- Ανακλαστικές ιδιότητες αντικειμένου (*ReflexiveObjectProperty*).
- Περιορισμοί πεδίου ορισμού (*ObjectPropertyDomain* και *DataPropertyDomain*).
- Περιορισμοί πεδίου τιμών (*ObjectPropertyRange* και *DataPropertyRange*).

- Γεγονότα (`SameIndividual`, `DifferentIndividuals`, `ClassAssertion`, `ObjectPropertyAssertion`, `DataPropertyAssertion`, `NegativeObjectPropertyAssertion`, `NegativeDataPropertyAssertion`).
- Συναρτησιακές ιδιότητες δεδομένων (`FunctionalDataProperty`).
- ένα περιορισμένο εύρος τύπων δεδομένων.

Τα παρακάτω γνωρίσματα δεν υπάρχουν στο *EL++*, γιατί οδηγούν σε δυσεπιλυσιμότητα:

- Καθολική ποσοτικοποίηση σε μια κλάση (`ObjectAllValuesFrom`) ή πεδίο τιμών δεδομένων (`DataAllValuesFrom`).
- Περιορισμούς πληθικότητας (`ObjectMaxCardinality`, `ObjectMinCardinality`, `ObjectExactCardinality`, `DataMaxCardinality`, `DataMinCardinality`, και `DataExactCardinality`).
- Ένωση (`ObjectUnionOf`, `DisjointUnion`).
- Συμπλήρωμα (`ObjectComplementOf`).
- Απαριθμήσεις με πολλά στοιχεία (`ObjectOneOf`, `DataOneOf`).
- Ξένες ιδιότητες (`DisjointObjectProperties`, `DisjointDataProperties`).
- Μη ανακλαστικές ιδιότητες αντικειμένου (`IrreflexiveObjectProperty`).
- Αντίστροφες ιδιότητες αντικειμένου (`InverseObjectProperties`).
- Συναρτησιακές ιδιότητες αντικειμένου (`FunctionalObjectProperty`).
- Συμμετρικές ιδιότητες αντικειμένου (`SymmetricObjectProperty`).
- Μη συμμετρικές ιδιότητες αντικειμένου (`AsymmetricObjectProperty`).

2.6.2. DL-Lite

Η DL-Lite είναι ένα συντακτικό προφίλ της OWL 2 στο οποίο μπορούν να εφαρμοστούν συνεπείς και πλήρεις αλγόριθμοι συλλογισμού με πολυπλοκότητα δεδομένων LOGSPACE. Το προφίλ αυτό βασίζεται στη Λογική Περιγραφής DL-Lite [Calvanese D. , Giacomo, Lembo, Lenzerini, & Rosati, 2005], η οποία αποτελεί υποσύνολο της OWL Lite και επιτυγχάνει την χαμηλή αυτή πολυπλοκότητα δεδομένων, ως εξής: Ο αλγόριθμος δέχεται ένα συζευγμένο ερώτημα q το οποίο το μετασχηματίζει, λαμβάνοντας υπόψη το TBox, σε ερώτημα SQL. Το ABox είναι αποθηκευμένο σε σχεσιακή βάση δεδομένων και οι απαντήσεις στο q λαμβάνονται μέσω της βάσης. Επειδή ο μετασχηματισμός του q δεν εξαρτάται από το ABox, η πολυπλοκότητα δεδομένων αποτίμησης και απάντησης του q είναι LOGSPACE, όσο δηλαδή στα κλασικά σχεσιακά συστήματα. Αξίζει επίσης να σημειωθεί ότι $\text{LOGSPACE} \subseteq \text{P}$ [Paradimitriou, 1995].

Επειδή η γλώσσα είναι τόσο απλοϊκή, ο διαχωρισμός αυτός μεταξύ TBox και ABox είναι συνεπής και επιπλέον οι ταξινομικές λειτουργίες μπορούν να διεξαχθούν αποδοτικά. Η παραπάνω προσέγγιση υλοποιείται στο σύστημα QuOnto (<http://www.dis.uniroma1.it/~quonto/>). Η DL-Lite περιλαμβάνει τα περισσότερα από τα βασικά γνωρίσματα των εννοιολογικών μοντέλων, όπως τα διαγράμματα κλάσεων UML και τα διαγράμματα ER και είναι κατάλληλη για την αποδοτική απάντηση ερωτημάτων σε μεγάλο όγκο δεδομένων.

Αρκετές διαφοροποιήσεις της DL-Lite έχουν περιγραφεί στη βιβλιογραφία. Εκείνη που παρουσιάζεται εδώ είναι η DL-Lite_r , καθώς επιτρέπει αξιώματα συμπερίληψης ιδιοτήτων. Οι άλλες διαφοροποιήσεις παραλείπουν τα αξιώματα συμπερίληψης προς όφελος της ύπαρξης συναρτησιακών και αντιστρόφως συναρτησιακών ιδιοτήτων αντικειμένου.

Οι παρακάτω κατασκευές μπορούν να χρησιμοποιηθούν στον ορισμό υποκλάσεων:

- Κλάσεις OWL, εκτός από την `owl:Thing`.
- Υπαρξιακή ποσοτικοποίηση (`ObjectSomeValuesFrom`), όπου η μόνη κλάση που επιτρέπεται είναι η `owl:Thing`.

Οι ακόλουθες κατασκευές μπορούν να χρησιμοποιηθούν στον ορισμό υπερκλάσεων:

- Κλάσεις OWL.
- Υπαρξιακή ποσοτικοποίηση (`ObjectSomeValuesFrom`), όπου η μόνη κλάση που επιτρέπεται είναι η `owl:Thing`.
- Συμπλήρωμα (`ObjectComplementOf`).

Όλα τα αξιώματα κλάσης στην DL-Lite περιορίζονται με ένα τρόπο που είναι συμβατός με αυτούς τους περιορισμούς. Για παράδειγμα, τα αξιώματα για το πεδίο ορισμού και τιμών των ιδιοτήτων επιτρέπεται να αναφέρονται μόνο στις υπερκλάσεις που αναφέρθηκαν παραπάνω.

Επιπρόσθετα, η DL-Lite επιτρέπει τα εξής αξιώματα ιδιοτήτων και γεγονότα:

- Αντίστροφες ιδιότητες αντικειμένου (`InverseObjectProperties`).
- Συμπερίληψη ιδιοτήτων (`SubObjectPropertyOf`), χωρίς αλυσίδες.
- Ισοδυναμία ιδιοτήτων (`EquivalentObjectProperties`).
- Πεδίο ορισμού ιδιοτήτων (`ObjectPropertyDomain`).
- Πεδίο τιμών ιδιοτήτων (`ObjectPropertyRange`).
- Ξένες ιδιότητες (`DisjointObjectProperties`).
- Συμμετρικές ιδιότητες (`SymmetricObjectProperty`).
- Γεγονότα (`SameIndividual`, `DifferentIndividuals`, `ClassAssertion`, `ObjectPropertyAssertion`).

Τα παρακάτω γνωρίσματα της OWL 2 δεν υπάρχουν στην DL-Lite:

- Υπαρξιακή ποσοτικοποίηση σε κλάση (`ObjectSomeValuesFrom`).
- Αυτοπάθεια (`ObjectExistsSelf`).
- Υπαρξιακή ποσοτικοποίηση σε ονοματικό (`ObjectHasValue`).
- Ονοματικά (`ObjectOneOf`).
- Καθολική ποσοτικοποίηση σε κλάση (`ObjectAllValuesFrom`).
- Περιορισμοί πληθικότητας (`ObjectMaxCardinality`, `ObjectMinCardinality`, `ObjectExactCardinality`).
- Τομή (`ObjectUnionOf`, `DisjointUnion`).
- Συμπερίληψη ιδιοτήτων (`SubObjectPropertyOf`) που να περιλαμβάνει αλυσίδες.
- Μεταβατικότητα (`TransitiveObjectProperty`).
- Ανακλαστικότητα (`ReflexiveObjectProperty`).
- Μη ανακλαστικότητα (`IrreflexiveObjectProperty`).
- Μη συμμετρικότητα (`AsymmetricObjectProperty`).
- (Αντιστρόφως) συναρτησιακές ιδιότητες (`FunctionalObjectProperty` και `InverseFunctionalObjectProperty`).

2.6.3. OWL-R

Η OWL-R αποτελεί ένα προφίλ της OWL 2 το οποίο, χρησιμοποιώντας τεχνολογίες βασισμένες σε κανόνες, επιτρέπει διαβαθμισμό συλλογισμό. Το εν λόγω προφίλ έχει σχεδιαστεί ώστε να μη χρειάζεται να συμπεραίνεται η ύπαρξη ατόμων, τα οποία δεν είναι ρητά δηλωμένα μέσα στη βάση γνώσης. Αυτός ο σχεδιαστικός στόχος καθιστά δυνατή μια απευθείας μετάφραση των συνθηκών σημασιολογίας της OWL σε κανόνες, για την οποία οι περισσότεροι μηχανισμοί συλλογισμού που βασίζονται σε κανόνες τερματίζουν σε πεπερασμένο χρόνο.

Η ευελιξία αποτελεί έναν ακόμα σχεδιαστικό στόχο για την OWL-R. Αφενός, η OWL-R μπορεί να εξυπηρετήσει εφαρμογές που μπορούν να ανταλλάξουν την πλήρη εκφραστικότητα της γλώσσας με την αποδοτικότητα. Αφετέρου, η OWL-R μπορεί επίσης να εξυπηρετήσει εφαρμογές RDF(S) που έχουν ανάγκη από την επιπλέον εκφραστικότητα της OWL. Για το λόγο αυτό, γίνεται αναφορά σε δύο διαφοροποιήσεις τις OWL-R.

Η πρώτη παραλλαγή της OWL-R είναι η *OWL-R DL* η οποία στοχεύει σε χρήστες που προτίθενται να θυσιάσουν λίγη εκφραστικότητα, προκειμένου να μπορούν να πραγματοποιήσουν συλλογισμό χρησιμοποιώντας συστήματα που βασίζονται σε κανόνες. Η OWL-R DL ορίζεται ως ένα συντακτικό υποσύνολο της OWL 2, δηλαδή επιβάλλει συντακτικούς περιορισμούς στα αξιώματα της γλώσσας. Για παράδειγμα, μια κλάση C_1 της OWL δεν μπορεί να οριστεί ως υποκλάση της ένωσης των κλάσεων C_2 και C_3 .

Η δεύτερη παραλλαγή της OWL-R είναι η *OWL-R Full* η οποία στοχεύει σε χρήστες του RDF(S) που θέλουν να το εμπλουτίσουν με πρόσθετες δομές. Επομένως, οι οντολογίες OWL-R Full είναι γραφήματα RDF που διερμηνεύονται κάτω από μια ασθενέστερη εκδοχή των εκτασιακών συνθηκών σημασιολογίας της OWL 2.

Η OWL-R DL εκφράζει στην πραγματικότητα τον φορμαλισμό των προγραμμάτων Λογικών Περιγραφής (βλ. ενότητα 3.2.2) που έχει πολυωνυμική πολυπλοκότητα, αλλά είναι ασθενέστερος εκφραστικά ακόμα και από την OWL Lite. Συνοπτικά, μια οντολογία OWL-R DL μπορεί να χρησιμοποιήσει τις περισσότερες από τις δομές της γλώσσας OWL 2 εκτός από τις ακόλουθες:

- Cardinality.
- minCardinality.
- NegativeObjectPropertyAssertion.
- NegativeDataPropertyAssertion.
- owl:complementOf.

Οι περιορισμοί αυτοί είναι ως συνέπεια του γεγονότος ότι στην *def-Horn* λογική δεν επιτρέπεται η ισότητα και η άρνηση (βλ. ενότητα 3.2.2). Επίσης, δεν μπορούν να χρησιμοποιηθούν ελεύθερα σε όλα τα σημεία των αξιωμάτων όλες οι κατασκευές της OWL-R DL. Για παράδειγμα, στο αξίωμα *SubClassOf*, η χρήση των κατασκευών στην αριστερή και στη δεξιά πλευρά του αξιώματος πρέπει να ακολουθεί το μοτίβο του πίνακα 2.6 (κάθε κατασκευή της αριστερής πλευράς μπορεί να συνδυάζεται με οποιαδήποτε κατασκευή της δεξιάς πλευράς).

Σε αντίθεση με την OWL-R DL, στην OWL-R Full δεν υπάρχουν συντακτικοί περιορισμοί στον τρόπο που μπορούν να χρησιμοποιηθούν οι κατασκευές της γλώσσας: κάθε γράφημα RDF σχηματίζει μια έγκυρη οντολογία OWL-R Full. Εντούτοις, η σημασιολογία των δομών

της γλώσσας αποδυναμώνεται στην OWL-R Full προκειμένου να μιμηθεί τα μοτίβα χρήσης της OWL-R DL. Για παράδειγμα, στην OWL, μια OWL κλάση C_1 είναι υποκλάση της C_2 αν και μόνον αν η επέκταση της C_1 είναι ένα υποσύνολο της επέκτασης της C_2 . Στην OWL-R Full, αυτή η συνθήκη του "αν και μόνον αν" περιορίζεται σε "μόνο αν".

Αριστερή Πλευρά	Δεξιά Πλευρά
μια κλάση OWL	μια κλάση OWL
μια ονοματική κλάση (OneOf)	τομή κλάσεων (ObjectIntersectionOf)
τομή κλάσεων (ObjectIntersectionOf)	υπαρξιακή ποσοτικοποίηση σε κλάση (ObjectAllValuesFrom)
ένωση κλάσεων (ObjectUnionOf)	περιορισμοί αριθμού ≤ 1 (ObjectMaxCardinality 1)
υπαρξιακή ποσοτικοποίηση σε κλάση (ObjectSomeValuesFrom)	υπαρξιακή ποσοτικοποίηση σε ονοματικό (ObjectHasValue)
υπαρξιακή ποσοτικοποίηση σε ονοματικό (ObjectHasValue)	

Πίνακας 2.6: Συντακτικός περιορισμός αξιωμάτων SubClassOf στην OWL-R DL.

Η εξασθενημένη αυτή μορφή συνεπαγωγής βασίζεται στην λεγόμενη *συνεπαγωγή RD^** (RD^* -entailment) [ter Horst, 2005] και επιβάλλεται στην OWL-R Full προκειμένου να είναι πλήρως συμβατή (όχι ασθενέστερη, αλλά επέκταση) με το RDF(S), που έχει ασθενέστερη σημασιολογία από την OWL. Η λογική που προκύπτει με τον τρόπο αυτό είναι γνωστή και ως *OWL-Horst* και υποστηρίζεται αποδοτικά από το σύστημα OWLIm (βλ. ενότητα 4.5.3). Η πολυπλοκότητα της συνεπαγωγής RD^* είναι NP-complete, ενώ γίνεται πολυωνυμική, αν δεν περιλαμβάνονται κενοί κόμβοι (π.χ. μη ρητώς δηλωμένα άτομα).

2.6.4. Υπολογιστικές Ιδιότητες

Ο πίνακας 2.7 συνοψίζει τα γνωστά αποτελέσματα πολυπλοκότητας για τις OWL 2, OWL DL, EL++, DL-Lite, και OWL-R. Αν η πολυπλοκότητα ενός προβλήματος χαρακτηρίζεται ως *ανοιχτή* (Open), με ένα αστεράκι (*), σημαίνει ότι η αποφασισιμότητά της είναι ακόμα ένα ζήτημα που δεν έχει κλείσει. Αν, ωστόσο, το αστεράκι παραλείπεται, τότε το πρόβλημα είναι γνωστό ότι είναι αποφασίσιμο, αλλά δεν έχουν ακόμη τεθεί τα ακριβή όρια πολυπλοκότητας.

Για τις γλώσσες που είναι *προτασιακά κλειστές* (δηλαδή που υποστηρίζουν, άμεσα ή έμμεσα, τομή, ένωση και συμπλήρωμα κλάσεων), όπως η OWL, τα προβλήματα της συνέπειας, ικανοποιησιμότητας, υπαγωγής και ελέγχου στιγμιότυπων μπορούν να αναχθούν το ένα στο άλλο. Κανένα όμως από τα προφίλ που περιγράφηκαν δεν είναι προτασιακά κλειστό, άρα τα προβλήματα αυτά μπορεί να έχουν διαφορετικές πολυπλοκότητες και να απαιτούν διαφορετικούς αλγορίθμους.

Γλώσσα	Προβλήματα Συλλογισμού	Ταξινομική Πολυπλοκότητα	Πολυπλοκότητα Δεδομένων	Πολυπλοκότητα Επερώτησης	Συνδυασμένη Πολυπλοκότητα
OWL	Συνέπεια, Ικαν/τα, Υπαγωγή, Έλεγχος Στιγμιότυπων	2NExp-complete	Ανοιχτή (NP-hard)	–	2NExp-complete
	Απάντηση Συζευγμένων Ερωτημάτων	Ανοιχτή*	Ανοιχτή*	Ανοιχτή*	Ανοιχτή*

OWL DL	Συνέπεια, Ικαν/τα, Υπαγωγή, Έλεγχος Στιγμιότυπων	NEXP- complete	Ανοιχτή (NP-Hard)	–	NEXP-complete
	Απάντηση Συζευγμένων Ερωτημάτων	Ανοιχτή*	Ανοιχτή*	Ανοιχτή*	Ανοιχτή*
EL++	Συνέπεια, Ικαν/τα, Υπαγωγή, Έλεγχος Στιγμιότυπων	P-complete	P-complete	–	P-complete
	Απάντηση Συζευγμένων Ερωτημάτων	P-complete	P-complete	NP-complete	PSPACE- complete
DL-Lite	Συνέπεια, Ικαν/τα, Υπαγωγή, Έλεγχος Στιγμιότυπων	P	LOGSPACE	–	P
	Απάντηση Συζευγμένων Ερωτημάτων	P	LOGSPACE	NP-complete	NP-complete
OWL-R	Συνέπεια, Ικαν/τα, Υπαγωγή, Έλεγχος Στιγμιότυπων	P-complete	P-complete	–	P-complete
	Απάντηση Συζευγμένων Ερωτημάτων	P-complete	P-complete	NP-complete	NP-complete

Πίνακας 2.7: Η πολυπλοκότητα των προφίλ της OWL 2.

Στην DL-Lite, ο έλεγχος στιγμιότυπων και η απάντηση συζευγμένων ερωτημάτων μπορεί να γίνει αξιοποιώντας την τεχνολογία σχεσιακών βάσεων δεδομένων, δηλαδή μέσω μετάφρασης σε επερωτήσεις SQL. Το γεγονός ότι η πολυπλοκότητα δεδομένων προχωράει πέρα από LOGSPACE, σημαίνει ότι η απάντηση ερωτημάτων και ο έλεγχος στιγμιότυπων απαιτούν μηχανές πιο ισχυρές από αυτές που προσφέρουν οι τεχνολογίες για τις σχεσιακές βάσεις δεδομένων.

3. ΣΥΛΛΟΓΙΣΜΟΣ ΣΤΟ ΣΗΜΑΝΤΙΚΟ ΙΣΤΟ

3.1. Εισαγωγή

Ο συλλογισμός είναι, όπως φάνηκε στο κεφάλαιο 1, κρίσιμης σημασίας για την SWKD. Ο συλλογισμός στην περίπτωση του Σημαντικού Ιστού μπορεί να οριστεί ως ακολούθως:

Συλλογισμός (*reasoning*) είναι ο συνδυασμός των γεγονότων που είναι εκφρασμένα μέσα σε μία οντολογία, προκειμένου να ελεγχθεί η ορθότητα ενός άλλου γεγονότος (π.χ. μιας ερώτησης) ή να παραχθούν νέα γεγονότα.

Η διαδικασία εξαγωγής νέων γεγονότων καλείται και *συμπερασμός* (*inferencing*). Στην περίπτωση που ο συλλογισμός οδηγεί σε αφαίρεση γεγονότων από την οντολογία, και όχι μόνο σε προσθήκη, κάνουμε λόγο για *μη μονοτονικό* συλλογισμό, που όμως δεν υποστηρίζεται μέχρι στιγμής από τις γλώσσες του Σημαντικού Ιστού. Για παράδειγμα, το γεγονός ότι όλα τα πουλιά πετούν, μας οδηγεί στο συμπέρασμα ότι αν το *x* είναι Πουλί, τότε πρέπει να πετάει. Αν όμως αργότερα προστεθεί το γεγονός ότι το *x* είναι Πιγκουίνος, τότε το προηγούμενο *προκαθορισμένο* γεγονός (*default*) πρέπει να αναιρεθεί, τουλάχιστον για τη συγκεκριμένη περίπτωση. Ο συλλογισμός ωστόσο θα συνεχίσει κανονικά, ενώ στην περίπτωση π.χ. των Λογικών Περιγραφής, θα προέκυπτε αποτέλεσμα μη ικανοποιησιμότητας της βάσης γνώσης.

Ο συλλογισμός είναι επομένως φυσικό να εξαρτάται από τον τρόπο με τον οποίο τα γεγονότα αναπαρίστανται, σε μορφή συμβατή με τη μηχανή. Ο υποκείμενος λοιπόν *φορμαλισμός* που ακολουθείται σε κάθε περίπτωση, καθορίζει σε μεγάλο βαθμό τις δυνατότητες του συλλογισμού. Από την άλλη, η φύση ενός προβλήματος αναπαράστασης γνώσης (π.χ. για ένα ορισμένο πεδίο) μπορεί να είναι τέτοια, ώστε να υπαγορεύει τη θέσπιση ενός νέου φορμαλισμού, που κωδικοποιεί πιο άμεσα και φυσικά το πρόβλημα, για μηχανική χρήση.

Στην πορεία της Τεχνητής Νοημοσύνης, πολλοί φορμαλισμοί (όπως π.χ. τα Σημαντικά Δίκτυα) έχουν οριστεί για αυτόν τον λόγο, ενώ αργότερα βρέθηκε ότι έχουν (ή δεν έχουν) επιθυμητές υπολογιστικές ιδιότητες. Πολλές μέθοδοι αναπαράστασης έχουν ως βάση την Κατηγορηματική Λογική ή Λογική Πρώτης Τάξης και αποτελούν υποσύνολα αυτής. Άλλες πάλι μπορεί να έχουν κοινά στοιχεία, αλλά καμία να μην είναι υποσύνολο της άλλης, να είναι δηλαδή *ορθογώνιες*, όπως, για παράδειγμα, οι Λογικές Περιγραφής με τη λογική Horn [Αντωνίου & Harmelen, 2004]. Σε κάθε περίπτωση όμως, η αύξηση της εκφραστικότητας (της γενικότητας) ενός φορμαλισμού, έρχεται συνήθως με το τίμημα της αύξησης της πολυπλοκότητας του συλλογισμού.

Στο κεφάλαιο αυτό ασχολούμαστε κυρίως με τον φορμαλισμό των Λογικών Περιγραφής, ως τον πιο πρόσφορο για υποστήριξη συλλογισμού στο Σημαντικό Ιστό, από άποψη εκφραστικότητας και αποδοτικότητας. Εισάγουμε την ορολογία που χρησιμοποιείται στις λογικές αυτές, τα αντίστοιχα προβλήματα συλλογισμού, τις διαδικασίες απόφασης ταμπλό και ορισμένες τεχνικές υλοποίησης και βελτιστοποίησης αλγορίθμων ταμπλό. Αρχικά όμως γίνεται μια επισκόπηση των διαφόρων εναλλακτικών φορμαλισμών που μπορούν να

χρησιμοποιηθούν για συλλογισμό με οντολογίες Ιστού ή/και τις επεκτάσεις τους. Το κεφάλαιο ολοκληρώνεται με την παρουσίαση της Λογικής Περιγραφής *SROIQ* που συνιστά την γλώσσα οντολογιών Ιστού OWL 1.1.

Περισσότερες πληροφορίες για τους, σχετικούς με το Σημαντικό Ιστό, φορμαλισμούς μπορούν να βρεθούν στο [Κουτσομητρόπουλος, 2004], στα περισσότερα εγχειρίδια Τεχνητής Νοημοσύνης (π.χ. [Russell & Norvig, 2005; Βλαχάβας, Κεφαλάς, Βασιλειάδης, Κόκκορας, & Σακελλαρίου, 2006]), καθώς και στο [Brachman & Levesque, 2004].

Τέλος, για λόγους πληρότητας, θα πρέπει να αναφερθούν οι προσεγγίσεις για *ασαφή* (fuzzy) συλλογισμό σε οντολογίες Ιστού, όπου τόσο στα δηλωμένα, όσο και στα εξαγόμενα γεγονότα ανατίθεται ένας βαθμός ασάφειας (δηλ. μια διαβάθμιση, από 0 έως 1) [Stoilos, Simou, Stamou, & Kollias, 2006]. Οι συγγραφείς δίνουν συνεπείς και πλήρεις αλγόριθμους συλλογισμού για την λογική *SHIN* (OWL Lite), εκτεταμένη με βαθμούς ασάφειας [Stoilos, Stamou, Pan, Tzouvaras, & Horrocks, 2007].

3.2. Φορμαλισμοί και Τεχνικές

Παρόλο που η OWL έχει στενή συγγένεια με τις Λογικές Περιγραφές, η επιλογή ενός τέτοιου συστήματος για ανακάλυψη γνώσης σε οντολογίες δεν αποτελεί μονόδρομο. Οι κυριότεροι εναλλακτικοί προς τις Λογικές Περιγραφές φορμαλισμοί που μπορούν να χρησιμοποιηθούν για συλλογισμό με οντολογίες περιλαμβάνουν:

- *Πλήρη Λογική Πρώτης Τάξης*: Οι Λογικές Περιγραφές αποτελούν εκφραστικό υποσύνολο της λογικής αυτής.
- *Κανόνες*: Υποσύνολα της OWL είναι συμβατά (αναγώγιμα) σε σύνολα κανόνων. Επίσης, οι επεκτάσεις της OWL με κανόνες έχουν στενή σχέση με αυτόν τον φορμαλισμό.

Στη συνέχεια, δίνεται αρχικά μία πολύ συνοπτική εισαγωγή στις έννοιες που διατρέχουν τη βιβλιογραφία σχετικά με τους παραπάνω φορμαλισμούς και γίνεται προσπάθεια να εντοπιστούν τα αντίστοιχα αποτελέσματα πολυπλοκότητας. Κατόπιν εξετάζεται η σχέση, καθώς και η επέκταση της OWL με κανόνες. Τέλος αναφέρονται οι ιδιαιτερότητες της χρήσης μηχανισμών συλλογισμού Πρώτης Τάξης για την OWL. Ως αποτέλεσμα, γίνονται εμφανή τα πλεονεκτήματα και τα μειονεκτήματα της υποστήριξης συλλογισμού για το Σημαντικό Ιστό με την υιοθέτηση των παραπάνω φορμαλισμών.

3.2.1. Ορισμοί

Η Λογική Πρώτης Τάξης (ΛΠΤ) είναι γενικά μη αποφασίσιμη. Για τον έλεγχο της ικανοποιησιμότητας μιας πρότασης ΛΠΤ μπορεί να χρησιμοποιηθεί μια διαδικασία γνωστή ως *ανάλυση* (resolution). Η μέθοδος της ανάλυσης είναι συνεπής, αλλά όχι πλήρης: Αν το σύνολο των προτάσεων ΛΠΤ είναι μη ικανοποιήσιμο, ο αλγόριθμος θα το εντοπίσει. Στην αντίθετη όμως περίπτωση δεν είναι εγγυημένο ότι η ανάλυση θα τερματίσει. Βάσει αυτού, η ανάλυση χαρακτηρίζεται ως *πλήρης ως προς διάψευση* (refutation-complete). Συστήματα που επιχειρούν το έλεγχο της ικανοποιησιμότητας προτάσεων ΛΠΤ (η οποία επαρκεί για την αναπαράσταση ορισμένων μαθηματικών προβλημάτων) ονομάζονται *μηχανισμοί*

απόδειξης θεωρημάτων (theorem-provers). Η πολυπλοκότητα της ανάλυσης στη ΛΠΤ είναι τουλάχιστον εκθετική (EXP-hard).

Η ΛΠΤ περιλαμβάνει λογικά και μη λογικά σύμβολα: Τα μη λογικά σύμβολα χωρίζονται σε *σύμβολα συναρτήσεων* (function symbols) και *σύμβολα κατηγορημάτων* (predicate symbols). Τα σύμβολα κατηγορημάτων που έχουν πολλαπλότητα 0 ονομάζονται *προτασιακά σύμβολα* (propositional symbols).

Αν περιορίσουμε τις προτάσεις ΛΠΤ ώστε να μην περιέχουν όρους (δηλαδή ούτε και μεταβλητές), και ποσοτικοποιητές και να περιέχουν μόνο προτασιακά σύμβολα λαμβάνουμε το λεγόμενο *προτασιακό υποσύνολο* της γλώσσας. Το πρόβλημα της αποφασισιμότητας σε αυτήν την περίπτωση είναι αποφασίσιμο, αλλά NP-complete. Η πολυπλοκότητα της ανάλυσης στην προτασιακή λογική είναι εκθετική.

Αν περιορίσουμε τις προτάσεις ΛΠΤ ώστε να περιέχουν το πολύ ένα θετικό λεκτικό (δηλαδή άτομο ή την άρνησή του) οι προτάσεις λέγονται *Horn*. Οι προτάσεις Horn προσομοιάζουν συνθήκες της μορφής "Αν...Τότε" (κανόνες), και αποτελούν τη βάση για τη γλώσσα προγραμματισμού PROLOG και άρα γενικά για τον λογικό προγραμματισμό. Το πρόβλημα του ελέγχου ικανοποιησιμότητας (γενικών) προτάσεων Horn είναι και αυτό μη αποφασίσιμο.

Για τον έλεγχο της ικανοποιησιμότητας προτάσεων Horn μπορεί να χρησιμοποιηθεί μια εξειδικευμένη μορφή ανάλυσης, γνωστή ως *ανάλυση SLD*. Όπως και η γενική ανάλυση, η ανάλυση SLD είναι συνεπής, αλλά πλήρης ως προς διάψευση. Στη γενική περίπτωση εφαρμόζεται με την λεγόμενη *προς τα πίσω ακολουθία εκτέλεσης* (backward chaining), η οποία μπορεί να επιστρέψει την μη ικανοποιησιμότητα σε εκθετικό χρόνο. Στην περίπτωση που οι προτάσεις Horn είναι ταυτόχρονα και προτασιακές, μπορεί να εφαρμοστεί η λεγόμενη *προς τα εμπρός ακολουθία εκτέλεσης* (forward chaining), η οποία είναι και συνεπής και πλήρης και τερματίζει σε χρόνο γραμμικό ως προς το μέγεθος της εισόδου. Επομένως η προτασιακή λογική Horn είναι αποφασίσιμη και μάλιστα σε γραμμικό χρόνο.

Η *Datalog* είναι μια γλώσσα επερώτησης για παραγωγικές βάσεις δεδομένων και αποτελεί συντακτικό υποσύνολο της PROLOG. Ο κύριος περιορισμός είναι ότι οι κανόνες δεν πρέπει να περιλαμβάνουν σύμβολα συναρτήσεων (ή οι συναρτήσεις να είναι πολλαπλότητας 0, δηλαδή σταθερές). Η αποτίμηση ερωτημάτων σε Datalog (ισοδύναμα, ο έλεγχος της ικανοποιησιμότητας των κανόνων Datalog) είναι συνεπής και πλήρης και μπορεί να γίνει αποδοτικά, ακόμα και για μεγάλες βάσεις, με χρήση προς τα εμπρός ακολουθίας εκτέλεσης.

Ένα *σύστημα παραγωγής* είναι ένα σύστημα συλλογισμού που εφαρμόζει προς τα εμπρός ακολουθία εκτέλεσης και χρησιμοποιεί κανόνες ορισμένης μορφής (*κανόνες παραγωγής*) για την αναπαράσταση της γνώσης. Τα συστήματα παραγωγής επίσης καλούνται και *συστήματα βασισμένα σε κανόνες*. Η προς τα εμπρός ακολουθία εκτέλεσης που εφαρμόζει ένα σύστημα παραγωγής, μπορεί να υλοποιηθεί με τον αλγόριθμο *RETE*, ο οποίος έχει πολυπλοκότητα γραμμική ανά επανάληψη.

Τέλος, οι *Λογικές Περιγραφές* (Description Logics) είναι το υποσύνολο εκείνο της ΛΠΤ όπου κάθε *τύπος* (formula) μπορεί να περιλαμβάνει το πολύ δύο (ελεύθερες) μεταβλητές.

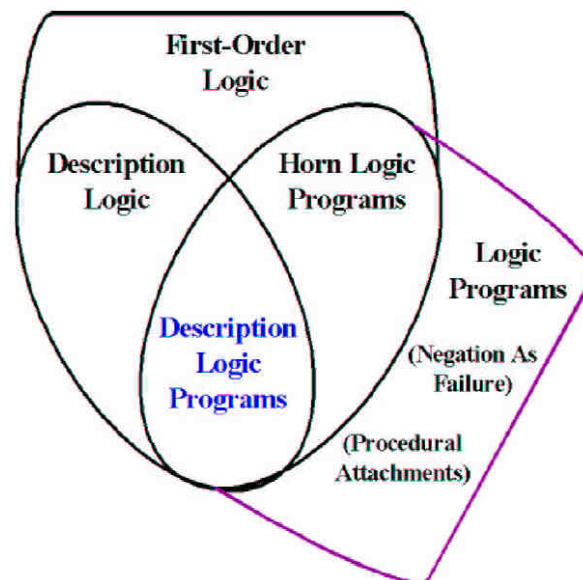
3.2.2. Προγράμματα Λογικών Περιγραφής

Τα Προγράμματα Λογικών Περιγραφής (Description Logic Programs, DLP) είναι στην ουσία ο συνδυασμός των Λογικών Περιγραφής με τον λογικό προγραμματισμό. Παρόλο που δεν έχουν μεγάλη εκφραστικότητα, πολλές οντολογίες του Σημαντικού Ιστού εμπίπτουν σε αυτό το υποσύνολο, για το οποίο η πολυπλοκότητα του συλλογισμού είναι στη χειρότερη περίπτωση πολυωνυμική. Στο [Groszof, Horrocks, Volz, & Decker, 2003] εισάγεται η έννοια των DLP και δείχνεται πώς μια (περιορισμένη) Λογική Περιγραφής μπορεί να αναχθεί σε ένα υποσύνολο λογικής Horn, επονομαζόμενο def-Horn:

- Μια πρόταση Horn που έχει ακριβώς ένα θετικό λεκτικό ονομάζεται *οριστική* (definitive) και είναι επίσης γνωστή ως *κανόνας Horn*.
- Μια πρόταση Horn χαρακτηρίζεται *χωρίς ισότητα* (equality-free) αν δεν εμφανίζεται σε αυτήν το κατηγορημα της ισότητας.
- Μια πρόταση Horn ονομάζεται *Datalog*, αν τα μόνα σύμβολα συναρτήσεων που περιλαμβάνει είναι πολλαπλότητας 0 (γνωστά και ως *σταθερές*). Τα ονόματα των στιγμιοτύπων σε μια οντολογία είναι στην πραγματικότητα τέτοιες συναρτήσεις. Οι Λογικές Περιγραφής έχουν εξορισμού τον Datalog περιορισμό.

Η λογική που επιτρέπει οριστικές, ελεύθερες ισότητας και Datalog προτάσεις Horn καλείται *def-Horn*. *Γεγονότα* ονομάζονται οι κανόνες εκείνοι που περιλαμβάνουν μόνο ένα θετικό λεκτικό χωρίς μεταβλητές (ground).

Ένα *πρόγραμμα λογικής* από την άλλη (Logic Program, LP) είναι μια επέκταση των προτάσεων Horn, με την έννοια ότι περιλαμβάνονται κάποια χαρακτηριστικά που δεν μπορούν να εκφραστούν σε ΛΠΤ και κυρίως:



Εικόνα 3.1: Αλληλεπικάλυψη Λογικών Περιγραφής και LP.

- *Άρνηση ως αποτυχία* (negation as failure): Στους κανόνες που συνιστούν ένα πρόγραμμα λογικής, η άρνηση σε ένα λεκτικό μπορεί να ερμηνευθεί είτε ως ότι το λεκτικό αυτό είναι ψευδές, είτε ως ότι είναι άγνωστο (δεν είναι ακόμα πιστευτό). Η άρνηση ως αποτυχία είναι μια μορφή μη μονοτονικότητας.

- *Διαδικαστικές επισυνάψεις* (procedural attachments): Η επαλήθευση κάποιων κανόνων μπορεί να οδηγήσει στην κλήση και εκτέλεση προγραμματιστικών διαδικασιών, όπως για παράδειγμα, στην προσθήκη ή αφαίρεση γεγονότων από τη βάση γνώσης.

Τα LP που δεν περιλαμβάνουν την άρνηση ως αποτυχία ονομάζονται *οριστικά* LP. Τα οριστικά, ελεύθερα μεταβλητών και Datalog LP καλούνται *def-LP*. Τα def-LP είναι στην πραγματικότητα ένα υποσύνολο def-Horn. Πρέπει δηλαδή να ισχύει ο περιορισμός ότι κάθε συμπέρασμα ενός def-LP πρέπει να έχει τη μορφή γεγονότος. Τα def-LP στη γενική περίπτωση έχουν πολυωνυμική πολυπλοκότητα και στην προτασιακή περίπτωση γραμμική.

Η τομή των def-LP με τις Λογικές Περιγραφές συνιστά τα *Προγράμματα Λογικών Περιγραφής*, που είναι λιγότερο εκφραστικά και από τους δύο φορμαλισμούς (εικόνα 3.1). Όσον αφορά τις Λογικές Περιγραφές, αυτό σημαίνει:

- Δεν είναι δυνατή η δήλωση της ύπαρξης ατόμων, των οποίων η ταυτότητα μπορεί να μην είναι γνωστή, δηλ. δεν μπορούμε να πούμε $\text{Person} \sqsubseteq \exists \text{hasFather.T}$.
- Δεν είναι δυνατή η έκφραση περιορισμών πληθικότητας με ισότητα (π.χ. $\geq 2R.C$).
- Δεν είναι δυνατή η έκφραση του συμπληρώματος κλάσης (λόγω της απαγόρευσης άρνησης στο σώμα του κανόνα).

Συμπερασματικά, τα DLP είναι λιγότερο εκφραστικά και από την OWL Lite, δηλαδή την Λογική Περιγραφή SHIF(D). Η υλοποίηση μηχανισμού εξαγωγής συμπερασμών για DLP έχει γίνει στην αρχική μορφή του συστήματος KAON, γνωστής ως *Bubo*.

Οι DLP κανόνες γίνονται δεκτοί από ένα σύστημα παραγωγής, το οποίο μπορεί να ελέγξει την ικανοποιησιμότητα του DLP υποσυνόλου με ασφάλεια. Αυτό συμβαίνει για παράδειγμα στο σύστημα FuXi (<http://code.google.com/p/python-dlp/wiki/FuXi>), που υλοποιείται σε Python, καθώς και στο OWLJesTab του Protégé, που αγνοεί τις ισότητες (sameAs, equivalentAs), καθώς και τις σχέσεις υπαγωγής. Και τα δύο παραπάνω συστήματα υλοποιούν τον αλγόριθμο RETE. Άλλα συστήματα που παρέχουν (μερική) υποστήριξη για OWL Lite και βασίζονται στον αλγόριθμο RETE είναι τα OWLLisaKB (<http://www.mindswap.org/~katz/OWLLisaKB/>) και το περιβάλλον Jena.

3.2.3. SWRL και DL-safe κανόνες

Τα DLP προγράμματα έχουν επομένως πολύ μικρή εκφραστικότητα. Ο συνδυασμός κανόνων και οντολογιών γίνεται με έναν τρόπο απωλεστικό, μειώνοντας την εκφραστικότητα των οντολογιών για να είναι συμβατή με κανόνες και όχι με αυξητικό τρόπο, επεκτείνοντας δηλαδή τις γλώσσες οντολογιών με ένα (ανώτερο) επίπεδο κανόνων, όπως προδιαγράφεται από την αρχιτεκτονική του Σημαντικού Ιστού [Horrocks, Parsia, Patel-Schneider, & Hendler, 2005].

Για το λόγο αυτό, στο [Horrocks I. , Patel-Schneider, Bechhofer, & Tsarkov, 2005] προτείνεται η *Γλώσσα Κανόνων Σημαντικού Ιστού* (Semantic Web Rule Language, SWRL) η οποία επεκτείνει την OWL DL με κανόνες Horn. Η γλώσσα αυτή είναι μη αποφασίσιμη. Ωστόσο, η SWRL μπορεί να αναχθεί σε ΛΠΤ και άρα μπορούν να χρησιμοποιηθούν μηχανισμοί απόδειξης θεωρημάτων για τον έλεγχο της ικανοποιησιμότητας. Από την άλλη, στο σύστημα CARIN [Levy & Rousset, 1998] επιτυγχάνεται η επέκταση μιας Λογικής

Περιγραφής, της *ALCNR* (ή *SN*) με κανόνες, διατηρώντας την αποφασισιμότητα. Η *ALCNR* είναι όμως πολύ λιγότερο εκφραστική από την *OWL*.

Γεννάται επομένως το ερώτημα, τι είδους κανόνες μπορούν να προστεθούν στην *OWL*, χωρίς να χάνεται η αποφασισιμότητα. Οι κανόνες αυτοί αποτελούν το λεγόμενο *DL-safe* υποσύνολο που υπαγορεύει:

- Οι μεταβλητές του κανόνα μπορούν να αντιστοιχιστούν μόνο σε άτομα που περιλαμβάνονται ρητά μέσα στο *ABox*. Έτσι π.χ. δεν μπορεί να εκφραστεί η έννοια «*Ο ιδιοκτήτης του αυτοκινήτου x είναι και ιδιοκτήτης κάποιου κινητήρα*» (ποιού κινητήρα;).
- Οι κανόνες Horn είναι χωρίς συναρτήσεις (ή αλλιώς, είναι *Datalog*).

Η επέκταση της *OWL DL* με *DL-safe* κανόνες δείχνεται ότι είναι αποφασίσιμη, ενώ εισάγεται αλγόριθμος για τον έλεγχο ικανοποιησιμότητας στην περίπτωση της *SHIQ* επαυξημένης με τέτοιους κανόνες [Motik, Sattler, & Struder, 2005]. Ο αλγόριθμος αυτός, ουσιαστικά ανάγει την *SHIQ* σε *Datalog* και τερματίζει σε εκθετικό χρόνο. Είναι επίσης βέλτιστος -αφού η λογική είναι *Exp-complete*- και υλοποιείται στο σύστημα *KAON2*.

Έχοντας μελετήσει τις εκφραστικές δυνατότητες της γλώσσας *OWL 1.1* που βασίζεται στη Λογική Περιγραφής *SROIQ* και αναλογιζόμενοι ιδιαίτερα τα χαρακτηριστικά και τις αλυσίδες ρόλων, είναι εύλογο αν η περιορισμένη αυτή μορφή κανόνων Horn μπορεί ενδεχομένως να ισοδυναμεί με εκφράσεις *SROIQ*. Πράγματι, πολύ πρόσφατα [Gasse, Sattler, & Haarslev, 2008; Krötzsch, Rudolph, & Hitzler, 2008], έχει φανεί ότι τέτοιοι κανόνες μπορούν να εσωτερικευθούν πλήρως σε αξιώματα μιας βάσης γνώσης *SROIQ*. Η αναγωγή αυτή είναι αποδοτική (γίνεται σε πολυωνυμικό χρόνο) και μάλιστα χωρίς τον περιορισμό οι μεταβλητές να αναφέρονται σε ρητά δηλωμένα άτομα, αρκεί να μην εμφανίζονται σε κύκλους. Επίσης η ιεραρχία ρόλων που προκύπτει μετά την αναγωγή των κανόνων πρέπει να παραμένει κανονική (*regular*) με την έννοια που δίνεται στο [Horrocks, Kutz, & Sattler, 2006].

3.2.4. Μηχανισμοί απόδειξης θεωρημάτων

Εφόσον οι Λογικές Περιγραφές είναι υποσύνολο της ΛΠΤ, μπορούν να χρησιμοποιηθούν μηχανισμοί απόδειξης θεωρημάτων για συλλογισμό με τέτοιες λογικές. Αυτό επιτυγχάνεται μεταφράζοντας ένα σύνολο από αξιώματα Λογικών Περιγραφής (όπως π.χ. αξιώματα συμπερίληψης και περιορισμούς αριθμού) στα αντίστοιχα αξιώματα ΛΠΤ, με άμεσο τρόπο, σύμφωνα με την σημασιολογία Πρώτης Τάξης των Λογικών Περιγραφής, φτιάχνοντας έτσι μια θεωρία ΛΠΤ. Αυτή η προσέγγιση εξετάζεται διεξοδικά στα [Horrocks & Voronkov, 2006; Tsarkov, Riazanov, Bechhofer, & Horrocks, 2004; Tsarkov & Horrocks, 2003]. Ισχύουν όμως τα ακόλουθα:

- Οι μηχανισμοί απόδειξης θεωρημάτων παρέχουν συλλογισμό για την πλήρη ΛΠΤ, η οποία είναι γενικά μη αποφασίσιμη. Συνεπώς οι μηχανισμοί αυτοί δεν είναι εγγυημένο ότι τερματίζουν.
- Οι βασισμένοι σε Λογικές Περιγραφής μηχανισμοί συλλογισμού έχουν αναπτυχθεί ειδικά για τέτοιες λογικές και χρησιμοποιούν βελτιστοποιήσεις, οι οποίες αυξάνουν τις επιδόσεις τους σε σχέση με τους μηχανισμούς απόδειξης θεωρημάτων.

- Οι Hustadt & Schmidt [Hustadt & Schmidt, 2000] έχουν αναπτύξει μια μετάφραση, αρκετά σύνθετη, τροπικών (modal) λογικών σε ΛΠΤ, η οποία, λόγω άμεσης αντιστοιχίας, εφαρμόζεται και για απλές Λογικές Περιγραφής. Η μετάφραση αυτή είναι τέτοια, ώστε το υποσύνολο ΛΠΤ που προκύπτει είναι εγγυημένα αποφασίσιμο. Για πιο εκφραστικές λογικές όμως, όπως αυτές που χρησιμοποιούνται στο Σημαντικό Ιστό, μια τέτοια μετάφραση χρειάζεται επέκταση και μάλιστα δεν είναι καν βέβαιο ότι κάτι τέτοιο είναι εφικτό.
- Για εκφραστικές λογικές, όπως αυτές που περιλαμβάνουν μεταβατικούς ρόλους (*SHIQ*) και αξιώματα ρόλων (*RIQ*) και οι οποίες είναι συνήθεις στις οντολογίες Ιστού, η μετάφραση σε ΛΠΤ απαιτεί τουλάχιστον τρεις μεταβλητές, καταλήγοντας σε υποσύνολο ΛΠΤ που είναι μη αποφασίσιμο. Γενικά, η προσέγγιση της απλής μετάφρασης, θεωρείται αφελής και οδηγεί σε μη πλήρη συλλογισμό.

Πέρα από τις παραπάνω ασυμβατότητες, η χρήση μηχανισμών απόδειξης θεωρημάτων για συλλογισμό με οντολογίες μπορεί να έχει ακόμα κάποια χρησιμότητα, κυρίως διότι:

- Υπάρχει η δυνατότητα ενσωμάτωσης συγκεκριμένων ως προς Λογικές Περιγραφής βελτιστοποιήσεων στους μηχανισμούς απόδειξης θεωρημάτων, με αποτέλεσμα να αυξάνουν τις επιδόσεις τους, όταν χειρίζονται τέτοιες λογικές. Αυτό για παράδειγμα έχει δοκιμαστεί στην περίπτωση του υψηλών επιδόσεων μηχανισμού Vampire.
- Για τις εκτός Λογικών Περιγραφής επεκτάσεις της OWL, όπως η SWRL, τις οποίες ούτως ή άλλως δεν μπορεί να χειριστεί ένα σύστημα Λογικών Περιγραφής, μπορεί να χρησιμοποιηθεί κάποιος μηχανισμός απόδειξης θεωρημάτων. Ο συλλογισμός στην περίπτωση αυτή θα είναι συνεπής, αλλά όχι πλήρης, κάτι που είναι αναμενόμενο, αφού, όπως φάνηκε, η SWRL είναι μη αποφασίσιμη.

Συστήματα που υποστηρίζουν το συλλογισμό σε OWL και είναι βασισμένα σε πλήρη ΛΠΤ αποτελούν τα Hoolet (<http://owl.man.ac.uk/hoolet/>), που χρησιμοποιεί την μηχανή απόδειξης θεωρημάτων Vampire και Surnia (<http://www.w3.org/2003/08/surnia/>), που βασίζεται στο OTTER.

3.3. Λογικές Περιγραφής

Η επιλογή των Λογικών Περιγραφής ως λογικό υπόβαθρο για συλλογισμό στο Σημαντικό Ιστό φαίνεται ότι έχει ισχυρά πλεονεκτήματα, όπως:

- Ύπαρξη αλγορίθμων, που αποφασίζουν πλέον το πλήρες σύνολο της εκφραστικότητας της OWL (μέχρι και OWL 1.1).
- Διαθεσιμότητα πρακτικών συστημάτων, που έχουν υλοποιημένους τους αλγόριθμους αυτούς και επιπλέον εφαρμόζουν μια σειρά από βελτιστοποιήσεις που βελτιώνουν δραματικά την επίδοση στις συνήθεις περιπτώσεις.
- Συγγένεια και συνέπεια των προτάσεων των γλωσσών οντολογιών Ιστού με τις Λογικές Περιγραφής. Για παράδειγμα η OWL 1.1 είναι ακριβώς η λογική *SROIQ(D)* και αυτή είναι η σημαντική που τις αποδίδεται, δεν διαθέτει δηλαδή πλέον τη σημασιολογία μοντέλου του RDF (βλ. και ενότητα 2.5.1).

Στα μειονεκτήματα συγκαταλέγονται:

- Υψηλή πολυπλοκότητα χειρότερης περίπτωσης των αλγορίθμων ελέγχου ικανοποιησιμότητας. Στις περισσότερες περιπτώσεις η πολυπλοκότητα αυτή ξεπερνά το (βέλτιστο) άνω όριο που έχει θεωρητικά καταρτιστεί.
- Για λιγότερο εκφραστικά υποσύνολα, όπως π.χ. η OWL Lite και, ιδιαίτερα, υποσύνολα αυτής, είναι σαφές ότι συστήματα που βασίζονται σε κανόνες και Datalog έχουν πολύ καλύτερη συμπεριφορά.

Στα επόμενα γίνεται μια εισαγωγή στο συντακτικό και τη σημασιολογία των Λογικών Περιγραφής και συνοψίζεται η ονοματολογία που ακολουθείται συνήθως για τις λογικές αυτές, ανάλογα με την εκφραστική ισχύ τους.

3.3.1. Συντακτικό και σημασιολογία

Οι Λογικές Περιγραφής είναι φορμαλισμοί που υποστηρίζουν τη λογική περιγραφή εννοιών και ρόλων. Αυθαίρετες περιγραφές εννοιών και ρόλων (στο εξής απλώς έννοιες και ρόλοι) μπορούν να σχηματιστούν από ονόματα ατομικών εννοιών και ρόλων με τη χρήση διαφόρων τελεστών, οι οποίοι εξαρτώνται από τη συγκεκριμένη λογική.

Ατομικά (*atomic*) ονομάζονται τα σύμβολα (εννοιών και ρόλων) που αποτελούν στοιχειώδεις περιγραφές, βάσει των οποίων κατασκευάζονται επαγωγικά σύνθετες περιγραφές. Αρχικά (*primitive*) ονομάζονται τα σύμβολα που εμφανίζονται μόνο στη δεξιά πλευρά του ορισμού μιας σύνθετης έννοιας ή ρόλου.

Διαισθητικά, ατομικές είναι οι έννοιες εκείνες στις οποίες αντιστοιχεί ένα συγκεκριμένο όνομα ή σύμβολο, ενώ αρχικές είναι οι έννοιες που δεν μπορούν να αναλυθούν περαιτέρω. Στα επόμενα, τα C και D συμβολίζουν αυθαίρετες έννοιες, τα R και S αυθαίρετους ρόλους, τα A και P ονόματα ατομικών εννοιών και ρόλων και το n έναν μη αρνητικό ακέραιο.

Για τις έννοιες, οι διαθέσιμοι τελεστές συνήθως περιλαμβάνουν κάποια ή όλα από τα συνήθη λογικά συνδετικά, *σύζευξη* (Π), *διάζευξη* (\sqcup) και *άρνηση* (\neg) (κατ' αντιστοιχία με τους συνολοθεωρητικούς τελεστές, *τομή*, *ένωση*, *συμπλήρωμα*). Επίσης, η καθολική *ανώτατη* έννοια (universal top concept), η οποία συμβολίζεται με T και ισοδυναμεί με $A\sqcup\text{-}A$, καθώς και η μη συνεπής λογικά *κατώτατη* (bottom) έννοια, η οποία συμβολίζεται με \perp και ισοδυναμεί με $A\sqcap\text{-}A$, είναι συνήθως προκαθορισμένες. Άλλοι συνήθεις τελεστές περιλαμβάνουν περιορισμένες μορφές ποσοτικοποίησης, που ονομάζονται *υπαρξιακοί περιορισμοί ρόλων* (existential role restrictions, $\exists R.C$) και *καθολικοί περιορισμοί ρόλων* (universal role restrictions, $\forall R.C$). Ορισμένες Λογικές Περιγραφής υποστηρίζουν επίσης *προσδιορισμένους περιορισμούς αριθμού* (qualified number restrictions, $\leq n.PC$ και $\geq n.PC$), οι οποίοι θέτουν περιορισμούς πληθικότητας σε ρόλους που συνδέουν στιγμιότυπα μίας έννοιας με στιγμιότυπα μιας άλλης έννοιας. Οι περιορισμοί πληθικότητας συνήθως περιορίζονται στις μορφές $\leq n.PT$ και $\geq n.PT$, που ονομάζονται *απροσδιόριστοι περιορισμοί αριθμού* (unqualified number restrictions) ή απλά περιορισμοί αριθμού και μπορούν να γραφούν συντετμημένα $\leq n.P$ και $\geq n.P$. Οι ρόλοι που μπορούν να εμφανιστούν σε έννοιες περιορισμού πληθικότητας περιορίζονται συνήθως σε ατομικούς, καθώς η εμφάνιση αυθαιρέτων ρόλων σε τέτοιες έννοιες οδηγεί σε μη αποφασισιμότητα.

Είναι επίσης δυνατόν να υποστηρίζονται τελεστές σχηματισμού ρόλων και, σε κάποιες πολύ εκφραστικές λογικές, οι ρόλοι μπορούν να είναι κανονικές εκφράσεις (regular expressions) που σχηματίζονται χρησιμοποιώντας τους τελεστές της ένωσης (union, \sqcup) της σύνθεσης (composition, \circ), της ανακλαστικής-μεταβατικής κλειστότητας (transitive-reflexive closure, $*$) της ταυτότητας (identity, id), και της αντιστροφής (inverse, $\bar{}$).

Στις έννοιες και τους ρόλους αποδίδεται η συνήθης κατά Tarski σημαντική της θεωρίας μοντέλων, ώστε η σημασία τους δίδεται από μια διερμήνευση $I = (D', ')$, όπου D' είναι το πεδίο (ένα σύνολο) και $'$ η συνάρτηση διερμήνευσης. Περισσότερες λεπτομέρειες για τη σύνταξη και τη σημασιολογία μπορούν να βρεθούν στο [Baader & Nutt, 2007].

Γενικά, μια βάση γνώσης Λογικών Περιγραφής (knowledge base, KB) αποτελείται από ένα σύνολο T ορολογικών (terminological) αξιωμάτων (ή TBox) και ένα σύνολο A δηλωτικών (assertional) αξιωμάτων (ή ABox). Τα αξιώματα στο T εκφράζουν γεγονότα για έννοιες και ρόλους, ενώ τα αξιώματα στο A εκφράζουν γεγονότα για ατομικά στιγμιότυπα εννοιών και ρόλων. Στην ενότητα αυτή, μια βάση γνώσης θα θεωρείται ότι αποτελείται μόνο από το ορολογικό τμήμα T , εφόσον θα ασχοληθούμε κυρίως με συλλογισμό σχετικά με έννοιες και ρόλους. Εξάλλου, για εκφραστικές λογικές, είναι δυνατή η αναγωγή του συλλογισμού για άτομα στο ορολογικό μόνο τμήμα, με μια τεχνική γνωστή ως απαλοιφή ABox (βλ. ενότητα 3.6.3). Η γνώση που περιλαμβάνεται στο TBox, χαρακτηρίζεται συνήθως εντασιακή (intensional), ενώ η γνώση που περιλαμβάνεται στο ABox, χαρακτηρίζεται εκτασιακή (extensional).

Μια ορολογική βάση γνώσης T αποτελείται συνήθως από ένα σύνολο αξιωμάτων της μορφής $C \sqsubseteq D$ και $C \equiv D$, όπου τα C και D είναι έννοιες. Μια διερμήνευση I ικανοποιεί το T αν, για κάθε αξίωμα $(C \sqsubseteq D) \in T$, $C' \subseteq D'$, και για κάθε αξίωμα $(C \equiv D) \in T$, $C' = D'$. Το T είναι ικανοποιήσιμο, αν υπάρχει κάποια μη κενή διερμήνευση που να το ικανοποιεί. Να σημειωθεί ότι το T μπορεί να περιέχει μόνο αξιώματα συμπερίληψης ή μόνο αξιώματα ισότητας, καθώς οι δύο μορφές μπορούν να αναχθούν η μια στην άλλη σύμφωνα με τις ακόλουθες ισοδυναμίες:

$$C \sqsubseteq D \Leftrightarrow T \equiv D \sqcup \bar{C}$$

$$C \equiv D \Leftrightarrow C \sqsubseteq D \text{ και } D \sqsubseteq C.$$

Μια έννοια C υπάγεται σε μία έννοια D ως προς το T (γράφεται $T \models C \sqsubseteq D$) αν $C' \subseteq D'$ σε κάθε διερμήνευση I που ικανοποιεί το T , μια έννοια C είναι ικανοποιήσιμη ως προς το T (γράφεται $T \models C \not\sqsubseteq \perp$) αν $C' \neq \emptyset$ σε κάποια διερμήνευση I που ικανοποιεί το T και μια έννοια C είναι μη ικανοποιήσιμη ως προς το T (γράφεται $T \models \bar{C}$) αν $C' = \emptyset$ για κάθε I που ικανοποιεί το T . Η υπαγωγή και η (μη) αποφασισιμότητα συνδέονται στενά. Αν $T \models C \sqsubseteq D$, τότε, σε όλες τις διερμηνεύσεις I που ικανοποιούν το T , $C' \subseteq D'$ και επομένως $C' \cap (\bar{D})' = \emptyset$.

Αντίστροφα, αν η C είναι μη ικανοποιήσιμη ως προς το T , τότε, σε κάθε I που ικανοποιεί το T , $C' = \emptyset$ και επομένως $C' \subseteq \perp'$. Η υπαγωγή και η (μη) ικανοποιησιμότητα μπορούν επομένως να αναχθούν η μια στην άλλη χρησιμοποιώντας τις ακόλουθες ισοδυναμίες:

$$T \models C \sqsubseteq D \Leftrightarrow T \models \bar{(C \sqcap \bar{D})}$$

$$T \models \bar{C} \Leftrightarrow T \models C \sqsubseteq \perp.$$

Σε ορισμένες Λογικές Περιγραφής το T μπορεί επίσης να περιέχει αξιώματα που ορίζουν ένα σύνολο από μεταβατικούς ρόλους \mathbf{R}_+ ή/και μια μερική διάταξη υπαγωγής στους ρόλους. Ένα αξίωμα $R \in \mathbf{R}_+$ σημαίνει ότι ο R είναι ένας μεταβατικός ρόλος, ενώ ένα αξίωμα $R \sqsubseteq S$ σημαίνει ότι ο R υπάγεται στον S . Μια διερμήνευση I ικανοποιεί το αξίωμα $R \in \mathbf{R}_+$ αν

το R' είναι μεταβατικά κλειστό (δηλαδή $(R')^+ = R'$, η μεταβατική κλειστότητα του ρόλου είναι ο εαυτός του) και το αξίωμα $R \sqsubseteq S$, αν $R' \subseteq S'$.

3.3.2. Εκφραστική ισχύς και ονοματολογία

Η εκφραστική δύναμη μιας Λογικής Περιγραφής χαρακτηρίζεται συνήθως από τρεις παράγοντες:

- Το σύνολο των κατασκευαστών εννοιών και ρόλων που αποδέχεται η λογική,
- εάν η λογική περιλαμβάνει υποστήριξη για τύπους δεδομένων (αριθμούς, συμβολοσειρές, ημερομηνίες) και για αφηρημένες έννοιες και
- εάν η λογική υποστηρίζει τόσο στιγμιότυπα (άτομα) όσο και έννοιες.

Οι καθιερωμένες λογικές βασίζονται σχεδόν όλες σε μια κοινή λογική γνωστή ως ALC^1 . Κάθε νέα λογική επαυξάνει την ALC με επεκτάσεις αυξανόμενης ισχύος. Ο παρακάτω πίνακας συνοψίζει τα χαρακτηριστικά της ALC όπως και των επεκτάσεών της, μαζί με κάποια παραδείγματα:

Συμβολισμός	Χαρακτηριστικό
ALC	Δημιουργία αρχικών εννοιών και ρόλων. Δημιουργία σύνθετων εννοιών με χρήση ένωσης, τομής και συμπληρώματος. Δημιουργία εκφράσεων εννοιών με υπαρξιακή ποσοτικοποίηση και περιορισμό τιμής. Δήλωση αξιωμάτων για τη βάση γνώσης, όπως <i>Γυναίκα</i> είναι ο <i>Θηλυκός Άνθρωπος</i> .
(D)	Απτά πεδία (Concrete domains). Τύποι δεδομένων, όπως αριθμοί, συμβολοσειρές και ημερομηνίες. Εμφανίζονται σχεδόν αποκλειστικά στους υπαρξιακούς ποσοτικοποιητές, σχηματίζοντας έννοιες όπως «οι άνθρωποι που γεννήθηκαν στις 16/3/2003»
F	Συναρτησιακοί Ρόλοι . Ο όρος <i>συναρτησιακός</i> (functional) προέρχεται από τα μαθηματικά και δηλώνει μια σχέση πολλά προς ένα. Κάθε συναρτησιακός ρόλος πρέπει να έχει το πολύ έναν πληρωτή. Για παράδειγμα ο ρόλος <i>έχειΜητέρα</i> μπορεί να χαρακτηριστεί ως συναρτησιακός, δηλώνοντας ότι ένας άνθρωπος έχει ακριβώς μία μητέρα. Οι συναρτησιακοί ρόλοι αποτελούν υποσύνολο των περιορισμών αριθμού. Όταν και οι δύο είναι παρόντες, τότε παραλείπεται το F από το χαρακτηρισμό της γλώσσας.
H	Ιεραρχία Ρόλων . Η επέκταση αυτή οργανώνει τους ρόλους σε σχέσεις ειδικότητας / γενικότητας. Οι ιεραρχίες ρόλων κατασκευάζουν αξιώματα της μορφής «ο ρόλος <i>έχει για</i> είναι ένα είδος του ρόλου <i>έχει παιδί</i> ». Αυτό επιτρέπει κάποιους πρόσθετους συμπερασμούς: αν, για παράδειγμα, ο Αλέξανδρος <i>έχει για</i> τον Δημήτρη, τότε επίσης, ο Αλέξανδρος <i>έχει παιδί</i> τον Δημήτρη.
I	Αντίστροφοι Ρόλοι . Το χαρακτηριστικό αυτό συσχετίζει δύο ρόλους. Όταν ο ένας από αυτούς ορίζει μια σχέση υποκειμένου, τότε ο άλλος ορίζει την αντίστοιχη σχέση αντικειμένου. Το χαρακτηριστικό αυτό επιτρέπει τη

¹ Η ALC (“**A**tttribute **L**anguage with **C**omplement”) μπορεί να οριστεί ως το τμήμα εκείνο της λογικής πρώτης τάξης που αποκτάται θέτοντας το συντακτικό περιορισμό οι τύποι να έχουν δύο μεταβλητές.

	<p>δήλωση ότι το όνομα ενός ρόλου είναι το αντίστροφο ενός άλλου και έτσι ο συμπερασμός μπορεί να γίνει σε αυτή τη βάση.</p> <p>Για παράδειγμα, δεδομένης της δήλωσης ότι <i>γεννήθηκε στην πόλη</i> είναι το αντίστροφο του <i>είναι η πόλη που γεννήθηκε ο</i>, και της δήλωσης ότι ο Δημήτρης <i>γεννήθηκε στην πόλη</i> Πάτρα, συνάγεται το γεγονός ότι η Πάτρα <i>είναι η πόλη που γεννήθηκε ο</i> Δημήτρης.</p> <p>Οι σχέσεις που περιγράφονται από τους αντίστροφους ρόλους δεν είναι κατ' ανάγκη δυαδικές. Είναι πιθανό ένας ρόλος να είναι ο αντίστροφος του εαυτού του. Παραδείγματα τέτοιων <i>συμμετρικών</i> ρόλων είναι οι <i>είναι κοντά σε</i> και <i>έχει φίλο</i>.</p>
O	<p>Ονοματικά (Nominals). Ειδικός τύπος ατομικής έννοιας που ορίζεται ως μονοσύνολο ενός ατόμου. Για παράδειγμα οι έννοιες Σαββατοκύριακο μπορεί να εκφραστεί ως: Σαββατοκύριακο \equiv {Σάββατο} \cup {Κυριακή} (ή ισοδύναμα, {Σάββατο, Κυριακή}), όπου τα Σάββατο και Κυριακή είναι άτομα και τα {Σάββατο} και {Κυριακή} είναι ονοματικά.</p>
Q	<p>Προσδιορισμένοι Περιορισμοί Αριθμού (Qualified Number Restrictions). Ένα συγκεκριμένο όριο που περιορίζει την πληθικότητα ενός ρόλου και ορίζει ότι οι πληρωτές του ρόλου, για αυτόν τον περιορισμό, ανήκουν σε μια συγκεκριμένη έννοια. Για παράδειγμα η έκφραση Άνθρωπος Π ≥ 2 έχει Παιδί.Θηλυκό δηλώνει τον άνθρωπο που έχει τουλάχιστον δύο παιδιά που είναι κορίτσια.</p>
N	<p>Απροσδιόριστοι Περιορισμοί Αριθμού. Προσδιορίζουν την πληθικότητα ενός ρόλου χωρίς όμως και να ορίζουν την έννοια από την οποία προέρχονται οι πληρωτές του ρόλου.</p>
R+	<p>Μεταβατικοί Ρόλοι (Transitive Roles). Οι μεταβατικοί ρόλοι μεταδίδουν τη σχέση μιας έννοιας σε μια άλλη, όπου αυτές μοιράζονται έναν κοινό ρόλο. Αυτό το χαρακτηριστικό επιτρέπει συλλογισμό της μορφής «Αν η Αχαΐα είναι γεωγραφική περιοχή της Πελοποννήσου και η Πελοπόννησος είναι γεωγραφική περιοχή της Ελλάδας τότε η Αχαΐα είναι γεωγραφική περιοχή της Ελλάδας»</p>

Πίνακας 3.1: Εκφραστική δύναμη των Λογικών Περιγραφής.

Κάποιες συχνά υλοποιημένες λογικές έχουν συντομευμένα ονόματα. Συγκεκριμένα η «καθιερωμένη» λογική $ALCR+$ συχνά συντέμνεται ως S . Έτσι, η $ALCHR+F$ γίνεται SHF και η $ALCQHIR+$ γίνεται $SHIQ$.

3.4. Συλλογισμός με Λογικές Περιγραφής

Στην ενότητα αυτή παρουσιάζονται οι υπηρεσίες συλλογισμού που χρησιμοποιούνται για να σχηματίσουν τον πυρήνα ενός συστήματος αναπαράστασης γνώσης βασισμένου σε Λογικές Περιγραφής. Δείχνεται επίσης πώς, με τη χρήση του *ξεδιπλώματος* (unfolding) και της *εσωτερίκευσης* (internalization), οι υπηρεσίες αυτές μπορούν συνήθως να αναχθούν στο πρόβλημα του καθορισμού της ικανοποιησιμότητας μίας έννοιας.

Γενικά, η μελέτη των τεχνικών υλοποίησης και βελτιστοποίησης, καθώς και συγκεκριμένων αλγορίθμων Λογικών Περιγραφής είναι κρίσιμης σημασίας διότι:

- δείχνει τα θεωρητικά όρια της επίδοσης των τεχνικών Ανακάλυψης Γνώσης στο Σημαντικό Ιστό.

- Αναγνωρίζει τις πηγές πολυπλοκότητας και βοηθά στη *στοχευμένη* σημασιακή προσαρμογή μιας οντολογίας, επικεντρώνοντας σε σημεία που προσφέρουν εκφραστικότητα με μικρό κόστος (βλ. ενότητα 5.6).

3.4.1. Υπηρεσίες συλλογισμού

Ο ορολογικός συλλογισμός σε ένα σύστημα αναπαράστασης γνώσης βασισμένο σε Λογικές Περιγραφές βασίζεται στον προσδιορισμό των σχέσεων υπαγωγής, ως προς τα αξιώματα σε μία βάση γνώσης. Εκτός από την απάντηση συγκεκριμένων ερωτημάτων υπαγωγής και ικανοποιησιμότητας, είναι συχνά χρήσιμο να υπολογίζεται και να αποθηκεύεται (συνήθως με τη μορφή διευθυνόμενου ακυκλικού γραφήματος) η μερική διάταξη υπαγωγής όλων των ονομάτων εννοιών που εμφανίζονται στη βάση γνώσης, μια διαδικασία γνωστή ως *ταξινόμηση* της βάσης γνώσης. Ορισμένα συστήματα έχουν επίσης τη δυνατότητα να ασχοληθούν με δηλωτικά αξιώματα, δηλαδή αυτά που αφορούν στιγμιότυπα εννοιών και ρόλων, και να προσφέρουν υπηρεσίες συλλογισμού όπως η *πραγμάτωση* (ο καθορισμός των εννοιών των οποίων ένα δοσμένο άτομο αποτελεί στιγμιότυπο) και η *ανάκτηση* (ο καθορισμός του συνόλου των ατόμων που αποτελούν στιγμιότυπα μιας δοσμένης έννοιας).

Στην πράξη, πολλά συστήματα χρησιμοποιούν αλγορίθμους ελέγχου υπαγωγής που δεν είναι σε θέση να καθορίσουν τις σχέσεις υπαγωγής ως προς μια αυθαίρετη βάση γνώσης. Αντίθετα, περιορίζουν τα είδη των αξιωμάτων που μπορούν να εμφανιστούν σε μια βάση γνώσης, ώστε να είναι δυνατή η πραγματοποίηση *αντικατάστασης απαλοιφής εξαρτήσεων* (dependency-elimination substitutions) πριν από τον υπολογισμό των σχέσεων υπαγωγής. Οι περιορισμοί αυτοί απαιτούν όλα τα αξιώματα να είναι *μοναδικοί*, *ακυκλικοί ορισμοί*. Ένα αξίωμα καλείται *ορισμός* της A αν είναι της μορφής $A \sqsubseteq D$ ή $A \equiv D$, όπου A ατομικό όνομα, είναι *μοναδικό* αν η βάση γνώσης δεν περιέχει άλλες πληροφορίες για την A και είναι *ακυκλικό* αν η D δεν αναφέρεται άμεσα ή έμμεσα (μέσω άλλων αξιωμάτων) στην A . Μια βάση γνώσης που ικανοποιεί αυτούς τους περιορισμούς χαρακτηρίζεται *ξεδιπλώσιμη* (unfoldable).

Ορισμοί της μορφής $A \sqsubseteq D$ ονομάζονται *αρχικοί* ή *αναγκαίοι*, αφού η D καθορίζει μια αναγκαία συνθήκη για τα στιγμιότυπα της A , ενώ ορισμοί της μορφής $A \equiv D$ ονομάζονται *μη αρχικοί* ή *ικανοί και αναγκαίοι*, αφού η D καθορίζει συνθήκες που είναι ικανές και αναγκαίες για τα στιγμιότυπα της A . Για να ξεχωρίζουν τα αξιώματα που δεν είναι ορισμοί, συχνά αποκαλούνται *γενικά* (general) αξιώματα. Ο περιορισμός μιας βάσης γνώσης σε αξιώματα-ορισμούς μπορεί να κάνει το συλλογισμό ευκολότερο, αλλά μειώνει σημαντικά την εκφραστική ισχύ της βάσης γνώσης. Ωστόσο, ακόμα και σε μια μη περιορισμένη (ή *γενική*) βάση γνώσης, οι ορισμοί και το ξεδίπλωμα είναι χρήσιμες ιδέες, αφού μπορούν να χρησιμοποιηθούν για τη βελτιστοποίηση των διαδικασιών συλλογισμού.

3.4.2. Συλλογισμός στο ABox

Το ABox περιέχει εκτασιακή γνώση για το πεδίο ενδιαφέροντος, δηλαδή δηλώσεις για τα άτομα, που συχνά καλούνται *δηλώσεις συμμετοχής* (membership assertions). Για παράδειγμα το

Θηλυκό Π Άνθρωπος (ANNA)

δηλώνει ότι το άτομο ANNA είναι ένας θηλυκός άνθρωπος. Αν η έννοια της γυναίκας οριστεί ως

$$\text{Γυναίκα} \equiv \text{Θηλυκό} \Pi \text{ Άνθρωπος}$$

Τότε από τις παραπάνω δηλώσεις μπορεί να συναχθεί ότι η ANNA είναι ένα στιγμιότυπο της έννοιας Γυναίκα. Ανάλογα, το

$$\text{έχειΠαιδί (ANNA, JACOB)}$$

δηλώνει ότι η ANNA έχει παιδί τον JACOB. Δηλώσεις της πρώτης περίπτωσης είναι γνωστές ως *δηλώσεις έννοιας*, ενώ δηλώσεις της δεύτερης περίπτωσης είναι γνωστές ως *δηλώσεις ρόλου*.

Όπως φάνηκε σε αυτά τα παραδείγματα, στο ABox συνήθως προσδιορίζεται γνώση στη μορφή δηλώσεων έννοιας και δηλώσεων ρόλου. Στις δηλώσεις έννοιας επιτρέπονται γενικά εκφράσεις έννοιας, ενώ στις δηλώσεις ρόλου συνήθως δεν επιτρέπεται ο ρόλος να είναι μία έκφραση, αλλά μόνο αρχικός (primitive) ρόλος. Εκφράσεις ρόλων συναντώνται μόνο στην περίπτωση των πολύ εκφραστικών γλωσσών.

Η βασική εργασία συλλογισμού σε ένα ABox είναι ο έλεγχος στιγμιότυπου (instance checking), όπου επαληθεύεται αν ένα άτομο είναι στιγμιότυπο (δηλαδή ανήκει) μιας συγκεκριμένης έννοιας. Αν και υπάρχουν και χρησιμοποιούνται και άλλες υπηρεσίες συλλογισμού, αυτές μπορούν να εκφραστούν σε σχέση με τον έλεγχο στιγμιότυπου. Μεταξύ αυτών συγκαταλέγονται η *συνέπεια της βάσης γνώσης* (knowledge base consistency), δηλαδή η επαλήθευση αν κάθε έννοια στη βάση γνώσης αποδέχεται τουλάχιστον ένα άτομο, η *πραγμάτωση* (realization) που βρίσκει την πιο συγκεκριμένη έννοια της οποίας κάποιο άτομο είναι στιγμιότυπο και η *ανάκτηση* (retrieval), που βρίσκει τα άτομα στη βάση γνώσης που είναι στιγμιότυπα μιας δοσμένης έννοιας. Όλα τα παραπάνω μπορούν να επιτευχθούν μέσω του ελέγχου στιγμιότυπου.

3.4.3. Ξεδίπλωμα

Δοσμένης μιας ξεδιπλώσιμης βάσης γνώσης T και μιας έννοιας C , της οποίας η ικανοποιησιμότητα πρόκειται να ελεγχθεί ως προς το T , είναι δυνατή η απαλοιφή από την C όλων των ονομάτων εννοιών που εμφανίζονται στο T χρησιμοποιώντας μια αναδρομική διαδικασία αντικατάστασης, γνωστή ως ξεδίπλωμα. Η ικανοποιησιμότητα της προκύπτουσας έννοιας είναι ανεξάρτητη από τα αξιώματα στο T και μπορεί επομένως να ελεγχθεί χρησιμοποιώντας μια διαδικασία απόφασης, η οποία μπορεί να προσδιορίσει την ικανοποιησιμότητα μίας μόνο έννοιας (ή, ισοδύναμα, την ικανοποιησιμότητα μιας έννοιας ως προς μια κενή βάση γνώσης).

Για ένα μη αρχικό όνομα έννοιας A , που ορίζεται στο T από ένα αξίωμα $A \equiv D$, η διαδικασία συνίσταται απλώς στην αντικατάσταση της A με την D όπου αυτή εμφανίζεται στην C και μετά στο αναδρομικό ξεδίπλωμα της D . Για ένα αρχικό όνομα έννοιας A , που ορίζεται στο T από ένα αξίωμα $A \sqsubseteq D$, η διαδικασία είναι ελαφρώς πιο περίπλοκη. Όπου η A εμφανίζεται στη C , αντικαθίσταται με την έννοια $A' \sqcap D$, όπου A' ένα νέο όνομα έννοιας που δεν εμφανίζεται στο T ή στη C , και η D ξεδιπλώνεται αναδρομικά. Η έννοια A' εκφράζει την «αρχικότητα» της A – όλα εκείνα τα χαρακτηριστικά που τη διαφοροποιούν από την D . Χρησιμοποιούμε το $\text{Unfold}(C, T)$ για να δηλώσουμε μια έννοια C ξεδιπλωμένη ως προς μια βάση γνώσης T .

Μια διαδικασία απόφασης που προσπαθεί να βρει μια διερμηνευση I , η οποία να ικανοποιεί την ξεδιπλωμένη έννοια μπορεί τώρα να χρησιμοποιηθεί, αφού κάθε τέτοια διερμηνευση θα ικανοποιεί επίσης το T . Αυτό δείχνεται εύκολα εφαρμόζοντας την διαδικασία ξεδιπλώματος σε όλες τις έννοιες που σχηματίζουν το δεξί μέρος των αξιωμάτων του T , έτσι ώστε να είναι κατασκευασμένες εξ' ολοκλήρου από ονόματα εννοιών που δεν ορίζονται στο T και είναι επομένως ανεξάρτητες από τα άλλα αξιώματα του T . Η διερμηνευση κάθε έννοιας που ορίζεται στο T μπορεί επομένως να θεωρηθεί ως η διερμηνευση της ξεδιπλωμένης έννοιας στο δεξί μέρος, όπως δίνεται από το I και τη σημασιολογία των τελεστών σχηματισμού εννοιών και ρόλων.

Ο συλλογισμός υπαγωγής μπορεί να γίνει ανεξάρτητος από το T χρησιμοποιώντας την ίδια τεχνική. Δοσμένων δύο εννοιών C και D , ο καθορισμός της υπαγωγής της C στην D ως προς το T είναι ισοδύναμος με τον προσδιορισμό της υπαγωγής της $\text{Unfold}(C, T)$ στην $\text{Unfold}(D, T)$ ως προς μια κενή βάση γνώσης:

$$T \models C \sqsubseteq D \Leftrightarrow \emptyset \models \text{Unfold}(C, T) \sqsubseteq \text{Unfold}(D, T).$$

Το ξεδίπλωμα δεν θα ήταν εφικτό γενικά, αν τα αξιώματα στο T δεν ήταν μοναδικοί, ακυκλικοί ορισμοί. Αν το T περιείχε πολλαπλά αξιώματα ορισμών για κάποια έννοια A , για παράδειγμα, $\{(A \equiv C), (A \equiv D)\} \subseteq T$, θα ήταν αδύνατο να αντικατασταθεί το A διατηρώντας ταυτόχρονα τη σημασία και των δύο αξιωμάτων. Αν το T περιείχε κυκλικά αξιώματα, για παράδειγμα $(A \sqsubseteq \exists R.A) \subseteq T$, τότε η προσπάθεια ξεδιπλώματος της A θα οδηγούσε σε μη τερματισμό. Αν το T περιείχε γενικά αξιώματα, για παράδειγμα $\exists R.C \sqsubseteq D$, δεν μπορεί να εγγραφεί ότι μια διερμηνευση που ικανοποιεί την ξεδιπλωμένη έννοια θα ικανοποιεί και τα αξιώματα αυτά.

3.4.4. Εσωτερίκευση

Ενώ είναι δυνατός ο σχεδιασμός ενός αλγορίθμου ικανού για συλλογισμό ως προς μια γενική βάση γνώσης, με πιο εκφραστικές λογικές, συγκεκριμένα εκείνες που επιτρέπουν τον ορισμό ενός καθολικού ρόλου, μια διαδικασία που καλείται *εσωτερίκευση* μπορεί να χρησιμοποιηθεί για να αναγάγει το πρόβλημα σε αυτό του καθορισμού της ικανοποιησιμότητας μίας μόνο έννοιας. Ένας πραγματικά καθολικός ρόλος είναι αυτός που διερμηνεύεται ως όλα τα ζεύγη στοιχείων στο πεδίο της διερμηνευσης (δηλαδή $\Delta' \times \Delta'$). Όμως, ένας ρόλος U χαρακτηρίζεται καθολικός ως προς μια ορολογία T , αν ορίζεται έτσι ώστε ο U να είναι μεταβατικά κλειστός και $P \sqsubseteq U$ για όλα τα ονόματα ρόλων P που εμφανίζονται στο T . Για μια λογική που υποστηρίζει τους τελεστές σχηματισμού ρόλων της ένωσης και της μεταβατικής-ανακλαστικής κλειστότητας, αυτό μπορεί να επιτευχθεί θεωρώντας ότι ο U είναι

$$(P_1 \sqcup \dots \sqcup P_n \sqcup P_1^- \sqcup \dots \sqcup P_n^-)^*,$$

όπου P_1, \dots, P_n είναι τα ονόματα ρόλων που εμφανίζονται στο T . Για μια λογική που υποστηρίζει μεταβατικά κλειστούς ρόλους και αξιώματα συμπερίληψης ρόλων, αυτό μπορεί να επιτευχθεί προσθέτοντας τα αξιώματα

$$(U \in R_i), (P_1 \sqsubseteq U), \dots, (P_n \sqsubseteq U), \dots, (P_1^- \sqsubseteq U), \dots, (P_n^- \sqsubseteq U)$$

στο T , όπου P_1, \dots, P_n είναι τα ονόματα ρόλων που εμφανίζονται στο T και U είναι ένα νέο όνομα ρόλου που δεν εμφανίζεται στο T . Σε κάθε περίπτωση, τα μέρη με τους αντίστροφους ρόλους χρειάζονται, μόνο αν η λογική υποστηρίζει τον τελεστή αντίστροφου ρόλου.

Τα αξιώματα εννοιών στο T μπορούν να αναχθούν σε αξιώματα της μορφής $T \sqsubseteq C$ χρησιμοποιώντας τις ισοδυναμίες:

$$A \equiv B \Leftrightarrow T \sqsubseteq (A \sqcup \neg B) \sqcap (\neg A \sqcup B)$$

$$A \sqsubseteq B \Leftrightarrow T \sqsubseteq \neg A \sqcup B.$$

Τα αξιώματα αυτά μπορούν στη συνέχεια να συνενωθούν για να δώσουν ένα αξίωμα $T \sqsubseteq C$, όπου:

$$C \equiv \prod_{(A_i \equiv B_i) \in T} ((A_i \sqcup \neg B_i) \sqcap (\neg A_i \sqcup B_i)) \sqcap \prod_{(A_j \equiv B_j) \in T} (\neg A_j \sqcup B_j).$$

Επειδή η διερμηνευση του T είναι ίση με το πεδίο ($T' = \Delta'$), το παραπάνω αξίωμα δηλώνει ότι κάθε στοιχείο του πεδίου θα πρέπει να ικανοποιεί την C . Κατά τον έλεγχο της ικανοποιησιμότητας μιας έννοιας D ως προς το T , ο περιορισμός αυτός στις πιθανές διερμηνεύσεις μπορεί να τεθεί ελέγχοντας την ικανοποιησιμότητα της $D \sqcap C \sqcap \forall U.C$ (ή απλώς $D \sqcap \forall U.C$ στην περίπτωση που ο U είναι μεταβατικά ανακλαστικά κλειστός). Αυτό βασίζεται στο γεγονός ότι, στις Λογικές Περιγραφής, οι ικανοποιήσιμες έννοιες έχουν πάντα μια διερμηνευση στην οποία κάθε στοιχείο συνδέεται με κάθε άλλο στοιχείο μέσω μιας ακολουθίας ρόλων (η ιδιότητα του καταρρεύσαντος μοντέλου, collapsed model property).

3.5. Αλγόριθμοι Ελέγχου Υπαγωγής

Με τη χρήση του ξεδιπλώματος και της εσωτερίκευσης, στις περισσότερες περιπτώσεις, ο ορολογικός συλλογισμός σε ένα σύστημα αναπαράστασης γνώσης βασισμένο σε Λογικές Περιγραφής μπορεί να αναχθεί σε συλλογισμό υπαγωγής ή ικανοποιησιμότητας. Οι αλγοριθμικές τεχνικές για τον υπολογισμό των σχέσεων υπαγωγής χωρίζονται σε δύο κύριες κατηγορίες: δομικές και λογικές.

3.5.1. Δομικοί αλγόριθμοι υπαγωγής

Δομικοί αλγόριθμοι υπαγωγής χρησιμοποιούνται στα πρώιμα συστήματα Λογικών Περιγραφής και χρησιμοποιούνται ακόμα σε συστήματα όπως τα CLASSIC και LOOM. Οι αλγόριθμοι αυτοί βασίζονται σε μία τεχνική γνωστή ως *ταίριασμα δομής* (structure-matching) [Brachman & Levesque, 2004], σύμφωνα με την οποία οι περιγραφές που πρόκειται να ελεγχθούν για υπαγωγή, πρώτα κανονικοποιούνται και κατόπιν συγκρίνεται η συντακτική δομή των κανονικών αυτών μορφών. Η υπαγωγή ισχύει όταν η μία μορφή εμπεριέχεται συντακτικά μέσα στην άλλη.

Για απλές λογικές, οι δομικοί αλγόριθμοι είναι συνεπείς και πλήρεις, ενώ παρουσιάζουν μόνο πολυωνυμική πολυπλοκότητα. Για πιο εκφραστικές λογικές όμως, χάνουν το χαρακτηριστικό της πληρότητας, αφού δεν μπορούν να αντιμετωπίσουν κατασκευαστές όπως η διάζευξη, η άρνηση και ο υπαρξιακός περιορισμός ($\exists R.C$) [Baader & Nutt, 2007]. Έτσι, ενώ τέτοιοι αλγόριθμοι μπορεί να είναι αποδοτικοί, έχουν αρκετά μειονεκτήματα:

- Το πιο σημαντικό μειονέκτημα είναι ότι, ενώ είναι εύκολο να δειχθεί η συνέπεια των δομικών κανόνων συμπερασμού (ποτέ δεν θα συναχθεί μια σχέση υπαγωγής που δεν ισχύει), ο αλγόριθμος συνήθως δεν είναι πλήρης (μπορεί δηλαδή να μην συμπεράνει όλες τις ισχύουσες σχέσεις υπαγωγής).

- Είναι δύσκολο να επεκταθούν για πιο εκφραστικές Λογικές Περιγραφής, συγκεκριμένα αυτές που περιλαμβάνουν άρνηση ή να υποστηρίξουν συλλογισμό με αυθαίρετες βάσεις γνώσης. Αυτή η έλλειψη εκφραστικής ισχύος περιορίζει την αξία τους στις εφαρμογές οντολογικής μηχανικής και τους καθιστά άχρηστους σε εφαρμογές συλλογισμού με σχήματα βάσεων δεδομένων.
- Αν και η αποδοχή της μη πληρότητας σε κάποιο βαθμό αποτελεί έναν τρόπο βελτίωσης της επίδοσης ενός μηχανισμού συλλογισμού Λογικών Περιγραφής, η επίδοση των μηχανισμών συλλογισμού που δεν είναι πλήρεις εξαρτάται σημαντικά από το βαθμό αυτό, που είναι ιδιαίτερα δύσκολο να ποσοτικοποιηθεί.

3.5.2. Λογικοί αλγόριθμοι

Οι αλγόριθμοι αυτοί χρησιμοποιούν μια απόδειξη διάψευσης (refutation): Η C υπάγεται στην D , αν μπορεί να δειχθεί ότι η ύπαρξη ενός ατόμου x που βρίσκεται στην έκταση της C ($x \in C'$), αλλά όχι στην έκταση της D ($x \notin D'$) είναι λογικά ασυνεπής. Όπως φάνηκε, ο έλεγχος αυτός αντιστοιχεί στον έλεγχο της λογικής (μη) ικανοποιησιμότητας της έννοιας $C \Pi \neg D$ (δηλαδή $C \sqsubseteq D$ αν και μόνο αν η $C \Pi \neg D$ είναι μη ικανοποιήσιμη). Σημειώνεται ότι ο σχηματισμός μιας τέτοιας έννοιας προϋποθέτει την ύπαρξη άρνησης στη λογική.

Διάφορες τεχνικές μπορούν να χρησιμοποιηθούν για τον έλεγχο της λογικής ικανοποιησιμότητας μίας έννοιας. Μια προφανής δυνατότητα είναι να χρησιμοποιηθούν υπάρχοντες μηχανισμοί συλλογισμού. Για παράδειγμα το σύστημα LOGICSWORKBENCH, ένα σύστημα συλλογισμού προτασιακής τροπικής λογικής (propositional modal logic) μπορεί να χρησιμοποιηθεί εκμεταλλεύμενοι τις αντιστοιχίες μεταξύ Λογικών Περιγραφής και τροπικών λογικών. Επίσης, μηχανισμοί απόδειξης θεωρημάτων ΛΠΤ μπορούν επίσης να χρησιμοποιηθούν, μέσω κατάλληλων μεταφράσεων των Λογικών Περιγραφής σε ΛΠΤ.

Υπάρχουν πλεονεκτήματα και μειονεκτήματα στην προσέγγιση της «επανάχρησης»: Αφενός, είναι πιο εύκολο να κατασκευαστεί ένα σύστημα πάνω σε έναν υπάρχοντα μηχανισμό συλλογισμού, και η απόδοση μπορεί να μεγιστοποιηθεί χρησιμοποιώντας μια προηγμένη υλοποίηση όπως το SPASS (ένας βελτιστοποιημένος μηχανισμός απόδειξης θεωρημάτων) ή τους βελτιστοποιημένους αλγορίθμους SAT. Η μετάφραση επίσης σε ΛΠΤ φαίνεται ότι μπορεί να αντιμετωπίσει αρκετές εκφραστικές Λογικές Περιγραφής, συγκεκριμένα αυτές που υποστηρίζουν σύνθετους τελεστές σχηματισμού ρόλων, όπως η άρνηση και η ταυτότητα.

Αφετέρου, είναι ενδεχομένως δύσκολο να επεκταθεί ο συλλογισμός για πιο εκφραστικές λογικές ή να προστεθούν βελτιστοποιήσεις που εκμεταλλεύονται συγκεκριμένα χαρακτηριστικά μιας Λογικής Περιγραφής, χωρίς να απαιτείται η επανυλοποίηση του μηχανισμού συλλογισμού. Για μια σύγκριση της χρήσης συστημάτων Λογικών Περιγραφής με συστήματα ΛΠΤ και συστήματα παραγωγής (κανόνων) για συλλογισμό με οντολογίες Ιστού βλ. και ενότητα 3.2).

Τα περισσότερα υλοποιημένα συστήματα που βασίζονται σε λογικό συλλογισμό, υλοποιούν σχεδιασμένες κατά περίπτωση διαδικασίες απόφασης ταμπλό (tableau). Οι αλγόριθμοι αυτοί προσπαθούν να αποδείξουν ότι η D υπάγει την C ξεκινώντας με ένα άτομο που ικανοποιεί την $C \Pi \neg D$ και δείχνοντας ότι κάθε προσπάθεια επέκτασης του γεγονότος αυτού σε πλήρη διερμήνευση (χρησιμοποιώντας ένα σύνολο κανόνων επέκτασης ταμπλό) οδηγεί σε λογική αντίφαση. Αν βρεθεί μια πλήρης και μη αντιφατική διερμήνευση

(μια διερμηνευση όπου κάποιο στοιχείο του πεδίου ανήκει στο C' , αλλά όχι στο D'), αυτό αποτελεί αντιπαράδειγμα που αναιρεί την υποτεθείσα σχέση υπαγωγής.

Η προσέγγιση αυτή έχει πολλά πλεονεκτήματα και έχει κυριαρχήσει στην πρόσφατη έρευνα:

- Διαθέτει ισχυρή θεωρητική βάση στη ΛΠΤ.
- Μπορεί σχετικά εύκολα να προσαρμοστεί για ένα εύρος λογικών απλά αλλάζοντας το σύνολο των κανόνων επέκτασης ταμπλό.
- Μπορεί να προσαρμοστεί σε πολύ εκφραστικές Λογικές Περιγραφές και σε συλλογισμό με αυθαίρετες βάσεις γνώσης χρησιμοποιώντας σύνθετους μηχανισμούς ελέγχου για την εξασφάλιση τερματισμού.
- Έχει δειχθεί ότι είναι βέλτιστη για ορισμένες λογικές, με την έννοια ότι η πολυπλοκότητα χειρότερης περίπτωσης του αλγορίθμου δεν είναι χειρότερη από τη γνωστή (θεωρητική) πολυπλοκότητα του προβλήματος της ικανοποιησιμότητας στη συγκεκριμένη λογική.

Έτσι, οι τεχνικές υλοποίησης και βελτιστοποίησης που θα εξεταστούν στην συνέχεια, υποθέτουν τη χρήση διαδικασιών απόφασης ταμπλό. Ωστόσο, πολλές τεχνικές είναι ανεξάρτητες του αλγορίθμου ελέγχου υπαγωγής και μπορούν εύκολα να προσαρμοστούν στις περισσότερες βασισμένες σε λογική μεθόδους. Το αντίστροφο επίσης ισχύει, αφού αρκετές από τις τεχνικές αυτές προέρχονται από άλλες λογικές διαδικασίες απόφασης και συγκεκριμένα εκείνες που επιχειρούν να βελτιστοποιήσουν την αναζήτηση που χρησιμοποιείται για να αντιμετωπιστεί ο μη ντετερμινισμός.

3.5.3. Αλγόριθμοι ταμπλό

Οι αλγόριθμοι ταμπλό επιχειρούν να αποδείξουν την ικανοποιησιμότητα μιας έννοιας D κατασκευάζοντας ένα μοντέλο, μια διερμηνευση δηλαδή I στην οποία το D' δεν είναι κενό.

Ένα ταμπλό είναι ένα γράφημα που αναπαριστά ένα τέτοιο μοντέλο, με κορυφές που αντιστοιχούν σε άτομα (στοιχεία του Δ') και ακμές που αντιστοιχούν σε σχέσεις μεταξύ ατόμων (στοιχεία του $\Delta' \times \Delta'$).

Ένας τέτοιος αλγόριθμος ξεκινά με ένα στιγμιότυπο που ικανοποιεί την D και επιχειρεί να κατασκευάσει ένα ταμπλό ή μια δομή από την οποία μπορεί ένα ταμπλό να κατασκευαστεί, συμπεραίνοντας την ύπαρξη άλλων ατόμων ή περιορισμών σε άτομα. Ο μηχανισμός συμπερασμού συνίσταται στην εφαρμογή ενός συνόλου κανόνων επέκτασης που αντιστοιχούν στις λογικές κατασκευές της γλώσσας και ο αλγόριθμος τερματίζει είτε όταν η δομή είναι πλήρης (δεν είναι δυνατοί περαιτέρω συμπερασμοί) είτε όταν έχουν αποκαλυφθεί προφανείς αντιφάσεις. Ο μη ντετερμινισμός αντιμετωπίζεται με αναζήτηση στις πιθανές διαφορετικές επεκτάσεις. Η έννοια είναι μη ικανοποιήσιμη, όταν κάθε επέκταση οδηγεί σε αντίφαση και είναι ικανοποιήσιμη, όταν κάποια πιθανή επέκταση οδηγήσει σε μια πλήρη, μη αντιφατική δομή.

Για την παρουσίαση των αλγορίθμων ταμπλό χρησιμοποιούνται διάφορες μορφές αναπαράστασης. Στα επόμενα, θα χρησιμοποιηθεί η σημειογραφία γραφημάτων με ετικέτες (labeled graphs), καθώς έχει εμφανή αντιστοιχία προς τις συνήθεις τεχνικές

υλοποίησης. Στη βασική του μορφή, ο συμβολισμός αυτός περιγράφει την κατασκευή ενός διευθυνόμενου γραφήματος (συνήθως δένδρου) στο οποίο κάθε κορυφή (ή κόμβος) x έχει ως ετικέτα ένα σύνολο εννοιών ($L(x) = \{C_1, \dots, C_n\}$) και κάθε ακμή $\langle x, y \rangle$ έχει ως ετικέτα ένα ρόλο ($L(\langle x, y \rangle) = R$). Αν μια έννοια C βρίσκεται στην ετικέτα ενός κόμβου x ($C \in L(x)$), τότε το γράφημα αναπαριστά ένα μοντέλο στο οποίο το άτομο που αντιστοιχεί στο x βρίσκεται στη διερμηνευση της C . Όταν μια ακμή $\langle x, y \rangle$ έχει στην ετικέτα το R ($L(\langle x, y \rangle) = R$), το γράφημα αναπαριστά ένα μοντέλο όπου η πλειάδα που αντιστοιχεί στο $\langle x, y \rangle$ βρίσκεται στη διερμηνευση του R .

Π-κανόνας	αν 1. $(C \sqcap D) \in L(x)$ 2. $\{C, D\} \notin L(x)$ τότε $L(x) \rightarrow L(x) \cup \{C, D\}$
Λ-κανόνας	αν 1. $(C \sqcup D) \in L(x)$ 2. $\{C, D\} \cap L(x) = \emptyset$ τότε είτε $L(x) \rightarrow L(x) \cup \{C\}$ ή $L(x) \rightarrow L(x) \cup \{D\}$
Ξ-κανόνας	αν 1. $\exists R.C \in L(x)$ 2. δεν υπάρχει y έτσι ώστε $L(\langle x, y \rangle) = R$ και $C \in L(y)$ τότε δημιούργησε έναν νέο κόμβο y και ακμή $\langle x, y \rangle$ με $L(y) = \{C\}$ και $L(\langle x, y \rangle) = R$
Υ-κανόνας	αν 1. $\forall R.C \in L(x)$ 2. υπάρχει κάποιο y έτσι ώστε $L(\langle x, y \rangle) = R$ και $C \notin L(y)$ τότε $L(y) \rightarrow L(y) \cup \{C\}$

Πίνακας 3.2: Κανόνες επέκτασης ταμπλό για την ALC.

Ένας κόμβος y ονομάζεται R -διάδοχος (R -successor) ενός κόμβου x , αν υπάρχει μια ακμή $\langle x, y \rangle$ με ετικέτα R , ο x ονομάζεται προκάτοχος (predecessor) του y , αν ο y είναι R -διάδοχος του x και ο x ονομάζεται πρόγονος (ancestor) του y , αν ο x είναι προκάτοχος του y ή υπάρχει κόμβος z ο οποίος είναι προκάτοχος του y και ο x είναι πρόγονος του z .

Μια αντίφαση ή σύγκρουση (clash) εντοπίζεται όταν $\{C, \neg C\} \subseteq L(x)$ για μια έννοια C και έναν κόμβο x .

Για τον έλεγχο της ικανοποιησιμότητας μίας έννοιας D ένας απλός αλγόριθμος αρχικοποιεί το δένδρο, ώστε να περιέχει έναν μόνο κόμβο x (που ονομάζεται ρίζα) με $L(x) = \{D\}$ και κατόπιν επεκτείνει το δένδρο εφαρμόζοντας κανόνες οι οποίοι είτε επεκτείνουν τις ετικέτες των κόμβων είτε προσθέτουν νέα φύλλα. Ένα σύνολο κανόνων επέκτασης για τη Λογική Περιγραφής ALC φαίνεται στον πίνακα 3.2, όπου τα C και D είναι έννοιες και ο R είναι ρόλος. Να σημειωθεί ότι:

- Οι έννοιες θεωρείται ότι βρίσκονται σε κανονική μορφή άρνησης (negation normal form, NNF), δηλαδή οι αρνήσεις εμφανίζονται μόνο σε ονόματα εννοιών. Αυθαίρετες έννοιες ALC μπορούν να μετατραπούν σε κανονική μορφή άρνησης ωθώντας τις αρνήσεις προς τα μέσα, χρησιμοποιώντας τους κανόνες De Morgan και τις ισοδυναμίες $\neg(\exists R.C) \Leftrightarrow (\forall R.\neg C)$ και $\neg(\forall R.C) \Leftrightarrow (\exists R.\neg C)$. Η διαδικασία αυτή μπορεί να επεκταθεί για πιο εκφραστικές λογικές χρησιμοποιώντας πρόσθετες ισοδυναμίες, όπως $\neg(\leq n R) \Leftrightarrow (\geq (n+1) R)$.

- Διαζευγμένες έννοιες $(C \sqcup D) \in L(x)$ οδηγούν σε μη ντετερμινιστική επέκταση. Στην πράξη αυτό αντιμετωπίζεται συνήθως με αναζήτηση: Δοκιμή κάθε πιθανής επέκτασης με τη σειρά, ώσπου να βρεθεί ένα πλήρως εκτεταμένο και χωρίς συγκρούσεις δένδρο ή ώσπου όλες οι πιθανότητες ναδειχθεί ότι οδηγούν σε αντιφάσεις. Σε πιο εκφραστικές λογικές, άλλες κατασκευές, όπως οι μέγιστοι περιορισμοί αριθμού ($\leq n R$), επίσης οδηγούν σε μη ντετερμινιστική επέκταση. Η αναζήτηση των μη ντετερμινιστικών επεκτάσεων είναι το κύριο αίτιο της δυσεπιλυσιμότητας (intractability) στους ταμπλό αλγορίθμους ελέγχου υπαγωγής.
- Οι έννοιες υπαρξιακού περιορισμού ρόλων $\exists R.C \in L(x)$ προκαλούν τη δημιουργία νέων R -διαδόχων, ενώ οι έννοιες καθολικού περιορισμού ρόλων $\forall R.C \in L(x)$ επεκτείνουν τις ετικέτες των R -διαδόχων.

Το δένδρο θεωρείται πλήρως εκτεταμένο, όταν κανένας κανόνας επέκτασης δεν μπορεί να εφαρμοστεί. Αν μπορεί να βρεθεί ένα πλήρως εκτεταμένο και χωρίς συγκρούσεις δένδρο, ο αλγόριθμος επιστρέφει *ικανοποιησιμο*, διαφορετικά επιστρέφει *μη ικανοποιησιμο*.

Πιο εκφραστικές λογικές απαιτούν αρκετές επεκτάσεις στο βασικό αυτό φορμαλισμό. Για παράδειγμα, σε λογικές που περιλαμβάνουν ταυτόχρονα αξιώματα συμπερίληψης ρόλων και κάποια μορφή περιορισμού πληθικότητας, είναι απαραίτητο να δοθούν ετικέτες στις ακμές με σύνολα ονομάτων ρόλων, αντί για ένα μόνο όνομα. Είναι επίσης συχνά απαραίτητο να προστεθεί εντοπισμός κύκλων (που συνήθως ονομάζεται *μπλοκάρισμα-blocking*) στις προϋποθέσεις κάποιων κανόνων συμπερασμού, ώστε να εξασφαλίζεται ο τερματισμός. Η ιδέα είναι να εμποδιστεί η επέκταση ενός κλαδιού, όταν η ίδια ετικέτα κόμβου επανεμφανίζεται στο κλαδί. Το μπλοκάρισμα μπορεί επίσης να οδηγήσει σε πιο σύνθετη αντιστοιχία μεταξύ της δομής που δημιουργείται από τον αλγόριθμο και του μοντέλου μιας ικανοποιήσιμης έννοιας, αφού το μοντέλο μπορεί να περιέχει κύκλους ή ακόμα και να είναι άπειρο.

3.5.4. Συλλογισμός με απτά πεδία

Οι γλώσσες οντολογιών Ιστού διαφέρουν επίσης από τις παραδοσιακές Λογικές Περιγραφής στο ότι επιτρέπεται η εκτεταμένη χρήση τύπων και τιμών δεδομένων, όπως για παράδειγμα αυτών που προέρχονται από το XML Schema. Οι δυνατότητες αυτές μπορεί να θεωρηθεί ότι αντιστοιχίζονται σε μία περιορισμένη μορφή της έννοιας των απτών πεδίων σε μια λογική περιγραφής.

Ένα απτό πεδίο D αποτελείται από ένα σύνολο Δ^D , το πεδίο του D και ένα σύνολο $pred(D)$, τα ονόματα των κατηγορημάτων του D . Κάθε όνομα κατηγορήματος $P \in pred(D)$ συσχετίζεται με μια πολλαπλότητα (arity) n και ένα n -αδικό κατηγορήμα $P^D \subseteq (\Delta^D)^n$.

Για παράδειγμα, το απτό πεδίο N έχει το σύνολο \mathbb{N} των φυσικών αριθμών ως πεδίο και το $pred(N)$ αποτελείται από τα δυαδικά ονόματα κατηγορημάτων $<, \leq, \geq, >$, καθώς και από τα μοναδιαία ονόματα κατηγορημάτων $<_n, \leq_n, \geq_n, >_n$, όπου $n \in \mathbb{N}$ και τα οποία διερμηνεύονται από τα αντίστοιχα κατηγορήματα του \mathbb{N} κατά τον προφανή τρόπο.

Τα προβλήματα συλλογισμού, ακόμα και σε απλές λογικές, με την παρουσία απτών πεδίων εύκολα μπορεί να γίνουν δυσεπίλυτα. Για παράδειγμα η Λογική Περιγραφής *ALC* συνδυασμένη με τον τελεστή μεταβατικής κλειστότητας και απτά πεδία είναι μη αποφασίσιμη [Baader, Küsters, & Wolter, 2007]. Ανάλογη είναι συνήθως η κατάσταση και όταν επιτρέπονται γενικά αξιώματα συμπερίληψης. Για το γεγονός αυτό φαίνεται να ευθύνεται κυρίως η σύγκριση μεταξύ των απτών τιμών και εν γένει τα δυαδικά κατηγορήματα.

Εφόσον όμως στην *OWL* και στις άλλες γλώσσες που αντιστοιχούν σε τέτοιες λογικές η χρήση τύπων δεδομένων δεν απαιτεί την πλήρη εκφραστικότητα των απτών πεδίων, είναι δυνατή η παροχή υπηρεσιών συλλογισμού για τιμές και τύπους, χωρίς (σημαντική) επιβάρυνση της πολυπλοκότητας του αλγορίθμου υπαγωγής για την αντίστοιχη λογική. Η περιορισμένη μορφή των απτών πεδίων που μπορεί να θεωρηθεί, βασίζεται στο γεγονός ότι οι «δύσκολοι» τελεστές της σύγκρισης, καθώς και η κατασκευή σύνθετων τύπων δεδομένων από τους απλούς με τη χρήση περιορισμών, δεν χρειάζεται να ενταχθεί φORMALISτικά στη Λογική Περιγραφής. Αντίθετα θεωρούμε ότι ένα υλοποιημένο σύστημα διαθέτει ένα ξεχωριστό, έξω-λογικό μέρος (σύστημα τύπων) που αναλαμβάνει τη δουλειά αυτή. Κατά τα λοιπά, το σύστημα συμπεριφέρεται σε έναν τύπο δεδομένων σαν να επρόκειτο για μια (απλοποιημένη) μορφή έννοιας και διερμηνεύει μόνο υπαρξιακούς και καθολικούς περιορισμούς που τον εμπλέκουν.

Στη συνέχεια επικεντρώνουμε στο πώς αντιμετωπίζεται ο συλλογισμός με απτά πεδία στην περίπτωση της *SHOQ(D)* [Horrocks & Sattler, 2001]. Η προσέγγιση που ακολουθείται μπορεί να επεκταθεί ομοίωμα και για τις περιπτώσεις της *SHOIQ(D)* και *SROIQ(D)* που αντιστοιχούν στις εκδόσεις της *OWL*.

Οι απτοί τύποι δεδομένων χρησιμοποιούνται για την αναπαράσταση λεκτικών τιμών, όπως αριθμοί και συμβολοσειρές. Ένα σύστημα τύπων ορίζει ένα σύνολο «αρχικών» τύπων δεδομένων, όπως *συμβολοσειρά* και *ακέραιος* και παρέχει το μηχανισμό για την παραγωγή νέων τύπων από τους υπάρχοντες. Για παράδειγμα στο XML Schema ο τύπος *nonNegativeInteger* προέρχεται από τον τύπο *integer* περιορίζοντας τις τιμές του *integer* να είναι μεγαλύτερες ή ίσες με το μηδέν.

Για την αναπαράσταση εννοιών της μορφής «άνθρωποι ηλικίας τουλάχιστον 21», η γλώσσα εννοιών μπορεί να επεκταθεί με ένα σύνολο D απτών τύπων δεδομένων και με έννοιες της μορφής $\exists T.d$ και $\forall T.d$. Υπάρχει δηλαδή ένα σύνολο τύπων δεδομένων D και με κάθε $d \in D$ συσχετίζεται ένα σύνολο $d^D \subseteq \Delta_D$, όπου Δ_D είναι το πεδίο όλων των τύπων δεδομένων. T είναι ένας απτός ρόλος που συσχετίζει άτομα του αφηρημένου με άτομα του απτού πεδίου ($T' \subseteq \Delta' \times \Delta_D'$). Οι παραπάνω έννοιες διερμηνεύονται ως εξής:

$$(\exists T.d)' = \{x \mid \exists y. \langle x, y \rangle \in T' \text{ και } y \in d^D\}$$

$$(\forall T.d)' = \{x \mid \forall y. \langle x, y \rangle \in T' \text{ συνεπάγεται } y \in d^D\}$$

Το σχήμα αυτό μπορεί να επεκταθεί για να περιλάβει δυαδικούς συνδυασμούς τύπων δεδομένων και προσδιορισμένους περιορισμούς αριθμού με τύπους, χωρίς πρόβλημα.

Ισχύουν οι ακόλουθες υποθέσεις:

1. Το πεδίο διερμίνευσης όλων των απτών τύπων Δ_D (το απτό πεδίο) είναι ξένο προς το πεδίο διερμίνευσης της γλώσσας εννοιών (το αφηρημένο πεδίο).

2. Υπάρχει συνεπής και πλήρης διαδικασία απόφασης που ελέγχει την κενότητα έκφρασης της μορφής $d_1^D \cap \dots \cap d_n^D$, όπου d_i είναι ένας (πιθανώς με άρνηση) απτός τύπος δεδομένων του D (όπου $\neg d$ διερμηνεύεται ως $\Delta_D \setminus d^D$).

Λέμε πως ένα σύνολο τύπων δεδομένων *συμμορφώνεται* εφόσον πληροί τα παραπάνω κριτήρια.

Το γεγονός ότι το αφηρημένο και το απτό πεδίο είναι μεταξύ τους ξένα, προέρχεται τόσο από φιλοσοφικές όσο και από πραγματιστικές αιτίες. Από τη μια, οι απτοί τύποι θεωρείται ότι είναι ήδη επαρκώς δομημένοι από το σύστημα τύπων, το οποίο μπορεί να περιλαμβάνει κάποιο μηχανισμό παραγωγής, καθώς και ενσωματωμένες σχέσεις διάταξης και σύγκρισης. Από την άλλη, επιτρέπει την ενασχόληση με αυθαίρετα συμμορφούμενα σύνολα τύπων, χωρίς να διακινδυνεύει την παγίωση της γλώσσας εννοιών ή τη συνέπεια και πληρότητα της διαδικασίας απόφασης.

Το σύστημα τύπων μπορεί να είναι περίπλοκο, όπως στην περίπτωση του XML Schema ή τόσο απλό όσο στην περίπτωση της γλώσσας OIL, όπου οι μόνοι αρχικοί τύποι είναι οι ακέραιοι και οι συμβολοσειρές και νέοι τύποι μπορούν να παραχθούν μέσω περιορισμών μέγιστου και ελάχιστου. Για παράδειγμα, μπορεί να οριστεί ο τύπος (min 21) και να χρησιμοποιηθεί στην έννοια Person $\Pi \text{Age}(\text{min } 21)$.

Είναι λοιπόν φανερό πως ένας αλγόριθμος ταμπλό μπορεί πολύ εύκολα να επεκταθεί για μια Λογική Περιγραφής που επαυξάνεται με αυτήν την περιορισμένη μορφή απτών πεδίων. Για την περίπτωση της SHOQ(D) οι επιπλέον κανόνες ταμπλό παρουσιάζονται στον πίνακα 3.3. Αντίστοιχα, οι συνθήκες σύγκρουσης επαυξάνονται για να συμπεριλάβουν και την περίπτωση όπου η ετικέτα ενός κόμβου $L(x)$ καταλήξει να περιέχει ξένους μεταξύ τους τύπους δεδομένων d_1, \dots, d_n , δηλαδή όταν $d_1^D \cap \dots \cap d_n^D$ είναι κενό. Λόγω της δεύτερης υπόθεσης ο έλεγχος της συνθήκης αυτής δεν επηρεάζει την αποφασιστικότητα του προβλήματος και μπορεί ενδεχομένως να γίνει από το σύστημα τύπων.

\exists_D-κανόνας	αν	1. $\exists T.d \in L(x)$
	2. δεν υπάρχει y έτσι ώστε $L(\langle x, y \rangle) = T$ και $d \in L(y)$	
τότε	δημιούργησε έναν νέο κόμβο y και ακμή $\langle x, y \rangle$ με $L(y) = \{d\}$ και $L(\langle x, y \rangle) = T$	
\forall_D-κανόνας	αν	1. $\forall T.d \in L(x)$
	2. υπάρχει κάποιο y έτσι ώστε $L(\langle x, y \rangle) = T$ και $d \notin L(y)$	
τότε	$L(y) \rightarrow L(y) \cup \{d\}$	

Πίνακας 3.3: Επέκταση των κανόνων ταμπλό για υποστήριξη απτών πεδίων στην περίπτωση της SHOQ(D).

Αξίζει όμως να σημειωθεί ότι ο συνδυασμός αντίστροφων ρόλων είτε με απτά πεδία είτε με ονοματικά οδηγεί σε πρόβλημα ικανοποιησιμότητας, με πολυπλοκότητα τουλάχιστον NEXP (NEXP-hard). Έτσι ούτε στην OWL, ούτε στην OWL 1.1 δεν επιτρέπεται να οριστεί ο αντίστροφος ενός απτού ρόλου. Για το λόγο αυτό στο [Horrocks, Kutz, & Sattler, 2006] αναγνωρίζεται ως σημαντική γραμμή περαιτέρω έρευνας η εξέταση τύπων δεδομένων που να επιτρέπουν αντίστροφες συναρτησιακές ιδιότητες: Τέτοιες ιδιότητες έχουν ενδιαφέρον γιατί επιτρέπουν την έκφραση απλών περιορισμών κλειδιού. Για παράδειγμα, μπορεί να είναι επιθυμητή η χρήση μιας ιδιότητας τύπων δεδομένων AMKA για τον αριθμό μητρώου κοινωνικής ασφάλισης ως κλειδιού για τους Έλληνες πολίτες.

3.6. Υλοποίηση και Βελτιστοποίηση Αλγορίθμων Ταμπλό

Η θεωρητική διατύπωση ενός αλγορίθμου ταμπλό είναι συνήθως σχεδιασμένη, ώστε να διευκολύνει περισσότερο την απόδειξη ορθότητας, παρά την υλοποίηση. Δεν παρέχει, για παράδειγμα, καμία πληροφορία για τη σειρά με την οποία εφαρμόζονται οι κανόνες ή για το πώς αντιμετωπίζεται ο μη ντετερμινισμός στην πράξη.

Η πρώτη απόκλιση θεωρίας και πράξης είναι ότι οι εκφράσεις εννοιών δεν είναι συνήθως σε NNF. Αντίθετα, οι αλγόριθμοι χρησιμοποιούν μια απλοποιημένη κανονική μορφή μαζί με έναν πρόσθετο κανόνα -τον --κανόνα- που επιτρέπουν ταχύτερη εύρεση των συγκρούσεων.

Δεύτερον, οι πρακτικοί αλγόριθμοι συνήθως καταγράφουν τις αλλαγές που γίνονται στο γράφημα επέκτασης (για παράδειγμα, ποιες έννοιες έχουν προστεθεί σε ετικέτες κόμβων ή τη δημιουργία νέων ακμών στο γράφημα) και διατηρούν μια διάταξη των εννοιών που πρέπει να επεκταθούν περαιτέρω. Επειδή μπορεί πολλοί κανόνες να είναι εφαρμόσιμοι σε μια δεδομένη στιγμή, πρέπει να επιλεγεί μια διάταξη εφαρμογής κανόνων σε μια ντετερμινιστική υλοποίηση.

Τρίτον, όποτε εφαρμόζεται ένας μη ντετερμινιστικός κανόνας, ένας ντετερμινιστικός αλγόριθμος πρέπει να δημιουργήσει ένα *σημείο διακλάδωσης* στο οποίο ο αλγόριθμος αποθηκεύει την κατάσταση της διαδικασίας συλλογισμού (που περιλαμβάνει, για παράδειγμα, το γράφημα επέκτασης). Στη συνέχεια ο αλγόριθμος δοκιμάζει (με κάποια σειρά) τις πιθανές εφαρμογές του κανόνα, επαναφέροντας την κατάσταση μετά από κάθε προσπάθεια, ώσπου να οδηγηθεί σε ένα πλήρες και χωρίς συγκρούσεις γράφημα ή να δειχθεί ότι όλα οδηγούν σε σύγκρουση. Στην πρώτη περίπτωση ο αλγόριθμος τερματίζει και επιστρέφει «ικανοποιησιμο» στη δεύτερη *υπαναχωρεί* (backtracks) στο προηγούμενο σημείο διακλάδωσης, αν υπάρχει ή διαφορετικά τερματίζει και επιστρέφει «μη ικανοποιησιμο».

Είναι φανερό επίσης πως, σε μια πρακτική υλοποίηση, υπονοούνται δύο διατάξεις: η σειρά με την οποία θα εφαρμοστούν οι κανόνες επέκτασης και η σειρά με την οποία θα διερευνηθούν οι διαφορετικές πιθανές επεκτάσεις ενός μη ντετερμινιστικού κανόνα. Οι διατάξεις που θα επιλεγούν μπορεί να επηρεάσουν σημαντικά το χρόνο και το χώρο που θα χρησιμοποιήσει ο αλγόριθμος και πολλές βελτιστοποιήσεις αποσκοπούν να βρουν «καλές» διατάξεις για αυτές τις περιπτώσεις [Tsarkov, Horrocks, & Patel-Schneider, 2007].

Γίνεται επομένως σαφές πως, κατά την υλοποίηση στην πράξη ενός ταμπλό αλγορίθμου ελέγχου ικανοποιησιμότητας, θα πρέπει να ληφθούν υπόψη και να εφαρμοστούν συγκεκριμένες τεχνικές βελτιστοποίησης. Στόχος είναι, αφενός, η πολυπλοκότητα του αλγορίθμου να μην ξεπερνά πολύ την αντίστοιχη θεωρητική πολυπλοκότητα του προβλήματος και αφετέρου, ο αλγόριθμος να εμφανίζει βελτιωμένη συμπεριφορά στις πιο συνήθεις περιπτώσεις.

Αξίζει επίσης να σημειωθεί ότι, ενώ ο έλεγχος ικανοποιησιμότητας για εκφραστικές Λογικές Περιγραφές απαιτεί, στη χειρότερη περίπτωση, χρόνο τουλάχιστον εκθετικό ως προς το μέγεθος της εισόδου, οι περισσότερες τεχνικές βελτιστοποίησης έχουν πολυωνυμική ή ακόμα και γραμμική πολυπλοκότητα χειρότερης περίπτωσης. Δύο πολύ συνηθισμένες τέτοιες τεχνικές που έχουν εμφανές αποτέλεσμα στην αύξηση της επίδοσης είναι το *οκνηρό ξεδίπλωμα* (lazy unfolding), όπου οι έννοιες ξεδιπλώνονται μόνο όταν απαιτείται (όταν για παράδειγμα μια έννοια εμφανιστεί στην ετικέτα ενός κόμβου) και η *απορρόφηση* (absorption), βάσει της οποίας τα γενικά αξιώματα μπορούν να

αντικατασταθούν (να απορροφηθούν) από αρχικούς ορισμούς. Ορισμένες άλλες τεχνικές παρατίθενται πιο αναλυτικά στη συνέχεια, ενώ για μια πιο διεξοδική περιγραφή ο αναγνώστης παραπέμπεται στα [Horrocks I. , 2007] και [Tsarkov, Horrocks, & Patel-Schneider, 2007].

3.6.1. Λεξική κανονικοποίηση και απλοποίηση

Η υπόθεση ότι η είσοδος ενός ταμπλό αλγορίθμου είναι σε NNF μπορεί να απλοποιεί την περιγραφή του αλγορίθμου, αλλά σημαίνει ότι η πρώτη (και η συνηθέστερη) μορφή σύγκρουσης, $\{C, \neg C\} \subseteq L(x)$ για κάποιο κόμβο x στο γράφημα, θα εντοπιστεί αν η C είναι όνομα έννοιας. Για παράδειγμα, κατά τον έλεγχο της ικανοποιησιμότητας της έννοιας $(A \sqcap B) \sqcap \neg(A \sqcap B)$ ο μετασχηματισμός σε NNF θα δώσει $(A \sqcap B) \sqcap (\neg A \sqcup \neg B)$. Στην πράξη αυτό σημαίνει ότι, παρά την προφανή αντίφαση, θα εφαρμοστεί αναζήτηση υπαναχώρησης για να καθοριστεί ότι η έννοια είναι μη ικανοποιήσιμη.

Για το λόγο αυτό οι πρακτικοί αλγόριθμοι δεν μετασχηματίζουν την είσοδο σε NNF, αλλά περιλαμβάνουν έναν \neg -κανόνα που πραγματοποιεί ένα μόνο βήμα κανονικοποίησης: η εφαρμογή του \neg -κανόνα στο $\neg(A \sqcap B) \in L(x)$ οδηγεί στην προσθήκη του $\neg A \sqcup \neg B$ στο $L(x)$ και ένα γράφημα περιέχει σύγκρουση αν περιέχει έναν κόμβο x με $\{C, \neg C\} \subseteq L(x)$, όπου C μια αυθαίρετη (μη ατομική) έννοια. Επίσης, για τη διευκόλυνση του εντοπισμού τέτοιων συγκρούσεων, η είσοδος κανονικοποιείται και απλοποιείται, ώστε λογικά ισοδύναμες έννοιες να είναι και συντακτικά ισοδύναμες. Αυτό επιτυγχάνεται εφαρμόζοντας (αναδρομικά) στις εκφράσεις εννοιών ένα σύνολο κανόνων επανεγγραφής και εφαρμόζοντας μια ολική διάταξη στα στοιχεία των διαζεύξεων. Για παράδειγμα οι \sqcap - και \exists -έννοιες ξαναγράφονται ως αρνημένες (negated) \sqcup - και \forall -έννοιες αντίστοιχα, οι συζεύξεις συγχωνεύονται όποτε είναι δυνατόν, οι συζευκτές έννοιες διατάσσονται και χρησιμοποιούνται λογικές ισοδυναμίες και μετασχηματισμοί για την απλοποίηση των εννοιών. Τα τελευταία χωρίζονται σε τρεις κατηγορίες: *απαλοιφή σταθεράς* (constant elimination), *συντακτικές ισοδυναμίες* (syntactic equivalences) και *σημαντικοί μετασχηματισμοί* (semantic transformations).

Η κατηγορία απαλοιφής σταθεράς περιλαμβάνει τις ακόλουθες ισοδυναμίες:

$$(C \sqcap T) \equiv C, (C \sqcap \perp) \equiv \perp, \forall R.T \equiv T$$

Η κατηγορία συντακτικής ισοδυναμίας περιλαμβάνει τις ακόλουθες ισοδυναμίες:

$$(C \sqcap C) \equiv C, \neg\neg C \equiv C, C \sqcap \neg C \equiv \perp, \geq 1R.C \equiv \exists R.C$$

Η κατηγορία των σημαντικών μετασχηματισμών εκμεταλλεύεται της σημασιικές πληροφορίες που έχουν ήδη συλλεχθεί κατά τη φάση της προεπεξεργασίας για να απλοποιήσει τις εκφράσεις εννοιών. Για παράδειγμα, αν το TBox περιέχει το αξίωμα $A \sqsubseteq B$ για ονόματα εννοιών A και B , η έκφραση έννοιας $A \sqcap B$ μπορεί να απλοποιηθεί σε A . Το ίδιο ισχύει και για αυθαίρετες εκφράσεις εννοιών.

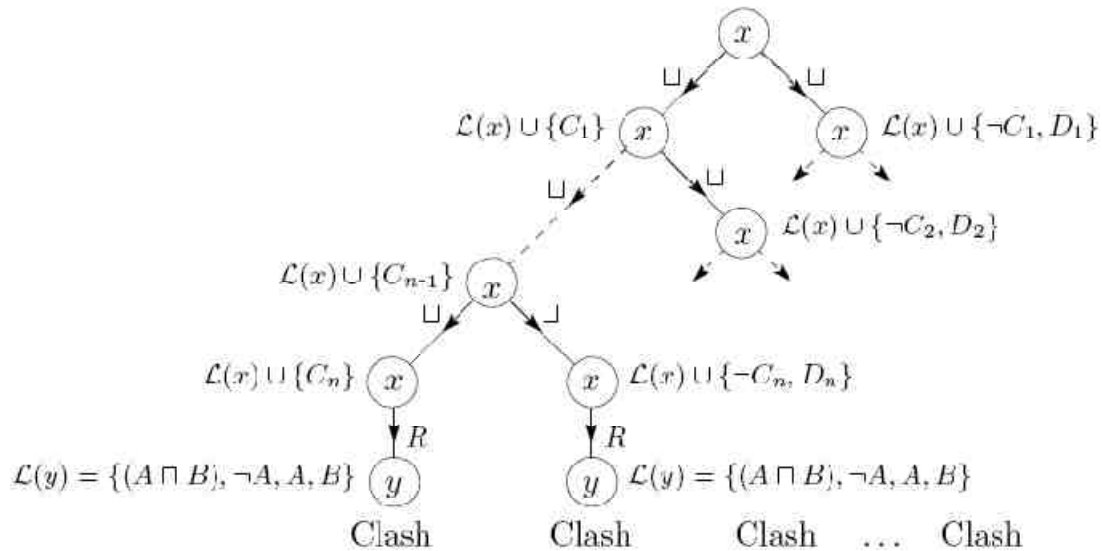
3.6.2. Υπαναχώρηση κατευθυνόμενη από εξάρτηση

Έστω η ακόλουθη έννοια C της μορφής:

$$(C_1 \sqcup D_1) \sqcap \dots \sqcap (C_n \sqcup D_n) \sqcap \exists R.(A \sqcap B) \sqcap \forall R.\neg A$$

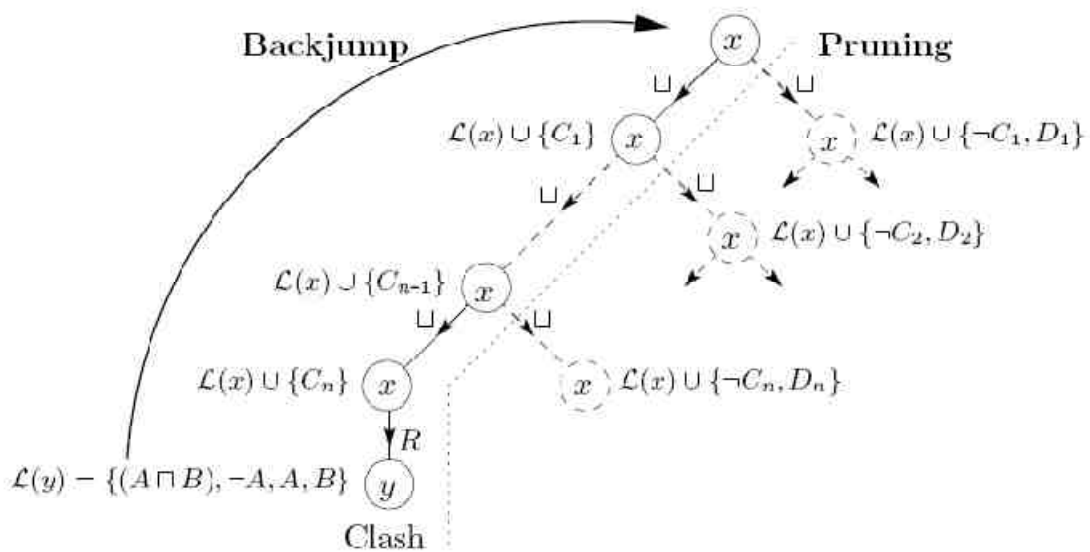
Σε μια κλασική προσέγγιση διάταξης κανόνων από πάνω προς τα κάτω (top-down, όπου οι παραγωγικοί κανόνες, δηλαδή αυτοί που δημιουργούν νέους κόμβους, εφαρμόζονται

τελευταίοι) οι διαζεύξεις στην έννοια αυτή θα επεκταθούν πριν από τον υπαρξιακό περιορισμό. Επομένως, ένας αλγόριθμος ταμπλό θα επεκτείνει πρώτα n διαζεύξεις και μετά θα βρει τη σύγκρουση λόγω του \exists -κανόνα. Στην περίπτωση της κανονικής (μη βελτιστοποιημένης) υπαναχώρησης, 2^n διαφορετικές επεκτάσεις διαζεύξεων θα δοκιμαστούν, προτού διαπιστωθεί ότι η έννοια είναι μη ικανοποιήσιμη. Το φαινόμενο αυτό, γνωστό και ως λυγισμός (thrashing) φαίνεται στην εικόνα 3.2.



Εικόνα 3.2: Λυγισμός κατά την αναζήτηση υπαναχώρησης.

Για την αποφυγή αυτής της εκθετικής συμπεριφοράς κατά τον έλεγχο της ικανοποιησιμότητας της C και παρόμοιων εννοιών, μια πιο σύνθετη λύση είναι να εφαρμοστεί μια μορφή υπαναχώρησης κατευθυνόμενης από εξάρτηση (dependency-directed backtracking), που ονομάζεται υπαναχώρηση με άλμα (backjumping).



Εικόνα 3.3: Κλάδεμα της αναζήτησης με χρήση άλματος προς τα πίσω.

Διαισθητικά, η υπαναχώρηση με άλμα λειτουργεί μαρκάροντας κάθε έννοια C που βρίσκεται στην ετικέτα ενός κόμβου x με ένα σύνολο εξάρτησης $Dep_C(x)$ που προσδιορίζει

τα σημεία διακλάδωσης (δηλαδή τις εφαρμογές ενός μη ντετερμινιστικού κανόνα) από τα οποία εξαρτάται. Στην περίπτωση που το γράφημα περιέχει κόμβο x με $\{C, \neg C\} \in L(x)$, χρησιμοποιούνται τα $Dep_C(x)$ και $Dep_{\neg C}(x)$ για να εντοπιστεί το πιο πρόσφατο σημείο διακλάδωσης b , από το οποίο εξαρτάται είτε η C είτε η $\neg C$. Ο αλγόριθμος μπορεί τότε να κάνει άλμα πίσω στο b περνώντας πάνω τα άλλα σημεία διακλάδωσης που μεσολαβούν, χωρίς να εξετάσει τις εναλλακτικές διακλαδώσεις και να κάνει μια διαφορετική μη ντετερμινιστική επιλογή, που μπορεί να μην οδηγήσει στην ίδια συνθήκη σύγκρουσης που εντοπίστηκε. Αν δεν υπάρχει τέτοιο b , η σύγκρουση δεν εξαρτάται από κάποια μη ντετερμινιστική επιλογή και ο αλγόριθμος τερματίζει, επιστρέφοντας «μη ικανοποιήσιμο».

Η εικόνα 3.3 παρουσιάζει την εφαρμογή της τεχνικής αυτής στην περίπτωση της έννοιας C που περιγράφηκε προηγουμένως. Η σύγκρουση που εντοπίζεται προέρχεται από τις έννοιες A και $\neg A$ για τις οποίες όμως τα σύνολα εξάρτησης είτε είναι κενά είτε $Dep_A(y) = Dep_{\exists R.(A \sqcap B)}(x)$ και $Dep_{\neg A}(y) = Dep_{\forall R.\neg A}(x)$, αφού η εμφάνισή τους προήλθε αποκλειστικά από την εφαρμογή ντετερμινιστικών κανόνων. Έτσι, το πιο πρόσφατο b είναι η κορυφή του δένδρου και ο αλγόριθμος κλαδεύει το δένδρο αναζήτησης, αποφεύγοντας τη δημιουργία και εξερεύνηση $2^n - 1$ R -διαδόχων.

3.6.3. Απαλοιφή ABox

Οι αλγόριθμοι ταμπλό για την ικανοποιησιμότητα μιας έννοιας Λογικών Περιγραφής συνήθως φαίνεται να αγνοούν την ύπαρξη των δηλωτικών αξιωμάτων, δηλαδή αυτών που αφορούν στα άτομα. Επίσης, οι διαδικασίες απόφασης επικεντρώνουν στην ικανοποιησιμότητα μίας έννοιας και δεν ασχολούνται με τα αντίστοιχα δηλωτικά προβλήματα, όπως η συνέπεια του ABox. Για συγκεκριμένες όμως εκφραστικές λογικές, κυρίως αυτές που υποστηρίζουν άρνηση και ονοματικά, όλα τα δηλωτικά αξιώματα μπορούν να αντικατασταθούν από ισοδύναμες ορολογικές εκφράσεις, μέσα από μια διαδικασία που είναι γνωστή ως απαλοιφή ABox. Τα δηλωτικά αξιώματα περιλαμβάνουν:

Μια ατομική δήλωση (individual assertion) έχει μια από τις παρακάτω μορφές: $a:C$, $(a,b):R$, $(a,b):\neg S$ ή $a \neq b$, για $a, b \in I$ (το σύνολο των ατομικών ονομάτων), έναν (πιθανώς αντίστροφο) ρόλο R , έναν (πιθανώς αντίστροφο) απλό ρόλο S και μια έννοια C . Το ABox A είναι ένα πεπερασμένο σύνολο ατομικών δηλώσεων.

Έτσι, έστω C η έννοια της οποίας η ικανοποιησιμότητα πρόκειται να ελεγχθεί. Για την απαλοιφή, το ABox A αντικαθίσταται από ένα νέο ABox A' που περιέχει μόνο ατομικές δηλώσεις της μορφής $a:D$. Προς το σκοπό αυτό, κάθε άτομο a αντικαθίσταται από ένα νέο ονοματικό o_a που δεν υπάρχει στο TBox ή στη C . Το A' προκύπτει στη συνέχεια αντικαθιστώντας κάθε ατομική δήλωση του A της μορφής $(a, b):R$ με $a \in \exists R.o_b$, κάθε $(a, b):\neg R$ με $a \in \forall R.\neg o_b$ και κάθε $a \neq b$ με $a:\neg o_b$. Έτσι, δοθείσης της C και του A' , η C' ορίζεται ως εξής:

$$C' = C \sqcap \prod_{a:D \in A'} \exists U.(o_a \sqcap D)$$

όπου U ο καθολικός ρόλος. Η C θα είναι ικανοποιήσιμη αν και μόνο αν η C' είναι ικανοποιήσιμη, αγνοώντας αυτή τη φορά το ABox.

Κατ' αντιστοιχία, το πρόβλημα της συνέπειας του ABox μπορεί να αναχθεί (και επομένως έχει την ίδια πολυπλοκότητα) στο πρόβλημα της ικανοποιησιμότητας μιας έννοιας, ως εξής:

Ένα ABox A είναι συνεπές ως προς ένα TBox αν η παρακάτω έννοια είναι ικανοποιήσιμη ως προς το ίδιο TBox:

$$\prod_{a:C \in A} \exists U. (\{a\} \sqcap C) \sqcap \prod_{(a,b):R \in A} \exists U. (\{a\} \sqcap \exists R. \{b\} \sqcap \prod_{(a,b):\neg R \in A} \exists U. (\{a\} \sqcap \forall R. \neg \{b\})$$

Όπου $\{a\}$, $\{b\}$ ονοματικά, δηλαδή έννοιες-μονοσύνολα που περιλαμβάνουν μόνο το άτομο a ή το b αντίστοιχα. Για τον ορισμό του καθολικού ρόλου U (ο οποίος και αυτός μπορεί να απαλειφθεί) βλ. και ενότητα 3.4.4.

3.7. Η Λογική Περιγραφής SROIQ

Όπως φάνηκε στην ενότητα 2.5, η OWL 1.1 (ή 2) έχει ως βάση τη Λογική Περιγραφής SROIQ. Στην πραγματικότητα, η SROIQ επεκτείνει τη Λογική Περιγραφής SHOIN, στην οποία βασίζεται η OWL DL, με τη λογική RIQ [Horrocks & Sattler, 2004]. Τα επιπλέον εκφραστικά γνωρίσματα που προστίθενται είναι:

- *Γενικά αξιώματα συμπερίληψης ρόλων* (general Role Inclusion Axioms, RIAs): Αν R , S και Q ονόματα ρόλων, τότε μπορεί να εκφραστεί ότι $R \circ S \sqsubseteq Q$. Με τον τρόπο αυτό υπονοείται η έκφραση των αλυσίδων ρόλων.
- *Ιδιότητες ρόλων*:
 - $\text{Tra}(R)$: Μεταβατικός ρόλος (υπάρχει και στην SHOIQ).
 - $\text{Sym}(R)$: Συμμετρικός ρόλος (υπάρχει και στην SHOIQ).
 - $\text{Asy}(R)$: Μη συμμετρικός ρόλος.
 - $\text{Ref}(R)$: Ανακλαστικός ρόλος.
 - $\text{Irr}(R)$: Μη ανακλαστικός ρόλος.
 - $\text{Dis}(R, S)$: Ξένοι ρόλοι.
- Εισάγεται ο *καθολικός ρόλος* (universal role) U : $U' \equiv \Delta' \times \Delta'$.
- Εισάγεται η ειδική έννοια $\exists R.\text{Self}$, ως η έννοια όλων των ατόμων που έχουν σχέση R με τον εαυτό τους (*τοπική ανακλαστικότητα*).
- *Αρνητικές δηλώσεις ρόλων*: Για a, b άτομα μπορεί να εκφραστεί ότι $\langle a, b \rangle: \neg R$.

Στα [Horrocks & Sattler, 2005; Horrocks & Sattler, 2007] και [Horrocks, Kutz, & Sattler, 2005; Horrocks, Kutz, & Sattler, 2006] δίνονται συνεπείς και πλήρεις αλγόριθμοι ταμπλό που αποφασίζουν το πρόβλημα ελέγχου της ικανοποιησιμότητας μιας έννοιας SHOIQ και SROIQ αντίστοιχα. Στη συνέχεια εξετάζουμε τους αλγορίθμους αυτούς, τις υποθέσεις πάνω στις οποίες βασίζονται, καθώς και την πολυπλοκότητά τους. Πρώτα όμως εντοπίζονται οι λόγοι που καθιστούν το συλλογισμό ιδιαίτερα δύσκολο στις περιπτώσεις αυτές.

3.7.1. Η δυσκολία του συλλογισμού

Παρά τον μη ντετερμινισμό σε έναν αλγόριθμο ταμπλό, οι περισσότερες Λογικές Περιγραφής έχουν δύο ενδιαφέρουσες ιδιότητες που τους επιτρέπουν να είναι άμεσα αποφασίσιμες: την *ιδιότητα μοντέλου δένδρου* (tree model property) και την *ιδιότητα πεπερασμένου μοντέλου* (finite model property).

(Ιδιότητα μοντέλου δένδρου) Αν μια οντολογία είναι συνεπής, τότε έχει ένα μοντέλο στο οποίο η δομή των σχέσεων σχηματίζει ή μπορεί να ειπωθεί ως δένδρο.

(ιδιότητα πεπερασμένου μοντέλου) *Αν μια οντολογία είναι συνεπής, τότε έχει ένα μοντέλο που είναι πεπερασμένο.*

Επειδή ένας αλγόριθμος ταμπλό λειτουργεί ψάχνοντας να βρει ένα δενδρικό γράφημα πλήρωσης, η ιδιότητα μοντέλου δένδρου είναι κρίσιμης σημασίας για την επιτυχία του, όπως η ιδιότητα πεπερασμένου μοντέλου είναι κρίσιμης σημασίας για τον τερματισμό του. Ήδη όμως από τη *SHIQ*, η αλληλεπίδραση μεταξύ αντίστροφων ρόλων και περιορισμών αριθμού (πληθικότητας) οδηγεί στην απώλεια της ιδιότητας πεπερασμένου μοντέλου, αφού υπάρχουν οντολογίες στη λογική αυτή που έχουν μόνο μη πεπερασμένα (άπειρα) μοντέλα. Στην περίπτωση αυτή εφαρμόζεται η τεχνική του μπλοκαρίσματος, ώστε ο αλγόριθμος ταμπλό να μπορεί να δημιουργήσει ένα πεπερασμένο, δενδρικής μορφής γράφημα, όπου κάθε (μπλοκαρισμένος) κόμβος, μπορεί να ξεδιπλωθεί (ungravel) σε άπειρα στοιχεία του μοντέλου.

Στην περίπτωση των ονοματικών, όταν δεν υπάρχουν ταυτόχρονα και περιορισμοί αριθμού ή αντίστροφοι ρόλοι ή όταν τα ονοματικά περιορίζονται σε άτομα του *ABox*, τότε ο αλγόριθμος ταμπλό μπορεί να δημιουργήσει ένα δάσος (forest), δηλαδή μια συλλογή από πολλά δενδρικά γραφήματα, καθένα από τα οποία έχει ως ρίζα ένα ονοματικό.

Η δυσκολία επομένως στην περίπτωση της *SHOIQ* (και άρα της *SROIQ*) προέρχεται από την ταυτόχρονη αλληλεπίδραση μεταξύ ονοματικών, περιορισμών αριθμού και αντίστροφων ρόλων που οδηγεί στη σχεδόν ολοκληρωτική απώλεια της ιδιότητας μοντέλου δένδρου. Για παράδειγμα, έστω μια οντολογία που περιέχει τα αξιώματα:

$$T \sqsubseteq \exists R^-.o, o \sqsubseteq (\leq 17 R.F)$$

Το πρώτο απαιτεί ότι, σε ένα μοντέλο αυτής της οντολογίας, κάθε στοιχείο θα πρέπει να έχει μια εισερχόμενη *R*-ακμή από το *o*. Το δεύτερο αξίωμα περιορίζει τον αριθμό των *R*-ακμών που κατευθύνονται από το *o* προς στιγμιότυπα της *F* σε το πολύ 17. Έμμεσα λοιπόν η έννοια *F* περιορίζεται να έχει το πολύ 17 άτομα. Αν επίσης προστεθεί το ακόλουθο αξίωμα:

$$F \sqsubseteq (\geq 17 R.F)$$

αυτό σημαίνει ότι κάθε στοιχείο του *F* θα πρέπει να συνδέεται με όλα τα υπόλοιπα, συν τον εαυτό του, μέσω της σχέσης *R*. Επιπλέον, η προσθήκη του αξιώματος:

$$F \sqsubseteq (\geq 1 S.F) \sqcap (\leq 1 S^-.F)$$

σημαίνει ότι, αφού η *F* έχει συγκεκριμένο αριθμό στοιχείων, πρέπει να υπάρχουν *S*-κύκλοι μεταξύ των στιγμιότυπων της *F*. Επομένως, ένας αλγόριθμος ταμπλό για την *SHOIQ* θα πρέπει να μπορεί να χειριστεί αυθαίρετες σχεσιακές δομές (π.χ. κύκλους).

Επιπρόσθετα, όπως και η *SHIQ*, η *SHOIQ* δεν έχει την ιδιότητα του πεπερασμένου μοντέλου. Για παράδειγμα, έστω το ακόλουθο αξίωμα:

$$N \sqsubseteq (\leq 1 P^-.T) \sqcap \exists P.N.$$

Η έννοια $\neg N \sqcap \exists P.N$ είναι ικανοποιήσιμη ως προς το παραπάνω αξίωμα, μόνο σε μοντέλα όπου τα στιγμιότυπα της *N* είναι άπειρα.

Έστω λοιπόν μια οντολογία που περιέχει όλα τα παραπάνω αξιώματα. Η συνέπεια της οντολογίας αυτής εξαρτάται κρίσιμα από τις σχέσεις μεταξύ των στιγμιότυπων της *F* και τις *N*. Για παράδειγμα, η προσθήκη των ακόλουθων αξιωμάτων:

$$N \sqsubseteq \exists V.F, F \sqsubseteq (\leq kV.N)$$

καθιστά την οντολογία ασυνεπή. Πράγματι, σύμφωνα με το πρώτο αξίωμα, κάθε ένα από τα άπειρα στιγμιότυπα της *N* πρέπει να συνδέεται, μέσω της *V*, με τουλάχιστον ένα από τα 17 στιγμιότυπα της *F*. Αυτό μπορεί να γίνει μόνο αν υπάρχει άπειρο πλήθος ακμών *V* από

την N στην F . Λόγω του δεύτερου αξιώματος όμως, το πλήθος των ακμών που μπορεί να δεχθεί η F από την N για κάθε ένα από τα 17 στιγμιότυπά της είναι το πολύ k .

Επομένως, ένας αλγόριθμος ταμπλό για τη $SHOIQ$ θα πρέπει να μπορεί να αντιμετωπίσει ταυτόχρονα αυθαίρετες σχεσιακές δομές, καθώς και πεπερασμένες δενδρικές δομές που αναπαριστούν άπειρα δένδρα και να εξασφαλίζει ότι ικανοποιούνται όλοι οι περιορισμοί - ιδιαίτερα οι περιορισμοί αριθμού στις σχέσεις μεταξύ αυτών των δύο κομματιών- ενώ ταυτόχρονα να εγγυάται τον τερματισμό.

3.7.2. Αλγόριθμοι ταμπλό για $SHOIQ$ και $SROIQ$

Βλέπουμε λοιπόν γιατί τα ονοματικά αποτελούν ένα μείζον πρόβλημα στο συλλογισμό με την OWL DL, και άρα με την OWL 1.1, όπως επιβεβαιώνεται και πειραματικά (βλ. π.χ. [Koutsomitropoulos, Fragakis, & Paratheodorou, 2006a]).

Το πρόβλημα αυτό αντιμετωπίζεται από τον αλγόριθμο ταμπλό που δίνεται για την $SHOIQ$ χωρίζοντας τους κόμβους του γραφήματος πλήρωσης σε δύο κατηγορίες:

- *Ονοματικοί κόμβοι* (nominal nodes) ονομάζονται οι κόμβοι που περιέχουν ένα ονοματικό στην ετικέτα τους.
- *Κόμβοι που μπορούν να μπλοκαριστούν* (blockable nodes) ονομάζονται οι κόμβοι που δεν είναι ονοματικοί. Σε αντίθεση με τους ονοματικούς κόμβους, οι blockable κόμβοι απαντώνται μόνο σε δενδρικές δομές που έχουν ως ρίζα έναν ονοματικό κόμβο. Ένα κλαδί ενός τέτοιου δένδρου μπορεί να καταλήγει σε ακμή που οδηγεί σε έναν ονοματικό κόμβο.

Για να μπορεί να τερματίσει, ο αλγόριθμος εισάγει επίσης έναν νέο κανόνα ταμπλό, τον NN -κανόνα, ο οποίος μαντεύει το ακριβές πλήθος των νέων ονοματικών κόμβων που μπορεί να προκύψουν από την αλληλεπίδραση μεταξύ ονοματικών, αντίστροφων ρόλων και περιορισμών αριθμού. Όταν εφαρμοστεί σε μια έννοια της μορφής $\leq nR.C$, ο κανόνας αυτός δημιουργεί, μη ντετερμινιστικά, από 1 έως n νέους ονοματικούς κόμβους, οι οποίοι είναι μεταξύ τους ξένοι.

Ο αλγόριθμος κάνει εκτεταμένη χρήση της τεχνικής του *μπλοκαρίσματος* για τους μη ονοματικούς κόμβους, όπου εμποδίζεται η επέκταση κόμβων (*μπλοκαρισμένοι κόμβοι*) που οδηγούν σε άπειρα μοντέλα, καθώς και της *συγχώνευσης* (merge), όπου, για να ικανοποιηθεί ένας περιορισμός μέγιστης πληθικότητας (\leq), συγχωνεύονται οι κόμβοι που πλεονάζουν. Οι κόμβοι που συγχωνεύονται, καθώς και τα κλαδιά του γραφήματος τα οποία ξεκινούν από αυτούς, λέμε ότι *κλαδεύονται* (prune). Να σημειωθεί ότι ένας ονοματικός κόμβος δεν μπορεί να μπλοκαριστεί, γιατί το ξεδίπλωμά του μπορεί να μην οδηγεί σε δενδρική δομή.

Ο έλεγχος της συνέπειας μιας βάσης γνώσης ως προς ένα TBox και μια ιεραρχία ρόλων \mathcal{R}_h , μπορεί να αναχθεί στον έλεγχο της ικανοποιησιμότητας μιας έννοιας μόνο, ως προς \mathcal{R}_h χρησιμοποιώντας την τεχνική της εσωτερίκευσης (ενότητα 3.4.4). Επίσης, για να ληφθούν υπόψη και οι δηλώσεις που αφορούν στα στιγμιότυπα, χρησιμοποιείται η τεχνική της απαλοιφής του ABox (ενότητα 3.6.3).

Στην περίπτωση της $SROIQ$, ο καθολικός ρόλος, που αποτελεί μέρος της εκφραστικότητας της γλώσσας και που μπορεί να εμφανίζεται στις παραπάνω αναγωγές, απαλείφεται

προσθέτοντας στο \mathcal{R}_h έναν νέο ρόλο U' , ο οποίος είναι συμμετρικός, μεταβατικός, ανακλαστικός και επιπλέον υπάγει όλους τους ρόλους που βρίσκονται στο \mathcal{R}_h .

Οι δηλώσεις που αφορούν τους ρόλους (τα χαρακτηριστικά των ρόλων) αποτελούν το \mathcal{R}_a . Το \mathcal{R}_a μπορεί να θεωρηθεί ότι περιέχει μόνο δηλώσεις της μορφής $\text{Ref}(R)$, $\text{Asy}(R)$ και $\text{Dis}(R,S)$ αφού:

$$\text{Trans}(R) \Leftrightarrow R \circ R \sqsubseteq R$$

$$\text{Sym}(R) \Leftrightarrow R^{\top} \sqsubseteq R$$

$$\text{Irr}(R) \Leftrightarrow \top \sqsubseteq \neg \exists R.\text{Self}$$

Το $R\text{Box } \mathcal{R} = \mathcal{R}_h \cup \mathcal{R}_a$ που προκύπτει με τον τρόπο αυτό ονομάζεται *ανηγμένο* (reduced).

Οι κανόνες επέκτασης ταμπλό για την περίπτωση της $SROIQ$, φαίνονται στον πίνακα 3.4.

Π-κανόνας	αν 1. $(C \sqcap D) \in L(x)$ 2. Ο x δεν είναι έμμεσα μπλοκαρισμένος 3. $\{C, D\} \not\subseteq L(x)$ τότε $L(x) \rightarrow L(x) \cup \{C, D\}$
ΛΙ-κανόνας	αν 1. $(C \sqcup D) \in L(x)$, 2. Ο x δεν είναι έμμεσα μπλοκαρισμένος 3. $\{C, D\} \cap L(x) = \emptyset$ τότε είτε $L(x) \rightarrow L(x) \cup \{C\}$ ή $L(x) \rightarrow L(x) \cup \{D\}$
Ξ-κανόνας	αν 1. $\exists S.C \in L(x)$ 2. Ο x δεν είναι μπλοκαρισμένος 3. Ο x δεν έχει S -γείτονα y με $C \in L(y)$ τότε δημιουργήσε έναν νέο κόμβο y και ακμή $\langle x, y \rangle$ με $L(y) = \{C\}$ και $L(\langle x, y \rangle) = S$
Self-Ref-κανόνας	αν 1. $\exists S.\text{Self} \in L(x)$ ή $\text{Ref}(S) \in \mathcal{R}_a$ 2. Ο x δεν είναι μπλοκαρισμένος 3. $S \notin L(\langle x, x \rangle)$ τότε δημιουργήσε μια ακμή $\langle x, x \rangle$ αν δεν υπάρχει ακόμα και θέσε $L(\langle x, x \rangle) \rightarrow L(\langle x, x \rangle) \cup S$
\forall_1-κανόνας	αν 1. $\forall S.C \in L(x)$ 2. Ο x δεν είναι έμμεσα μπλοκαρισμένος 3. $\forall B_S.C \notin L(x)$ τότε $L(x) \rightarrow L(x) \cup \{\forall B_S.C\}$
\forall_2-κανόνας	αν 1. $\forall B(p).C \in L(x)$ 2. Ο x δεν είναι έμμεσα μπλοκαρισμένος 3. $p \xrightarrow{S} q$ μέσα στο $B(p)$ 4. Υπάρχει S -γείτονας y του x με $\forall B(q).C \notin L(y)$ τότε $L(y) \rightarrow L(y) \cup \{\forall B(q).C\}$
\forall_3-κανόνας	αν 1. $\forall B.C \in L(x)$ 2. Ο x δεν είναι έμμεσα μπλοκαρισμένος 3. $\varepsilon \in L(B)$ 4. $C \notin L(x)$ τότε $L(x) \rightarrow L(x) \cup \{C\}$
ch-κανόνας	αν 1. $(\leq n S.C) \in L(x)$ 2. Ο x δεν είναι έμμεσα μπλοκαρισμένος 3. Ο x έχει S -γείτονα y με $\{C, \neg C\} \cap L(y) = \emptyset$ τότε είτε $L(y) \rightarrow L(y) \cup \{C\}$ ή $L(y) \rightarrow L(y) \cup \{\neg C\}$
\geq-κανόνας	αν 1. $(\geq n S.C) \in L(x)$

		<ol style="list-style-type: none"> 2. Ο x δεν είναι μπλοκαρισμένος 3. Δεν υπάρχουν n ασφαλείς S-γείτονες y_1, \dots, y_n του x με $C \in L(y_i)$ και $y_i \neq y_j$ για $1 \leq i < j \leq n$ <p>τότε δημιουργήσε n νέους κόμβους y_1, \dots, y_n με $L(\langle x, y_i \rangle) = \{S\}$, $L(y_i) = \{C\}$ και $y_i \neq y_j$ για $1 \leq i < j \leq n$</p>
≤-κανόνας	αν	<ol style="list-style-type: none"> 1. $(\leq nR.C) \in L(z)$ 2. Ο z δεν είναι έμμεσα μπλοκαρισμένος 3. $\#S^G(z, C) > n$ 4. Υπάρχουν δύο S-γείτονες x και y του z με $C \in L(x) \cap L(y)$ και όχι $x \neq y$ <p>τότε αν x είναι ονοματικός κόμβος, Merge(y, x) αλλιώς αν y είναι ονοματικός κόμβος ή πρόγονος του x, Merge(x, y) αλλιώς Merge(y, x)</p>
ο-κανόνας	αν	για κάποιο $o \in N$, υπάρχουν δύο κόμβοι x, y με $o \in L(x) \cap L(y)$ και όχι $x \neq y$
	τότε	Merge(x, y)
NN-κανόνας	αν	<ol style="list-style-type: none"> 1. $(\leq nS.C) \in L(x)$ 2. Ο x δεν είναι ονοματικός κόμβος 3. Υπάρχει blockable S-γείτονας y του x τέτοιος ώστε $C \in L(y)$ και x είναι διάδοχος του y 4. Δεν υπάρχει $(\leq lS.C) \in L(x)$ τέτοιο ώστε $l \leq n$ και υπάρχουν l ονοματικοί S-γείτονες z_1, \dots, z_l του x με $C \in L(z_i)$ και $z_i \neq z_j$ για όλα τα $1 \leq i < j \leq l$ <p>τότε</p> <ol style="list-style-type: none"> 1. μάντεψε m με $1 \leq m \leq n$ και θέσε $L(x) \rightarrow L(x) \cup \{(\leq mS.C)\}$ 2. δημιουργήσε m νέους κόμβους y_1, \dots, y_m με $L(\langle x, y_i \rangle) = \{S\}$, $L(y_i) = \{C, o_i\}$, με $o_i \in N$ να είναι νέο στο \mathbf{G} και $y_i \neq y_j$ για $1 \leq i < j \leq m$
≤_o-κανόνας	αν	<ol style="list-style-type: none"> 1. $(\leq mS.C) \in L(x)$, ο x είναι ονοματικός κόμβος και υπάρχει blockable S-γείτονας y του x με $C \in L(y)$ 2. Υπάρχουν m ονοματικοί S-γείτονες z_1, \dots, z_m του x με $C \in L(z_i)$ και $z_i \neq z_j$ για κάθε $1 \leq i < j \leq m$ 3. Υπάρχει ονοματικός S-γείτονας z του x με $C \in L(z)$ και όχι $y \neq z$ <p>τότε Merge(y, z)</p>

Πίνακας 3.4: Κανόνες επέκτασης ταμπλό για τη SROIQ.

Ισχύουν οι παρακάτω ορισμοί:

Ένας κόμβος y καλείται R -γείτονας ενός κόμβου x αν ο y είναι R -διάδοχος του x ή αν ο x είναι $\text{Inn}(R)$ διάδοχος του y .

Ένας κόμβος x καλείται έμμεσα μπλοκαρισμένος, αν είναι blockable και έχει μπλοκαρισμένο πρόγονο.

Ένας R -γείτονας y ενός κόμβου x καλείται ασφαλής (safe), αν ο x είναι blockable ή ο x είναι ονοματικός κόμβος και ο y δεν είναι μπλοκαρισμένος.

Το \mathbf{G} συμβολίζει το γράφημα πλήρωσης, το $\#M$ την πληθικότητα ενός συνόλου M και το N το σύνολο των ονοματικών της οντολογίας.

$S^G(x, C) = \{y \mid y \text{ είναι } S\text{-γείτονας του } x \text{ και } C \in L(y)\}$.

Στην περίπτωση της *SROIQ*, για μια ιεραρχία ρόλων που είναι κανονική (regular, βλ. ενότητα 3.7.3) και έναν (πιθανώς αντίστροφο) ρόλο S , κατασκευάζεται ένα μη τετερμινιστικό πεπερασμένο αυτόματο (NFA) \mathcal{B}_S . Τα αυτόματα αυτά κατασκευάζονται έτσι ώστε να ισχύει η ακόλουθη πρόταση:

Η διερμήνευση I είναι ένα μοντέλο του \mathcal{R}_h αν και μόνο αν για κάθε (πιθανώς αντίστροφο) ρόλο S που εμφανίζεται στο \mathcal{R}_h , για κάθε λέξη $w \in L(\mathcal{B}_S)$ και για κάθε $\langle x, y \rangle \in w'$ έχουμε $\langle x, y \rangle \in S'$.

Το $L(\mathcal{B}_S)$ συμβολίζει τη γλώσσα που δέχεται το αυτόματο, και η λέξη w συμβολίζει μια αλυσίδα ρόλων, όπου κάθε της γράμμα είναι κι ένα όνομα ρόλου. Δηλαδή το \mathcal{B}_S κατασκευάζεται έτσι ώστε, για κάθε λέξη (=αλυσίδα ρόλων) που δέχεται το αυτόματο, να ισχύει $w \sqsubseteq S$.

Οι καθολικοί περιορισμοί (\forall) πάνω σε NFAs που χρησιμοποιούνται στον αλγόριθμο έχουν το νόημα της *ενδιάμεσης μνήμης*, για την απομνημόνευση όλων των πιθανών μονοπατιών (αλυσίδων ρόλων) τα οποία μπορούν τελικά να οδηγήσουν στην εφαρμογή του περιορισμού $\forall R.C$ (δηλαδή στην προσθήκη της C στην ετικέτα κάποιου απογόνου).

Αν $o_1 \dots o_l$ είναι τα ονοματικά που εμφανίζονται στην έννοια D , της οποίας η ικανοποιησιμότητα πρόκειται να ελεγχθεί, τότε ο αλγόριθμος ξεκινάει με $l + 1$ κόμβους (ρίζες) $r_0 \dots r_l$ και θέτει $L(r_0) = \{D\}$ και $L(r_i) = \{o_i\}$ για κάθε $1 \leq i \leq l$. Ο αλγόριθμος τερματίζει αν βρεθεί ένα πλήρως εκτεταμένο γράφημα πλήρωσης χωρίς συγκρούσεις ή αν όλα τα μη τετερμινιστικά μονοπάτια οδηγήσουν σε σύγκρουση. Μια σύγκρουση εντοπίζεται όταν ισχύει ένα από τα παρακάτω:

- $\perp \in L(x)$.
- Για κάποιο όνομα έννοιας $A \{A, \neg A\} \subseteq L(x)$.
- Το x είναι S -γείτονας του x και $\neg \exists S.\text{Self} \in L(x)$.
- Υπάρχει $\text{Dis}(R, S) \in \mathcal{R}_\sigma$ και y είναι ταυτόχρονα R - και S -γείτονας του x .
- Υπάρχει $\text{Asy}(R) \in \mathcal{R}_\sigma$ και y είναι R -γείτονας του x και x είναι R -γείτονας του y .
- Υπάρχει κάποια έννοια $(\leq n R.C) \in L(x)$ και $\{y_0, \dots, y_n\} \subseteq S^G(x, C)$ με $y_i \neq y_j$ για κάθε $0 \leq i < j \leq n$.
- Για κάποιο $o \in N_l$, $x \neq y$ και $o \in L(x) \cap L(y)$.

Να σημειωθεί ότι οι δύο τελευταίοι κανόνες δεν υπάρχουν στην μορφή αυτή ούτε στην τεχνική αναφορά [Horrocks, Kutz, & Sattler, 2005] ούτε στην μεταγενέστερη ανακοίνωση [Horrocks, Kutz, & Sattler, 2006] (μάλιστα ο \leq_σ -κανόνας δεν αναφέρεται καν). Αναφέρονται όμως έτσι στο άρθρο [Horrocks & Sattler, 2007] που αφορά στη *SROIQ*, ενώ, ύστερα από επικοινωνία με τους συγγραφείς εξακριβώθηκε ότι είναι αναγκαίοι για τον τερματισμό και διορθώνουν μια παράλειψη του αρχικού *SROIQ* αλγορίθμου που ανακοινώθηκε στο [Horrocks & Sattler, 2005]. Αυτή η παράλειψη φαίνεται ότι έχει διαδοθεί και στους αλγορίθμους για τη *SROIQ* και διορθώνεται εδώ.

3.7.3. Υποθέσεις και πολυπλοκότητα

Μια βασική υπόθεση που δέχεται ο αλγόριθμος είναι οι ρόλοι που εμφανίζονται στους περιορισμούς αριθμού, καθώς και αυτοί που εμφανίζονται σε δηλώσεις της μορφής $\text{Irr}(R)$,

$Asy(R)$ και $Dis(R)$ να είναι απλοί. Στην περίπτωση της $SHOIQ$ απλοί είναι οι ρόλοι που δεν είναι μεταβατικοί, ούτε έχουν μεταβατικούς υπορόλους. Αυτό μεταξύ άλλων σημαίνει ότι δεν επιτρέπεται να εκφραστεί ένας ρόλος που είναι ταυτόχρονα συναρτησιακός και μεταβατικός, αφού η συναρτησιακότητα είναι ειδική περίπτωση περιορισμού αριθμού. Στην περίπτωση της $SROIQ$ ο ορισμός του απλού ρόλου γενικεύεται για να αποκλείσει και τους ρόλους εκείνους που υπάγουν αλυσίδες ρόλων.

Στην περίπτωση της $SHOIQ$ επιτρέπονται μόνο απλοί ρόλοι, γιατί ο συνδυασμός τους με περιορισμούς αριθμού οδηγεί σε μη αποφασισιμότητα. Πρόσφατα όμως έχειδειχθεί ότι υπάρχουν περιπτώσεις στις οποίες ο περιορισμός αυτός είναι δυνατόν και να μην ισχύει [Kazakov, Sattler, & Zolin, 2007]. Στην περίπτωση της $SROIQ$ δεν είναι ξεκάθαρο αν η απαγόρευση μη απλών ρόλων σε δηλώσεις ρόλων είναι απαραίτητη για την αποφασισιμότητα, ωστόσο διευκολύνει την επέκταση των προηγούμενων αλγορίθμων και τον συνδυασμό τους.

Ένας δεύτερος περιορισμός, ειδικά για τα RIAs όπως γενικεύονται στη $SROIQ$, είναι η ιεραρχία ρόλων να είναι κανονική (regular). Αυτό σημαίνει ότι οι ρόλοι που απαρτίζουν μια αλυσίδα η οποία υπάγεται σε ένα όνομα ρόλου R πρέπει να βρίσκονται σε μια αυστηρή μερική διάταξη ($<$) ως προς τον ρόλο R . Η $<$ είναι εξ ορισμού μεταβατική και μη ανακλαστική. Συνολικά, οι διάταξη των ρόλων σε όλα τα RIAs δεν πρέπει να παραβιάζει αυτές τις ιδιότητες.

Όσον αφορά την πολυπλοκότητα, ας σημειωθεί πρώτα ότι όλες οι αναγωγές και απλοποιήσεις που αναφέρθηκαν προηγουμένως (εσωτερίκευση, απαλοιφή,...) μπορούν να γίνουν πολυωνυμικά, άρα δεν επιβαρύνουν την πολυπλοκότητα. Στη συνέχεια, αν D η έννοια της οποίας την ικανοποιησιμότητα ελέγχει ο αλγόριθμος, το μέγεθος της ετικέτας κάθε κόμβου στο \mathbf{G} εξαρτάται από το μέγεθος του συνόλου των «σχετικών υπο-εννοιών» της D , $cl(D, \mathcal{R})$ (η «κλειστότητα» της D). Το μέγεθος του συνόλου αυτού είναι, για την περίπτωση της $SHOIQ$, πολυωνυμικό ως προς την είσοδο.

Έστω $m = |cl(D)|$, k ο αριθμός των ρόλων και των αντιστρόφων τους σε όλη την οντολογία, n ο μεγαλύτερος αριθμός που εμφανίζεται μέσα σε έναν περιορισμό μέγιστης πληθικότητας και l το πλήθος των ονοματικών που εμφανίζονται στην D . Έστω επίσης ότι $\lambda = 2^{2^{m+k}}$. Το πλήθος των κόμβων του γραφήματος πλήρωσης που θα δημιουργήσει ο αλγόριθμος φράσσεται από $O((mn)^\lambda)$ είναι δηλαδή, ασυμπτωτικά, διπλά εκθετικό ως προς το μέγεθος της εισόδου.

Στην περίπτωση της $SROIQ$, η επιβάρυνση για την αυξημένη εκφραστικότητα σε σχέση με την $SHOIQ$ προέρχεται από την κατασκευή των μη ντετερμινιστικών αυτομάτων, γιατί το μέγεθος της $cl(D)$, εξαρτάται επίσης και από το μέγεθος των αυτομάτων. Κάθε ένα από τα αυτόματα αυτά έχει μέγεθος $O(2^{d_x})$, όπου d_x το βάθος της ιεραρχίας ρόλων. Επίσης, η μετατροπή των αυτομάτων σε ντετερμινιστικά γίνεται με αντίστοιχη εκθετική αύξηση, δηλαδή μέγεθος $O(2^{2^{d_x}})$. Έτσι, ο αλγόριθμος της $SROIQ$ ξεκινά να αποφασίσει την ικανοποιησιμότητα μιας έννοιας D της οποίας η κλειστότητα έχει μέγεθος τουλάχιστον εκθετικό ως προς το βάθος της ιεραρχίας. Παρόλο που στο [Horrocks, Kutz, & Sattler, 2006] επικρατεί ο ισχυρισμός ότι η κατασκευή των αυτομάτων μπορεί να θεωρηθεί βήμα προεπεξεργασίας και άρα το μέγεθος της κλειστότητας παραμένει πολυωνυμικό ως προς την είσοδο (D), πρέπει να ληφθούν υπόψη οι ακόλουθες παρατηρήσεις:

- Η εργασία [Horrocks, Kutz, & Sattler, 2006] παρουσιάζει ένα ταμπλό για τον έλεγχο της ικανοποιησιμότητας μιας έννοιας μόνο (C_0). Αν ζητηθεί η ικανοποιησιμότητα κάποιας άλλης έννοιας, η κλειστότητα πρέπει να επαναυπολογιστεί για την συγκεκριμένη έννοια και τα αυτόματα να κατασκευαστούν εκ νέου.
- Αν ο αλγόριθμος αφορούσε στην ικανοποιησιμότητα βάσης γνώσης, τότε θα έπρεπε να υπολογιστεί η κλειστότητα όλου του TBox, όπως γίνεται στο [Horrocks & Sattler, 2007].
- Εκτός από την κατασκευή των αυτομάτων, ο αλγόριθμος συμπεριφέρεται όπως αυτός για τη *SHOIQ*, δηλαδή με πολυπλοκότητα 2- NExp.

Ένα ενδιαφέρον χαρακτηριστικό της διαδικασίας απόφασης για *SHOIQ* και για *SROIQ*, αλλά και γενικά των βασισμένων σε ταμπλό προσεγγίσεων, αποτελεί το γεγονός ότι η συμπεριφορά τους διαβαθμίζεται ως προς την εκφραστικότητα που πραγματικά αξιοποιείται μέσα σε μια οντολογία (“pay as you go”): Αν οι περιορισμοί αριθμού είναι συναρτησιακοί (δηλαδή το n είναι το πολύ 1) ο *NN*-κανόνας γίνεται ντετερμινιστικός· αν δεν υπάρχουν αλληλεπιδράσεις μεταξύ ονοματικών, αντίστροφων ρόλων και περιορισμών αριθμού ο *NN*-κανόνας δεν εφαρμόζεται ποτέ κ.ο.κ.

4. ΜΕΘΟΔΟΛΟΓΙΑ ΑΝΑΚΑΛΥΨΗΣ ΓΝΩΣΗΣ ΣΤΟ ΣΗΜΑΝΤΙΚΟ ΙΣΤΟ

4.1. Εισαγωγή

Διάφορες λύσεις έχουν κατά καιρούς προταθεί εστιάζοντας στη θεμελίωση ή βελτίωση των υπηρεσιών συλλογισμού. Γλώσσες περιγραφής οντολογιών, συστήματα και μηχανισμοί συμπερασμού, καθώς και υλοποιήσεις με δυνατότητες συλλογισμού έχουν αναπτυχθεί, προκειμένου να βελτιώσουν τις δυνατότητες ανάκτησης και να καταστήσουν δυνατή την ανακάλυψη γνώσης, όπου είναι εφικτό. Η μελέτη αυτών των προσεγγίσεων αποκαλύπτει ότι δεν υπάρχει κάποια συνεκτική και τυποποιημένη μεθοδολογία που να ακολουθείται στη γενική περίπτωση. Επιπρόσθετα, η εκφραστικότητα που επιτυγχάνεται ποικίλει, χωρίς πάντα να ικανοποιούνται οι εκφραστικές ανάγκες του Σημαντικού Ιστού. Τέλος, οι τρέχουσες προσεγγίσεις για τη σύνθεση και την διεξαγωγή ευφυών ερωτημάτων δεν φαίνεται να είναι αρκετά δηλωτικές: Ο χρήστης είναι συνήθως αυτός που επιβαρύνεται με το έργο της συλλογής των κατάλληλων πληροφοριών που είναι απαραίτητες για την κατασκευή του ερωτήματος. Διαφαίνεται λοιπόν ένα σοβαρό έλλειμμα στον τρόπο με τον οποίο ο τελικός χρήστης ή αυτοματοποιημένοι πράκτορες μπορούν να αλληλεπιδρούν με τις οντολογίες και να τις αξιοποιούν συνθέτοντας ερωτήματα και λαμβάνοντας απαντήσεις σε ορισμένη μορφή.

Ο βασικός στόχος του κεφαλαίου αυτού είναι η πρόταση, η δοκιμή και η τεκμηρίωση μιας μεθοδολογίας για τη διεξαγωγή SWKD. Η μεθοδολογία αυτή απαντά στο ερώτημα:

«Ποιες σχεδιαστικές αποφάσεις πρέπει να ληφθούν προκειμένου για (πριν από) το σχεδιασμό και ανάπτυξη μιας υπηρεσίας ανακάλυψης γνώσης για το Σημαντικό Ιστό;»

Μπορεί να διαιρεθεί σε τρία στάδια:

1. την επιλογή του κατάλληλου φορμαλισμού που θα επιτρέψει τη χρήση εργαλείων τεχνητής νοημοσύνης για διεξαγωγή συλλογισμού σε έγγραφα του Σημαντικού Ιστού,
2. την ανάδειξη βασικών κριτηρίων που πρέπει να ληφθούν υπόψη, ώστε ο συλλογισμός να είναι κατάλληλος για το περιβάλλον του Ιστού και
3. την επιλογή ενός συγκεκριμένου μηχανισμού συμπερασμού που να ανταποκρίνεται στα κριτήρια που τίθενται στο προηγούμενο βήμα.

Η μεθοδολογία αυτή έχει καταρτιστεί έχοντας ως απώτερο στόχο να ελέγξει την ορθότητα της παρακάτω ερευνητικής υπόθεσης:

Υπόθεση 1: *Οι τεχνολογίες και η λογική θεωρία του Σημαντικού Ιστού μπορούν να συνεισφέρουν αποδοτικά και εκφραστικά στο πρόβλημα της SWKD.*

Δείχνουμε ότι ο βαθμός ικανοποίησης της παραπάνω υπόθεσης εξαρτάται από συγκεκριμένα θεωρητικά, οργανωτικά και τεχνικά ζητήματα. Τα ζητήματα αυτά, που αναλύονται στην ενότητα 4.6, αποτελούν στιγμιότυπο της τρέχουσας κατάστασης στο χώρο και άρα μπορούν να αλλάζουν· ωστόσο, η μεθοδολογία είναι έτσι σχεδιασμένη, ώστε τα στάδιά της να μην εξαρτώνται από τις τρέχουσες συνθήκες. Αυτό σημαίνει ότι τα κρίσιμα σχεδιαστικά ερωτήματα που πρέπει πρώτα απ' όλα να απαντηθούν δεν αλλοιώνονται, όμως οι απαντήσεις τους μπορούν εύκολα να προσαρμοστούν στην εκάστοτε κατάσταση.

Για παράδειγμα, αν η έρευνα της πολυπλοκότητας απόφασης σε κάποιον από τους θεωρούμενους εναλλακτικούς φορμαλισμούς οδηγήσει σε αποδοτικότερους αλγορίθμους, δεν υπάρχει πια ανάγκη για την υιοθέτηση των Λογικών Περιγραφής ως βάσης. Αν πάλι αναπτυχθεί μια μηχανή συμπερασμού ταχύτερη και με υψηλότερη εκφραστικότητα από αυτές που εξετάζονται, η χρήση της μεθοδολογίας θα οδηγήσει στην αξιοποίησή της.

Το παραπάνω παράδειγμα είναι φυσικά υποθετικό και υπεραπλουστευμένο, δεδομένου ότι υπάρχει, όπως θα φανεί, συγκεκριμένη οριοθέτηση των βελτιώσεων που μπορούν να επιτευχθούν, αλλά είναι ενδεικτικό του *προσαρμοστικού* χαρακτήρα της μεθοδολογίας.

Η πειραματική επαλήθευση της Μεθοδολογίας Ανακάλυψης Γνώσης γίνεται μέσω της υλοποίησης και ανάπτυξης μιας πρότυπης Διεπαφής Ανακάλυψης Γνώσης (Knowledge Discovery Interface, KDI). Η KDI είναι μια διαδικτυακή υπηρεσία, που επιτρέπει την υποβολή ευφύων ερωτημάτων σε οντολογικά έγγραφα και έχει αναπτυχθεί ως υλοποίηση των αποφάσεων και κριτηρίων που τίθενται σε κάθε στάδιο της μεθοδολογίας. Στην περίπτωση της, έχει υιοθετηθεί ένας διαισθητικός τρόπος αλληλεπίδρασης που, διαφανώς προς το χρήστη, σχηματίζει διαισθητικά το ερώτημά του και το αναλύει σε λειτουργίες χαμηλού επιπέδου του υποκείμενου μηχανισμού συμπερασμών.

Αρχικά, στην επόμενη ενότητα γίνεται μια επισκόπηση της βιβλιογραφίας σχετικά με τις προσεγγίσεις συλλογισμού και ανακάλυψης γνώσης στον Παγκόσμιο Ιστό. Αφού προηγηθεί μια συζήτηση για τον αναγκαίο για το περιβάλλον του Ιστού τρόπο διεξαγωγής των συμπερασμών (στάδιο 2), στην οποία εξετάζονται και τα συζευγμένα ερωτήματα, στη συνέχεια παρουσιάζονται και αξιολογούνται διάφοροι μηχανισμοί συμπερασμού βασισμένοι κυρίως σε Λογικές Περιγραφής (στάδιο 3). Κατόπιν συζητούνται και συγκρίνονται ορισμένα συστήματα διαχείρισης οντολογιών που προσφέρουν, μεταξύ άλλων, επιθυμητά χαρακτηριστικά για ένα περιβάλλον παραγωγής, όπως αυτό της *διαχρονικότητας* (persistence). Στη συνέχεια αναλύονται οι περιορισμοί που ελέγχουν την αποδοτικότητα και εκφραστικότητα της μεθοδολογίας. Το κεφάλαιο ολοκληρώνεται με την παρουσίαση της πρότυπης υλοποίησης KDI. Θεωρούμε ότι το πρώτο στάδιο της μεθοδολογίας έχει αναλυθεί στο προηγούμενο κεφάλαιο και δεν θα μας απασχολήσει εκτενώς εδώ.

Το κεφάλαιο αυτό βασίζεται και επεκτείνει τις ανακοινώσεις [Koutsomitropoulos, Fragakis, & Papatheodorou, 2005; Koutsomitropoulos, Meidanis, Kandili, & Papatheodorou, 2006; Koutsomitropoulos, Fragakis, & Papatheodorou, 2006b], καθώς και τα κεφάλαια [Koutsomitropoulos, Fragakis, & Papatheodorou, 2006a; Koutsomitropoulos, Meidanis, Kandili, & Papatheodorou, 2008].

4.2. Προηγούμενη Εργασία

Αν και η ιδέα του Σημαντικού Ιστού μόνο πρόσφατα έχει συστηματοποιηθεί, η ανάγκη για εξαγωγή συμπερασμάτων και εν γένει νοήμονα συμπεριφορά στο Διαδίκτυο υπάρχει σχεδόν από την καταβολή του Παγκόσμιου Ιστού. Στις πρώτες προσπάθειες που επιχειρούν να αξιοποιήσουν σημασιολογικές τεχνικές για ανακάλυψη γνώσης στο Διαδίκτυο θα πρέπει να ενταχθεί η ιδέα για εμπλουτισμό των HTML εγγράφων με οντολογική πληροφορία κατανοήσιμη και από τη μηχανή [Luke, Spector, & Rager, 1996]. Η ιδέα αυτή άλλωστε αποτέλεσε το έναυσμα και οδήγησε στην εξέλιξη της γλώσσας SHOE [Heflin, Hendler, & Luke, 1999] και στην ανάπτυξη σημασιολογικών εργαλείων αναζήτησης βασισμένων σε αυτή [Hendler & Heflin, 2000].

Έχοντας επίγνωση των περιορισμών της ανακάλυψης γνώσης σε ένα αυθαίρετο περιβάλλον, όπως ο Παγκόσμιος Ιστός, και κατανοώντας τα θετικά αποτελέσματα που έχει δώσει η ανάκτηση πληροφορίας στο πρόβλημα της διαδικτυακής αναζήτησης, πιο πρόσφατες έρευνες δίνουν βάρος στο συνδυασμό των δύο αυτών προσεγγίσεων. Χαρακτηριστικό παράδειγμα αποτελεί το σύστημα OWLIR [Shah, Finin, Joshi, Cost, & Mayfield, 2002], το οποίο αποσκοπεί στη βελτίωση της επίδοσης της ανάκτησης δικτυακών εγγράφων, εμπλουτίζοντας τα έγγραφα με σημασιολογική σήμανση κατά DAML+OIL και αξιοποιώντας ένα σύστημα συμπερασμού βασισμένο σε κανόνες. Ο εμπλουτισμός των εγγράφων επιτυγχάνεται χρησιμοποιώντας μια οντολογία ανωτέρου επιπέδου ως βάση και κατόπιν ένα σύστημα επεξεργασίας και εξαγωγής κειμένου, που εκμεταλλεύεται κυρίως τη δομή των εγγράφων. Η αναζήτηση διεκπεραιώνεται με τη χρήση κλασικών μεθόδων ανάκτησης και στη συνέχεια το σύνολο των αποτελεσμάτων εκλεπτύνεται, λαμβάνοντας υπόψη τα αποτελέσματα του συστήματος συμπερασμού.

Βελτίωση των αποτελεσμάτων της αναζήτησης, η οποία θα λαμβάνει υπόψη τις σημασιολογικές συσχετίσεις των εγγράφων και των οντοτήτων που περιγράφονται σε αυτά, επιδιώκει και το σύστημα TAP [Guha & McCool, 2003]. Στην περίπτωση του TAP ωστόσο, δεν διεξάγεται συμπερασμός, αλλά τα RDF ή OWL έγγραφα, τα οποία συνιστούν τη γνωσιακή βάση, αντιμετωπίζονται ως δομημένα σύνολα μεταδεδομένων. Τα σύνολα αυτά ισοδυναμούν με διευθυνόμενα γραφήματα, με τις ακμές να αντιστοιχούν σε σχέσεις, ενώ οι κορυφές, με τη βοήθεια μιας βάσης γνώσης, ανάγονται σε πραγματικούς δικτυακούς πόρους. Ακόμη, στο TAP δεν γίνεται εμπλουτισμός των δικτυακών εγγράφων με σήμανση όπως στο OWLIR, αλλά η οντολογική γνώση προέρχεται είτε από μια τοπική βάση γνώσης, είτε, μελλοντικά, από τις διασκορπισμένες στον Ιστό οντολογίες.

Η ανάπτυξη και συντήρηση μιας βάσης γνώσης, ιδιαίτερα όταν αυτή πρέπει να είναι αρκετά λεπτομερής, ώστε να παρέχει χρήσιμα αποτελέσματα, είναι μια επίπονη διαδικασία, ένα μεγάλο μέρος της οποίας είναι αναγκαίο να γίνεται χειρωνακτικά. Είτε η βάση γνώσης συντηρείται τοπικά είτε η γνωσιακή πληροφορία ενσωματώνεται μέσα στα έγγραφα, το πρόβλημα της απόκτησης και διατήρησης χρήσιμης γνώσης αποτελεί μια από τις σημαντικότερες προκλήσεις που έχει να αντιμετωπίσει μια υπηρεσία του Σημαντικού Ιστού.

Στην περίπτωση του συστήματος Artequakt [Alani, et al., 2003] το πρόβλημα αυτό αντιμετωπίζεται μέσα από μια προσέγγιση αυτοματοποιημένης εξαγωγής γνώσης από δικτυακούς πόρους. Χρησιμοποιώντας τεχνικές επεξεργασίας φυσικής γλώσσας, οι οποίες εφαρμόζονται σε δικτυακά έγγραφα, επιτυγχάνεται η πληθύσωση μιας οντολογίας που

βασίζεται στο CIDOC-CRM, μιας και το σύστημα εξειδικεύει στο πεδίο των καλλιτεχνών και των καλλιτεχνικών δημιουργημάτων. Η οντολογία αυτή έχει το ρόλο του «εννοιολογικού σχήματος» για τις πληροφορίες που θα πρέπει να εξαχθούν από τα έγγραφα και να επαυξηθούν τη βάση γνώσης. Πρέπει εντούτοις να παρατηρηθεί ότι το Artequakt δεν χρησιμοποιεί τεχνικές συμπερασμού και άρα τεχνικές ανακάλυψης γνώσης, ενώ η οντολογική πληροφορία που διατηρεί είναι οργανωμένη σε XML.

Ως εφαρμογή επίδειξης των υπηρεσιών ανακάλυψης γνώσης που θα καταστήσει δυνατές ο Σημαντικός Ιστός, αναπτύχθηκε το σύστημα Wine Agent [Hsu & McGuinness, 2003]. Το σύστημα αυτό χρησιμοποιεί μια συγκεκριμένη οντολογία πεδίου (Wine Ontology) κωδικοποιημένη σε DAML+OIL/OWL και επιδεικνύει την διεξαγωγή συμπερασμών πάνω σε αυτή. Για τη διεξαγωγή των συμπερασμών, το Wine Agent χρησιμοποιεί ένα πρόγραμμα απόδειξης θεωρημάτων, δηλαδή ένα σύστημα συλλογισμού βασισμένο σε ΛΠΤ (JTP).

4.3. Καταλληλότητα για τον Ιστό

Η ανακάλυψη γνώσης, αν και δεν μπορεί να ταυτιστεί με την παραδοσιακή διαδικασία ανάκτησης δεδομένων όπως αυτή τυποποιείται από τις γλώσσες και τα πρωτόκολλα ερωτημάτων, π.χ. σε μια βάση δεδομένων, ωστόσο προϋποθέτει το σχηματισμό και την υποβολή κάποιου τύπου ευφύων ερωτημάτων, η απάντηση των οποίων θα συνθέτει την αναζητούμενη γνώση. Για παράδειγμα, ένα τυπικό ερώτημα σε SQL ή XQuery δεν είναι κατάλληλο για την άντληση απαντήσεων από έναν μηχανισμό συμπερασμού. Ούτε όμως και οι πλέον πρόσφατες προτάσεις για τις γλώσσες ερωτημάτων στο Σημαντικό Ιστό, όπως η SPARQL, δεν φαίνεται να επαρκούν για τη μοντελοποίηση τέτοιων δηλωτικών ερωτημάτων, αφού δεν μπορούν να συλλάβουν την πλήρη εκφραστικότητα της OWL.

Στην ενότητα αυτή εξετάζονται ορισμένες προϋποθέσεις που πρέπει να πληροί μια μέθοδος παραγωγικής απόκτησης πληροφορίας στο Σημαντικό Ιστό, ώστε η ανακάλυψη γνώσης να είναι προσαρμοσμένη στις ιδιαιτερότητες ενός διαδικτυακού περιβάλλοντος. Αρχικά επιχειρηματολογούμε για την αναγκαιότητα υποστήριξης συλλογισμού και στο ABox. Κατόπιν, παρουσιάζονται τα λεγόμενα *συζευγμένα ερωτήματα*, ως πιο ισχυρά εκφραστικώς και παρατίθενται σχετικά αποτελέσματα πολυπλοκότητας. Τέλος, εξετάζεται η γλώσσα SPARQL-DL, στην οποία αναγνωρίζουμε θετικά στοιχεία ως βάση για την τυποποίηση μιας γλώσσας επερωτήσεων κατάλληλης για έγγραφα OWL και OWL1.1.

4.3.1. Η μέθοδος των συμπερασμών

Η μελέτη της βιβλιογραφίας σχετικά με τη χρήση οντολογιών και Λογικών Περιγραφής για σημασιολογικό ταίριασμα των περιγραφών των Υπηρεσιών Ιστού [González-Castillo, Trastour, & Bartolini, 2001; Paolucci, Kawamura, Payne, & Sycara, 2002], καθώς και άλλων πηγών [Network Inference, 2003], οδηγεί στην αναγνώριση μιας συνήθους μεθόδου ερωτημάτων, η οποία εκμεταλλεύεται τις σχέσεις υπαγωγής μεταξύ των εννοιών. Σύμφωνα με τη μέθοδο αυτή, το ερώτημα μοντελοποιείται ως μία νέα έννοια, χρησιμοποιώντας τους γνωστούς κατασκευαστές των Λογικών Περιγραφής και κατόπιν ταξινομείται στην ιεραρχία. Οι έννοιες που υπάγονται στη νέα έννοια-ερώτημα, καθώς και αυτές που τυχόν ταυτίζονται με αυτή, αποτελούν απόλυτες απαντήσεις στο ερώτημα. Οι έννοιες που υπάγουν την

έννοια-ερώτημα μπορούν επίσης να θεωρηθούν σχετικές απαντήσεις, οι οποίες δεν πληρούν βέβαια τα κριτήρια του ερωτήματος, αλλά η απόστασή τους από το ερώτημα μέσα στην ιεραρχία καθορίζει και το βαθμό στον οποίο τα ικανοποιούν. Περαιτέρω, κάθε στιγμιότυπο μπορεί να μοντελοποιηθεί ως ατομική έννοια, οπότε ερωτήματα που απαιτούν τη χρήση στιγμιότυπων μπορούν να διεξαχθούν με τον ίδιο τρόπο [Horrocks & Tessaris, 2002]. Όπως φαίνεται, η παραπάνω μέθοδος, που μπορεί να χαρακτηριστεί ως *ταξινομική*, βασίζεται αποκλειστικά στις λειτουργίες συμπερασμού που παρέχονται μόνο για το TBox μιας βάσης γνώσης.

Αν για το ταίριασμα δομημένων περιγραφών των Υπηρεσιών Ιστού αρκεί η δημιουργία τεχνητών κλάσεων και η ταξινόμηση τους στην ιεραρχία, η ανακάλυψη πόρων στο Σημαντικό Ιστό απαιτεί την αξιοποίηση της επιπλέον εκφραστικότητας που παρέχουν τα στιγμιότυπα. Οι ευφυείς λειτουργίες που περιλαμβάνει το ABox, όπως αυτή του ελέγχου στιγμιότυπων, είναι κρίσιμης σημασίας όταν η μοντελοποίηση του πεδίου χρειάζεται τη λεπτομερέστερη ανάλυση που τα στιγμιότυπα καθιστούν δυνατή. Κάτι τέτοιο είναι ιδιαίτερα αληθινό σε ένα περιβάλλον όπως ο Παγκόσμιος Ιστός, το οποίο περιλαμβάνει σημασιακές δομές απροσδιόριστης και αυθαίρετης λεπτομέρειας. Σε κάθε περίπτωση, η εισαγωγή στιγμιότυπων σε μια οντολογία ενεργοποιεί συμπερασμούς και εκφράσεις που θα ήταν αδύνατο να επιτευχθούν με την αποκλειστική χρήση της ταξινομίας των εννοιών.

Έστω για παράδειγμα η περίπτωση, όπου επιθυμούμε να εκφράσουμε το γεγονός ότι ένα σπαθί είναι κατασκευασμένο από σίδηρο. Η περιγραφή OWL για αυτό θα είναι:

```
<owl:Class rdf:ID="σπαθί">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#είναιΦτιαγμένοΑπό"/>
      <owl:someValuesFrom rdf:resource="#σίδηρος"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

Η έκφραση που περιγράφει ότι ο σίδηρος είναι μέταλλο θα είναι:

```
<owl:Class rdf:ID="σίδηρος">
  <rdfs:subClassOf rdf:resource="#Μέταλλο" />
</owl:Class>
```

Μπορούμε έτσι να ανακτήσουμε όλα τα μεταλλικά αντικείμενα:

```
<owl:Restriction>
  <owl:onProperty rdf:resource="#είναιΦτιαγμένοΑπό"/>
  <owl:someValuesFrom rdf:resource="#Μέταλλο"/>
</owl:Restriction>
```

Τέτοιου τύπου ερωτήματα μπορούν εύκολα να απαντηθούν με τον παραπάνω τρόπο, χρησιμοποιώντας μόνο το TBox της οντολογίας. Τι γίνεται όμως στην περίπτωση που επιθυμούμε να μάθουμε από τι υλικό είναι φτιαγμένο το σπαθί; Αν χρησιμοποιηθεί συλλογισμός και στο ABox, η απάντηση είναι άμεση, χρησιμοποιώντας ένα ερώτημα πληρωτή ρόλου.

Είναι λοιπόν σκόπιμο να εφαρμόζεται μια *βασισμένη σε στιγμιότυπα* μέθοδος για τη διεξαγωγή των συμπερασμών, η οποία θα πρέπει τουλάχιστον να εντάσσει στον πυρήνα της τις δύο βασικές λειτουργίες του ABox: τον *έλεγχο στιγμιότυπου*, που βρίσκει τις έννοιες στις οποίες ανήκει ένα στιγμιότυπο και την *ανάκτηση πληρωτών ρόλου*, όπου, δοσμένου ενός στιγμιότυπου και ενός ρόλου, συμπεραίνονται τα σχετιζόμενα στιγμιότυπα μέσω του ρόλου αυτού. Μέσω των δύο αυτών λειτουργιών, αλλά και με την υποβοήθηση που παρέχουν οι συμπερασμοί στο TBox, επιτυγχάνεται τόσο η ανάκτηση δεδομένων που ρητά έχουν δηλωθεί στην πληροφορία μιας οντολογίας, όσο και η ανακάλυψη γνώσης που προκύπτει λογικά από την πληροφορία αυτή. Επομένως, η υποστήριξη για συλλογισμό τόσο στο TBox, όσο και στο ABox αποτελεί ένα σημαντικό κριτήριο για την επιλογή ενός κατάλληλου μηχανισμού συλλογισμού. Η βασισμένη σε στιγμιότυπα μέθοδος υλοποιείται στην KDI, που παρουσιάζεται στην ενότητα 4.7.

4.3.2. Συζευγμένα ερωτήματα

Ακόμα και η ανάκτηση στιγμιότυπων του ABox μπορεί να θεωρηθεί ως περιορισμένη μορφή επερώτησης: παρόλο που είναι δυνατό να χρησιμοποιηθούν σύνθετες εκφράσεις εννοιών ως ερωτήματα, δεν μπορούν να εκφραστούν σύνθετες, μη δενδρικές σχεσιακές δομές. Για παράδειγμα, μια έννοια Λογικών Περιγραφής δεν μπορεί να εκφράσει αυθαίρετες κυκλικές δομές, αφού, για τη διατήρηση της αποφασισιμότητας, χρειάζεται αυτή ακριβώς η ιδιότητα μοντέλου δένδρου (tree model property, βλ. ενότητα 3.7.1). Από την άλλη, τα *συζευγμένα ερωτήματα* είναι συνήθη στην κοινότητα των βάσεων δεδομένων και έχουν ισχυρότερες εκφραστικές δυνατότητες από την ανάκτηση στιγμιότυπων. Για παράδειγμα, έστω μια βάση γνώσης που περιέχει τη δήλωση (ΞέχειΓιο.(ΞέχειΚόρη.Τ))(Μαρία), που δηλώνει ότι το άτομο (ή η σταθερά, σε όρους Πρώτης Τάξης) Μαρία έχει ένα γιο που έχει μια κόρη. Επομένως, η Μαρία είναι γιαγιά. Επίσης, υποθέτουμε ότι οι ρόλοι έχειΓιο και έχειΚόρη έχουν έναν μεταβατικό υπερ-ρόλο έχειΑπόγονο. Αυτό σημαίνει ότι η Μαρία σχετίζεται μέσω του ρόλου έχειΑπόγονο με το (άνωνυμο) εγγόνι της. Είναι σαφές ότι, για αυτή τη βάση γνώσης, η Μαρία αποτελεί την απάντηση στο συζευγμένο ερώτημα έχειΓιο(x,y) \wedge έχειΚόρη(y,z) \wedge έχειΑπόγονο (x,z), όπου θεωρούμε ότι το x είναι μια διακεκριμένη μεταβλητή (επίσης ονομάζεται *ελεύθερη*) και τα y, z είναι μη διακεκριμένες (υπαρξιακά ποσοτικοποιημένες) μεταβλητές. Είναι λοιπόν ακριβώς αυτή η ύπαρξη *μεταβλητών* και, ακόμα περισσότερο, η δημιουργία *κύκλων* μέσω αυτών που καθιστά ένα συζευγμένο ερώτημα ισχυρό και γενικά δυσκολότερο από τον κλασσικό συλλογισμό TBox/ABox.

Αν όλες οι μεταβλητές στο ερώτημα είναι μη διακεκριμένες, τότε η απάντηση είναι απλώς αληθής ή ψευδής και το ερώτημα ονομάζεται *δυναδικό* (Boolean). Το ερώτημα q είναι αληθές, αν η βάση γνώσης K *συνεπάγεται* το ερώτημα ($K=q$). Για ερωτήματα που περιλαμβάνουν ελεύθερες μεταβλητές, η απάντηση συνίσταται σε εκείνες τις πλειάδες ονομάτων ατόμων για τις οποίες η βάση γνώσης συνεπάγεται το ερώτημα που προκύπτει αντικαθιστώντας τις ελεύθερες μεταβλητές με τα άτομα της κάθε πλειάδας. Αυτό έχει την έννοια ότι για να ελεγχθεί η *απάντηση ερωτήματος* για ένα μη δυναδικό ερώτημα, θα πρέπει να ελεγχθούν, εξαντλητικά, όλοι οι πιθανοί συνδυασμοί ονομάτων ατόμων στις θέσεις των ελεύθερων μεταβλητών. Αν m το πλήθος των ελεύθερων μεταβλητών και k η δύναμη του

ABox, το πλήθος των συνεπαγωγών δυαδικών ερωτημάτων που θα πρέπει να ελεγχθούν είναι k^m , δηλαδή εκθετικό ως προς το μέγεθος του ερωτήματος.

Ένας συνηθισμένος περιορισμός είναι να επιτρέπονται ως αναθέσεις μεταβλητών μόνο στιγμιότυπα των οποίων το όνομα υπάρχει στο ABox. Επομένως, όλες οι μεταβλητές μπορούν να αντικατασταθούν από *βασικούς όρους* (ground terms). Ο περιορισμός αυτός ουσιαστικά αποτελεί τη λεγόμενη *υπόθεση κλειστού κόσμου* (closed world assumption-CWA), σε αντίθεση με τις Λογικές Περιγραφές, όπου αν κάτι δεν είναι ρητά δηλωμένο στη βάση γνώσης δεν θεωρείται και ψευδές. Αυτή είναι και η προσέγγιση που ακολουθείται από τους περισσότερους υλοποιημένους μηχανισμούς συλλογισμού. Έτσι, το πρόβλημα της απάντησης ερωτήματος ανάγεται εύκολα στη λειτουργία ανάκτησης στιγμιότυπων, αντικαθιστώντας και τις μη διακεκριμένες μεταβλητές με τα ονόματα ατόμων του ABox και ελέγχοντας τη συνεπαγωγή κάθε μέρους της σύζευξης χωριστά. Με τον περιορισμό αυτό όμως, η απάντηση στο ερώτημα του προηγούμενου παραδείγματος θα ήταν ψευδής, εφόσον δεν υπάρχουν άλλα στιγμιότυπα εκτός από τη Μαρία μέσα στο ABox.

Στο [Glimm, Horrocks, Lutz, & Sattler, 2008] δείχνεται ότι η πολυπλοκότητα της συνεπαγωγής συζευγμένων ερωτημάτων για τη Λογική Περιγραφή *SHIQ* (λίγο πιο εκφραστικής από την OWL Lite) είναι 2Exp-complete και εισάγεται αντίστοιχος αλγόριθμος, ο οποίος αναγάγει το πρόβλημα στην ικανοποιησιμότητα μιας βάσης γνώσης *SHIQ*^π (Η *SHIQ* με την επιπρόσθετη υποστήριξη συζεύξεων ρόλων). Η πολυπλοκότητα αυτή είναι μεγαλύτερη από την πολυπλοκότητα της λειτουργίας ελέγχου στιγμιότυπων για τη λογική (Exp-complete). Επίσης στο [Glimm, Horrocks, & Sattler, 2006] παρουσιάζεται αλγόριθμος για τη συνεπαγωγή ερωτημάτων στη *SHOIQ* (OWL DL) με τον ουσιαστικό όμως περιορισμό ότι δεν επιτρέπονται μεταβατικοί ρόλοι μέσα σε κυκλικά ερωτήματα. Ο αλγόριθμος ουσιαστικά αναγάγει το πρόβλημα στην ικανοποιησιμότητα μιας βάσης γνώσης *SHOIQ*_↓, μιας επέκτασης δηλαδή της *SHOIQ*. Συνακόλουθα, η πολυπλοκότητά του θα είναι τουλάχιστον όσο αυτή για την ικανοποιησιμότητα σε *SHOIQ* (NExp-complete). Χωρίς τον περιορισμό για τους μεταβατικούς ρόλους, αποτελεί ανοιχτό πρόβλημα αν η συνεπαγωγή συζευγμένων ερωτημάτων είναι αποφασίσιμη στη *SHOIQ* και άρα και στη *SROIQ*, που αντιστοιχεί στην OWL 1.1. Θυμίζουμε επίσης ότι οι ταμπλό αλγόριθμοι ελέγχου ικανοποιησιμότητας για τις παραπάνω λογικές επιτρέπουν στους περιορισμούς αριθμού μόνο *απλούς* ρόλους (δηλ. που δεν είναι μεταβατικοί, ούτε έχουν μεταβατικούς υπορόλους). Τέλος, η επέκταση συζευγμένων ερωτημάτων με ανισότητα είναι μη αποφασίσιμη [Calvanese, Giacomo, & Lenzerini, 1998].

4.3.3. Η SPARQL-DL

Λόγω των ιδιαίτερων χαρακτηριστικών του Παγκόσμιου Ιστού και της κατανομής της πληροφορίας σε αυτόν, είναι επομένως σκόπιμο να υποστηρίζεται η υποβολή και απάντηση επερωτήσεων σε συζευγμένη μορφή, η οποία άλλωστε προσομοιάζει και του πολύ δημοφιλούς τρόπου ανάκτησης στις βάσεις δεδομένων. Εκτός όμως από τις σχετικά υψηλές πολυπλοκότητες και τους περιορισμούς που υπάρχουν για τις λογικές που αντιστοιχούν στις διαλέκτους της OWL, υπάρχει και έλλειμμα σχετικά με κάποιον τυποποιημένο τρόπο σχηματισμού, υποβολής και επεξεργασίας των ερωτημάτων αυτών.

Η γλώσσα SPARQL για παράδειγμα [Prud'hommeaux & Seaborne, 2008] έχει σημασιολογία που βασίζεται στην έννοια των RDF γραφημάτων και άρα δεν μπορεί να

αντιστοιχηθεί ξεκάθαρα στην OWL. Από την άλλη, τα ερωτήματα τύπου ASK του πρωτοκόλλου DIG/1.1 περιορίζονται μόνο σε ατομικά (και όχι συζευγμένα) ερωτήματα, ενώ η γλώσσα nRQL που υποστηρίζεται από το RACER είναι κατάλληλη για συζευγμένα ερωτήματα που αφορούν μόνο στο ABox. Αξίζει να σημειωθεί ότι το πρωτόκολλο DIG/1.2, όπως αυτή τη στιγμή υλοποιείται επίσης στην εμπορική εκδοχή του συστήματος RACER (1.9.2) φαίνεται να υποστηρίζει συζευγμένα ερωτήματα βασικής (grounded) μορφής. Η συγκεκριμένη επέκταση πρόκειται να αποτελέσει μέρος του επερχόμενης έκδοσης του πρωτοκόλλου, DIG/2.0 (βλ. και ενότητα 4.6.2).

Ερώτημα	Τύπος ερωτήματος	Παράδειγμα
Q1	TBox	SubClassOf (?c, ex:Student)
Q2	ABox	Type(?x, and(ex:Student, ex:Employee)), PropertyValue(?x, ex:name, ?y)
Q3	ABox με μη διακεκριμένες μεταβλητές	PropertyValue(?x, ex:hasPublication, _:y), PropertyValue(_:y, ex:publishedAt, _:z), Type(_:z, not(ex:Workshop))
Q4	ABox/TBox	Type(?x, ex:Student), Type(?x, ?c) SubClassOf(?c, ex:Employee)
Q5	ABox/RBox	ObjectProperty(?p), PropertyValue(ex:John, ?p, ?u)

Πίνακας 4.1: Παραδείγματα ερωτημάτων SPARQL-DL.

Οι Sirin και Parsia [Sirin & Parsia, 2007] εισάγουν τη γλώσσα SPARQL-DL, ως υποσύνολο της SPARQL, η οποία επιτρέπει την έκφραση, υπό προϋποθέσεις, και μη βασικών συζευγμένων ερωτημάτων και επιπρόσθετα την ανάμιξή τους με ταξινομικά ερωτήματα (βλ. π.χ. τα ερωτήματα Q4 και Q5 στον πίνακα 4.1), για τη γλώσσα OWL DL. Ταυτόχρονα, η SPARQL-DL επιτρέπει να γίνονται ερωτήσεις που αφορούν σε λειτουργίες του ABox και που όμως δεν μπορούν να εκφραστούν σε Λογική Πρώτης Τάξης. Για παράδειγμα το ερώτημα:

ex:a rdf:type ?C

το οποίο επιστρέφει τις έννοιες στις οποίες ανήκει το στιγμιότυπο a.

Τα μη βασικά ερωτήματα εκφράζονται χρησιμοποιώντας στη θέση των μη διακεκριμένων μεταβλητών κενούς κόμβους (b-nodes). Στο ερώτημα Q3 του πίνακα, Τα $_y$, $_z$ είναι κενοί κόμβοι. Ο περιορισμός έγκειται στο ότι η λύση ενός SPARQL-DL ερωτήματος αντιστοιχίζει τις μεταβλητές μόνο σε κατονομασμένα στιγμιότυπα (ή ονόματα ατομικών εννοιών ή ονόματα ατομικών ρόλων ή λεκτικά) και όχι σε κενούς κόμβους. Για παράδειγμα αν η βάση γνώσης περιέχει μόνο τη δήλωση

(∃ex:p.ex:C)(ex:a)

το ερώτημα $\text{PropertyValue}(?x, ?y)$ δεν θα επιστρέψει κανένα αποτέλεσμα, γιατί δεν είναι γνωστό το όνομα του ατόμου στο οποίο μπορεί το $?y$ να αντιστοιχηθεί. Αντίθετα, το ερώτημα $\text{PropertyValue}(?x, _y)$ είναι επιτυχές. Αν επιτρέπονταν κενοί κόμβοι στην απάντηση, τότε το $?y$ θα μπορούσε να αντιστοιχηθεί σε ένα μοναδικό αναγνωριστικό κενού κόμβου, π.χ. $_b$. Ακόμα, ο κενός αυτός κόμβος θα μπορούσε να χρησιμοποιηθεί σε μετέπειτα ερωτήματα, π.χ. $\text{Type}(_b, ?C)$ («σε ποιά έννοια ανήκει το $_b$;») το οποίο θα επέστρεφε ex:C .

Η SPARQL-DL έχει υλοποιηθεί και υποστηρίζεται, με διάφορες βελτιστοποιήσεις, από τις πρόσφατες εκδόσεις του Pellet. Ωστόσο, η απάντηση ερωτημάτων που περιέχουν κύκλους μη διακεκριμένων μεταβλητών δεν υποστηρίζεται για την OWL DL, κάτι που είναι λογικό,

εφόσον το αντίστοιχο πρόβλημα είναι ανοιχτό. Όπως όμως φάνηκε στα προηγούμενα, με την απαγόρευση μεταβατικών ρόλων σε τέτοια ερωτήματα, το πρόβλημα θα ήταν εφικτό να αντιμετωπιστεί. Επίσης δεν είναι επιτρεπτή η χρήση μεταβλητών (διακεκριμένων ή μη) μέσα σε σύνθετες εκφράσεις εννοιών. Από άποψη επίδοσης, η ύπαρξη μη διακεκριμένων μεταβλητών στο ερώτημα φαίνεται ότι επιβαρύνει σημαντικά το χρόνο απόκρισης του Pellet [Kremen & Sirin, 2008].

Η SPARQL-DL μπορεί να επεκταθεί ομαλά και προς την OWL 1.1. Τα πρόσθετα χαρακτηριστικά ρόλων που αυτή η γλώσσα φέρει, μπορούν να αντιμετωπιστούν με την προσθήκη αντίστοιχων ατόμων ερώτησης (query atoms), όπως π.χ. *Reflexive(p)* για τον έλεγχο της ανακλαστικότητας μιας ιδιότητας *p*. Επίσης, η παρονομασία οδηγεί σε διάκριση των ατόμων ερώτησης που αφορούν την ιεραρχία των ιδιοτήτων, δηλαδή *subObjectPropertyOf* και *subDataPropertyOf*, αντί για *subPropertyOf*, επιλύοντας έτσι την αμφισημία στο επίπεδο του ερωτήματος. Αν για παράδειγμα η ιδιότητα *ex:p* ορίζεται ταυτοχρόνως και ως ιδιότητα αντικειμένου και ως τύπου δεδομένων, τότε είναι πιθανό τα ερωτήματα *subObjectPropertyOf(?x, ex:p)* και *subDataPropertyOf(?x, ex:p)* να έχουν διαφορετικά αποτελέσματα.

4.4. Αξιολόγηση Μηχανισμών Συμπερασμού

Έχοντας συζητήσει (στο κεφάλαιο 3) τα πλεονεκτήματα και τα μειονεκτήματα των Λογικών Περιγραφής ως τον υποκείμενο φορμαλισμό συλλογισμού για το Σημαντικό Ιστό, στη ενότητα αυτή θα εξεταστούν έξι δημοφιλή συστήματα συμπερασμού που θα μπορούσαν να χρησιμοποιηθούν για την παροχή υπηρεσιών συλλογισμού σε OWL έγγραφα: τα Cerebra, FaCT, FaCT++, RACER, Pellet και KAON2. Το KAON2 είναι το μόνο που δεν βασίζεται σε Λογικές Περιγραφής και έχει συμπεριληφθεί για λόγους πληρότητας και προκειμένου να αναδειχθούν οι διαφορές με τις εναλλακτικές προσεγγίσεις συλλογισμού. Η αξιολόγησή των συστημάτων, που συνοψίζεται στον πίνακα 4.2, διεξάγεται λαμβάνοντας υπόψη την διαθεσιμότητα, την εκφραστικότητα, την υποστήριξη για την OWL, το συλλογισμό με στιγμιότυπα και τις δυνατότητες διασύνδεσης που παρέχονται από τα εν λόγω συστήματα.

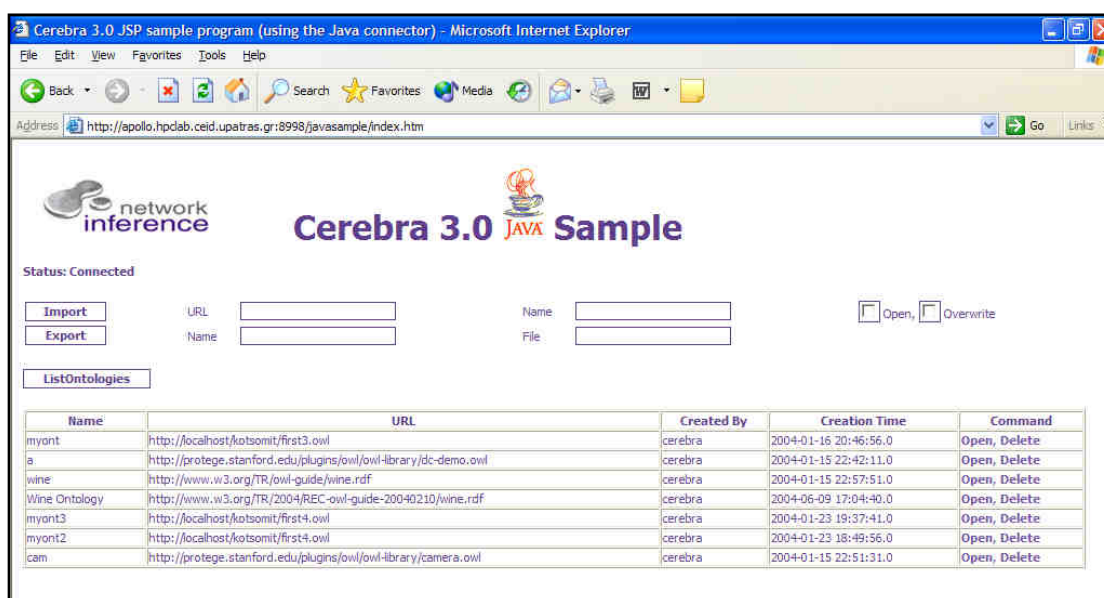
	Διαθεσιμότητα	Συνδεσιμότητα	Εκφραστική Ισχύς	Εγγενής Υποστήριξη για OWL (σύνταξη)	Συλλογισμός με Στιγμιότυπα (ABox)
Cerebra	Εμπορικό	RMI, SOAP	SHIF	Ναι	Όχι
FaCT	Δωρεάν	CORBA, DIG/1.0	SHIQ	Όχι	Όχι
FaCT++	Δωρεάν	JNI, DIG/1.1	SROIQ(D)	Όχι	Ναι
Racer	Δωρεάν (πριν την 1.8)	TCP, DIG/1.2	SHIQ(D)	Ναι	Ναι
Pellet	Δωρεάν	DIG/1.1	SROIQ(D)	Ναι	Ναι
KAON2	Δωρεάν	RMI, DIG/1.1	SHIQ(D)	Ναι	Ναι

Πίνακας 4.2: Συγκριτική αξιολόγηση μηχανισμών συμπερασμού βασισμένων σε Λογικές Περιγραφής.

4.4.1. Cerebra

Το Cerebra, της Cerebra Inc. (πρώην Network Inference, τώρα εξαγορασμένο από την webMethods και ενσωματωμένο στο προϊόν της Fabric) είναι ένα εμπορικό σύστημα, που υποστηρίζει συλλογισμό, καθώς και υπηρεσίες διαχείρισης οντολογιών (εικόνα 4.1). Το Cerebra διαφέρει από τα παραδοσιακά, βασισμένα σε Λογικές Περιγραφής συστήματα στο ότι παρέχει επιπλέον χαρακτηριστικά που είναι επιθυμητά σε ένα επιχειρησιακό περιβάλλον, για την ανάπτυξη αξιόπιστων συστημάτων παραγωγής με κάποιες ευφυείς δυνατότητες, αλλά πάντως όχι στην ισχύ των προσφερόμενων συμπερασμών.

Πράγματι, μια ενδιαφέρουσα ιδιότητα του Cerebra είναι η προσθήκη διαχρονικότητας (persistence) στις βάσεις γνώσης που είναι ικανό να επεξεργάζεται. Το Cerebra μπορεί να φορτώσει OWL έγγραφα είτε από το τοπικό σύστημα αρχείων είτε απευθείας από το Διαδίκτυο, με την παροχή του αντίστοιχου URL. Κατόπιν, οι πληροφορίες της οντολογίας αποθηκεύονται, ακολουθώντας μια εσωτερική μοντελοποίηση, σε μια σχεσιακή βάση δεδομένων, από όπου μπορούν να φορτωθούν ξανά, όποτε χρειαστεί.



Εικόνα 4.1: Φόρτωση και διαχείριση οντολογιών στο σύστημα Cerebra.

Η αρχιτεκτονική του συστήματος υποστηρίζει τη διαχείριση πολλών αιτημάτων ταυτόχρονα: Διαφορετικοί χρήστες μπορούν να προσπελάζουν και να θέτουν ερωτήματα σε διαφορετικές οντολογίες την ίδια στιγμή. Επίσης παρέχεται πλήρης υποστήριξη για την οδηγία `<owl:imports>` στα OWL έγγραφα: Αν κάποια OWL οντολογία απαιτεί την συμπερίληψη των πληροφοριών και από άλλους χώρους ονομάτων, το Cerebra επιχειρεί να τις προσπελάσει μέσω του Παγκόσμιου Ιστού και, αν επιτύχει, τις αποθηκεύει μαζί με την αρχική οντολογία.

Το Cerebra επιτρέπει τη σύνδεση εφαρμογών πελάτη που είναι βασισμένες σε Java ή σε .NET. Εκτός όμως από αυτό, οποιαδήποτε υπηρεσία Ιστού μπορεί να αξιοποιήσει τη λειτουργικότητα του συστήματος, μέσω της διεπαφής SOAP που υποστηρίζει. Η διεπαφή SOAP εγκαθίσταται ως υπηρεσία Ιστού στον εξυπηρετητή Tomcat.

Οι κατασκευασμένοι σε Java πελάτες μπορούν να διασυνδεθούν με το σύστημα είτε μέσω RMI είτε μέσω SOAP, χρησιμοποιώντας τις κλάσεις που παρέχει το Cerebra για το σκοπό

αυτό. Στην περίπτωση του .NET, το Cerebra παρέχει μια βιβλιοθήκη .dll μέσω της οποίας μπορεί να επιτευχθεί σύνδεση με τη SOAP διεπαφή του συστήματος. Και στις δύο περιπτώσεις παρέχεται ένα API που δίνει τη δυνατότητα επεξεργασίας, διαχείρισης και υποβολής ερωτημάτων στις οντολογίες. Η υποβολή ερωτημάτων σε μια οντολογία, ιδιαίτερα αυτών που αφορούν τα στιγμιότυπα, όσο και οι προκύπτουσες απαντήσεις, ακολουθούν, ως ένα βαθμό, το πρότυπο XQuery.

Δεν υπάρχει κάποια τεκμηρίωση στην οποία να δηλώνεται η συλλογιστική ισχύς του Cerebra. Είναι όμως γνωστό ότι το σημασιολογικό μοντέλο που ακολουθεί εσωτερικά για τη διεξαγωγή συμπερασμών βασίζεται στις Λογικές Περιγραφές. Η πειραματική αξιολόγηση του συστήματος έδειξε ότι, όσον αφορά το ταξινομικό μέρος μιας οντολογίας, δηλαδή το TBox, το Cerebra υποστηρίζει σχεδόν όλους τους κατασκευαστές και τα αξιώματα για τις κλάσεις και τους ρόλους (συμπεριλαμβανομένων των συνολοθεωρητικών τελεστών), κάτι που θα ανήγαγε την εκφραστική του ικανότητα στο επίπεδο της OWL DL. Περαιτέρω όμως πειραματικός έλεγχος που διεξήχθη στα πλαίσια της παρούσας εργασίας έδειξε τα ακόλουθα:

- Την αδυναμία αναγνώρισης των συμμετρικών ρόλων, κάτι που επισημάνθηκε κατά την πειραματική αξιολόγηση και αναγνωρίστηκε ως σφάλμα του συστήματος.
- Την αδυναμία έκφρασης ελάχιστης πληθικότητας μεγαλύτερης της μονάδας (π.χ. $\text{minCardinality}=2$), κάτι που χρησιμοποιείται στην μοντελοποίηση περιορισμών αριθμού.
- Το σημαντικότερο, ότι δεν υποστηρίζεται συμπερασμός στο πεδίο των στιγμιότυπων μιας οντολογίας (ABox). Εξαιρεση ίσως αποτελεί η λειτουργία `instanceProperty`, που όμως, δοθείσης μιας κλάσης και ενός ρόλου, επιστρέφει όλα τα ζεύγη στιγμιότυπων που συμπεραίνεται ότι σχετίζονται μέσω του δοσμένου ρόλου και που το αριστερό όρισμα προέρχεται από τη δοσμένη κλάση.

Τα παραπάνω κατατάσσουν την εκφραστικότητα του Cerebra το πολύ στο επίπεδο της Λογικής Περιγραφής *SHIF*. Από την άλλη, το σχεσιακό μοντέλο που εκμεταλλεύεται το Cerebra επιτρέπει την υποβολή πολύ ισχυρών ερωτημάτων ανάκτησης για τα στιγμιότυπα, που βασίζονται στο συντακτικό της XQuery. Τα ερωτήματα αυτά μπορούν να εμπλέκουν και τύπους δεδομένων, όπως συμβολοσειρές και αριθμούς, καθώς και τελεστές μεταξύ αυτών (ισότητα, σύγκριση), αλλά τα αποτελέσματα βασίζονται στην πληροφορία που είναι ρητά μόνο εκφρασμένη μέσα στην οντολογία και όχι σε αυτήν που μπορεί να συμπεραθεί.

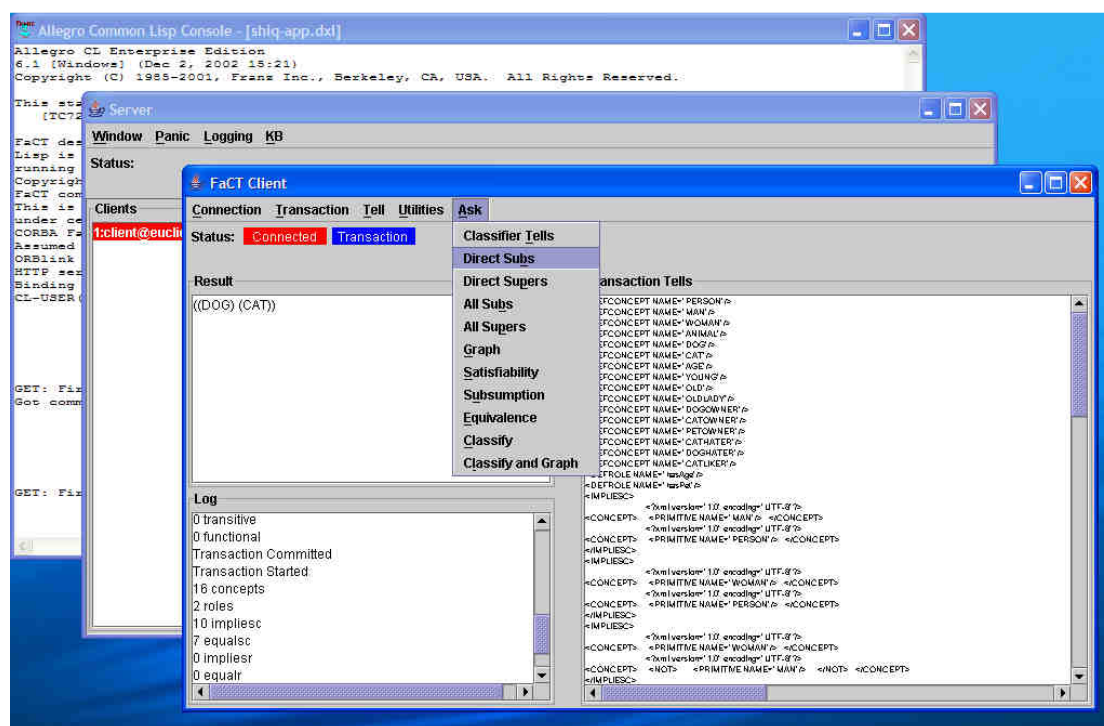
4.4.2. FaCT

Το σύστημα FaCT [Horrocks & Sattler, 2002] (Fast Classification of Terminologies - Ταχεία Ταξινόμηση Ορολογιών) αποτελεί ελεύθερα διαθέσιμο λογισμικό συλλογισμού το οποίο αναπτύχθηκε στο πανεπιστήμιο του Manchester υπό τον καθηγητή Ian Horrocks. Το σύστημα είναι υλοποιημένο σε Common Lisp, ενώ η διανομή του περιλαμβάνει αυτόνομα εκτελέσιμα τόσο για Windows όσο και για Linux. Αρχικά το FaCT υποστήριζε τη Λογική Περιγραφής *SHF*, ενώ αργότερα εξελίχθηκε για να συμπεριλάβει την *SHIF* και τέλος την *SHIQ*.

Οι πιο πρόσφατες εκδόσεις του FaCT επιτρέπουν τη διασύνδεση με άλλες εφαρμογές, ακολουθώντας το μοντέλο πελάτη - εξυπηρετητή μέσω μιας CORBA διεπαφής [Bechhofer,

Horrocks, Patel-Schneider, & Tessaris, 1999], το οποίο όμως απαιτεί περιβάλλον εκτέλεσης Java. Επιπλέον το FaCT υποστηρίζει και το πρότυπο DIG/1.0 που προδιαγράφει έναν απλό τρόπο επικοινωνίας με προγράμματα συμπερασμού Λογικών Περιγραφής, μέσω της ανταλλαγής XML αιτημάτων και αποκρίσεων πάνω από το πρωτόκολλο HTTP.

Το FaCT υλοποιεί βελτιστοποιημένους, πλήρεις και συνεπείς αλγορίθμους για το πρόβλημα της υπαγωγής στις Λογικές Περιγραφής που αναφέρθηκαν παραπάνω. Αν και πρωτοποριακό σύστημα για την εποχή του, με επιδόσεις που το κατέτασσαν αρκετά ψηλά σε σχέση με άλλα παραδοσιακά συστήματα Λογικών Περιγραφής, το FaCT δεν υποστηρίζει συλλογισμό στο ABox, γεγονός που αυτόματα το καθιστά ακατάλληλο για τη γλώσσα OWL. Πράγματι, στα πλαίσια της πειραματικής αξιολόγησης του συστήματος, δοκιμάστηκε η μετατροπή μιας απλής OWL οντολογίας στην ενδιάμεση μορφή που υποστηρίζει το FaCT. Η μετατροπή αυτή έγινε χρησιμοποιώντας ένα εργαλείο που διατίθεται μέσω της ιστοσελίδας του ευρωπαϊκού IST έργου WonderWeb (<http://wonderweb.semanticweb.org>), στο οποίο εξάλλου εντάσσεται και η ανάπτυξη της επόμενης έκδοσης του συστήματος (FaCT++). Από την μετατροπή αυτή φάνηκαν τα εξής:



Εικόνα 4.2: Φόρτωση οντολογίας και υποβολή ερωτήματος στο σύστημα FaCT χρησιμοποιώντας την εφαρμογή πελάτη της διανομής.

- Τα άτομα μετατρέπονται σε αρχικές έννοιες.
- Οι σχέσεις μεταξύ των ατόμων δεν διατηρούνται.
- Οι νέες έννοιες που δημιουργήθηκαν για να αναπαραστήσουν τα άτομα υπάγονται τώρα στις έννοιες στις οποίες ανήκαν τα άτομα αρχικά.

Εκτός όμως από την αδυναμία υποστήριξης ABox και τύπων δεδομένων (απλών πεδίων) που καθιστούν το FaCT σημασιολογικά ανεπαρκές έστω και για την OWL Lite, το σύστημα είναι και συντακτικά ασύμβατο με την OWL. Εκτός από την ενδιάμεση, Lisp μορφή των βάσεων γνώσης, το FaCT υποστηρίζει οντολογίες σε XML μορφή, ακολουθώντας όμως ένα

ιδιαίτερο σχήμα που δεν συμβαδίζει με κάποιο πρότυπο αναπαράστασης οντολογιών σε XML (όπως π.χ. η γλώσσα XOL) και φυσικά, δεν μπορεί να αντιστοιχιστεί προς και από την OWL χωρίς σημαντική απώλεια πληροφορίας.

4.4.3. FaCT++

Πολλά από τα παραπάνω μειονεκτήματα αντιμετωπίζονται στην επόμενη έκδοση του συστήματος, FaCT++ [Tsarkov & Horrocks, 2006], το οποίο έχει αναπτυχθεί στα πλαίσια του έργου WonderWeb. Το FaCT++ είναι κατά βάση μια επανυλοποίηση του FaCT σε C++ με διαφορετική όμως εκφραστικότητα και με στόχο τελικά την υποστήριξη της OWL DL.

Το FaCT++ υλοποιεί τη διαδικασία απόφασης για τη *SHOIQ* [Horrocks & Sattler, 2005] και επίσης τις προτεινόμενες επεκτάσεις συνόλων κανόνων για την OWL 1.1. Το γεγονός αυτό, σε συνδυασμό με τη νέα αρχιτεκτονική συλλογισμού με τύπους δεδομένων του συστήματος, κατατάσσουν την εκφραστικότητα του FaCT++ στο επίπεδο της λογικής *SROIQ(D)*.

Οι πρώιμες εκδόσεις του FaCT++ (πριν την 1.0) δεν υποστηρίζουν εγγενώς την XML/OWL σύνταξη. Ωστόσο, παρέχεται ένα εργαλείο για μετατροπή στην ενδιάμεση Lisp μορφή που υποστηρίζει το FaCT++. Όσον αφορά τα άτομα (και κατ' επέκταση τα ονοματικά), παρόλο που διαπιστώθηκε ότι επιβιώνουν της μετατροπής, ωστόσο η υποστήριξη για αυτά είναι προς το παρόν πολύ περιορισμένη, καθώς αντιμετωπίζονται συλλήβδην ως αρχικές έννοιες.

Οι πρόσφατες εκδόσεις φαίνεται ότι μιμούνται την αφηρημένη σύνταξη της OWL κατά τρόπο τύπου ASK/TELL, αλλά η πλήρης υποστήριξη για το OWL συντακτικό έχει αφεθεί ως στόχος της υλοποίησης [Tsarkov & Horrocks, 2006]. Επιπρόσθετα, τα εκτελέσιμα του συστήματος δεν φαίνεται να λειτουργούν σε κατάσταση μαζικής λειτουργίας (batch mode) ή να προσφέρουν δυνατότητες φόρτωσης από την γραμμή εντολών. Η εκτέλεση του FaCT++ ενεργοποιεί τον εξυπηρετητή DIG σε μια καθορισμένη από το χρήστη θύρα.

Ως πρόγραμμα C++ και προκειμένου για την υποστήριξη του OWL API (το Jena δεν υποστηρίζεται ακόμα), το FaCT++ έχει επεκταθεί με μια Εγγενή Διεπαφή Java (Java Native Interface, JNI) [Horridge, Tsarkov, & Redmond, 2006]. Ακόμα κι έτσι όμως απαιτείται η απευθείας στη μνήμη ανάπτυξη του προγράμματος.

4.4.4. RACER

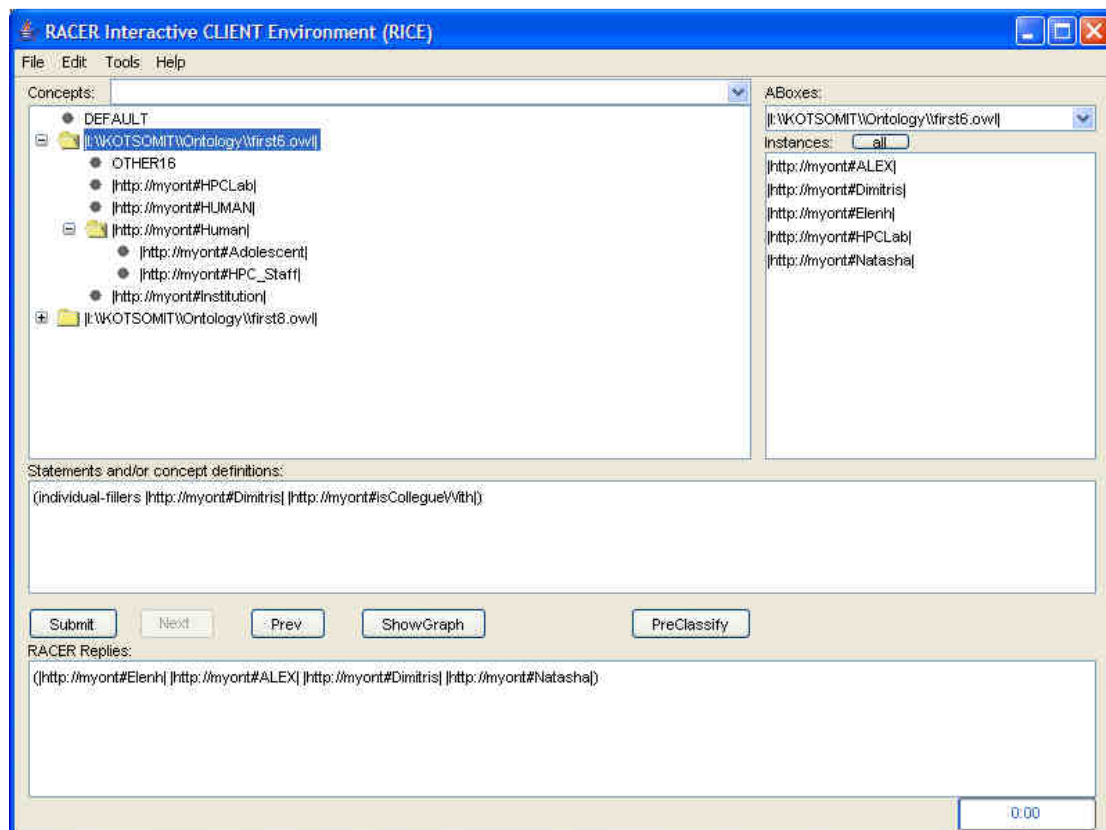
Το σύστημα RACER [Haarslev & Möller, 2003] (Renamed ABox and Concept Expression Reasoner - Μετονομασμένο Πρόγραμμα Συλλογισμού για ABox και Εκφράσεις Εννοιών) είναι ένας μηχανισμός συμπερασμού για πολύ εκφραστικές Λογικές Περιγραφές, και αποτελεί το πρώτο λογισμικό της κατηγορίας του που υποστηρίζει συλλογισμό και στο ABox μιας βάσης γνώσης. Το γεγονός αυτό αποτέλεσε για αρκετό καιρό το κύριο πλεονέκτημα του συστήματος σε σχέση με τους υπόλοιπους μηχανισμούς συμπερασμού.

Το RACER αναπτύσσεται από τους καθηγητές Volker Haarslev και Ralf Moeller στο πανεπιστήμιο Concordia του Montreal και στο πολυτεχνείο του Αμβούργου, αντίστοιχα. Είναι ελεύθερα διαθέσιμο για ερευνητικούς σκοπούς, ενώ έχει συσταθεί εταιρία για την εμπορική εκμετάλλευση του συστήματος (Racer v1.8+ ή RacerPro). Είναι γραμμένο σε Common Lisp, ενώ διατίθεται σε αυτόνομο εκτελέσιμα για διάφορες πλατφόρμες λειτουργικών συστημάτων, συμπεριλαμβανομένων των Windows και Linux.

Η διασύνδεση του RACER με άλλες εφαρμογές επιτυγχάνεται είτε μέσω της διεπαφής TCP/IP του συστήματος, είτε μέσω HTTP, αφού το σύστημα υποστηρίζει πλήρως το πρότυπο DIG/1.2. Στην περίπτωση του TCP, έχουν αναπτυχθεί και είναι διαθέσιμες προγραμματιστικές διεπαφές (APIs) τόσο σε Java, όσο και σε C/C++. Παράλληλα, το RACER μπορεί να λειτουργήσει και σε «κατάσταση αρχείων», όπου τα αρχεία της οντολογίας και των επιθυμητών ερωτημάτων φορτώνονται ως παράμετροι από τη γραμμή εντολών.

Εκτός από την Lisp μορφή των βάσεων γνώσης, το RACER μπορεί να φορτώσει και να επεξεργαστεί εγγενώς οντολογίες γραμμένες σε XML, RDF, RDFS, DAML+OIL και τέλος OWL (από την έκδοση 1.7.7 και μετά). Η υποκείμενη Λογική Περιγραφής για την οποία παρέχει συλλογισμό το σύστημα είναι η SHIQ(D), συμπεριλαμβανομένων των στιγμιότυπων (ABox). Έτσι, όπως επιβεβαιώθηκε και πειραματικά, η σχέση του RACER με την OWL (DL), μπορεί να χαρακτηριστεί από τα παρακάτω σημεία:

- Το RACER υποστηρίζει πλήρως το συντακτικό της OWL.
- Η σημαντική της γλώσσας είναι πιο εκφραστική από τη γλώσσα του RACER στο θέμα των *ονοματικών (O)*, που δεν υποστηρίζονται από το σύστημα.
- Το RACER υπερτερεί της εκφραστικότητας της OWL στο θέμα των *προσδιορισμένων περιορισμών αριθμού (Q έναντι N)* και στο θέμα των *απτών πεδίων* (για τα οποία, όπως φάνηκε, η OWL 1.0 δεν παρέχει παρά μόνο στοιχειώδη υποστήριξη).



Εικόνα 4.3: Υποβολή ερωτήματος στο σύστημα RACER μέσω της γραφικής διεπαφής RICE.

Το RACER για παράδειγμα ενσωματώνει αλγορίθμους για διεξαγωγή συμπερασμών που μπορεί να βασίζονται σε σχέσεις min/max μεταξύ ακεραίων, σε γραμμικές πολυωνυμικές εξισώσεις και ανισότητες πραγματικών, σε μη γραμμικές πολυωνυμικές εξισώσεις

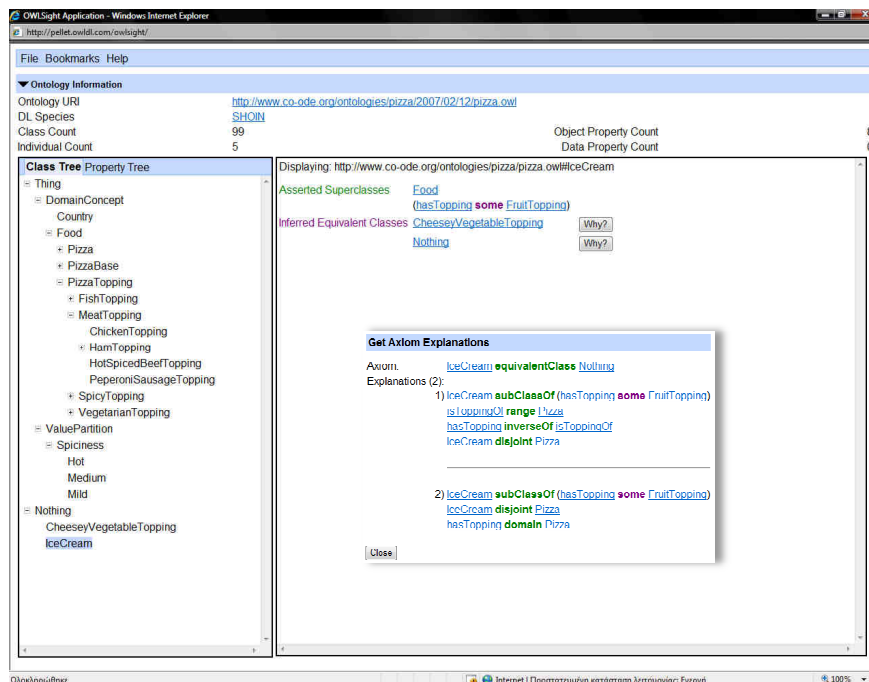
μιγαδικών και σε συγκρίσεις μεταξύ συμβολοσειρών. Από την άλλη η OWL επιτρέπει μόνο την έκφραση της ισότητας ενός ατόμου με ένα στιγμιότυπο του απτού πεδίου.

Το βασικότερο λοιπόν πρόβλημα που εμποδίζει την πλήρη συμβατότητα του RACER με την OWL DL είναι τα ονοματικά. Το RACER αντιμετωπίζει το πρόβλημα αυτό κατασκευάζοντας μία καινούρια έννοια για κάθε ονοματικό και καθιστώντας το άτομο που εμφανίζεται στο ονοματικό, στιγμιότυπο της νέας έννοιας. Έτσι ο συλλογισμός σε αυτές τις περιπτώσεις, ενώ είναι συνεπής, δεν παραμένει και πλήρης.

Για παράδειγμα, έστω η έκφραση *hasValue* της OWL για έναν συναρτησιακό ρόλο *R* και ένα άτομο *o*. Η έκφραση αυτή μπορεί να διερμηνευτεί στις Λογικές Περιγραφής ως $\exists R.\{o\}$, όπου το $\{o\}$ είναι ένα ονοματικό. Αν λοιπόν $\langle x,o \rangle \in R'$ τότε το RACER θα μπορέσει σωστά να συμπεράνει ότι $x \in (\exists R.\{o\})'$, όμως θα επιστρέψει NIL αν του ζητηθεί να αποδείξει το αντίστροφο.

4.4.5. Pellet

Το Pellet [Sirin E. , Parsia, Grau, Kalyanpur, & Katz, 2007] είναι ένας βασισμένος σε ταμπλό μηχανισμός συλλογισμού για λογικές περιγραφής γραμμένος σε Java. Το Pellet ξεκίνησε ως ερευνητικό πρόγραμμα του εργαστηρίου MindSwarp στο UMBC και αποτελεί το πρώτο σύστημα που υλοποίησε τη διαδικασία απόφασης για τη *SHOIQ*, υποστηρίζοντας έτσι την OWL DL στο σύνολό της. Το Pellet αποτελεί τώρα λογισμικό ανοιχτού κώδικα και συντηρείται από την Clark & Parsia LLC.



Εικόνα 4.4: Εμφάνιση πληροφοριών και εξήγηση ασυνέπειας από το Pellet, μέσα από την εφαρμογή OWLSight.

Το σύστημα μπορεί να λειτουργήσει ως αυτόνομη εφαρμογή από τη γραμμή εντολών, απ' όπου μπορεί να φορτώσει απευθείας και να διαχειριστεί έγγραφα OWL. Ως Java εφαρμογή προσφέρει επίσης ένα API που είναι δυνατό να χρησιμοποιηθεί σε συνδυασμό με άλλες διεπαφές όπως το Jena και το OWL API (WonderWeb). Για να αξιοποιηθεί ως δομικό

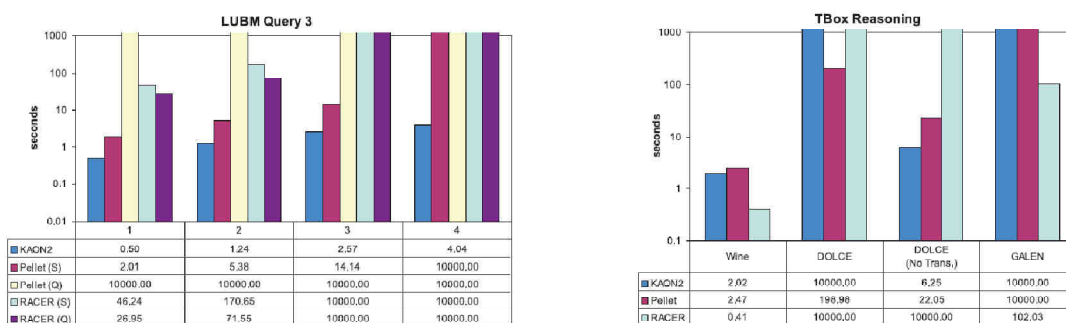
στοιχείο μιας κατανομημένης εφαρμογής, το Pellet παρέχει διεπαφή μόνο για DIG/1.1, που όμως έχει περιορισμένη υποστήριξη για την OWL. Οι βιβλιοθήκες του Pellet είναι επίσης ενσωματωμένες στο εργαλείο SWOOP και στο Protégé 4.0 (σε κατάσταση alpha).

Το Pellet επιδεικνύει συνεπή και πλήρη συμπεριφορά συλλογισμού για την OWL DL, γεγονός που σημαίνει ότι μπορεί να διαχειριστεί τα ονοματικά επιτυχώς. Επίσης, μετά την έκδοση 1.4, έχει προστεθεί υποστήριξη για τις επεκτάσεις κανόνων που έχουν προταθεί με την OWL 1.1. Προσφέρεται επιπλέον και συλλογισμός με τύπους δεδομένων του XML Schema, επομένως η υποκείμενη λογική που υποστηρίζεται είναι η *SROIQ(D)*.

4.4.6. ΚΑΟΝ2

Το ΚΑΟΝ2 (**K**arlsruhe **O**ntology **2**, <http://kaon2.semanticweb.org>) αποτελεί μια ολοκληρωμένη υποδομή για τη διαχείριση, επερώτηση και συλλογισμό με οντολογίες Ιστού. Το σύστημα αυτό έχει αναπτυχθεί σε Java και παρέχει ένα API για την προγραμματιστική διαχείριση των οντολογιών. Μπορεί όμως να λειτουργήσει και αυτόνομα, ως εξυπηρετητής, αφού υποστηρίζει τη διεπαφή RMI της Java, αλλά και το DIG, φορτώνοντας οντολογικά έγγραφα γραμμένα στην RDF/XML σύνταξη της OWL. Είναι ελεύθερα διαθέσιμο, αλλά διατίθεται επιπλέον και η εμπορική του έκδοσή με το όνομα OntroBroker.

Η σημαντικότερη διαφοροποίηση από τους προηγούμενους μηχανισμούς συμπερασμού είναι ότι το ΚΑΟΝ2 δεν υλοποιεί αλγορίθμους ταμπλό, αλλά μια διαδικασία απόφασης βασισμένη σε ανάλυση Πρώτης Τάξης. Η διαδικασία αυτή αναγάγει επομένως τη βάση γνώσης Λογικής Περιγραφής σε πρόγραμμα διαζευγμένης Datalog (disjunctive Datalog program). Στην τρέχουσα έκδοσή του, το σύστημα υλοποιεί τον (βέλτιστο) αλγόριθμο για την αναγωγή της *SHIQ(D)* λογικής περιγραφής [Hustadt, Motik, & Sattler, 2004]. Ως επέκταση αυτού, έχει δημοσιευθεί αντίστοιχος, βασισμένος σε ανάλυση αλγόριθμος και για την *SHOIQ(D)* [Kazakov & Motik, 2006], που όμως δεν φαίνεται να έχει ακόμα υλοποιηθεί. Το ΚΑΟΝ2 επίσης υποστηρίζει το DL-safe υποσύνολο της SWRL (βλ. ενότητα 3.2.3).

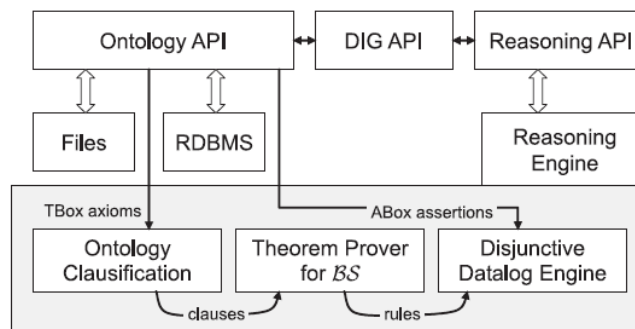


Εικόνα 4.5: Χρόνοι απόκρισης ερωτημάτων για ΚΑΟΝ2, RACER και Pellet.

Η προσέγγιση αυτή υπερτερεί σε σχέση με τις ταμπλό πρακτικές στο γεγονός ότι προσφέρει αξιοσημείωτη βελτίωση της επίδοσης κατά την απόκριση σε συζευγμένα ερωτήματα που απευθύνονται σε μεγάλα ABox (εικόνα 4.5, αριστερά), τα οποία όμως αντιστοιχούν σε σχετικά απλά TBox [Motik & Sattler, 2006]. Αντίθετα, ερωτήματα σε TBox, όπως ο έλεγχος των σχέσεων υπαγωγής και η ταξινόμηση μιας οντολογίας αποκρίνονται σε χρόνο πολύ χειρότερο σε σχέση με τις ταμπλό υλοποιήσεις (εικόνα 4.5, δεξιά). Στο σημείο

αυτό αξίζει επίσης να θυμηθούμε ότι ένας αλγόριθμος ταμπλό πραγματοποιεί συλλογισμό για στιγμιότυπα ανάγοντας το ABox σε ορολογικά αξιώματα (βλ. και ενότητα 3.6.3). Αντίθετα το KAON2 διατηρεί χωριστά δομικά στοιχεία για την ταξινόμηση του ABox και για το συλλογισμό με τα στιγμιότυπα (εικόνα 4.6), ενώ εφαρμόζει ειδικές βελτιστοποιήσεις για τα τελευταία (μαγικά σύνολα και βελτιστοποιήσεις join-order).

Το KAON2 υποστηρίζει (μερικώς) την SPARQL για το σχηματισμό και την υποβολή ερωτημάτων σε οντολογίες. Επίσης, προσφέρει μια μονάδα για αντιστοίχιση και εξαγωγή μιας σχεσιακής βάσης δεδομένων σε οντολογία, συμπεριλαμβανομένων των στιγμιότυπων. Στις αδυναμίες του συστήματος, όπως εντοπίζονται από τους δημιουργούς του, συγκαταλέγονται:



Εικόνα 4.6: Η αρχιτεκτονική του KAON2.

- Δεν υπάρχει υποστήριξη για ονοματικά. Αν μια οντολογία τα περιέχει, κάθε ενέργεια συλλογισμού θα οδηγήσει σε σφάλμα.
- Το KAON2 δεν μπορεί να χειριστεί μεγάλες τιμές σε περιορισμούς πληθικότητας. Υπάρχει περίπτωση όπου ακόμα και μια μέγιστη τιμή του 2 έχει οδηγήσει σε αδυναμία απόκρισης οποιουδήποτε ερωτήματος.

4.5. Διαχείριση και Επερώτηση Οντολογικών Εγγράφων

Αν με την υλοποίηση της KDI επιδεικνύεται, με πρότυπο τρόπο, η εφαρμογή της μεθοδολογίας ανακάλυψης γνώσης, ωστόσο είναι κατανοητό ότι σε κρίσιμα περιβάλλοντα παραγωγής η εκφραστικότητα ίσως να μην παίζει τον πρώτο ρόλο. Πράγματι, η αποδοτική διαχείριση μεγάλου αριθμού οντολογικών εγγράφων, η συντήρησή τους και η αποδοτική ανακάλυψη γνώσης μέσα από αυτά ξεφεύγουν από τους στόχους της βασικής μεθοδολογίας, αλλά αποτελούν βασική προϋπόθεση για την επιτυχία ενός ρεαλιστικού συστήματος. Για το λόγο αυτό, στα επόμενα εξετάζουμε μια σειρά διαδεδομένων προγραμματιστικών υλοποιήσεων που έχουν σαν κύριο στόχο την αποτελεσματική διαχείριση και αξιοποίηση οντολογιών: τα Sesame, Jena και OWLim. Ο πίνακας 4.3 συνοψίζει τα χαρακτηριστικά των συστημάτων αυτών.

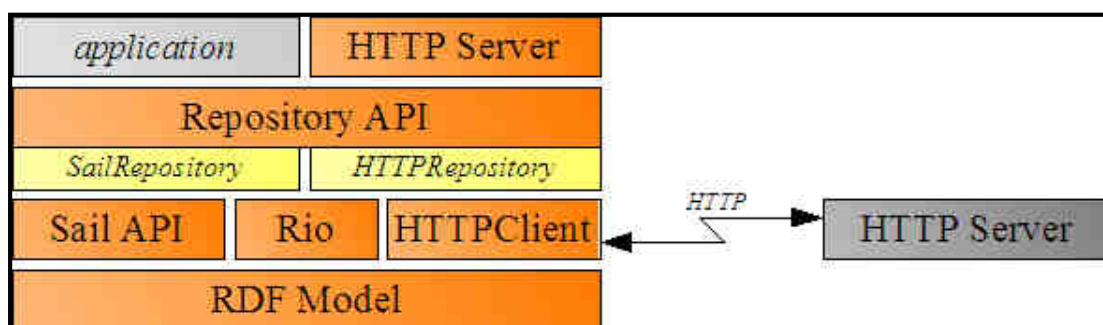
	Jena 2.5	Sesame 2.0	OWLim 2.9.1
Τελευταία Έκδοση	1/2007	12/2007	9/2007
Γλώσσες Επερωτήσεων	SPARQL, RDQL	SPARQL, SeRQL	SeRQL, RQL, RDQL
Μοντέλο	Memory, DB, File	Memory, DB, File,	όπως Sesame

Αποθήκευση	Native, RDF Repository		(1.2.7)
Υποστηριζόμενα Συστήματα Βάσεων Δεδομένων	MySQL, PostgreSQL, Oracle, BerkeleyDB, MS SQL Server, DB2, HSQLDB	MySQL	όπως Sesame (1.2.7)
Υποστηριζόμενοι Μορφότυποι	RDF/XML, N-Triples, N3, Turtle	RDF/XML, N-Triples, N3, Turtle, Trig, TriX, JSON for SPARQL results	RDF/XML, N-Triples, N3
Υποστήριξη Συλλογισμού	RDFS, OWL-Lite	RDFS, OWL-Lite	RDFS, OWL DLP, OWL Horst
Εξυπηρετητής RDF	SPARQL protocol	SPARQL protocol	όπως Sesame (1.2.7)
Άδεια	BSD	LGPL	LGPL

Πίνακας 4.3: Σύγκριση Jena, Sesame και OWLim.

4.5.1. Sesame

Το Sesame [Broekstra, Kampman, & Harmelen, 2002] είναι ένα πλαίσιο ανοικτού κώδικα που προσφέρεται για την αποδοτική αποθήκευση και επερώτηση εγγράφων RDF, παρέχοντας ταυτοχρόνως πλήρη υποστήριξη στο RDF Schema. Επιπρόσθετα, παρέχει ένα σύνολο βιβλιοθηκών (API) για την διαχείριση δεδομένων RDF, ενώ μπορεί να λειτουργήσει και ως εξυπηρετητής βάσεων δεδομένων για την προσπέλαση απομακρυσμένων αποθεμάτων. Μερικά από τα χαρακτηριστικά του είναι η υλοποίηση μηχανισμών εξαγωγής συμπερασμών, η υποστήριξη γνωστών γλωσσών επερωτήσεων για RDF, καθώς και δημοφιλών πρωτοκόλλων για την υλοποίηση του μοντέλου επικοινωνίας πελάτη-εξυπηρετητή. Οι δυνατότητες αποθήκευσης που προσφέρει ποικίλουν και μπορεί να είναι εσωτερικά στη μνήμη, σε βάση δεδομένων ή σε αρχείο. Ωστόσο, ο σχεδιασμός και η υλοποίηση του δεν εξαρτώνται από την εκάστοτε χρησιμοποιούμενη μέθοδο αποθήκευσης, συνεπώς το Sesame μπορεί να αναπτυχθεί πάνω από οποιοδήποτε αποθηκευτικό μέσο, χωρίς να χρειάζεται να τροποποιηθεί ο μηχανισμός επερωτήσεων ή κάποιο άλλο λειτουργικό του κομμάτι. Τέλος, το εν λόγω πλαίσιο, επειδή είναι υλοποιημένο σε Java, μπορεί να εκτελεστεί πάνω από οποιαδήποτε πλατφόρμα ή λειτουργικό σύστημα που διαθέτει εικονική μηχανή της Java.



Εικόνα 4.7: Η αρχιτεκτονική του Sesame 2.0.

Το Sesame έχει δυο κύριες διεπαφές επικοινωνίας: το Sail API και το Repository API (εικόνα 4.7). Το Storage And Inference Layer (SAIL) API είναι ένα σύνολο βιβλιοθηκών συστήματος, χαμηλού επιπέδου, που υλοποιεί μεθόδους αποθήκευσης για RDF και μηχανισμούς εξαγωγής συμπερασμών. Ο ρόλος του είναι να αποκρύπτει τις λεπτομέρειες

της υλοποίησης από το αμέσως πιο πάνω επίπεδο, ώστε τελικά να μπορούν να χρησιμοποιηθούν διάφορες μέθοδοι αποθήκευσης και συμπερασμών.

Το Repository API αποτελεί ένα σύνολο βιβλιοθηκών υψηλότερου επιπέδου. Πρόκειται για το βασικό API που απευθύνεται στους προγραμματιστές εφαρμογών. Παρέχει διάφορες μεθόδους για τη μεταφόρτωση αρχείων, την διεξαγωγή επερωτήσεων, καθώς και την εξαγωγή και τη διαχείριση δεδομένων. Επίσης, είναι διαθέσιμο σε δύο μορφές: την τοπική και την απομακρυσμένη. Η τοπική έκδοση του Repository API μπορεί να χρησιμοποιηθεί για τοπικά αποθέματα που τρέχουν πάνω στην ίδια εικονική μηχανή της Java (Java Virtual Machine), ενώ η απομακρυσμένη εξυπηρετεί περιπτώσεις όπου χρησιμοποιείται το μοντέλο επικοινωνίας πελάτη-εξυπηρετητή, για τη σύνδεση της εφαρμογής με έναν εξυπηρετητή Sesame.

4.5.2. Jena

Το Jena [McBride, 2002] είναι ένα ολοκληρωμένο εργαλείο για την ανάπτυξη εφαρμογών στο Σημαντικό Ιστό. Παρέχει ένα προγραμματιστικό περιβάλλον για το RDF, το RDF Schema, την OWL και τη γλώσσα επερωτήσεων SPARQL.

Η δεύτερη έκδοση του Jena (Jena2) αποτελεί βελτιωμένη έκδοση της πρώτης και συμφωνεί με τις τελευταίες προδιαγραφές του RDF. Η ουσιαστική συνεισφορά του Jena2 είναι το πλούσιο σύνολο βιβλιοθηκών που παρέχει για τη διαχείριση τέτοιου είδους δεδομένων. Ως κεντρική, εσωτερική διεπαφή το σύστημα αυτό προσφέρει μια αφαιρετική υλοποίηση του γραφήματος RDF η οποία χρησιμοποιείται σε κάθε περίπτωση γραφημάτων, είτε αυτά αναπτύσσονται εσωτερικά στη μνήμη, είτε μέσα σε μια βάση δεδομένων ή ακόμα και αν προκύπτουν μέσα από διαδικασίες εξαγωγής συμπερασμών.

Το Jena υποστηρίζει συλλογισμό για το RDFS, την OWL Lite, καθώς και το DIG 1.1, προσφέρει συντακτικούς αναλυτές και εγγραφείς για τους μορφότυπους RDF/XML, N-Triples, N3 και Turtle, ενώ, ως προς τις μεθόδους αποθήκευσης, ο χρήστης μπορεί να επιλέξει ανάμεσα στο να χρησιμοποιήσει τη μνήμη ή να προτιμήσει ένα σύστημα βάσης δεδομένων, προκειμένου να επιτύχει μόνιμη και ασφαλή διατήρηση των εγγράφων του. Τέλος, το Jena2 υποστηρίζει πλήρως τη γλώσσα επερωτήσεων SPARQL, τη χρήση της οποίας προτείνει στους χρήστες του, ενώ διατηρεί ακόμα υποστήριξη για μια πιο παλιά και λιγότερο δυναμική γλώσσα, την RDQL.

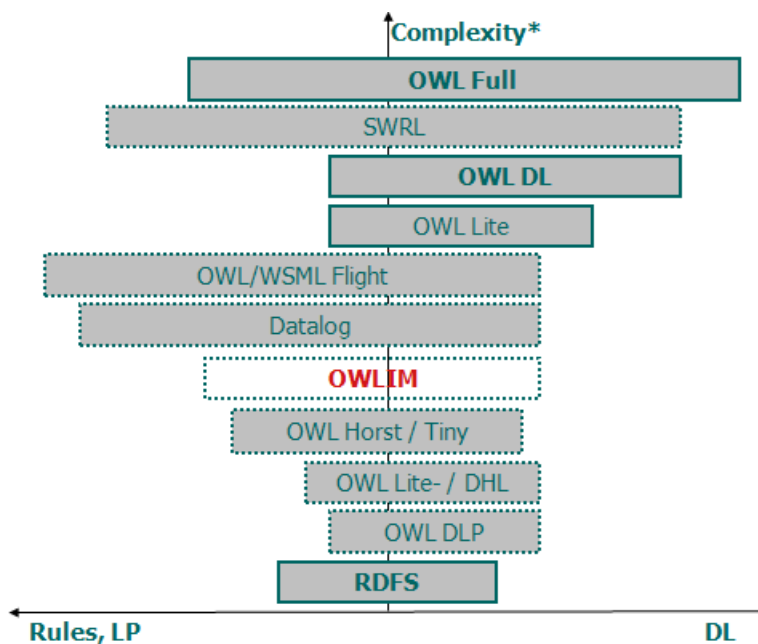
4.5.3. OWLim

Το OWLim [Kiryakov, Ognyanov, & Manon, 2005] είναι ένα πακέτο λογισμικού που βασίζεται στη πλατφόρμα Sesame, για την αποθήκευση, διαχείριση και επερώτηση οντολογιών γραμμένων σε OWL. Στην πραγματικότητα το OWLim αποτελεί μια υλοποίηση του Επιπέδου Αποθήκευσης και Συμπερασμού (SAIL) για το περιβάλλον Sesame, με έμφαση στην υποστήριξη του συντακτικού της OWL και στην παροχή υπηρεσιών συλλογισμού για τη γλώσσα αυτή. Στα βασικά πλεονεκτήματα του OWLim συγκαταλέγονται τα χαρακτηριστικά της διαχρονικότητας, της κλιμακωσιμότητας για μεγάλο όγκο δηλώσεων και της ταχύτητας απόκρισης των ερωτημάτων, στοιχεία ιδιαίτερα επιθυμητά σε ένα περιβάλλον παραγωγής.

Για την υποστήριξη του συλλογισμού το OWLim βασίζεται στο TRREE (Triple Reasoning and Rule Entailment Engine), το οποίο είναι ένας μηχανισμός συμπερασμού βασισμένος σε

κανόνες. Το TRREE πραγματοποιεί συλλογισμό εφαρμόζοντας προς τα εμπρός ακολουθία εκτέλεσης και υλοποιώντας *πλήρη υλικοποίηση* (total materialization). Η προς τα εμπρός ακολουθία εκτέλεσης ξεκινά με τα δηλωμένα στην οντολογία γεγονότα και μπορεί να υπολογίσει την εγκυρότητα ενός συγκεκριμένου ερωτήματος, να συμπεράνει ένα συγκεκριμένο τμήμα της γνώσης (π.χ. μόνο την ταξινομία) ή να υπολογίσει όλους τους πιθανούς συμπερασμούς (*κλειστότητα συμπερασμού*). Ο επανυπολογισμός της κλειστότητας του συμπερασμού ύστερα από την τροποποίηση της βάσης γνώσης (προσθήκη, αλλαγή ή διαγραφή γεγονότων) ονομάζεται *πλήρης υλικοποίηση*. Η στρατηγική αυτή των συμπερασμών έχει ως αποτέλεσμα:

- Η προσθήκη νέων γεγονότων είναι σχετικώς αργή, γιατί απαιτεί την επέκταση της κλειστότητας συμπερασμού, όχι όμως τον πλήρη επανυπολογισμό της, αφού χρησιμοποιείται προς τα εμπρός ακολουθία εκτέλεσης.
- Η διαγραφή γεγονότων είναι αργή, αφού απαιτείται κάθε φορά ο πλήρης επανυπολογισμός της κλειστότητας, γιατί μπορεί να υπάρχουν συμπερασμοί που να μην ισχύουν πλέον.
- Οι λειτουργίες επερώτησης και ανάκτησης είναι γρήγορες (συγκρίσιμες με ένα ΣΔΒΔ), δεδομένου ότι το προϊόν του συλλογισμού είναι ήδη άμεσα διαθέσιμο.



Εικόνα 4.8: Κατάταξη της εκφραστικότητας του OWLIm.

Το γεγονός ότι το TRREE είναι βασισμένο σε κανόνες σημαίνει ότι, για την παροχή συλλογισμού με οντολογικά έγγραφα, απαιτείται εκ των προτέρων να είναι διαθέσιμα κατάλληλα σύνολα κανόνων. Τα προκαθορισμένα σύνολα κανόνων για το OWLIm υποστηρίζουν την πλήρη εκφραστικότητα του RDFS, καθώς και ένα υποσύνολο της OWL, γνωστό ως *OWL Horst*. Η ταξινόμηση της εκφραστικότητας του OWLIm ως προς τις διάφορες διαλέκτους της OWL και των σχετικών με αυτήν γλωσσών φαίνεται στην εικόνα 4.8.

Το OWLIm διατίθεται σε δύο εκδόσεις: Το ελεύθερα διαθέσιμο SwiftOWLIm και την εμπορική εκδοχή BigOWLIm. Το SwiftOWLIm πραγματοποιεί συλλογισμό και αποτίμηση των

ερωτημάτων απευθείας μέσα στη μνήμη, δηλαδή συντηρεί το οντολογικό μοντέλο, πλήρως εκτεταμένο και με τις προκύπτουσες συνεπαγωγές, μέσα στην κύρια μνήμη του συστήματος. Παράλληλα, αποθηκεύει εφεδρικά αντίγραφα της εκάστοτε κατάστασης του αποθέματος ως N-Triples, αλλά όχι και την κλειστότητα των συμπερασμών. Αυτό σημαίνει ότι η αρχικοποίηση απαιτεί συντακτική επεξεργασία των N-Triples και επανυπολογισμό της κλειστότητας. Αντίθετα, το BigOWLIm αποθηκεύει, διαχειρίζεται και ενημερώνει το εκτεταμένο μοντέλο με διαχρονικό τρόπο, με τη χρήση δυαδικών αρχείων. Επιπλέον, προσφέρει μια σειρά βελτιστοποιήσεων για το συμπέρασμα (όπως για παράδειγμα, η ειδική μεταχείριση της έκφρασης owl:sameAs).

4.5.4. Υποστήριξη Διαχρονικότητας και Περιορισμοί

Για λόγους κλιμακωσιμότητας και επιδόσεων είναι προτιμότερο το οντολογικό μοντέλο να μην ταξινομείται κάθε φορά που υποβάλλεται κάποιο ερώτημα, αλλά μόνο όταν υπάρχουν αλλαγές ή ανά τακτά χρονικά διαστήματα. Αυτό είναι ιδιαίτερα αληθές, λαμβάνοντας υπόψη την πολυπλοκότητα των αλγορίθμων συλλογισμού (για παράδειγμα, 3-NEXP). Επομένως, το πλήρως εκτεταμένο μοντέλο είναι σκόπιμο να αποθηκεύεται είτε στο σύστημα αρχείων είτε σε κάποιο σχεσιακό σχήμα, παρέχοντας έτσι διαχρονικότητα. Τέτοιες υπηρεσίες φαίνεται ότι υποστηρίζονται και από τα τρία εξεταζόμενα συστήματα Jena, Sesame και OWLim. Εντούτοις:

- Το OWLim βασίζεται σε συγκεκριμένη μονάδα συλλογισμού (TRREE) που υποστηρίζει συλλογισμό για ένα μικρό υπερσύνολο της OWL Lite. Από την άλλη όμως, το OWLim έχει το επιθυμητό χαρακτηριστικό της υλικοποίησης, που αυξάνει την ταχύτητα απόκρισης των ερωτημάτων.
- Αντίστοιχο συλλογισμό περιορισμένης έκτασης υποστηρίζουν και τα άλλα δύο συστήματα, με το Sesame να περιορίζεται επιπρόσθετα μόνο στο RDFS.
- Για την υποστήριξη πιο εκφραστικών συμπερασμών, μόνο το Jena μπορεί να συνδεθεί με κάποιον εξωτερικό μηχανισμό μέσω DIG ή απ' ευθείας μέσα στη μνήμη (π.χ. Pellet). Όμως, το Jena δεν είναι ακόμα συμβατό με τις επεκτάσεις της OWL 1.1.
- Η SPARQL, τη δεδομένη χρονική στιγμή, υποστηρίζεται μόνο από τα Jena και Sesame2. Το OWLim περιλαμβάνει την υποστήριξη για SPARQL ως μελλοντική εργασία.

4.6. Αποδοτικότητα και Εκφραστικότητα

Δύο είναι οι κύριες παράμετροι που μας απασχολούν κατά τον σχεδιασμό και την εφαρμογή μιας μεθοδολογίας ανακάλυψης γνώσης για το Σημαντικό Ιστό: Η εκφραστικότητα και η αποδοτικότητα.

Ως εκφραστικότητα εννοούμε τη δυνατότητα αξιοποίησης όσο το δυνατόν πιο πλούσιων και εκφραστικών περιγραφών, όχι με την ποσοτική, αλλά με την ποιοτική έννοια, δηλαδή τη δυνατότητα χρήσης μοντέλων που αναπαριστούν όσο το δυνατόν πιο «λεπτεπίλεπτα» την εννοιολογική σύλληψη.

Υπό την έννοια αυτή η OWL DL είναι πιο εκφραστική από την OWL Lite και η OWL 1.1 είναι πιο εκφραστική από την OWL DL. Ωστόσο, υπό την ίδια έννοια, το RDF(S) δεν είναι πιο εκφραστικό από την OWL, γιατί, ενώ επιτρέπει αυθαίρετες περιγραφές, δεν διαθέτει τους σημασιακούς κατασκευαστές της OWL, όπως π.χ. οι περιορισμοί αριθμού.

Ως αποδοτικός ορίζεται στη σχετική με την πολυπλοκότητα βιβλιογραφία ένας συνεπής και πλήρης αλγόριθμος ο οποίος τερματίζει σε πολυωνυμικό χρόνο και άρα εδώ η αποδοτικότητα νοείται από την άποψη της πολυπλοκότητας.

Όμως, για το συλλογισμό με Λογικές Περιγραφής μια τέτοια κατάσταση είναι εξωπραγματική, όπως θα φανεί στη συνέχεια. Είναι παρόλα αυτά ενδιαφέρον να εξεταστούν οι παράμετροι εκείνες και οι περιορισμοί στην αποδοτικότητα της διαδικασίας ανακάλυψης γνώσης, αλλά και να εξακριβωθεί η σχέση μεταξύ εκφραστικότητας και απόδοσης. Εξάλλου, εκτός από την πολυπλοκότητα του αλγορίθμου συλλογισμού και του εκάστοτε εργαλείου που τον υλοποιεί, υπάρχουν και άλλοι παράγοντες που επηρεάζουν την αποδοτικότητα, όπως η επιβάρυνση που θέτει η επικοινωνία μεταξύ των συστημάτων.

Στα επόμενα, δείχνεται ότι οι τεχνολογίες και η λογική θεωρία του Σημαντικού Ιστού μπορούν να συνεισφέρουν αποδοτικά και εκφραστικά στο πρόβλημα της ανακάλυψης γνώσης στο Σημαντικό Ιστό (Υπόθεση 1) στο βαθμό που επιτρέπεται από συγκεκριμένους περιορισμούς, οι οποίοι μπορούν να διαιρεθούν σε τρεις κατηγορίες: α) *θεωρητικοί* (θεωρητικοί περιορισμοί, μη αποφασιστικότητα, νέοι αλγόριθμοι), β) *οργανωτικοί* (σχεδιασμός, πολιτικές και πρότυπα) και γ) *τεχνικοί* (τεχνικοί περιορισμοί μηχανισμών, διεπαφές, επίδοση in-memory, APIs). Η διάκριση αυτή δεν είναι αυστηρή, αλλά όπως είναι φυσικό και θα φανεί στη συνέχεια, αίτια της μίας κατηγορίας επηρεάζουν περιορισμούς των άλλων δύο και αντιστρόφως: για παράδειγμα, η πολιτική που ακολουθείται για την λήψη μιας απόφασης μπορεί να δημιουργήσει μια τεχνική αδυναμία σε κάποιο σύστημα και η έλλειψη ενός ισχυρού ερευνητικού αποτελέσματος μπορεί να δημιουργήσει δισταγμό στη λήψη μιας απόφασης.

4.6.1. Θεωρητικοί περιορισμοί-Πηγές πολυπλοκότητας

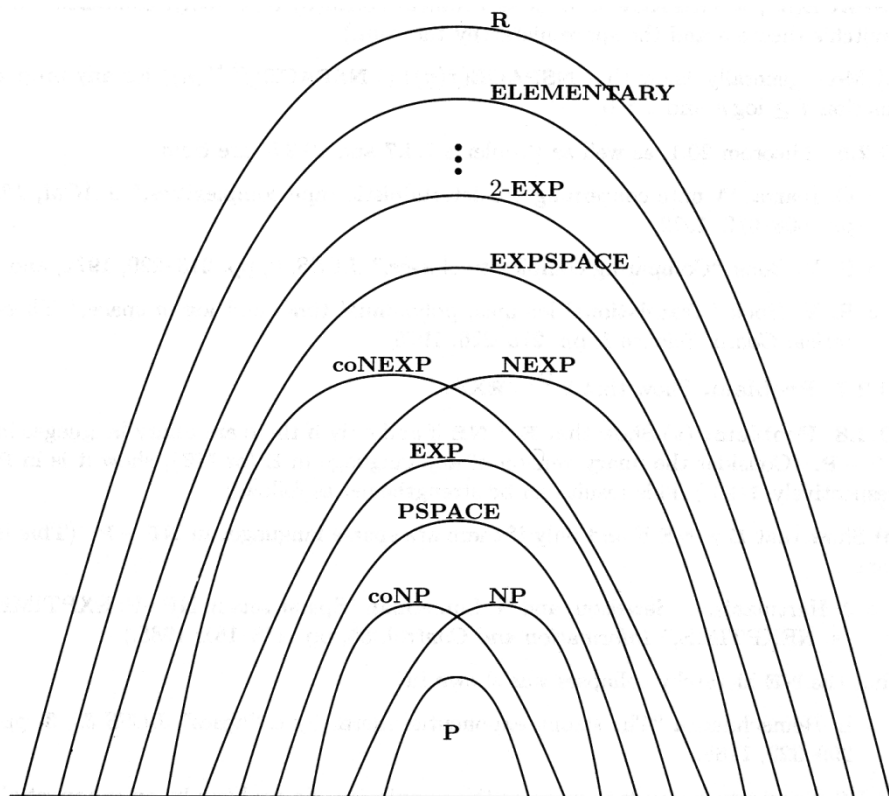
Είναι σαφές και από την υποενότητα 3.5.3 ότι η κύρια πηγή πολυπλοκότητας των αλγορίθμων ταμπλό είναι οι κανόνες που οδηγούν σε μη ντετερμινιστική επέκταση του γραφήματος πλήρωσης. Ο επόμενος πίνακας παρουσιάζει συνοπτικά τους κανόνες αυτούς και το αποτέλεσμα που έχουν στην πορεία του αλγορίθμου για την ικανοποιησιμότητα μιας βάσης γνώσης *SHOIQ*.

Ο μη ντετερμινισμός αντιμετωπίζεται στη γενική περίπτωση δοκιμάζοντας μία προς μία όλες τις μη ντετερμινιστικές εκδοχές, ώστε να προκύψει ένας υλοποιήσιμος ντετερμινιστικός αλγόριθμος. Στην περίπτωση του ταμπλό παρουσιάστηκαν (βλ. ενότητα 3.6) ορισμένες τεχνικές βελτιστοποίησης, με κυριότερη αυτή της υπαναχώρησης, όπου σε συγκεκριμένες μόνο περιπτώσεις, μπορεί να εξοικονομηθεί ένα σοβαρό τμήμα από το μη ντετερμινιστική επέκταση. Θεωρητικά όμως, κάθε τάξη μη ντετερμινισμού προσθέτει και μία εκθετική τάξη στον υλοποιήσιμο αλγόριθμο, όπως επιβεβαιώνεται και από το θεώρημα της χρονικής ιεραρχίας για τις κλάσεις πολυπλοκότητας [Paradimitriou, 1995], βάσει του οποίου $NEXPC2-EXPC2-NEXPC\dots$ και μάλιστα *γνήσια* υποσύνολα (εικόνα 4.9).

Κανόνας	Αποτέλεσμα
LI-κανόνας	Εισάγει μια μη ντετερμινιστική διάζευξη, κάθε φορά που συναντάται ο τελεστής ένωσης (LI) στην ετικέτα ενός κόμβου. Θα προκαλέσει τη δημιουργία 2 διαφορετικών μονοπατιών.
ch-κανόνας	Εισάγει μια μη ντετερμινιστική διάζευξη, όταν συναντάται περιορισμός αριθμού $\leq nS.C$. Θα προκαλέσει τη δημιουργία 2 διαφορετικών μονοπατιών.
NN-κανόνας	Απαιτεί το «μάντεμα» του ακριβούς πλήθους m των πληρωτών του ρόλου R . Αν η πληθικότητα στον περιορισμό είναι n τότε είναι πιθανό να χρειαστεί η δοκιμή n διαφορετικών μονοπατιών, καθένα από τα οποία δημιουργεί m διακλαδώσεις στο γράφημα.

Πίνακας 4.4: Οι μη ντετερμινιστικοί κανόνες στο ταμπλό της *SHOIQ*.

Έτσι, παρόλο που οι διαδικασίες απόφασης για *SHOIQ* και για *SROIQ* που έχουν παρουσιαστεί στη βιβλιογραφία απέχουν από το βέλτιστο, με την έννοια ότι οι αντίστοιχοι αλγόριθμοι δεν συμβαδίζουν απολύτως με τα αντίστοιχα θεωρητικά αποτελέσματα, δεν θα πρέπει να αναμένεται δραματική βελτίωση ακόμα και σε πλήρως βελτιστοποιημένους αλγόριθμους. Εξάλλου, τα παραπάνω αφορούν στη χειρότερη περίπτωση η οποία σπάνια φαίνεται ότι είναι και συνήθης, τουλάχιστον όσον αφορά τις οντολογίες Ιστού [Horrocks I., 2007].



Εικόνα 4.9: Η χρονική ιεραρχία των κλάσεων πολυπλοκότητας.

Ένα ακόμα σημείο που χρίζει προσοχής είναι ότι, δεν οφείλεται στους κανόνες ενός ταμπλό που, για εκφραστικές Λογικές Περιγραφής, απαιτούνται μη ντετερμινιστικές διαδικασίες απόφασης. Ο μη ντετερμινισμός φαίνεται ότι είναι εγγενής σε τέτοιες λογικές, όπως και αν αντιμετωπιστεί το πρόβλημα του συλλογισμού, είτε δηλαδή με αλγόριθμους

ταμπλό, είτε με μεθόδους ανάλυσης, είτε με (μερική) αναγωγή σε κανόνες και Datalog. Το γεγονός αυτό επιβεβαιώνεται από το ότι τόσο τα αντίστοιχα θεωρητικά αποτελέσματα πολυπλοκότητας [Tobies, 2001; Horrocks, Kutz, & Sattler, 2006], όσο και η πολυπλοκότητα της βασισμένης σε ανάλυση διαδικασίας απόφασης για την *SHOIQ* [Kazakov & Motik, 2006] καθώς και για Datalog συγκλίνουν στο θέμα του μη ντετερμινισμού.

Κατά τη διαδικασία σχεδιασμού και υλοποίησης μιας οντολογίας, είναι επομένως χρήσιμο να ληφθούν υπόψη οι συνθήκες που πυροδοτούν τέτοιους κανόνες, ώστε κατά το δυνατόν να αποφευχθούν. Κατά την επαύξηση της εκφραστικής ισχύος ενός οντολογικού μοντέλου, όπως συμβαίνει με την τεχνική της *σημασιακής προσαρμογής* (βλ. κεφ. 5) επιτυγχάνεται εντασιακή επέκταση του εκάστοτε μοντέλου με τη προσθήκη νέων κατασκευαστών και περιορισμών τέτοιων, ώστε να προσφέρεται υψηλή εκφραστικότητα με μικρό σχετικά κόστος, εφόσον οι παρεμβάσεις γίνονται στοχευμένα.

Ο πίνακας 4.5 συνοψίζει τα θεωρητικά και πρακτικά αποτελέσματα για την πολυπλοκότητα του συλλογισμού στις διάφορες εκδοχές της OWL και επιβεβαιώνει το (αναμενόμενο, βλ. και [Brachman & Levesque, 2004]) αντιθετικό ισοζύγιο που υπάρχει ανάμεσα στην αποδοτικότητα και την εκφραστικότητα. Το αποτέλεσμα της πληρότητας για την *SROIQ* πρόκειται να ανακοινωθεί [Kazakov Y. , 2008], ενώ τα αποτελέσματα για τις άλλες λογικές περιλαμβάνονται στο [Tobies, 2001].

Είδος OWL	Αντίστοιχη Λογική	Θεωρητική Πολυπλοκότητα	Πολυπλοκότητα Διαδικασίας Απόφασης (χειρ. περίπτωση)
OWL Lite	SHIF(D)	Exp-complete	Exp
OWL DL	SHOIN(D)	NExp-complete	2-NExp (3-Exp)
OWL 1.1	SROIQ(D)	2NExp-complete	3-NExp (4-Exp) ¹

Πίνακας 4.5: Η πολυπλοκότητα του συλλογισμού στην OWL.

Αξίζει τέλος να σημειωθεί ότι, στον παραπάνω πίνακα, το βέλτιστο αποτέλεσμα για την OWL Lite προέρχεται από την αναγωγή της *SHIQ* σε Datalog [Hustadt, Motik, & Sattler, 2004] που υλοποιείται στο KAON2 και όχι από αλγόριθμο ταμπλό (όπως π.χ. στο FaCT), που έχει πολυπλοκότητα χειρότερης περίπτωσης 2-NExp [Horrocks & Sattler, 1999]. Εντούτοις, για μεγάλα TBox, ο ταμπλό αλγόριθμος συμπεριφέρεται καλύτερα, λόγω των βελτιστοποιήσεων.

4.6.2. Οργανωτικοί περιορισμοί

Ήδη, λόγω των θεωρητικών περιορισμών και των άφθονων πηγών πολυπλοκότητας στο συλλογισμό με οντολογίες Ιστού προκύπτει ότι το πρόβλημα της SWKD απέχει μακράν από το να λύνεται αποδοτικά, δηλαδή σε πολυωνυμικό χρόνο. Εκτός όμως από τις θεωρητικές παραμέτρους, η αξιοποίηση των τεχνολογιών του Σημαντικού Ιστού για ανακάλυψη γνώσης εξαρτάται επίσης και από μια σειρά περιπτωσιακών συνιστωσών, που άπτονται οργανωτικού χαρακτήρα: για παράδειγμα, οι σχεδιαστικές αποφάσεις και οι πολιτικές προτυποποίησης που έχουν ληφθεί και ακολουθούνται στην πορεία εξέλιξης των οντολογικών γλωσσών, των διεπαφών προσπέλασης και επερώτησης, καθώς και των διαθέσιμων μηχανισμών συμπερασμού σίγουρα τέμνουν την μεθοδολογία ανακάλυψης γνώσης στο Σημαντικό Ιστό και καθορίζουν το βαθμό υλοποίησής της.

¹ Εφόσον η κατασκευή των αυτομάτων **δεν** θεωρηθεί μέρος της εισόδου («βήμα προεπεξεργασίας»).

Διεπαφές και Πρωτόκολλα

Με τις πρώτες προσπάθειες τυποποίησης μιας οντολογικής γλώσσας για τον Παγκόσμιο Ιστό, όπως η DAML+OIL και αργότερα η OWL, εμφανίστηκαν και αντίστοιχες προτάσεις για τον τρόπο που οι γλώσσες αυτές επρόκειτο να αξιοποιηθούν, μέσω μηχανισμών επερώτησης. Αυτό είχε ως αποτέλεσμα την πρόταση στη βιβλιογραφία των γλωσσών DQL (DAML Query Language) και OWL-QL [Fikes, Hayes, & Horrocks, 2002; Fikes, Hayes, & Horrocks, 2003], ως πρωτοκόλλων αρμόδιων για τη ανταλλαγή πληροφοριών και ερωταποκρίσεων ως προς οντολογικά έγγραφα. Εντούτοις, οι πολλά υποσχόμενες προδιαγραφές αυτές δεν φαίνεται να βρήκαν το δρόμο τους σε κάποια αξιοσημείωτη υλοποίηση, πόσο μάλλον στην παραγωγή, πιθανότατα για τους λόγους που εντοπίζονται στο [Koutsomitropoulos, Fragakis, & Paratheodorou, 2006b]. Αντίθετα, χρειάστηκε να περάσει κάποιος καιρός για την προτυποποίηση μιας γλώσσας επερώτησης για το Σημαντικό Ιστό, που όμως βασίζεται γενικά στο RDF και όχι στην OWL, τη SPARQL [Prud'hommeaux & Seaborne, 2008].

Η γλώσσα αυτή, που προήλθε μέσα από έναν επιστημονικό ανταγωνισμό ανάμεσα σε διαφορετικές υποψηφιότητες για προτυποποίηση (βλ. για παράδειγμα τη σύγκριση μεταξύ SPARQL και RDQL [Hutt, 2005]), αναδείχθηκε οριστικά σε πρότυπο του W3C μέσα από μία όχι τελειώς απροβλημάτιστη διαδικασία (βλ. και την επαναφορά της γλώσσας από υποψήφια σύσταση σε κείμενο εργασίας <http://www.w3.org/TR/2006/WD-rdf-sparql-query-20061004/>). Η SPARQL, ενώ πετυχαίνει να ορίσει έναν τυπικό τρόπο σχηματισμού και υποβολής συζευγμένων (conjunctive) ερωτημάτων σε RDF, αφήνει εκτός εύρους τις δυνατότητες του συμπερασμού, τα ταξινομικά ερωτήματα και τις ιδιομορφίες των κατ'εξοχήν οντολογικών γλωσσών OWL και OWL 1.1. Είναι επομένως λογικό το έλλειμμα αυτό να αναγνωρίζεται και να οδηγεί στην προσπάθεια προσαρμογής της γλώσσας αυτής συγκεκριμένα για την OWL DL (SPARQL-DL) και τελικά την OWL 1.1 (SPARQL-OWL) [Sirin & Parsia, 2007].

Επιπρόσθετα, το πρότυπο DIG, ένα πολύ διαδεδομένο XML πρωτόκολλο ερωταποκρίσεων για συστήματα Λογικών Περιγραφής, δεν εξελίχθηκε έγκαιρα ώστε να συμπεριλάβει την πλήρη εκφραστικότητα του Σημαντικού Ιστού, όπως αυτή τυποποιήθηκε στην OWL. Η τελευταία επίσημη έκδοση 1.1 για παράδειγμα, ενώ υποστηρίζεται από όλους σχεδόν τους σύγχρονους μηχανισμούς συμπερασμού, δεν μπορεί να χειριστεί τα ονοματικά [Dickinson, 2004]. Η έκδοση 2.0, που θα αντιμετώπιζε αυτά τα προβλήματα, βρίσκεται εδώ και αρκετό καιρό *in flux* [Horridge, Tsarkov, & Redmond, 2006] και μόνο πρόσφατα ανακοινώθηκε η σύσταση ομάδας εργασίας που θα την προδιαγράψει, ώστε να υποστηρίξει ακόμα και την OWL 1.1 [Bechhofer S., 2006].

Ως αποτέλεσμα των παραπάνω ανακολουθιών, είναι επόμενο:

- Οι μέχρι τώρα μεθοδολογικές προσεγγίσεις για επερώτηση οντολογικών εγγράφων να είναι πολύ περιορισμένες.
- Οι υλοποιήσεις που προσπαθούν να είναι συμβατές με κάποιο από τα πρωτόκολλα είτε υστερούν στην απόδοση, λόγω της επικοινωνιακής επιβάρυνσης (overhead), είτε σε εκφραστικότητα, λόγω της μη επικαιροποίησης.
- Συχνότερη είναι η εμφάνιση ad-hoc τεχνικών, που όμως δεν βοηθούν στη διαλειτουργικότητα.

Η *de facto* αυτή κατάσταση υπήρξε ακριβώς το έναυσμα για μια από τις πρώτες προσπάθειες πρότασης μιας μεθοδολογίας SWKD [Koutsomitropoulos, Fragakis, & Paratheodorou, 2005].

Σχεδιασμός γλωσσών

Μια μεθοδολογία ανακάλυψης γνώσης στο Σημαντικό Ιστό είναι φυσικό να εξαρτάται και από τα εκφραστικά μέσα που παρέχονται για την αναπαράσταση και επικοινωνία της κατανεμημένης στο Διαδίκτυο πληροφορίας. Ενώ τέτοια μέσα μπορεί να υπάρχουν αρκετά ως αποτέλεσμα ιδιωτικών προσπαθειών, έχει τελειώς διαφορετική σπουδαιότητα μια γλώσσα που προτείνεται ως σύσταση από έναν διεθνή οργανισμό, όπως το W3C.

Οι πολιτικές λήψης αποφάσεων και τα καταστατικά των ομάδων εργασίας για την προδιαγραφή της OWL έχουν επομένως ιδιαίτερη σημασία, καθώς επηρεάζουν τον βαθμό υιοθέτησης από την ερευνητική κοινότητα και τη βιομηχανία λογισμικού. Για παράδειγμα, υπάρχουν περιπτώσεις που η γλώσσα αποκτά ή απορρίπτει χαρακτηριστικά ύστερα από διεξαγωγή ψηφοφορίας μεταξύ των μελών της ομάδας εργασίας. Ένα τέτοιο χαρακτηριστικό είναι π.χ. η καθιέρωση σημασιολογίας για το χαρακτηριστικό της παρονομασίας, μια ιδιότητα εν εξελίξει της OWL 1.1, η οποία έχει συμφωνηθεί προς το παρόν να παραμείνει μια συντακτική σύμβαση.

Αξίζει επίσης να σημειωθεί ότι η OWL 1.1 με τα πολλά υποσχόμενα νέα εκφραστικά χαρακτηριστικά, μόλις τώρα μπαίνει στο δρόμο της τυποποίησης ενώ η OWL (1.0) είναι επίσημη σύσταση εδώ και κάποια χρόνια. Παρόλα αυτά, έχει ενδιαφέρον ότι η νέα γλώσσα προτίθεται να ενσωματώσει δυνατότητες οι οποίες, ενώ είχαν εξεταστεί ήδη από την εποχή της αρχικής OWL, δεν υλοποιήθηκαν ή αναβλήθηκαν (βλ. OWL Working Group Charter, postponed issues, <http://www.w3.org/2007/06/OWLCharter>), λόγω αιτιών με ξεκάθαρη πολιτική διάσταση, όπως: έλλειψης χρόνου (το καταστατικό και η κοινότητα πίεζε για πρότυπο, οι κριτικές ήταν έντονες), έλλειψης αλγορίθμων συλλογισμού και υποστηρικτικών εργαλείων τη δεδομένη στιγμή, καθώς και μη συναίνεση των μελών της ομάδας εργασίας σε όλα τα ζητήματα. Οι νέες αυτές δυνατότητες, εκτός από το γεγονός ότι πλέον υποστηρίζονται θεωρητικά και αναμένεται να υλοποιηθούν με επέκταση των υπαρχόντων εργαλείων, έχουν απαιτηθεί από «μειζονες χρήστες της OWL» [Patel-Schneider & Horrocks, 2006] και δεν είναι κρυφό ότι αυτοί προέρχονται από την ιατρική κοινότητα και τη φαρμακευτική βιομηχανία.

Ακόμη, η τρέχουσα πρόταση για την Γλώσσα Οντολογιών Ιστού, ως μέρος της επίσημης αρχιτεκτονικής του Σημαντικού Ιστού, αφήνει έξω τις εκφραστικές δυνατότητες του λογικού προγραμματισμού και του μη-μονοτονικού συλλογισμού. Η τοποθέτηση του επιπέδου των κανόνων, δηλαδή της SWRL, ως επέκταση της OWL (που με τη σειρά της βασίζεται στις Λογικές Περιγραφές) και όχι παράλληλα (όπως θα εκφραζόταν από την WRL), έχει θεωρηθεί αδικαιολόγητη εμμονή και έχει οδηγήσει σε διαμάχη [Kifer, De Bruijn, Boley, & Fensel, 2005; Horrocks, Parsia, Patel-Schneider, & Hendler, 2005].

Διαθεσιμότητα μηχανισμών συμπερασμού

Τα υποστηρικτικά εργαλεία για την OWL, με έμφαση σε αυτά που παρέχουν υπηρεσίες συλλογισμού, η κατάσταση ανάπτυξής τους και η διαθεσιμότητά τους είναι κρίσιμης σημασίας για την ανακάλυψη γνώσης. Ερευνητικά λογισμικά για Λογικές Περιγραφές όπως

το FaCT, πήρε κάποιο χρόνο και αρκετό χρήμα για να προσαρμοστούν στον κόσμο των οντολογιών Ιστού και να εξελιχθούν για να συμπεριλάβουν την πλήρη εκφραστικότητα (βλ. FaCT++). Η προσπάθεια αποδέσμευσής τους από τα στενά ακαδημαϊκά πλαίσια, ως μέσο για δημιουργία προσόδων δεν τελεσφόρησε πάντα: Το RACER Pro, παρόλο που ακολούθησαν τρεις εκδόσεις από την εμπορευματοποίησή του, δεν υποστηρίζει ακόμα ονοματικά, το CEREBRA πουλήθηκε στην WebMethods και δεν είναι πια διαθέσιμο ούτε υποστηρίζεται αυτόνομα και το περιβάλλον SNOBASE της IBM δεν φαίνεται να βρήκε το δρόμο του σε κάποιο προϊόν.

Οι λόγοι για την κατάσταση αυτή συνοψίζονται κυρίως σε τρία σημεία:

- Τα εργαλεία αυτά εξαρτώνται από μια ρευστή, όπως φάνηκε, διαδικασία τυποποίησης, που συνεχώς εξελίσσεται και δεν αφήνει περιθώρια για δημιουργία εμπορικής αξίας.
- Πρόκειται για τεχνολογίες ερευνητικής αιχμής και δεν υπήρχε αγορά διαθέσιμη ή αρκετά ώριμη για να τα υποδεχτεί.
- Η τεχνολογική εξέλιξη που είναι αναγκαία για την επιτυχία τέτοιων συστημάτων εξαρτάται σε μεγάλο βαθμό από την πανεπιστημιακή έρευνα σε διεθνές επίπεδο, η οποία μπορεί να κεφαλαιοποιηθεί μόνο με την προϋπόθεση του ανοιχτού κώδικα.

Αντίθετα, το σύστημα Pellet για παράδειγμα, υποστήριξε ήδη την πλήρη εκφραστικότητα της OWL DL, πριν ακόμα αποδεσμευθεί από τους κόλπους του UMBC. Η εταιρεία που πλέον έχει αναλάβει τη συντήρηση και επικαιροποίησή του, αφήνει το λογισμικό και τον κώδικά του ελεύθερα διαθέσιμα και προωθεί ως προϊόν τις υπηρεσίες υποστήριξής του, ακολουθώντας ένα επιχειρηματικό μοντέλο που φαίνεται να κερδίζει έδαφος τα τελευταία χρόνια (βλ. π.χ. τη Linux διανομή Ubuntu). Ως αποτέλεσμα, το Pellet έχει γρήγορα βρει θέση ως μέρος άλλων δημοφιλών εμπορικών προϊόντων όπως το TopBraid Composer.

4.6.3. Τεχνικοί περιορισμοί

Όπως φάνηκε στην ενότητα 4.4, τα μόνα από τα εξεταζόμενα συστήματα που υποστηρίζουν την πλήρη εκφραστικότητα της OWL (DL) είναι τα FaCT++ και Pellet. Ταυτοχρόνως είναι και τα μόνα που επεκτείνονται για την υποστήριξη της εξελισσόμενης OWL 1.1. Από την άλλη, το σύστημα RACER, παρόλο που υποστηρίζει πλήρως μόνο το *SHIQ* υποσύνολο της OWL, θεωρείται ότι παρέχει ιδιαίτερα βελτιστοποιημένες υπηρεσίες συλλογισμού με στιγμιότυπα (ABox) και απτά πεδία, όπως ακέραιοι και πραγματικοί αριθμοί.

Αν όμως πρώτο μέλημα είναι η εκφραστικότητα, θα μπορούσε κανείς αφελώς να συμπεράνει ότι το σύστημα RACER δεν έχει πλέον καμία αξία, εφόσον, αφενός δεν υποστηρίζει πλήρως την OWL και αφετέρου υπάρχουν ήδη διαθέσιμα συστήματα που το κάνουν. Μια όμως πιο προσεκτική μελέτη των δύο παραπάνω συστημάτων αποκαλύπτει ότι έχουν υποστήριξη μόνο για την έκδοση 1.1 του πρωτοκόλλου DIG (κάτι που είναι λογικό, αφού η έκδοση 2.0 δεν έχει οριστικοποιηθεί ακόμα) το οποίο επικοινωνεί μέσω HTTP και συγχρόνως *δεν παρέχουν καμιά άλλη συνδεσιμότητα*, τύπου TCP/IP. Επομένως, για να μπορούν αυτοί οι μηχανισμοί συλλογισμού να αξιοποιηθούν από ένα άλλο εργαλείο ή μέσα σε μια ολοκληρωμένη εφαρμογή, είναι αναγκαίο να χρησιμοποιηθεί κάποιο προγραμματιστικό API, για παράδειγμα το Jena ή το OWL 1.1 (τέως Manchester) API, το

οποίο θα διασυνδέει τους μηχανισμούς αυτούς απευθείας μέσα στην μνήμη (direct in-memory implementations) [Horridge, Bechhofer, & Norppens, 2007].

Η προσέγγιση αυτή μπορεί να έχει το πλεονέκτημα ότι αποφεύγει το φόρτο της ανταλλαγής μηνυμάτων που θα είχε το DIG, αλλά σίγουρα είναι ανεπαρκής για την ανάπτυξη αληθινά αποκεντρωμένων εφαρμογών και υπηρεσιών για το Σημαντικό Ιστό. Ελλείψει λοιπόν DIG 2.0, τα FaCT++ και Pellet δεν μπορούν να χρησιμοποιηθούν για την ανάπτυξη μιας κατανεμημένης διαδικτυακής υπηρεσίας για ανακάλυψη γνώσης στο Σημαντικό Ιστό, που να υποστηρίζει πλήρως την OWL DL. Ακόμα όμως και όσον αφορά την παράμετρο της αποδοτικότητας, η φυσική κατανομή σε διαφορετικά υπολογιστικά συστήματα του μηχανισμού συμπερασμού και των συνεργαζόμενων εργαλείων, μάλλον θα επιτάχυνε συνολικά την εφαρμογή, ιδιαίτερα στην περίπτωση της υψηλής εκφραστικότητας, δεδομένης της πολλαπλά εκθετικής πολυπλοκότητας (τόσο σε επεξεργαστική ισχύ, όσο και σε μνήμη).

Αυτός είναι και ο λόγος που στο κεφάλαιο 5 έχουμε υιοθετήσει το RACER ως υπόβαθρο για βασισμένο σε Λογικές Περιγραφής συλλογισμό. Το RACER υπήρξε κυρίαρχο ως προς την εκφραστικότητα και τις δυνατότητες διασύνδεσης, όταν το Pellet δεν υπήρχε καν. Τώρα το RACER, όντας ελεύθερα διαθέσιμο για μη εμπορικούς σκοπούς, είναι ο μόνος μηχανισμός συμπερασμού με την εγγύτερη στην OWL DL εκφραστικότητα, που προσφέρει μια ανεξάρτητη, λειτουργική και συμβατή με IP διεπαφή επικοινωνίας. Βέβαια, για τον συλλογισμό με τους νέους κατασκευαστές της OWL 1.1, όπως συμβαίνει στο κεφάλαιο 6, δεν υπάρχει άλλος τρόπος παρά να χρησιμοποιηθεί εσωτερικά κάποιος άλλος μηχανισμός.

4.7. Πειραματική Επαλήθευση - KDI

Η KDI είναι μια δικτυακή εφαρμογή, που παρέχει υπηρεσίες υποβολής ευφώνων ερωτημάτων σε έγγραφα οντολογιών Ιστού. Χαρακτηρίζεται ως *Διεπαφή*, για να δοθεί έμφαση στο γεγονός ότι προσφέρει στο χρήστη ένα διαισθητικό τρόπο για την σύνθεση και την υποβολή ερωτημάτων. Για τη διεξαγωγή των συμπερασμών, η KDI αλληλεπιδρά με το σύστημα RACER, το οποίο επιλέχθηκε για τους λόγους που συνοψίζονται στην ενότητα 4.6.3: Παρά την περιορισμένη εκφραστικότητά του, το RACER παρέχει τη δυνατότητα κατασκευής γύρω από αυτό κατανεμημένων εφαρμογών.

Πράγματι, από σχεδιαστικής άποψης, η KDI αποσκοπεί στο να είναι διπλά κατανεμημένη:

- i. τόσο ως προς την φυσική και λογική κατανομή των βάσεων γνώσης από τις οποίες αντλούνται οι πληροφορίες (οντολογίες - `owl:import`) όσο και
- ii. μέσω της δυνατότητας λογικής και φυσικής κατανομής των επιχειρησιακών συνιστωσών (tiers) της εφαρμογής: της τελικής διεπαφής χρήστη (browser), της επιχειρησιακής λογικής (servlets, javabeans, tomcat) και του συστήματος διαχείρισης *Βάσης Γνώσης* (μηχανισμός συμπερασμού).

Μια τέτοια πραγματικά δικτυακά κατανεμημένη αρχιτεκτονική, κατ' αναλογία του παραδοσιακού 3-tier μοντέλου, δεν είναι ακόμα εφικτή με την πλειοψηφία των πλέον πρόσφατων και εκφραστικών μηχανισμών Λογικών Περιγραφής (λόγω περιορισμών στις δυνατότητες διεπαφής τους). Ωστόσο, η προσέγγιση αυτή συνεισφέρει πιο ουσιαστικά στην

αξιοποίηση των αποτελεσμάτων από χρήστες και εφαρμογές, οδηγώντας στην εξαγωγή πιο προφανούς αξίας από τα οντολογικά δεδομένα [Hendler J. , 2008].

Η KDI έχει αναπτυχθεί ως υλοποίηση των αποφάσεων και κριτηρίων που τίθενται σε κάθε στάδιο της μεθοδολογίας. Ως τέτοια έχει τριπλή στόχευση:

- Να είναι αρκούντως *εκφραστική*, προκειμένου να επιτρέπει ισχυρούς συμπερασμούς στα οντολογικά έγγραφα.
- Να είναι *δηλωτική*, όντας ανεξάρτητη από το συγκεκριμένο σχήμα ή τα περιεχόμενα της οντολογίας.
- Να είναι *διαισθητική*, υποβοηθώντας το χρήστη στη σύνθεση του ερωτήματός του με φιλικό τρόπο.

Με τη χρήση της KDI παρέχονται, στο κεφάλαιο 5, μια σειρά από πειραματικά αποτελέσματα πάνω στο προσαρμοσμένο μοντέλο CIDOC-CRM τα οποία επαληθεύουν, μέχρι ορισμένο βαθμό, τις υποθέσεις που έχουν γίνει σχετικά κυρίως με την παράμετρο της *εκφραστικότητας*. Ταυτόχρονα δίνουν το έναυσμα για την αναζήτηση και εξέταση της υποστήριξης των νέων προτεινόμενων επεκτάσεων της λογικής θεωρίας του Σημαντικού Ιστού, δηλαδή της γλώσσας OWL 1.1.

Στη συνέχεια, γίνεται μια γενική περιγραφή της KDI και των βασικότερων τεχνολογιών που χρησιμοποιήθηκαν, μαζί με μια σύντομη επισκόπηση της λειτουργικότητάς της. Κατόπιν εξετάζεται η προγραμματιστική αρχιτεκτονική της διεπαφής, δίνοντας βαρύτητα στην τυπική σχέση μεταξύ της σύνθεσης ερωτήματος και των λειτουργιών (functions) χαμηλού επιπέδου του RACER στις οποίες αναλύεται το ερώτημα.

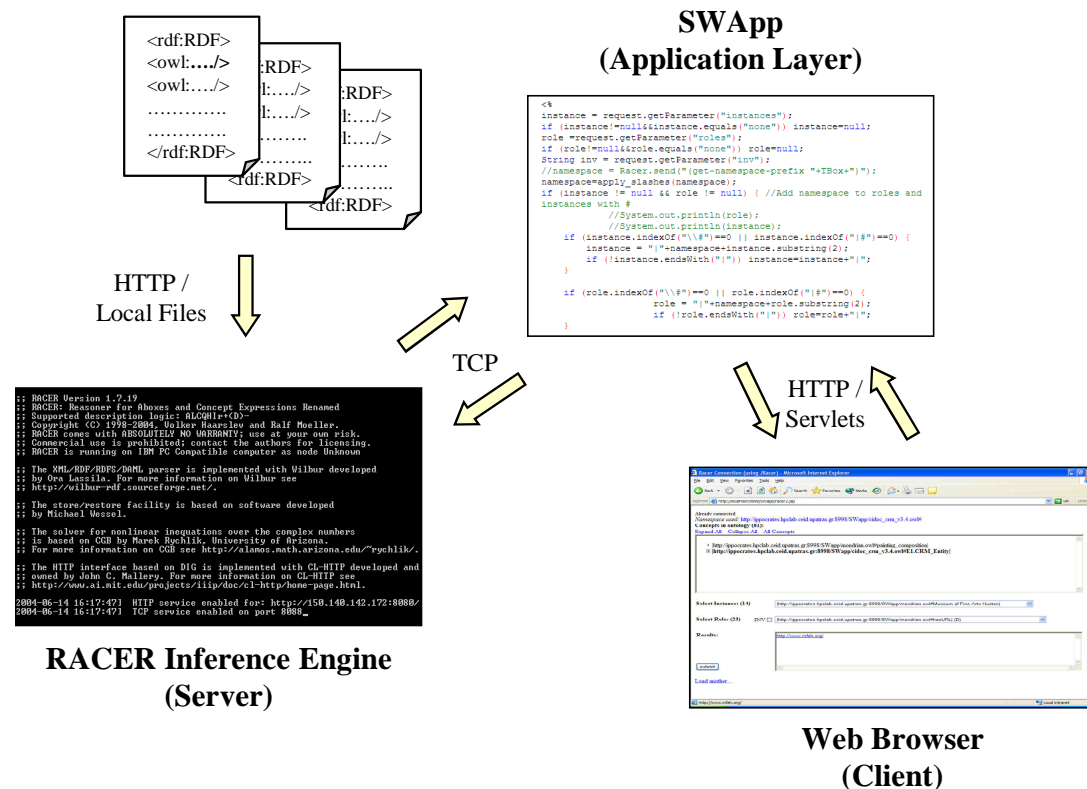
4.7.1. Γενική περιγραφή

Ο σχεδιασμός της διεπαφής ακολουθεί το παραδοσιακό μοντέλο των τριών επιπέδων (3-tier) με μια όμως σημαντική παραλλαγή: Ο ρόλος του τρίτου επιπέδου, που τυπικά αναλαμβάνει ένας εξυπηρετητής δεδομένων (δηλ. μια βάση δεδομένων), τώρα διεξάγεται από μια *βάση γνώσης* και το αντίστοιχο σύστημα διαχείρισης αυτής (βλ. εικόνα 4.10). Να σημειωθεί ότι καθένα από τα παρακάτω τρία επίπεδα μπορεί να είναι και φυσικά καταναμημένο σε διαφορετικά υπολογιστικά συστήματα.

Η διεπαφή μπορεί να φορτώσει OWL έγγραφα, τα οποία είτε είναι διαθέσιμα στο Διαδίκτυο, είτε βρίσκονται στο τοπικό σύστημα αρχείων του χρήστη. Ένα προσωρινό αντίγραφο κάθε εγγράφου αποθηκεύεται πρώτα τοπικά στον εξυπηρετητή εφαρμογής και κατόπιν φορτώνεται από τον *εξυπηρετητή βάσης γνώσης* (RACER). Για κάθε οντολογία, το RACER δημιουργεί και αποθηκεύει ένα εσωτερικό μοντέλο στη μνήμη, αφού πρώτα το ταξινομήσει, διαδικασία η οποία εκτελείται μόνο μία φορά, κατά την αρχική φόρτωση της οντολογίας. Επιπλέον, το RACER φορτώνει τυχόν άλλα έγγραφα που η αρχική οντολογία εισαγάγει, μέσω της οδηγίας `<owl:imports>`.

Η επιχειρησιακή λογική της εφαρμογής υλοποιείται με τη χρήση της γλώσσας προγραμματισμού Java και των τεχνολογιών JSP, JavaBeans και Java Servlets. Το ρόλο του *εξυπηρετητή εφαρμογής* έχει το ελεύθερα διαθέσιμο σύστημα Tomcat (έκδοση 5.0). Η επιχειρησιακή λογική κυρίως ασχολείται με την φόρτωση των εγγράφων, την κατάλληλη εμφάνιση της οντολογικής πληροφορίας στο χρήστη, τη σύνθεση και υποβολή των ερωτημάτων και το σχηματισμό και την παρουσίαση των αποτελεσμάτων. Η άντληση των

οντολογικών δεδομένων και η λήψη των αποτελεσμάτων των συμπερασμών στους οποίους αποδομείται κάθε ερώτημα, γίνονται επικοινωνώντας με το RACER, μέσω του πρωτοκόλλου TCP/IP. Η επικοινωνία αυτή διευκολύνεται σε μεγάλο βαθμό με την αξιοποίηση των κλάσεων και των μεθόδων που παρέχει η αφηρημένη προγραμματιστική διεπαφή JRacer. Η διεπαφή αυτή τροποποιήθηκε σε ορισμένα σημεία, κυρίως όσον αφορά την επεξεργασία των συνδέσμων των δικτυακών εγγράφων και την αντιμετώπιση των συνώνυμων εννοιών.



Εικόνα 4.10: Ο σχεδιασμός τριών επιπέδων της KDI.

Ο χρήστης τελικά αλληλεπιδρά με το *επίπεδο πελάτη*, που συνίσταται στην εμφάνιση JSP ιστοσελίδων από τον φυλλομετρητή του. Η επικοινωνία με το επίπεδο εφαρμογής γίνεται μέσω του πρωτοκόλλου HTTP και τη χρήση φορμών, ενώ παράλληλα χρησιμοποιούνται servlets για την διαχείριση αιτημάτων πολλαπλών χρηστών και για τον έλεγχο της ταυτόχρονης πρόσβασης. Επιπρόσθετα, όταν μια οντολογία που έχει φορτωθεί δεν χρησιμοποιείται από κανέναν χρήστη, διαγράφεται από τη μνήμη για καλύτερη διαχείριση των πόρων του συστήματος.

Η διεπαφή μπορεί να φορτώσει OWL (DL) έγγραφα που είναι διαθέσιμα είτε στο τοπικό σύστημα αρχείων είτε στο Διαδίκτυο. Μετά την επιτυχή σύνδεση με το RACER, η οντολογία φορτώνεται και οι πληροφορίες της εμφανίζονται από τον φυλλομετρητή. Ο χρήστης μπορεί να πλοηγηθεί στην ιεραρχία των κλάσεων, η οποία οπτικοποιείται σε δενδρική δομή, και να επιλέξει οποιαδήποτε από τις διαθέσιμες κλάσεις. Μόλις γίνει μια επιλογή, η σελίδα επαναφορτώνεται, εμφανίζοντας τώρα δύο μενού, όπου το ένα περιλαμβάνει όλα τα στιγμιότυπα της επιλεγμένης κλάσης και το άλλο τις ιδιότητες των οποίων το πεδίο ορισμού βρίσκεται μέσα στην κλάση αυτή. Ο χρήστης μπορεί να επιλέξει ένα στιγμιότυπο και ένα ρόλο και να υποβάλει το ερώτημά του, πατώντας το αντίστοιχο κουμπί. Υπάρχει

επίσης επιλογή για την αντιστροφή του επιλεγμένου ρόλου, καταλήγοντας έτσι σε διαφορετικό ερώτημα.

Η KDI μπορεί να φορτώσει OWL DL έγγραφα οποιασδήποτε δομής και περιεχομένου. Επιπλέον, ο χρήστης έχει τη δυνατότητα πλοήγησης μεταξύ των διαθέσιμων κλάσεων, των στιγμιότυπων τους και των αντίστοιχων ιδιοτήτων. Με τον τρόπο αυτό, όλες οι αναγκαιές για τη σύνθεση ενός ερωτήματος πληροφορίες καθίστανται διαθέσιμες. Μια τέτοια *δηλωτική* συμπεριφορά είναι κρίσιμης σημασίας για τη διαδικασία ανακάλυψης γνώσης στο Σημαντικό Ιστό: ο χρήστης θα πρέπει να μπορεί να υποβάλει ερωτήματα σε οποιαδήποτε οντολογία, ακόμη και σε αυτές που συναντά για πρώτη φορά.

Η διεπαφή βοηθά το χρήστη να συνθέσει ένα ερώτημα επιλέγοντας μια κλάση, ένα στιγμιότυπο και ένα ρόλο, με φιλικό τρόπο. Αφού το ερώτημα συντεθεί, αναλύεται στη συνέχεια σε αρκετές λειτουργίες χαμηλού επιπέδου (αιτήματα συλλογισμού) που υποβάλλονται κατόπιν στο RACER. Η διαδικασία αυτή είναι εντελώς διαφανής προς το χρήστη, αποκρύπτοντας τις λεπτομέρειες της πραγματικής επερώτησης προς τη βάση γνώσης. Αυτή η *διαισθητική* σύνθεση ευφύων ερωτημάτων, βελτιώνει σε μεγάλο βαθμό τη διαδικασία ανακάλυψης γνώσης, διευκολύνοντας το χρήστη να θέτει ακριβή και ορθά ερωτήματα.

4.7.2. Αρχιτεκτονική

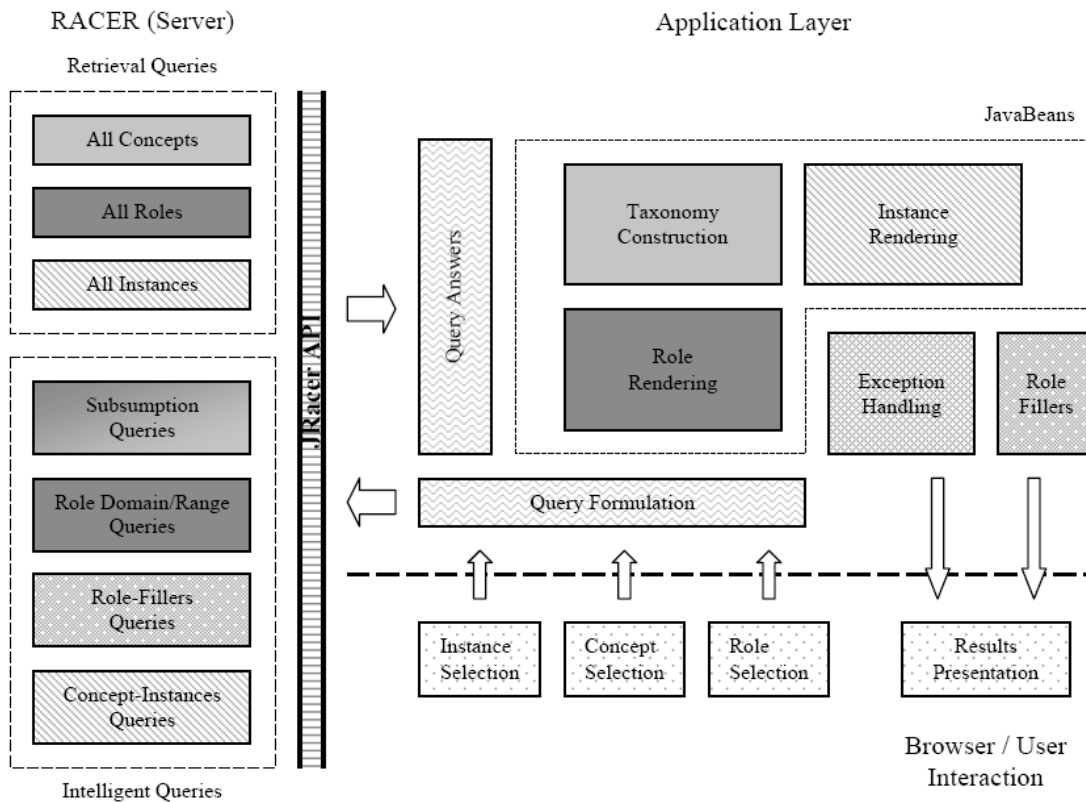
Στο σημείο αυτό εξετάζεται η αρχιτεκτονική των προγραμματιστικών συνιστωσών που απαρτίζουν την KDI. Η εικόνα 4.11 παρουσιάζει τα τμήματα της υλοποιημένης επιχειρησιακής λογικής (δεξιά), καθώς και τη σχέση και εξάρτηση των τμημάτων αυτών με τις λειτουργίες χαμηλού επιπέδου του RACER (αριστερά): Το μοτίβο στο σχήμα μιας λειτουργίας υποδηλώνει τις συνιστώσες στις οποίες εμπλέκεται.

Όπως αναφέρθηκε, το επίπεδο εφαρμογής αναλαμβάνει όλες τις λειτουργίες που σχετίζονται με την δόμηση, την επεξεργασία και εμφάνιση των πληροφοριών της οντολογίας, τη σύνθεση και υποβολή των ερωτημάτων, την εμφάνιση των αποτελεσμάτων και τη διαχείριση των εξαιρέσεων. Η κατασκευή της ιεραρχίας και η συλλογή των κατάλληλων στιγμιότυπων και ρόλων υλοποιούνται ως μέθοδοι *set* και *get* ενός *JavaBean*. Η κατασκευή της ιεραρχίας και η συλλογή όλων των ρόλων γίνονται μόνο μία φορά, κατά την φόρτωση της οντολογίας και διαρκούν καθ' όλη τη διάρκεια της συνόδου του χρήστη με τη διεπαφή (*session*).

Για την κατασκευή της ιεραρχίας, αρχικά λαμβάνονται τα ονόματα όλων των εννοιών της οντολογίας (“All Concepts”) και στη συνέχεια η διάταξή τους γίνεται με τη χρήση ερωτημάτων υπαγωγής (“Subsumption Queries”) προς το σύστημα RACER και τη βοήθεια μιας συνάρτησης αναζήτησης πρώτα κατά βάθος. Η συλλογή των στιγμιότυπων εμπλέκει τη συνάρτηση *all-instances*, καθώς και ερωτήματα που επιστρέφουν τα στιγμιότυπα μιας συγκεκριμένης έννοιας (“Concept-Instances Queries”). Αντίστοιχα, η εμφάνιση των ρόλων γίνεται με τη συνάρτηση *all-roles*, με ερωτήματα που επιστρέφουν το πεδίο τιμών ή ορισμού (“Role-Domain/Range Queries”) και με ερωτήματα υπαγωγής.

Η διαχείριση των εξαιρέσεων και η διαδικασία εύρεσης των πληρωτών ρόλου έχουν υλοποιηθεί ως κώδικας ενσωματωμένος στη JSP σελίδα. Η εύρεση των πληρωτών λαμβάνει υπόψη της αν έχει ενεργοποιηθεί η επιλογή INV (αντίστροφος), καθώς και αν πρόκειται για ιδιότητα τύπων δεδομένων και χρησιμοποιεί τα αποτελέσματα που προέρχονται από

αντίστοιχα ερωτήματα (“Role-Fillers Queries”) στο σύστημα RACER. Τα αποτελέσματα ή τα τυχόν σφάλματα μορφοποιούνται με συνδυασμό JSP και HTML κώδικα και εμφανίζονται στο φυλλομετρητή του χρήστη.



Εικόνα 4.11: Η αρχιτεκτονική της KDI.

Γενικά, ο σχηματισμός των ερωτημάτων προς το RACER γίνεται με την υλοποίηση κώδικα που ενσωματώνεται είτε στην σελίδα JSP είτε στις αντίστοιχες μεθόδους του JavaBean. Στο σχηματισμό αυτό συνεισφέρει η επιλογή από το χρήστη έννοιας, στιγμιότυπου και ρόλου (και αντίστροφου), που γίνεται ορατή στην εφαρμογή με τη χρήση και υποβολή HTML φορμών. Η υποβολή των ερωτημάτων προς το σύστημα RACER και η λήψη των απαντήσεων γίνονται με τη βοήθεια των κατάλληλων μεθόδων του JRacer API.

5. ΣΗΜΑΣΙΑΚΗ ΠΡΟΣΑΡΜΟΓΗ

5.1. Εισαγωγή

Οι οντολογίες παίζουν βασικό ρόλο στην ιδέα του Σημαντικού Ιστού. Παρέχουν έναν τρόπο όχι μόνο για την επικοινωνία δομικών πτυχών της πληροφορίας και υψηλού επιπέδου δεδομένων για αυτήν, αλλά και για την κατανόηση και τον ευφυή χειρισμό της από μια υπολογιστική μηχανή. Επιπρόσθετα, οι οντολογίες στο Σημαντικό Ιστό αποτελούν δικτυακά προσβάσιμη και συχνά κατανεμημένη γνώση, γεγονός το οποίο καθεαυτό γεννά μια σύγχρονη και ενδιαφέρουσα πρόκληση, αναφορικά με τον τρόπο που οι οντολογίες αυτές πρόκειται να αξιοποιηθούν, τόσο από τεχνική, όσο και από εννοιολογική σκοπιά.

Οι οντολογίες είναι συνήθως σχεδιασμένες, ώστε να αναπαριστούν μια συγκεκριμένη περιοχή της ανθρώπινης γνώσης, αποκαλούμενη *πεδίο γνώσης* (knowledge domain). Τέτοιες οντολογίες πεδίου επιχειρούν συνήθως να πετύχουν έναν διττό στόχο: πρώτον, να είναι όσο το δυνατόν πιο διεξοδικές, ώστε να καλύπτουν κάθε πιθανή πτυχή του θεωρούμενου πεδίου· δεύτερον, να μην είναι υπερβολικά συγκεκριμένες, με την έννοια ότι δεν θα παραχωρήσουν τη γενικότερη χρησιμότητα της οντολογίας. Άλλωστε, σύμφωνα με τον Guarino [Guarino, 1998], μια οντολογία μπορεί μόνο να προσεγγίσει την εννοιολογική σύλληψη της γνώσης του πεδίου (βλ. και ενότητα 2.2).

Το *Εννοιολογικό Μοντέλο Αναφοράς CIDOC* (CIDOC-CRM, Conceptual Reference Model) [Crofts, Doerr, & Gill, 2003] συνιστά μια τέτοια οντολογία που επιχειρεί να μοντελοποιήσει το πεδίο γνώσης της πολιτιστικής κληρονομιάς. Όπως κάθε πεδίο γνώσης που συλλαμβάνει ο ανθρώπινος νους, έτσι και η πολιτιστική κληρονομιά είναι πολύ δύσκολο να μοντελοποιηθεί με ακρίβεια. Επιπλέον και λόγω της φύσης τους, οι πληροφορίες πολιτιστικής κληρονομιάς είναι συνήθως κρυμμένες σε βιβλιοθήκες και αρχεία μουσείων και, όταν είναι δικτυακά διαθέσιμες, είναι ελάχιστα έως καθόλου δομημένες. Ακόμη, το CIDOC-CRM έχει πρόσφατα αναδειχθεί σε πρότυπο κατά ISO (ISO-21127), γεγονός που ενισχύει περαιτέρω τη σημασία της χρήσης του ως κοινού εννοιολογικού υπόβαθρου μεταξύ εφαρμογών πολιτιστικής κληρονομιάς.

Από την άλλη, ο Σημαντικός Ιστός προσφέρει ένα εύρος δελεαστικών δυνατοτήτων, από τη διαχείριση γνώσης στον Ιστό, στη σημασιακή περιγραφή πόρων και στην ανακάλυψη κατανεμημένης γνώσης. Επομένως, η περίτεχνη αναπαράσταση της γνώσης σε μια οντολογία, σε συνδυασμό με ευφυή εργαλεία συλλογισμού, καθορίζουν το βαθμό στον οποίο μπορεί να συμπεραθεί νέα, χρήσιμη γνώση, η οποία υπονοείται ανάμεσα στις οντολογικές γραμμές.

Χρησιμοποιώντας το CIDOC-CRM ως εννοιολογική βάση, δημιουργούμε πρώτα την κατανοήσιμη από τη μηχανή εκδοχή του, εκφράζοντάς το σε OWL. Η διαδικασία αυτή δεν εκφυλίζεται σε έναν απλό συντακτικό μετασχηματισμό. Αντίθετα, εκμεταλλευόμενοι τις πιο εκφραστικές, αλλά, ταυτοχρόνως, αποφασίσιμες δομές της OWL [Horrocks & Sattler, 2005], προχωρούμε στον εμπλουτισμό και την αναγωγή του μοντέλου, πετυχαίνοντας έτσι μια καλύτερη προσέγγιση της εννοιολογικής σύλληψης.

Όμως, μια τέτοια διαδικασία δεν μπορεί να συνεχιστεί επ' άπειρον: Υπάρχει πάντα ο κίνδυνος το μοντέλο να γίνει πολύ συγκεκριμένο, θέτοντας έτσι την εφαρμοσιμότητά του σε κίνδυνο. Για να αποφευχθεί κάτι τέτοιο, ενσωματώνουμε τις συγκεκριμένες ως προς OWL ισχυρές δηλώσεις σε διαφορετικά OWL έγγραφα, που περιλαμβάνουν απτά στιγμιότυπα των εννοιών και ρόλων του CRM. Η προσέγγιση αυτή, επιδεικνύει αφενός τις δυνατότητες κατανεμημένης ανακάλυψης γνώσης που είναι εγγενείς στις οντολογίες Ιστού· αφετέρου όμως προτείνει ένα παράδειγμα *σημασιακά επαυξημένων προφίλ εφαρμογής* (semantically enhanced application profiles), όπου η διαχωριστική γραμμή ανάμεσα στο πρότυπο και στην απαιτούμενη από την εφαρμογή ακρίβεια είναι λεπτή και κρίσιμη. Αυτό το είδος της *προσαρμογής* (profiling) ανάγει την συνήθη, σχετική με μεταδεδομένα έννοια του όρου, όπου θεωρείται ως η συνάθροιση ετερογενών πεδίων μεταδεδομένων [Dunal, Hodgins, Sutton, & Weibel, 2002], σε ένα υψηλότερο επίπεδο: Δεν ασχολείται τόσο πολύ με την οριζόντια επέκταση της οντολογίας, αλλά επιχειρεί να την επεκτείνει με σημασιακό τρόπο, όπως μπορεί να υπαγορεύεται από μια συγκεκριμένη εφαρμογή, διατηρώντας ταυτόχρονα τη διαλειτουργικότητα. Μας ενδιαφέρει επομένως να ελέγξουμε την ορθότητα της παρακάτω υπόθεσης:

Υπόθεση 2: *Η τεχνική της σημασιακής προσαρμογής επιτυγχάνει την εντασιακή επέκταση (εμβάθυνση) ενός μοντέλου, χωρίς όμως να βλάπτεται η γενική του εφαρμοσιμότητα.*

Προς το σκοπό αυτό, η κύρια ιδέα είναι να διαχωριστεί η τεχνική σε *διακριτές φάσεις*, κάθε μία από τις οποίες αντανακλά σε αντίστοιχο *φυσικό* διαχωρισμό του σημασιακού προφίλ σε πολλαπλά οντολογικά έγγραφα.

Το επόμενο βήμα είναι η αξιοποίηση της προκύπτουσας οντολογίας, κυρίως μέσω της δυνατότητας να εφαρμοστούν στην πράξη τα πλεονεκτήματα των σημασιακών επεκτάσεων. Όσο κι αν η γλώσσα ή το μοντέλο αναφοράς είναι τυποποιημένα, δεν ισχύει το ίδιο και για την διαδικασία με την οποία το βήμα αυτό μπορεί να επιτευχθεί. Για το λόγο αυτό, εφαρμόζουμε τη μεθοδολογία που αναλύθηκε στο προηγούμενο κεφάλαιο και χρησιμοποιούμε την KDI, ώστε να καταστεί δυνατή η υποβολή πιο εκφραστικών, βασισμένων σε συλλογισμό ευφυών επερωτήσεων στην προσαρμοσμένη μορφή του CRM.

Στην επόμενη ενότητα γίνεται μια επισκόπηση της ιδέας και των προσεγγίσεων που σχετίζονται με την προσαρμογή μεταδεδομένων εφαρμογής. Κατόπιν, εισάγεται το Εννοιολογικό Μοντέλο Αναφοράς CIDOC, η δομή και η σημασιολογία του. Μετά συζητείται η διαδικασία του μετασχηματισμού και της προσαρμογής του CIDOC-CRM, επισημαίνοντας τις βελτιώσεις και εισάγοντας τελικά την τεχνική της σημασιακής προσαρμογής. Στη συνέχεια, παρατίθενται ορισμένοι συμπερασμοί που διεξήχθησαν στο CRM χρησιμοποιώντας την KDI, καθώς και τα αποτελέσματά τους. Τέλος εξηγείται γιατί η τεχνική της σημασιακής προσαρμογής, όπως εισάγεται στον παρόν κεφάλαιο και εφαρμόζεται στο CIDOC-CRM, είναι *στοχευμένη*, με την έννοια ότι αυξάνεται η εκφραστικότητα με τέτοιο τρόπο, ώστε η πολυπλοκότητα του συλλογισμού να παραμένει η μικρότερη δυνατή.

Το κεφάλαιο αυτό βασίζεται και επεκτείνει τις ανακοινώσεις [Koutsomitropoulos & Papatheodorou, 2007; Koutsomitropoulos, Paloukis, & Papatheodorou, 2007a], το κεφάλαιο [Koutsomitropoulos, Paloukis, & Papatheodorou, to appear], καθώς και το άρθρο [Koutsomitropoulos, Paloukis, & Papatheodorou, 2007b].

5.2. Προσαρμογή μεταδεδομένων εφαρμογής

Η ανάγκη για αποτελεσματική περιγραφή πόρων στα ηλεκτρονικά αρχεία γρήγορα οδήγησε στην συνειδητοποίηση της έλλειψης ενιαίων τρόπων για την αναπαράσταση και διατήρηση πληροφοριών σχετικών με τέτοιους πόρους. Οι πληροφορίες αυτές, γνωστές και ως *μεταδεδομένα*, μπορούν επομένως να είναι οργανωμένες σε συμπαγή σχήματα μεταδεδομένων που τα παράγουν και τα διαχειρίζονται αρχές περιεχομένου, οργανισμοί και ειδικοί πεδίου. Η γλώσσα XML διευκόλυνε αυτή τη διαδικασία παρέχοντας ένα επίσημο συντακτικό για την έκφραση τόσο των σχημάτων, όσο και των ίδιων των μεταδεδομένων σε μορφή αναγνώσιμη από τη μηχανή.

Ωστόσο, καθώς τα σχήματα αυτά άρχισαν να αυξάνονται μέρα με τη μέρα, εστιάζοντας σε ένα ορισμένο πεδίο ενδιαφέροντος ή μια ορισμένη λειτουργία, συνέβαινε συχνά οι ανάγκες ενός συγκεκριμένου προγραμματιστή να μην ικανοποιούνταν από κάποιο υπάρχον σχήμα ή ορισμένα πεδία που να θεωρεί επιθυμητά, να είναι διασκορπισμένα σε πολλές πρότυπες υλοποιήσεις. Η προσαρμογή μεταδεδομένων εφαρμογής εμφανίστηκε λοιπόν ως η φυσική λύση για την υπερπήδηση αυτών των εμποδίων, σεβόμενη ταυτοχρόνως τον λόγο ύπαρξης των προτύπων. Όπως ορίζεται στο [Duval, Hodgins, Sutton, & Weibel, 2002]:

Προσαρμογή μεταδεδομένων εφαρμογής είναι η συγκέντρωση πεδίων μεταδεδομένων, επιλεγμένων από ένα ή περισσότερα σχήματα και ο συνδυασμός τους σε ένα ενιαίο σχήμα. Τα προφίλ εφαρμογής παρέχουν το μέσο για την έκφραση των αρχών της δομικότητας (modularity) και της επεκτασιμότητας. Ο σκοπός ενός προφίλ εφαρμογής είναι να προσαρμόσει ή να συνδυάσει υπάρχοντα σχήματα σε ένα πακέτο που είναι προσαρμοσμένο για τις λειτουργικές απαιτήσεις μιας συγκεκριμένης εφαρμογής, διατηρώντας ταυτόχρονα τη διαλειτουργικότητα με τα αρχικά σχήματα-βάσεις.

Υπάρχουν διάφοροι τρόποι για την ανάπτυξη ενός προφίλ μεταδεδομένων [Heery & Patel, 2000]: Ο πιο προφανής είναι να συμπεριληφθούν στο ίδιο σχήμα πεδία επιλεγμένα από διαφορετικά πρότυπα, κατάλληλα για τη συγκεκριμένη εφαρμογή. Αν χρειάζεται να οριστούν νέα πεδία, αυτό θα πρέπει να είναι απολύτως αναγκαίο και τα νέα πεδία θα πρέπει να αναφέρονται σε δικό τους χώρο ονομάτων. Άλλη τεχνική είναι ο περιορισμός του εύρους τιμών των πεδίων, για παράδειγμα, η παροχή ενός συγκεκριμένου ελεγχόμενου λεξιλογίου από όπου ένα πεδίο θα δέχεται τιμές ή η υπαγόρευση συγκεκριμένων μορφότυπων για τις τιμές. Τέλος, ένα προφίλ μπορεί να εκλεπτύνει τα υπάρχοντα στα πρότυπα πεδία. Μπορεί για παράδειγμα να περιλαμβάνει τον ορισμό υπο-πεδίων που στοχεύουν στον περιορισμό της σημασίας ενός ορισμού ή να εισάγει ορισμένες εξειδικεύσεις (qualifications).

Έστω το ακόλουθο παράδειγμα προσαρμογής εφαρμογής [Powell & Johnston, 2003]:

```
<?xml version="1.0"?>
<record xmlns="http://example.org/learningapp/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://example.org/learningapp/schema.xsd"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:ims="http://www.imsglobal.org/xsd/imsmd_v1p2">
```

```

<dc:title>
  Frog maths
</dc:title>
<dc:description>
  Simple maths games for 5-7 year olds.
</dc:description>
<ims:typicallearningtime>
  <ims:datetime>
    0000-00-00T00:15
  </ims:datetime>
</ims:typicallearningtime>
</record>

```

Το παραπάνω αποτελεί στιγμιότυπο της ανάμιξης πεδίων του Dublin Core με μαθησιακά μεταδεδομένα IMS. Είναι αξιοσημείωτο ότι το πιο σημαντικό μέσο για την πρακτική υλοποίηση των προφίλ εφαρμογής είναι η *χώροι ονομάτων*. Στο παραπάνω παράδειγμα, το dc αναπαριστά πεδία από το σύνολο Dublin Core, ενώ το ims δηλώνει μεταδεδομένα που προέρχονται από το IMS. Οι χώροι ονομάτων παίζουν κρίσιμο ρόλο, όχι μόνο για την αναγνώριση της προέλευσης διακριτών σχημάτων, αλλά επίσης και ως μέσο για το διαχωρισμό και κατόπιν την ανάμιξη διαφορετικών πεδίων και λεξιλογίων.

Είναι ξεκάθαρο ότι τα σχήματα μεταδεδομένων επιχειρούν να αποτυπώσουν και να μεταδώσουν ανθρωπίνως αντιληπτή γνώση στην πιο στοιχειώδη, μονοσήμαντη, μηχανική μορφή: Μια οριζόντια συνάθροιση ορισμών (πιθανώς με υπο-πεδία) με καθορισμένους μορφότυπους και περιορισμούς τιμών, που είναι εκφρασμένη (συνήθως) σε XML. Ως προς αυτό, τα πρότυπα μεταδεδομένων είναι απολύτως επιτυχή· ταυτόχρονα όμως η αναπαράσταση γνώσης σε αυτά θεωρείται φτωχή και απέχει από την μηχανική κατανοησιμότητα.

Ο Σημαντικός Ιστός και οι οντολογίες του δίνουν την ευκαιρία για ακριβέστερη μοντελοποίηση ενός πεδίου γνώσης και επομένως, επιτρέπουν την αναβάθμιση των μεταδομένων από μηχανικά αναγνώσιμη σε μηχανικά κατανοήσιμη κατάσταση. Στην πραγματικότητα, οι οντολογίες είναι σχήματα μεταδεδομένων με επακριβώς καθορισμένη σημασία και πλουσιότερες σχέσεις μεταξύ των πεδίων και των εννοιών ενός εννοιολογικού μοντέλου. Τα εργαλεία του Σημαντικού Ιστού ανοίγουν μια σειρά δυνατοτήτων που μπορούν να διευκολύνουν περαιτέρω την ανάπτυξη εμπλουτισμένων προφίλ μεταδεδομένων. Οι δυνατότητες αυτές περιλαμβάνουν μια νέα μέθοδο για την δημιουργία ενός προφίλ μεταδεδομένων εφαρμογής, όχι μόνο με το συνδυασμό, την εκλέπτυνση ή τον περιορισμό πεδίων, αλλά μέσω της *σημασιακής επαύξησης* του μοντέλου, και αυτό ακριβώς είναι που επιχειρείται να τεκμηριωθεί στην ενότητα 5.4.

5.3. Το Εννοιολογικό Μοντέλο CIDOC

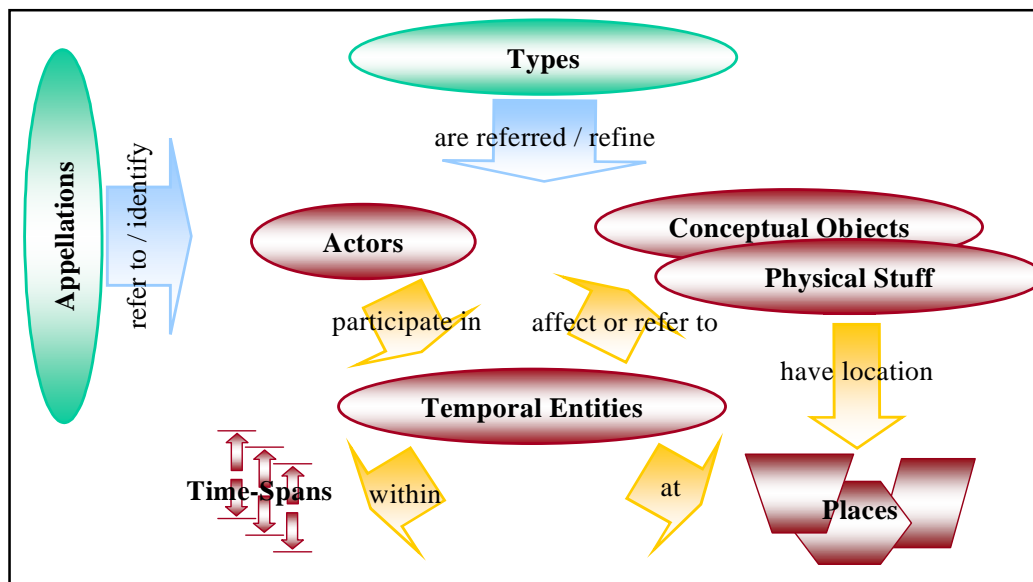
Το CIDOC-CRM μπορεί να θεωρηθεί οντολογία πεδίου με την έννοια που δίνεται στο [Guarino, 1998]. Καλύπτει επομένως μόνο μια ορισμένη περιοχή ενδιαφέροντος και όχι την εν γένει γνώση. Παράλληλα, το CIDOC-CRM έχει σχεδιαστεί ώστε να είναι επεκτάσιμο, ευέλικτο και απαλλαγμένο από τυπικές υλοποιήσεις. Για το λόγο αυτό, είναι εύκολο να εναρμονιστεί με άλλες οντολογίες ανωτάτου επιπέδου ή άλλα εννοιολογικά σχήματα

προκειμένου να εξυπηρετήσει τις ανάγκες μοντελοποίησης συγκεκριμένων οργανισμών, καθώς και ολοκλήρωσης της πληροφορίας από ετερογενείς σημασιολογικά πηγές [Doerr, Hunter, & Lagoze, 2003].

Στην ενότητα αυτή δίνεται μια σύντομη επισκόπηση του CRM, όπως εισάγεται στο [Crofts, Doerr, & Gill, 2003]. Στη συνέχεια συζητούνται οι μηχανικά αναγνώσιμες υλοποιήσεις του και η αντίστοιχη εκφραστικότητα που τις διέπει, ενώ στο τέλος παρατίθενται ορισμένες σχετικές εφαρμογές του CRM και προσεγγίσεις.

5.3.1. Εννοιολογική δομή

Η δομή του CRM μπορεί να περιγραφεί καλύτερα ξεκινώντας από τις ευρείες κλάσεις. Αυτές οι κλάσεις υψηλού επιπέδου προέκυψαν από τη διαδικασία λογικής ομαδοποίησης των κοινών ιδιοτήτων. Οι ομάδες αυτές αφορούν θεμελιώδεις έννοιες όπως η αναγνώριση, η συμμετοχή, η τοποθεσία, ο σκοπός, το κίνητρο, η χρήση κλπ. Στην εικόνα 5.1 παρουσιάζεται μια επισκόπηση της δομής του CRM όπου οι *Χρονικές Οντότητες* (Temporal Entities) και επομένως τα συμβάντα, κατέχουν κεντρική θέση.



Εικόνα 5.1: Ένα ποιοτικό μετα-σχήμα του CIDOC-CRM [Doerr, 2003].

Όλα τα μονοπάτια ιδιοτήτων προς τις ημερομηνίες περνούν μέσα από τις *Χρονικές Οντότητες*, όπως συμβαίνει και με τα περισσότερα μονοπάτια ιδιοτήτων για τις τοποθεσίες. Οι ιδιότητες τοποθεσιών που παρακάμπτουν τις χρονικές οντότητες, θα πρέπει να γίνουν κατανοητές ως σύντομοι δρόμοι των χρονικών οντοτήτων. Ανάλογα, οι *Δράστες* (Actors) μπορούν να ειπωθούν ότι συνδέουν υλικά και άυλα πράγματα (Physical Stuff, Conceptual Objects) μόνο μέσω των Χρονικών Οντοτήτων.

Κάθε στιγμιότυπο μιας κλάσης μπορεί να αναγνωριστεί από ορισμένες *Ονομασίες* (Appellations). Αυτές συνίστανται σε ονόματα, ετικέτες, τίτλους και άλλα μέσα αναγνώρισης που χρησιμοποιούνται στο ιστορικό περιβάλλον. Η αμφίσημη αυτή σχέση των αντικειμένων με τα ονόματά τους, μοντελοποιείται ως μέρος της ιστορικής διαδικασίας απόκτησης γνώσης. Η έννοια της αναγνώρισης που χρησιμοποιείται εδώ, δεν θα πρέπει να συγχέεται με αυτή των αναγνωριστικών βάσης δεδομένων σε υλοποιήσεις του μοντέλου, που δεν περιλαμβάνονται στην οντολογία.

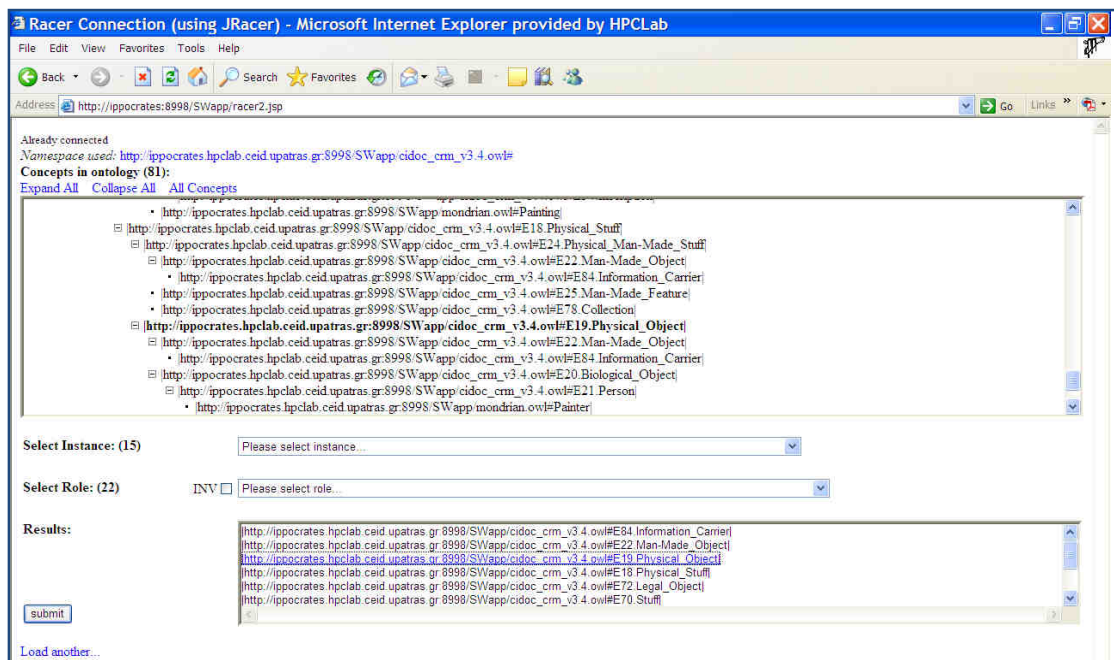
Όλα τα στιγμιότυπα μπορούν να εκλεπυνθούν σε πιο λεπτομερείς κατηγορίες μέσω της χρήσης των *Τύπων* (Types). Οι τύποι συνήθως αποτελούνται από ένα εύρος ιδιοτήτων που αναφέρονται γενικά σε πράγματα ενός συγκεκριμένου είδους, όπως για παράδειγμα «βιβλία για Ιστορία» σε αντίθεση με «βιβλία για την Επανάσταση».

Οι ιδιότητες του CRM μπορούν να ομαδοποιηθούν στις ακόλουθες κατηγορίες:

- *Αναγνώριση* αντικειμένων του πραγματικού κόσμου με ονόματα του πραγματικού κόσμου.
- *Ταξινόμηση* των αντικειμένων του πραγματικού κόσμου.
- *Μερική ανάλυση* και δόμηση των Εννοιολογικών και Φυσικών Αντικειμένων, των Περιόδων, των Δραστών, των Τοποθεσιών και των Χρόνων.
- *Τοποθέτηση* των περιόδων στο χωροχρόνο και των φυσικών αντικειμένων στο χώρο.
- *Επιρροή* των αντικειμένων σε δραστηριότητες και προϊόντα και αντιστρόφως.
- *Αναφορά* των πληροφοριακών αντικειμένων σε οποιοδήποτε αντικείμενο του πραγματικού κόσμου.

5.3.2. Υλοποιήσεις και εκφραστικότητα

Το CIDOC-CRM βρίσκεται αυτή τη στιγμή (*Μάιος 2008*) στην έκδοση 4.2.4. Η έκδοση 4.0 είναι στην πραγματικότητα η έκδοση 3.4.10. Οι αλλαγές που ενσωματώνονται σε επόμενες εκδόσεις αφορούν κυρίως τη διαγραφή, την προσθήκη και την μετονομασία εννοιών και ρόλων, σε μικρή κλίμακα. Από τις διαθέσιμες υπολογιστικές μορφές, επιλέχθηκε η RDF(S) ως πλουσιότερη από σημασιολογική άποψη και εγγύτερη στην OWL.



Εικόνα 5.2: Η ταξινόμηση του CIDOC-CRM, όπως φαίνεται μέσα από την KDI.

Από τον Ιανουάριο του 2005 υπάρχει μια μετάφραση σε OWL του RDF εγγράφου του CRM [Balzer, 2006]. Η έκδοση αυτή όμως προσθέτει εκφραστικότητα σχετική κυρίως με ρόλους

(αντιστροφή, μεταβατικότητα κλπ), και από σημασιολογική άποψη, δεν ξεπερνά το επίπεδο της Λογικής Περιγραφής *SHIF* (OWL Lite). Αυτό συμβαίνει διότι, παρά το γεγονός ότι προσθέτει και πληροφορία για ξένες κλάσεις, τα μόνα χαρακτηριστικά της OWL DL που δεν μπορούν, έστω και έμμεσα, να εκφραστούν στην OWL Lite είναι τα ονοματικά και οι περιορισμοί πληθικότητας μεγαλύτερης της μονάδας [Horrocks, Patel-Schneider, & Harmelen, 2003].

Ιδιαίτερα πρόσφατα επίσης (10 Μαΐου 2008) είναι διαθέσιμη μια νέα εκδοχή του μοντέλου σε OWL, από το πανεπιστήμιο του Erlangen, που βασίζεται στην έκδοση 4.2.4 [Oischinger, Schiemann, & Goerz, 2008]. Το ενδιαφέρον χαρακτηριστικό της είναι ότι χρησιμοποιεί πιο εκφραστικούς κατασκευαστές, διαθέσιμους στην OWL, όπως η υπαρξιακή ποσοτικοποίηση (`owl:someValuesFrom`). Μάλιστα, η εκφραστικότητα της υλοποίησης αυτής ισοδυναμεί σχεδόν απόλυτα με την εκφραστικότητα που προστίθεται κατά τη φάση της εντασιακής ενδυνάμωσης πυρήνα (βλ. ενότητα 5.4.2), μια φάση που αποτελεί μέρος της τεχνικής της σημασιακής προσαρμογής και έχει τεκμηριωθεί σε προγενέστερες δημοσιεύσεις.

Το CRM είναι ένα σχετικά περίπλοκο μοντέλο που περιλαμβάνει, στην τελευταία έκδοση, 87 έννοιες και 148 ρόλους, χωρίς τους αντίστροφους αυτών (δηλαδή συνολικά 296 ρόλους) (εικόνα 5.2). Εκφραστικά, χρησιμοποιούνται δομές που καθιστά δυνατές το RDF(S) και συνοψίζονται παρακάτω:

- Τόσο οι έννοιες όσο και οι ρόλοι είναι οργανωμένοι σε ιεραρχίες.
- Για κάθε ρόλο, ορίζονται οι έννοιες που αποτελούν το πεδίο ορισμού και το πεδίο τιμών του ρόλου.
- Για κάθε ρόλο, ορίζεται και ο αντίστροφός του, ως ξεχωριστός ρόλος, αφού δεν μπορεί να εκφραστεί άμεσα η έννοια του αντίστροφου στο RDF(S).
- Οι ρόλοι δεν διακρίνονται σε ιδιότητες αντικειμένων και ιδιότητες τύπων δεδομένων όπως στην OWL, αλλά για τις ιδιότητες τύπων δεδομένων χρησιμοποιείται ως πεδίο τιμών το `rdf:Literal`.

5.3.3. Σχετικές εφαρμογές και προσεγγίσεις

Από τις εφαρμογές του CRM πιο σχετική μπορεί να θεωρηθεί η χρήση του στο σύστημα *Artequakt* [Alani, et al., 2003]. Το *Artequakt* προσπαθεί να αμβλύνει το πρόβλημα της συντήρησης βάσης γνώσης χρησιμοποιώντας μια αυτοματοποιημένη προσέγγιση εξαγωγής γνώσης. Για την εξαγωγή πληροφορίας, η οποία, στα πλαίσια του συστήματος, σχετίζεται με τους καλλιτέχνες και τον καλλιτεχνικό κόσμο και χρησιμοποιείται για την ενημέρωση της βάσης γνώσης, το *Artequakt* εφαρμόζει επεξεργασία φυσικής γλώσσας πάνω σε δικτυακά έγγραφα. Η αποθηκευμένη γνώση χρησιμοποιείται μετά για την παραγωγή εξατομικευμένων βιογραφιών καλλιτεχνών. Το CIDOC-CRM έχει το ρόλο του «εννοιολογικού σχήματος» για τις εξαγόμενες πληροφορίες, θα πρέπει όμως να σημειωθεί ότι δεν αξιοποιούνται τεχνικές ανακάλυψης γνώσης, π.χ. για την αναζήτηση.

Το πρόγραμμα *Sculpteur* (<http://www.sculpteurweb.org>) στοχεύει επίσης στη δημιουργία ενός σημασιακού υπερ-επιπέδου σε μια ψηφιακή βιβλιοθήκη τρισδιάστατων πολιτιστικών αντικειμένων. Οι ιδιότητες και τα χαρακτηριστικά των αντικειμένων αυτών είναι οργανωμένα βάσει του CIDOC-CRM. Συλλογισμός διεξάγεται από πράκτορες ταξινόμησης οι

οποίοι, εφόσον πρώτα έχουν εκπαιδευτεί κατάλληλα, μπορούν να ταξινομήσουν τα αντικείμενα μέσα στην οντολογική δομή.

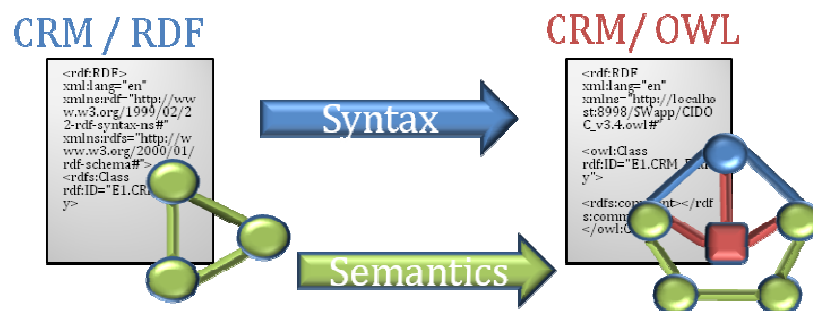
Τέλος, κοινό έναυσμα με την τεχνική της σημασιακής προσαρμογής έχει η προσέγγιση της έκφρασης προφίλ εφαρμογής, με χρήση της γλώσσας OWL/XDD [Ratanajairan, Nantajeewarawat, & Wuwongse, 2006]. Συνδυάζοντας κατασκευαστές της OWL με βασισμένες σε κανόνες εκφράσεις XML μορφής, υποστηρίζεται ότι είναι δυνατό να αναπαρασταθούν προφίλ εφαρμογής με αναλυτικούς σημασιακούς περιορισμούς. Η μέθοδος αυτή εφαρμόζεται για τον ορισμό ήδη μοντελοποιημένων πεδίων εφαρμογής, όπως για παράδειγμα το προφίλ εφαρμογής Dublin Core για βιβλιοθήκες (DC-Lib).

5.4. Η προσαρμογή του CIDOC-CRM

Στη συνέχεια συζητείται η διαδικασία του μετασχηματισμού του CIDOC-CRM ως προς πρώτον, τη συντακτική μετεγγραφή του και δεύτερον τη σημασιακή επαύξηση και προσαρμογή του (εικόνα 5.3). Για τη δημιουργία ενός σημασιακού προφίλ εφαρμογής για το CIDOC-CRM, ακολουθείται μια διττή προσέγγιση: Αρχικά γίνεται η μετάφρασή του στην κατάλληλη εκφραστική μορφή (δηλαδή στην OWL). Μετά προχωρούμε στην προσαρμογή καθεαυτή, πρώτα με την ενδυνάμωση της *εντασιακής του γνώσης* (intensional knowledge), δηλαδή της γενικής για το πεδίο γνώσης [Nardi & Brachman, 2007] και κατόπιν με την εκλέπτυνση του μοντέλου για τις ανάγκες μιας συγκεκριμένης εφαρμογής.

5.4.1. Συντακτικός μετασχηματισμός

Προκειμένου για το μετασχηματισμό της οντολογίας σε OWL σύνταξη, χρησιμοποιήθηκε αρχικά το σύστημα RACER (βλ. ενότητα 4.4.4). Το RACER έχει τη δυνατότητα να φορτώνει και να επεξεργάζεται βάσεις γνώσης εκφρασμένες σε ποικίλες μορφές, συμπεριλαμβανομένων των RDF(S) και OWL. Το RACER μπορεί να φορτώσει TBox που είναι εκφρασμένα σε RDF(S) με τη χρήση της εντολής `rdfs-read-tbox-file`. Στη συνέχεια το TBox μπορεί να εξαχθεί στην επιλεγμένη μορφή, με τη χρήση της εντολής `save-tbox` και την παράμετρο `:syntax`.



Εικόνα 5.3: Η διαδικασία μετασχηματισμού του CIDOC-CRM.

Πράγματι, ακολουθώντας τα παραπάνω βήματα, προέκυψε ένα τυπικό OWL έγγραφο που αναπαριστούσε με ακρίβεια την αρχική οντολογία. Διαπιστώθηκε όμως ότι το RACER προσέθετε περιττές και επιπλέον δηλώσεις που σε πολλά σημεία επικαλύπτονταν εκφραστικά. Για παράδειγμα:

- Για κάθε ρόλο και έννοια προστέθηκαν ετικέτες από τον χώρο ονομάτων του OIled και συγκεκριμένα οι `oiled:creationDate` και `oiled:Creator`, οι οποίες δεν απαιτούνταν και δεν περιλαμβάνονταν στο αρχικό έγγραφο.
- Για τον ορισμό μιας έννοιας ως πεδίο ορισμού ή πεδίο τιμών ενός ρόλου, χρησιμοποιείται κάθε φορά ο τελεστής `owl:UnionOf`, εκφράζοντας τα πεδία αυτά ως ένωση εννοιών με ένα μόνο μέλος (τη συγκεκριμένη έννοια).
- Ο ορισμός των πεδίων ορισμού και τιμών ενός ρόλου, και στην OWL, προέρχεται από τον χώρο ονομάτων του RDF(S) (`rdfs:domain`, `rdfs:range`). Το RACER, παρόλο που διατηρεί αυτές τις δηλώσεις, προσθέτει εντούτοις και άλλες ισοδύναμες, που προέρχονται από την έκφραση τέτοιου είδους περιορισμών στις Λογικές Περιγραφές. Αυτές οι ισοδύναμες δηλώσεις εμπλέκουν περιορισμούς αριθμού και περιορισμούς τιμής και μπορούν να συμβολιστούν στην OWL.

Η διαδικασία αυτή είχε ως αποτέλεσμα την μετατροπή του αρχικού εγγράφου, μεγέθους 60KB, σε ένα OWL έγγραφο, μεγέθους 478KB. Έτσι, προτιμήθηκε τελικά η χειρωνακτική μετατροπή του RDF έγγραφου, κατά την οποία διατηρήθηκαν οι κοινές εκφράσεις μεταξύ RDF(S) και OWL (π.χ. `rdfs:subClassOf` και `rdf:resource`), ενώ αντικαταστάθηκαν ορισμένα προθέματα χώρων ονομάτων και γενικά τροποποιήθηκε η ονοματολογία (π.χ. `owl:Class` αντί για `rdfs:Class` και `owl:ObjectProperty` ή `owl:DatatypeProperty` αντί για `rdf:Property`). Με τον τρόπο αυτό, ολοκληρώθηκε η φάση της συντακτικής μετατροπής της οντολογίας, που οδήγησε στη δημιουργία ενός OWL εγγράφου μεγέθους 63KB με όνομα `cidoc_crm_v3.4.owl`.

5.4.2. Σημασιακή εμπάθυνση και εκλέπτυνση

Η δεύτερη φάση της διαδικασίας αναγωγής του CRM περιλαμβάνει τη σημασιακή του επαύξηση με την προσθήκη εκφραστικών δομών της OWL, σε επίπεδο OWL DL, ώστε να καταστεί δυνατό ένα ικανοποιητικό επίπεδο συλλογισμού, καθώς και τη συμπλήρωσή του με ορισμένα συγκεκριμένα στιγμιότυπα. Ο πίνακας 5.1 συνοψίζει τα εκφραστικά κέρδη της τεχνικής της σημασιακής προσαρμογής στο CRM.

Η φάση αυτή διεξήχθη σε δύο βήματα: Πρώτον, προστέθηκαν εκφράσεις που αφορούν στο μοντέλο καθαυτό, ώστε να αποτυπωθεί καλύτερα η επιδιωκόμενη σημασία των ιδιοτήτων και των κλάσεων, με την αξιοποίηση του λεξιλογίου της OWL. Δεύτερον, προστέθηκαν επιπλέον υποκλάσεις και, σε αυτές, περαιτέρω *σημασιακοί περιορισμοί* που στην πραγματικότητα προσαρμόζουν το μοντέλο για τη συγκεκριμένη περίπτωση των πινάκων και των ζωγράφων. Ως ενδεικτικό σενάριο εφαρμογής έχει επιλεγεί η μοντελοποίηση γεγονότων από τη ζωή και το έργο του Ολλανδού ζωγράφου Piet Mondrian. Τα βήματα αυτά έχουν αναλυτικά ως εξής:

Εντασιακή εμπάθυνση πυρήνα

Στο βήμα αυτό δεν προσθέτουμε νέες κλάσεις ή οντότητες που να επεκτείνουν το CRM. Αντιθέτως, επιχειρείται καλύτερη προσέγγιση της κεντρικής (core) εννοιολογικής σύλληψης του μοντέλου χρησιμοποιώντας δηλώσεις OWL που επιτρέπουν την ακριβέστερη υλοποίησή της. Συγκεκριμένα:

- Μοντελοποιούνται περιορισμοί μέγιστης και ελάχιστης πληθικότητας για τους ρόλους με τη χρήση απροσδιόριστων περιορισμών αριθμού (`owl:minCardinality`, `owl:maxCardinality`)
- Μοντελοποιούνται αντίστροφοι ρόλοι, με τη χρήση του τελεστή `owl:inverseOf`.
- Δίνεται ένα παράδειγμα συμμετρικού ρόλου με τη χρήση της δήλωσης `rdf:type="&owl:Symmetric"`.

Ως ένα βαθμό, η προσθήκη περιορισμών πληθικότητας στις ιδιότητες μπορεί να θεωρηθεί ενέργεια προσαρμογής, εφόσον το μοντέλο ξεκάθαρα προδιαγράφει ότι τέτοιο ποσοτικοποιητές παρέχονται μόνο ως σημασιακές διευκρινήσεις. Εντούτοις, με τον τρόπο αυτό επιτυγχάνουμε τη μετατόπιση της επιδιωκόμενης σημασίας από τις σημειώσεις και τα σχόλια του μοντέλου σε επίπεδο σημασιακής δέσμευσης [Gruber, 1993b]. Αξίζει επίσης να σημειωθεί ότι, με το RDF(S) να αποτελεί την πιο διαδεδομένη υλοποίηση του CRM, δεν υπάρχει τρόπος να εκφραστούν τέτοιοι περιορισμοί. Για τους στόχους της διατριβής, δεν έχουμε ποσοτικοποιήσει εξαντλητικά τις ιδιότητες του CRM, αλλά έχουμε εφαρμόσει περιορισμούς σε ορισμένες μόνο, αυτές που χρησιμοποιούνται στο συγκεκριμένο παράδειγμα για τον Mondrian.

Είναι σαφές ότι οι παραπάνω προσθήκες εκλεπτύνουν τον πυρήνα του μοντέλου, ασχέτως αν κάτι τέτοιο επιδιώκεται ή όχι στις προδιαγραφές του. Η προσαρμογή κατά τον τρόπο αυτό επιτυγχάνει επομένως να επεκτείνει την *εντασιακή γνώση* του σχήματος, χρησιμοποιώντας κατασκευές και μέσα που είναι διαθέσιμα μόνο στην υποδομή του Σημαντικού Ιστού.

Κατασκευή	Παράδειγμα	Συμπερασμοί
Εντασιακή Ενδυνάμωση (crm_core_profile.owl)		
Πληθικότητες	<pre><owl:cardinality rdf:datatype = "xsd:nonNegativeInteger">4 </owl:cardinality></pre>	Ανακάλυψη ασυνεπειών στις σχέσεις. Συμπερασμός ταύτισης στιγμιοτύπων (όταν <code>cardinality=1</code>).
Αντίστροφοι Ρόλοι	<pre><owl:ObjectProperty rdf:about="&crm;P7F.took_place_at"> <owl:inverseOf rdf:resource="&crm;P7B.witnessed"/> </owl:ObjectProperty></pre>	Ανακάλυψη σχέσεων μεταξύ στιγμιοτύπων.
Συμμετρικοί Ρόλοι	<pre><owl:SymmetricProperty rdf:about="&crm;P139F.has_alternative_form"> </owl:SymmetricProperty></pre>	Ανακάλυψη σχέσεων μεταξύ στιγμιοτύπων.
Εκλέπτυνση Εφαρμογής (crm_paint_profile.owl)		
Απτά Πεδία (Τύποι Δεδομένων)	<pre><owl:DatatypeProperty rdf:ID="hasURL"> <rdfs:subPropertyOf rdf:resource= "&crm;P55F.has_current_location"/> <rdfs:range rdf:resource= "http://www.w3.org/2000/01/rdf-schema# Literal"/> </owl:DatatypeProperty></pre>	Συλλογισμός με τύπους δεδομένων. Ανάκτηση πόρων URI.
Υπαρξιακή Ποσοτικοποίηση	<pre><owl:Restriction> <owl:onProperty rdf:resource="&crm;P94F.has_created"/></pre>	Ορισμός τεχνητών κλάσεων και ανακάλυψη

	<pre> <owl:someValuesFrom rdf:resource="#Painting"/> <rdfs:subClassOf rdf:resource="#Painting_Event"/> </owl:Restriction> </pre>	στιγμιότυπων που ανήκουν σε αυτές βάσει των σχέσεών τους.
Καθολική Ποσοτικοποίηση	<pre> <owl:Restriction> <owl:onProperty rdf:resource="&crm;P94F.has_created"/> <owl:allValuesFrom rdf:resource="#Painting"/> </owl:Restriction> </pre>	Ανακάλυψη σχέσεων μεταξύ στιγμιότυπων βάσει του αν ανήκουν στην τεχνητή κλάση.
Ονοματικά	<pre> <owl:Restriction> <owl:onProperty rdf:resource="&crm;P2F.has_type"/> <owl:hasValue rdf:resource="#painting_composition"/> </owl:Restriction> </pre>	Συμπερασμός ανηκόντων στιγμιότυπων. Ανακάλυψη σχέσεων μεταξύ στιγμιότυπων. Συμπερασμός ταύτισης στιγμιότυπων.

Πίνακας 5.1: Εκφραστικά κέρδη της σημασιακής προσαρμογής του CIDOC-CRM ως προς την OWL.

Εκλέπτυνση εφαρμογής

Στο βήμα αυτό, δημιουργούμε ορισμένα στιγμιότυπα εννοιών και ρόλων του CRM που αφορούν τη συγκεκριμένη εφαρμογή. Περιλαμβάνουμε επίσης δηλώσεις αξιωμάτων και γεγονότων που μόνο η OWL επιτρέπει την έκφρασή τους, καθώς και νέους ρόλους και έννοιες που χρησιμοποιούν αυτήν την εκφραστικότητα.

- Προσθέτουμε τις κλάσεις: 'Painting', ως υποκλάση της κλάσης του CRM 'Visual_Item', 'Painting_Event', ως υποκλάση της 'Creation_Event' και 'Painter', ως υποκλάση της 'Person'.
- Προσθέτουμε την ιδιότητα τύπων δεδομένων 'hasURL', ως υπο-ιδιότητα της 'has_current_location'.
- Χαρακτηρίζουμε σημασιακά τις παραπάνω έννοιες, βάσει υπαρξιακής και καθολικής ποσοτικοποίησης, χρησιμοποιώντας τις εκφράσεις owl:hasValue, owl:someValuesFrom και owl:allValuesFrom που ενεργοποιούν τελικά πιο ισχυρούς συμπερασμούς.

Για παράδειγμα (βλ. επίσης ενότητα 5.5):

```

<owl:Class rdf:ID="Painter">
<rdfs:subClassOf rdf:resource="&crm;E21.Person"/>
<owl:equivalentClass>
<owl:Restriction>
<owl:onProperty rdf:resource="&crm;P14B.performed"/>
<owl:someValuesFrom rdf:resource="#Painting_Event"/>
</owl:Restriction>
</owl:equivalentClass>
</owl:Class>

```

Το παραπάνω απόσπασμα ορίζει μια νέα κλάση, την 'Painter' και δηλώνει ότι 'Painter' είναι κάθε 'Person' που έχει 'performed' τουλάχιστον ένα 'Painting_Event'. Μπορούμε επομένως να δημιουργήσουμε το στιγμιότυπο 'Mondrian' ως εξής:

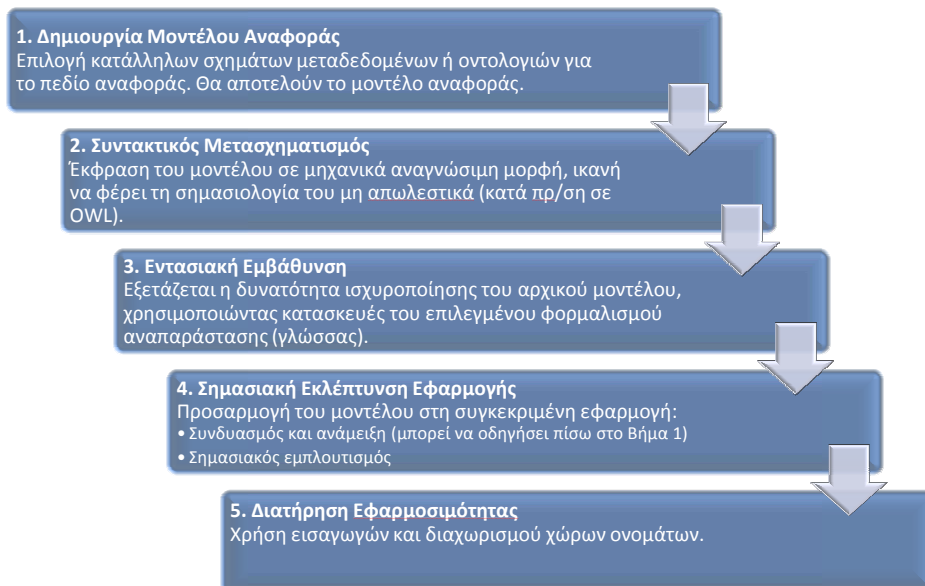
```
<crm:E21.Person rdf:ID="Mondrian">
  <crm:P14B.performed>
    <Painting_Event rdf:ID="Mondrian's Composition"/>
  </crm:P14B.performed>
</crm:E21.Person>
```

Δεδομένου του ορισμού της κλάσης 'Painter', ένας μηχανισμός συμπερασμού μπορεί τώρα να εξαγάγει ότι ο 'Mondrian' είναι πράγματι ζωγράφος ('Painter').

Αυτή είναι λοιπόν η άλλη κατεύθυνση της σημασιακής προσαρμογής: Προσθέτουμε νέα πεδία που φέρουν το δικό τους χώρο ονομάτων, αλλά επίσης τα εμπλέκουμε μεταξύ τους, καθώς και με τους ορισμούς του ίδιου του μοντέλου, θέτοντας έτσι *σημασιακές εκλεπτύνσεις* για τη δική μας συγκεκριμένη περίπτωση.

5.4.3. Η τεχνική της σημασιακής προσαρμογής

Η παραπάνω συζήτηση εισάγει τη διαδικασία της δημιουργίας σημασιακών προφίλ εφαρμογής και προτείνει μια γενικότερη μέθοδο για τις εφαρμογές μεταδεδομένων του Σημαντικού Ιστού. Αν και η τεχνική αυτή εφαρμόζεται συγκεκριμένα στο CIDOC-CRM, είναι ευνόητο πως ταιριάζει και σε κάθε άλλο πεδίο ενδιαφέροντος. Όπως φαίνεται στην εικόνα 5.4, ανεξαρτήτως από το πεδίο που επιλέγεται, πρώτο βήμα είναι η ανεύρεση μιας κατάλληλης, μηχανικά αναγνώσιμης υλοποίησης για το μοντέλο, η οποία, προς το παρόν, παρέχεται από τις προδιαγραφές της OWL.



Εικόνα 5.4: Η διαδικασία για την ανάπτυξη σημασιακών προφίλ εφαρμογής.

Δοσμένης μιας δόκιμης σύνταξης, είναι σκόπιμο να εξεταστούν οι δυνατότητες για καλύτερη αποτύπωση της εντασιακής γνώσης του μοντέλου, αξιοποιώντας ενδεχομένως το ιδιαίτερο λεξιλόγιο που προσφέρει η γλώσσα αναπαράστασης. Με τον τρόπο αυτό

ισχυροποιείται η εννοιολογική σύλληψη του μοντέλου και διασφαλίζονται οι δυνατότητές του.

Σε κάποιο σημείο, είναι πιθανό το μοντέλο να κριθεί ανεπαρκές για τις ανάγκες της συγκεκριμένης εφαρμογής. Όπως συμβαίνει και με την παραδοσιακή προσαρμογή μεταδεδομένων εφαρμογής, μπορεί να χρειαστεί να εξεταστούν και άλλα οντολογικά σχήματα, καθώς και σχήματα μεταδεδομένων και να αναμιχθούν με το αρχικό, επιστρέφοντας έτσι στο πρώτο βήμα. Επιπλέον, νέες κατάλληλες σημασιακές κατασκευές μπορεί να επινοηθούν, στενεύοντας έτσι τη σημασιολογία της επιδιωκόμενης εφαρμογής.

Κύριο μέλημα κατά την ανάπτυξη ενός προφίλ εφαρμογής είναι να εξασφαλιστεί ότι το αρχικό σχήμα δεν επηρεάζεται και ότι η γενική του εφαρμοσιμότητα διατηρείται. Για να επιτευχθεί αυτό, εκτός από τους χώρους ονομάτων, η OWL παρέχει ένα μηχανισμό ρητής συμπερίληψης, μέσω τη δήλωσης `owl:imports`. Στην προκειμένη περίπτωση, επιλέχθηκε οι σημασιακές προσθήκες να συμπεριληφθούν σε τρία νέα έγγραφα OWL: `crm_core_profile.owl` για την εμβάθυνση πυρήνα, `crm_paint_profile.owl` για την εκλέπτυνση εφαρμογής και `mondrian.owl` ως στιγμιότυπο των ανωτέρω¹. Με τον τρόπο αυτό διατηρείται το αρχικό μοντέλο και επίσης καταδεικνύονται οι δυνατότητες του Σημαντικού Ιστού για ολοκλήρωση οντολογιών και κατανεμημένη ανακάλυψη γνώσης.

Η διαλειτουργικότητα και η προς τα πίσω συμβατότητα με το αρχικό μοντέλο είναι επίσης ένα σημαντικό θέμα που μπορεί να αντιμετωπιστεί χρησιμοποιώντας αυτήν την προσέγγιση. Στις προσπάθειές της να φέρει τα σύνολο μεταδεδομένων Dublin Core (DC) πιο κοντά στην πραγματικότητα του Σημαντικού Ιστού, η Πρωτοβουλία για τα Μεταδεδομένα DC (DCMI, Dublin Core Metadata Initiative) αντιμετωπίζει ένα τέτοιο πρόβλημα: Ορίζοντας τα πεδία του DC ως ιδιότητες RDF και θέτοντας σε αυτές περιορισμούς, τα παλιότερα (legacy) μεταδεδομένα μπορεί να θεωρηθούν άκυρα από εφαρμογές συμπερασμού [Nilsson & Baker, 2008], όπως για παράδειγμα συμβαίνει με τα `dc:creator` και `dc:contributor`. Για να ξεπεράσει αυτό το εμπόδιο, το DCMI φαίνεται να ακολουθεί μια παρόμοια τακτική [Nilsson, Powell, Johnston, & Naeve, 2008]: Προσπαθεί να προσαρμόσει σημασιακά το μοντέλο του DC, ορίζοντας πεδία ορισμού και τιμών για τις ιδιότητες, και μετά να διατηρήσει τη συμβατότητα χρησιμοποιώντας εισαγωγές και διαχωρισμό ονομάτων. Φυσικά η ομάδα του DCMI είναι πιο αποφασιστική, με την έννοια ότι σκοπεύει να εκλεπτύνει *πλήρως* το μοντέλο και όχι μόνο να το προσαρμόσει για μια συγκεκριμένη εφαρμογή.

5.5. Αποτελέσματα

Στα επόμενα παρουσιάζονται τα αποτελέσματα ορισμένων πειραματικών συμπερασμών που διεξήχθησαν στην προσαρμοσμένη, OWL μορφή του CRM, ώστε να αποτιμηθούν οι «σημασιακές επιδόσεις» της τεχνικής της σημασιακής προσαρμογής. Για τη διεξαγωγή των συμπερασμών αυτών έχει χρησιμοποιηθεί η KDI.

Στόχος είναι να διερευνηθεί πόσο ισχυρός (με την έννοια της εκφραστικότητας) μπορεί να καταστεί ο συλλογισμός με τη χρήση του σημασιακού προφίλ, σε σύγκριση με το αρχικό μοντέλο. Προς το σκοπό αυτό αξίζει να σημειωθεί ότι όλοι οι συμπερασμοί που

¹ Όλα τα έγγραφα είναι διαθέσιμα στη διεύθυνση:
<http://ippocrates.hpclab.ceid.upatras.gr:8998/SWapp/>

παρουσιάζονται είναι *μοναδικοί*, με την έννοια ότι καθίστανται δυνατοί αποκλειστικά εξ' αιτίας του προφίλ. Τα αποτελέσματα ολοκληρώνονται με την παρουσίαση ενός ρεαλιστικού σεναρίου χρήσης, στο οποίο ο χρήστης, μέσα από μια σειρά συμπερασμών, οδηγείται στην ανακάλυψη και ανάκτηση ενός υπαρκτού δικτυακού πόρου.

Οι συμπερασμοί που ακολουθούν μπορούν να διακριθούν σε δύο κατηγορίες: σε *θετικούς συμπερασμούς* όπου, βάσει των αξιωμάτων εννοιών και ρόλων, καθώς και των γεγονότων της οντολογίας συμπεραίνονται νέα, μη ρητώς εκφρασμένα γεγονότα και σε *αρνητικούς συμπερασμούς* όπου, βάσει των οντολογικών αξιωμάτων και γεγονότων εντοπίζονται καταστάσεις μη ικανοποιησιμότητας σε έννοιες και στιγμιότυπα.

Για κάθε παράδειγμα παρατίθεται το OWL απόσπασμα στο οποίο βασίζεται ο συμπερασμός, και εξηγείται σχηματικά η διαδικασία του συλλογισμού με χρήση φορμαλισμού Λογικών Περιγραφής. Για οικονομία χώρου, αντί για πλήρεις χώρους ονομάτων χρησιμοποιείται το πρόθεμα `crm`: για οντότητες που προέρχονται από τα έγγραφα `cidoc_crm_v3.4.owl` και `crm_core_profile.owl`, το πρόθεμα `crm_p`: για οντότητες που ανήκουν στο `crm_paint_profile.owl` και το προκαθορισμένο πρόθεμα '#' για οντότητες που αναφέρονται μέσα στο έγγραφο στο οποίο ορίζονται. Ο ορισμός των στιγμιοτύπων γίνεται στο `mondrian.owl`, το οποίο εισάγει τα άλλα τρία. Οι σχέσεις και οι δηλώσεις που ισχύουν στα αποτελέσματα αυτά αναπαρίστανται επίσης σχηματικά, σύμφωνα με το ακόλουθο υπόμνημα: ένα τετράγωνο δηλώνει ένα στιγμιότυπο, ο κύκλος συμβολίζει μια έννοια, ένα διευθυνόμενο βέλος μεταξύ δύο στιγμιοτύπων i_1 και i_2 μια σχέση $R(i_1, i_2)$, ένα βέλος μεταξύ εννοιών μια σχέση υπαγωγής προς την κατεύθυνση του βέλους και ένα βέλος μεταξύ ενός στιγμιοτύπου και μιας έννοιας δηλώνει ότι το στιγμιότυπο ανήκει ('isA') σε αυτήν την έννοια.

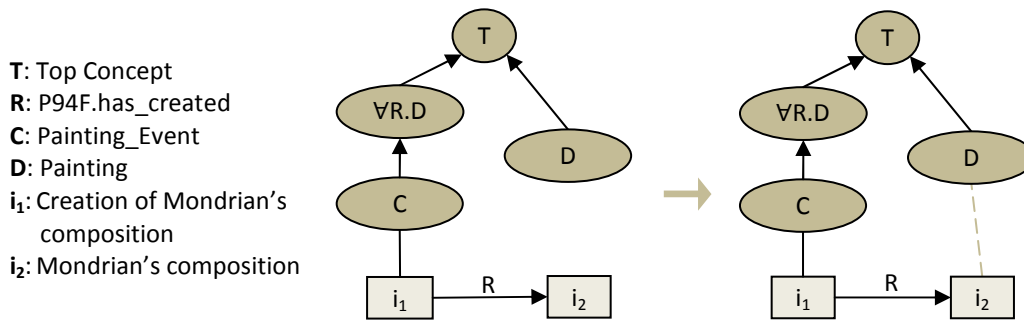
5.5.1. Θετικοί συμπερασμοί

Το παρακάτω απόσπασμα κώδικα προέρχεται από τα `crm_paint_profile.owl` και `mondrian.owl` και δηλώνει ότι ένα 'Painting_Event' είναι στην πραγματικότητα ένα 'Creation_Event' που 'has_created' μόνο αντικείμενα 'Painting':

```
<owl:Class rdf:ID="Painting_Event">
  <rdfs:subClassOf rdf:resource="&crm;E65.Creation_Event"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="&crm;P94F.has_created"/>
      <owl:allValuesFrom rdf:resource="#Painting"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>

<crm_p:Painting_Event rdf:ID="Creation of Mondrian's
composition">
  <crm:P94F.has_created rdf:resource="#Mondrian's composition"/>
</crm_p:Painting_Event>
```

Το απόσπασμα αυτό αναπαρίσταται σχηματικά στο αριστερό μέρος της εικόνας 5.5.



Εικόνα 5.5: Παράδειγμα συμπερασμού χρησιμοποιώντας περιορισμό τιμής.

Το 'Creation of Mondrian's composition' (i_1) είναι ρητά δηλωμένο ως 'Painting_Event' που 'has_created' (R) το 'Mondrian's composition' (i_2). Τώρα, ρωτώντας την ΚΔΙ να αποφανθεί «*τί είναι πίνακας;*», συμπεραίνει ότι το i_2 είναι πράγματι πίνακας ('Painting') (δεξί μέρος της εικόνας 5.5), διερμηνεύοντας ορθά τον περιορισμό τιμής στο ρόλο R .

Όσο απλό κι αν φαίνεται, το παραπάνω είναι στην πραγματικότητα ένας πολύ ισχυρός συμπερασμός. Χωρίς τον περιορισμό τιμής στο ρόλο 'has_created', το 'Mondrian's composition' θα μπορούσε να είναι απλώς ένα στιγμιότυπο του κόσμου, για παράδειγμα ένα βιβλίο, μια καρέκλα, ένας άνθρωπος ή μια χρονική περίοδος. Ακριβώς και μόνο αυτός ο περιορισμός - διαθέσιμος μόνο στην OWL DL και άρα εκφράσιμος μόνο ύστερα από τη δημιουργία του σημασιακού προφίλ - είναι που επιτρέπει στο 'Mondrian's composition' να ανακαλυφθεί ότι είναι 'Painting' και όχι οτιδήποτε άλλο. Στο επόμενο όμως παράδειγμα, το 'Mondrian's composition' δηλώνεται σαφώς ως 'Painting' από την αρχή.

Το δεύτερο παράδειγμα θετικού συμπερασμού εμπλέκει τη χρήση ονοματικών. Το ακόλουθο απόσπασμα από τα `crm_paint_profile.owl` και `mondrian.owl` δηλώνει ότι ένα 'Painting' είναι ένα 'Visual_Item' που έχει 'Type' 'painting_composition':

```

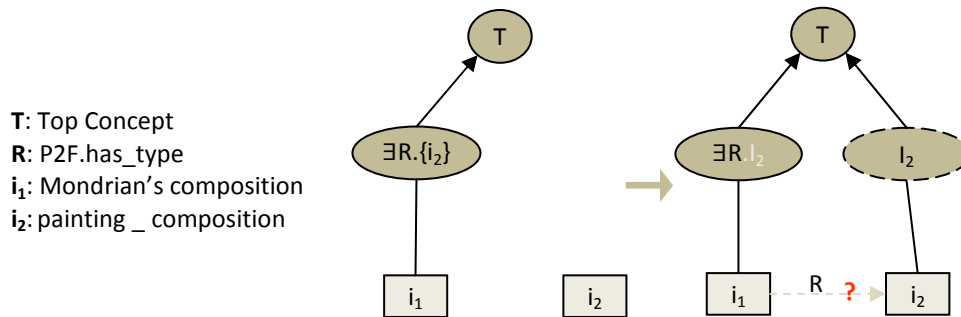
<owl:Class rdf:ID="Painting">
  <owl:subClassOf rdf:resource="&crm;E36.Visual_Item"/>
  <owl:equivalentClass>
    <owl:Restriction>
      <owl:onProperty rdf:resource="&crm;P2F.has_type"/>
      <owl:hasValue rdf:resource="#painting_composition"/>
    </owl:Restriction>
  </owl:equivalentClass>
</owl:Class>
<crm:E55.Type rdf:ID="painting_composition"/>

<crm_p:Painting rdf:ID="Mondrian's composition" />
  
```

Το παραπάνω απόσπασμα απεικονίζεται στο αριστερό μέρος της εικόνας 5.6. Το 'Mondrian's composition' (i_1) είναι ρητά δηλωμένο ως στιγμιότυπο του 'Painting', το οποίο ορίζεται μέσω του περιορισμού `hasValue` πάνω στο 'has_type' (R). Το 'painting_composition' δηλώνεται ως αντικείμενο 'Type'. Παρόλο που το γεγονός ότι το 'Mondrian's composition' 'has_type' 'Painting' προκύπτει ευθέως, η ΚΔΙ δεν μπορεί να το συμπεράνει και επιστρέφει `null` όταν ερωτάται «*ποιος είναι ο τύπος του 'Mondrian's composition'?*» (δεξί μέρος της εικόνας 5.6).

Το παράδειγμα αυτό καταδεικνύει πόσο δύσκολο είναι για τους μηχανισμούς συμπερασμού να πραγματοποιήσουν συλλογισμό που περιλαμβάνει και ονοματικά.

Δεδομένου του ονοματικού $\{i_2\}$, το RACER δημιουργεί μια νέα, συνώνυμη έννοια l_2 και καθιστά το i_2 στιγμιότυπο της l_2 . Στη συνέχεια αντικαθιστά τον περιορισμό `hasValue` με έναν υπαρξιακό ποσοτικοποιητή στην έννοια l_2 και επομένως είναι αδύνατο να συμπεράνει ότι η σχέση $R(i_1, i_2)$ πράγματι ισχύει.



Εικόνα 5.6: Παράδειγμα συμπερασμού με χρήση υπαρξιακής ποσοτικοποίησης και ονοματικών.

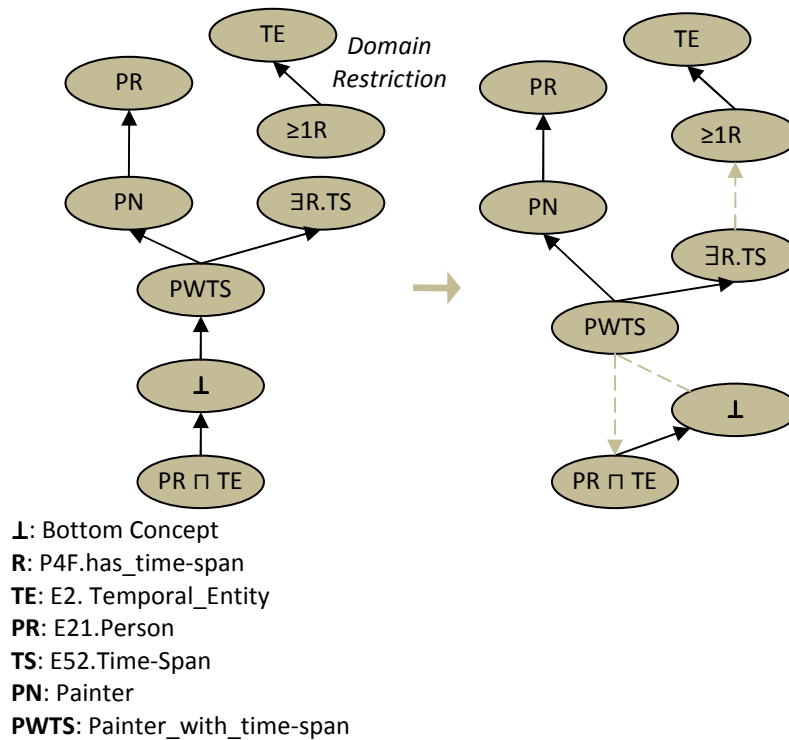
5.5.2. Αρνητικοί συμπερασμοί

Στο CRM, τα χρονικά γεγονότα (temporal events) μπορεί να έχουν ένα χρονικό εύρος (time-span). Φυσιολογικά, ένα 'Person' (άνθρωπος) δεν μπορεί να έχει time-span, εκτός και αν είναι ταυτόχρονα και χρονική οντότητα. Στη συνέχεια δηλώνουμε ότι τα 'Person' και 'Temporal_Entity' είναι ξένες μεταξύ τους έννοιες και επιχειρούμε να ορίσουμε την κλάση των 'Ζωγράφων με χρονικό εύρος' ('Painter_with_time-span').

```
<owl:ObjectProperty rdf:ID="P4F.has_time-span">
  <rdfs:domain rdf:resource="#E2.Temporal_Entity"/>
</owl:ObjectProperty>
<owl:Class rdf:about="#E2.Temporal_Entity">
  <owl:disjointWith rdf:resource="#E21.Person"/>
</owl:Class>
<owl:Class rdf:about="#Painter">
  <rdfs:subClassOf rdf:resource="#E21.Person"/>
</owl:Class>
<owl:Class rdf:ID="Painter_with_time-span">
  <rdfs:subClassOf rdf:resource="#Painter"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#E21.P4F.has_time-span"/>
      <owl:someValuesFrom rdf:resource="#E52.Time-Span"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

Το παραπάνω απόσπασμα απεικονίζεται στην εικόνα 5.7. Ένας 'Painter_with_time-span' ορίζεται ως 'Painter' (γνωστή υποκλάση του 'Person') που 'has_time-span' κάποια στιγμιότυπα 'Time-Span'. Ωστόσο, τα άτομα που έχουν 'time-span' απαιτείται να ανήκουν στην κλάση 'Temporal_Entity', όπως επιβάλλει ο αντίστοιχος περιορισμός πεδίου ορισμού. Επομένως, εκτός από το να είναι 'Person', ένας 'Painter_with_time-span' πρέπει να είναι και 'Temporal_Entity'. Από την άλλη τα 'Person' και 'Temporal_Entity' είναι ξένα, άρα η τομή τους ισοδυναμεί με την κατώτατη (πάντα κενή) έννοια. Έτσι, ένας 'Painter_with_time-

span' δεν μπορεί να υφίσταται, εφόσον η κλάση του συμπεραίνεται ισοδύναμη με την κατώτατη έννοια. Η KDI εντοπίζει σωστά την μη ικανοποιησιμότητα της κλάσης αυτής και την εμφανίζει με κόκκινο χρώμα στην ταξινόμια.



Εικόνα 5.7: Εντοπισμός μη ικανοποιήσιμων εννοιών.

5.5.3. Ένα σενάριο χρήσης

Στην ενότητα αυτή παρουσιάζεται ένα σενάριο χρήσης, που ωφελείται από την προσέγγιση της σημασιακής προσαρμογής και εμπλέκει το σημασιακό προφίλ που περιγράφηκε στην ενότητα 5.4. Το σενάριο αυτό ασχολείται με τον κόσμο των ζωγράφων και των πινάκων ζωγραφικής. Προς το σκοπό αυτό, συνδέουμε τις οντολογίες με πραγματικούς ψηφιακούς πόρους χρησιμοποιώντας αναφορές URI μέσα στα OWL έγγραφα που δημιουργήθηκαν, μιμούμενοι έτσι μια υποτιθέμενη ψηφιακή συλλογή. Οι πόροι αυτοί περιλαμβάνουν ψηφιακά αντίγραφα των πινάκων του Mondrian.

Δείχνεται πώς, μέσα από μια σειρά συμπερασμών, η KDI μπορεί να βοηθήσει το χρήστη να ανακαλύψει χρήσιμες πληροφορίες σε μορφή κειμένου, καθώς και άλλο ψηφιακό περιεχόμενο (μια ψηφιακή φωτογραφία). Στο σενάριο αυτό, θεωρείται δεδομένο ότι φυσικά, οι πόροι που πρόκειται να ανακαλυφθούν δεν έχουν περιγραφεί πλήρως, δηλαδή λείπουν τμήματα από το μονοπάτι της ανακάλυψης, τα οποία πρόκειται να καλυφθούν από συμπερασμούς. Είναι προφανές ότι μια τέτοια κατάσταση, όπως αναπαράγεται εδώ, είναι αντιπροσωπευτική της ημιδομημένης και ελλιπούς (αν και πολλαπλασιαστικής) φύσης των καταναμημένων στον Ιστό πληροφοριών και μεταδεδομένων.

Η έμφαση δίνεται κυρίως όχι στην αλληλεπίδραση με την KDI, αλλά στους συμπερασμούς που λαμβάνουν χώρα και παρέχεται, για λόγους συντομίας, το ισοδύναμο, σε Λογικές Περιγραφής, των OWL αποσπασμάτων που εμπλέκονται σε κάθε συμπερασμό.

OWL Έγγραφο	Απόσπασμα σε ΛΠ σύνταξη
mondrian.owl	$\&crm;P107B.is_current_or_former_member_of$ (Mondrian, De Stijl)
cidoc_crm_v3.4.owl	$\top \sqsubseteq \forall \&crm;P107B.is_current_or_former_member_of.\&crm;E74.Group$
Υποβολή ερωτήματος και απάντηση	
Στιγμιότυπο	De Stijl
Ρόλος	isA
Αποτέλεσμα	$\&crm;E74.Group$ / $\&crm;E39.Actor$ $\&crm;E77.Persistent_Item$ $\&crm;E1.CRM_Entity$

Πίνακας 5.2: «Τι είναι το De Stijl;»

Ας υποθέσουμε αρχικά ότι ένας χρήστης, που ασχολείται με τις καλές τέχνες, γνωρίζει ότι υπήρξε κάποτε ένα διάσημο καλλιτεχνικό ρεύμα, γνωστό ως 'De Stijl', αλλά δε γνωρίζει πολλά για αυτό. Πρώτα, ζητά από την KDI να βρει «τι είναι ('isA') το De Stijl», με άλλα λόγια, να διεξάγει έλεγχο στιγμιοτύπων σε αυτό (πίνακας 5.2).

Εξαιτίας του περιορισμού πεδίου ορισμού, το 'De Stijl' συμπεραίνεται ότι είναι ένα 'Group'. Ποια ήταν λοιπόν τα μέλη του; Ζητείται από την KDI να επιστρέψει την αντίστροφη σχέση ως προς την 'P107B.is_current_or_former_member_of' χρησιμοποιώντας το 'De Stijl' ως όρισμα (πίνακας 5.3).

OWL Έγγραφο	Απόσπασμα σε ΛΠ σύνταξη
mondrian.owl	$\&crm;P107B.is_current_or_former_member_of$ (Mondrian, De Stijl)
crm_core_profile.owl	$\&crm;P107B.is_current_or_former_member_of \bar{\equiv}$ $\&crm;P107F.has_current_or_former_member$
Υποβολή ερωτήματος και απάντηση	
Στιγμιότυπο	De Stijl
Ρόλος	$\&crm;P107F.has_current_or_former_member$
Αποτέλεσμα	<i>Mondrian</i>

Πίνακας 5.3: «Ποια είναι τα μέλη του De Stijl;»

Ο Mondrian αποδεικνύεται ότι ήταν σημαντικό μέλος του 'De Stijl', λόγω της δήλωσης inverseOf στο προφίλ πυρήνα. Ο χρήστης γνωρίζει ότι ο Mondrian ήταν καλλιτέχνης, αλλά με τι να έχει ασχοληθεί; Ήταν γλύπτης, μουσικός ή ζωγράφος;

OWL Έγγραφο	Απόσπασμα σε ΛΠ σύνταξη
crm_core_profile.owl	$\&crm;P14F.carried_out_by \bar{\equiv} \&crm;P14B.performed$ Painting $\equiv \exists \&crm;P2F.has_type.\{painting_composition\}$
crm_paint_profile.owl	$\exists \&crm;P94F.has_created.Painting \sqsubseteq Painting_Event$ $\#Painter \equiv \exists \&crm;P14B.performed.Painting_Event$
mondrian.owl	$\&crm;P94F.has_created$ (Creation of composition1, composition1) $\&crm;P2F.has_type$ (composition1, $\&crm;p;painting_composition$) $\&crm;P14F.carried_out_by$ (Creation of composition1, Mondrian)
Υποβολή ερωτήματος και απάντηση	
Στιγμιότυπο	Mondrian
Ρόλος	isA
Αποτέλεσμα	$\&crm;p;Painter$ /

Πίνακας 5.4: «Ποιο είναι το επάγγελμα του Mondrian;»

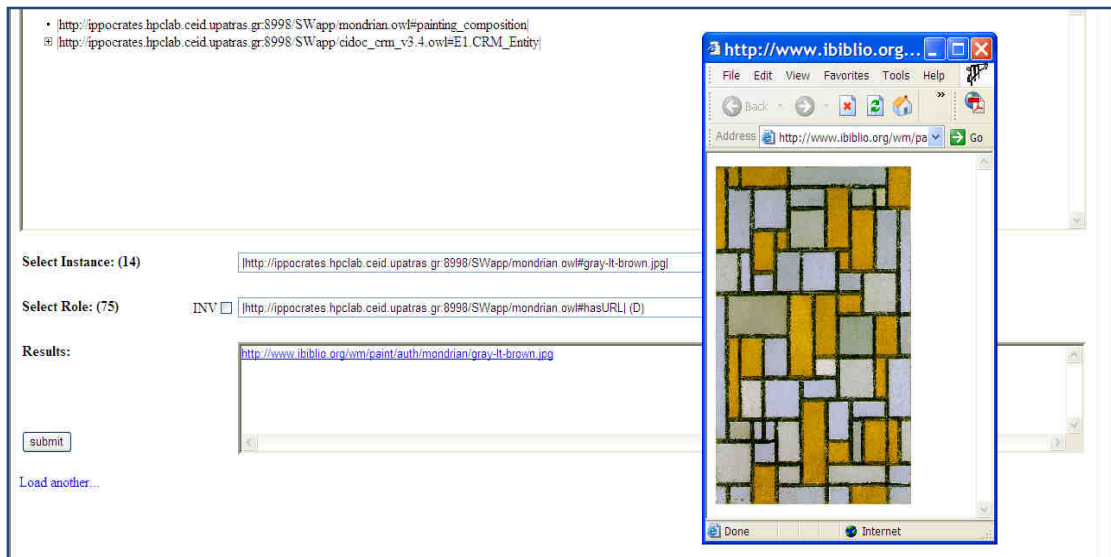
Όπως φαίνεται στον πίνακα 5.4, ύστερα από μια αρκετά περίπλοκη διαδικασία συλλογισμού, που εμπλέκει επίσης και ονοματικά, ο Mondrian βρίσκεται ότι είναι πράγματι

‘Painter’. Το ‘Creation of composition1’ έχει ‘carried_out_by’ από τον Mondrian και το ίδιο ‘has_created’ την composition1 που ‘has_type’ ‘painting_composition’. Λόγω αυτού και του ονομαστικού περιορισμού στο ‘has_created’ (παρομοίως προς την εικόνα 5.6), η composition1 συμπεραίνεται ότι είναι μέλος της κλάσης ‘Painting’ και επομένως το ‘Creation of composition1’ είναι ένα ‘Painting_Event’. Επιπλέον, ένας ‘Painter’ ορίζεται ως το ‘Person’ εκείνο που έχει ‘performed’ τουλάχιστον ένα ‘Painting_Event’ και το ‘carried_out_by’ είναι το αντίστροφο του ‘performed’, οπότε και προκύπτει το ζητούμενο συμπέρασμα.

Να σημειωθεί ότι πουθενά στην οντολογία δεν ορίζεται ότι ο Mondrian είναι ‘Painter’. Χωρίς αυτή τη διαδρομή του συλλογισμού, θα ήταν απλώς ένας πόρος/στιγμιότυπο δίχως τύπο. Επιπρόσθετα, όλες αυτές οι λεπτομέρειες αποκρύπτονται από τον χρήστη.

Έχοντας βρει το επάγγελμά του, ο χρήστης επιθυμεί τώρα να μάθει περισσότερα για τα έργα του Mondrian. Κάνει κλικ πάνω στο προηγούμενο αποτέλεσμα (‘Painter’) και αυτόματα στην ιεραρχία επιλέγεται η κλάση ‘Painter’. Αυτό έχει ως αποτέλεσμα ότι οι διαθέσιμοι ρόλοι περιορίζονται μόνο στους πιο σχετικούς, δηλαδή σε αυτούς που έχουν την επιλεγμένη κλάση ή τις υπάγουσες αυτήν στο πεδίο ορισμού τους και εμφανίζονται βάσει αναζήτησης πρώτα κατά βάθος.

Βάσει των δηλώσεων του πίνακα 5.4, ο χρήστης βρίσκει ότι η composition1 σχετίζεται με τον Mondrian και ότι είναι μία ‘Painting’, γνωστή υποκλάση του ‘Visual_Item’. Πού μπορεί λοιπόν κανείς να τη δει; Ένα ερώτημα αντίστροφης σχέσης στο ‘shows_visual_item’ αποκαλύπτει το όνομα αρχείου ‘gray-It-brown.jpg’. Η εύρεση της τοποθεσίας του αρχείου αυτού είναι άμεση, χρησιμοποιώντας την ιδιότητα τύπων δεδομένων ‘hasURL’, που επιστρέφει έναν σύνδεσμο ο οποίος, αν ακολουθηθεί, οδηγεί στην εικόνα 5.8.



Εικόνα 5.8: Ανάκτηση δικτυακού πόρου με χρήση ιδιότητας τύπων δεδομένων.

5.6. Στοχευμένη προσαρμογή

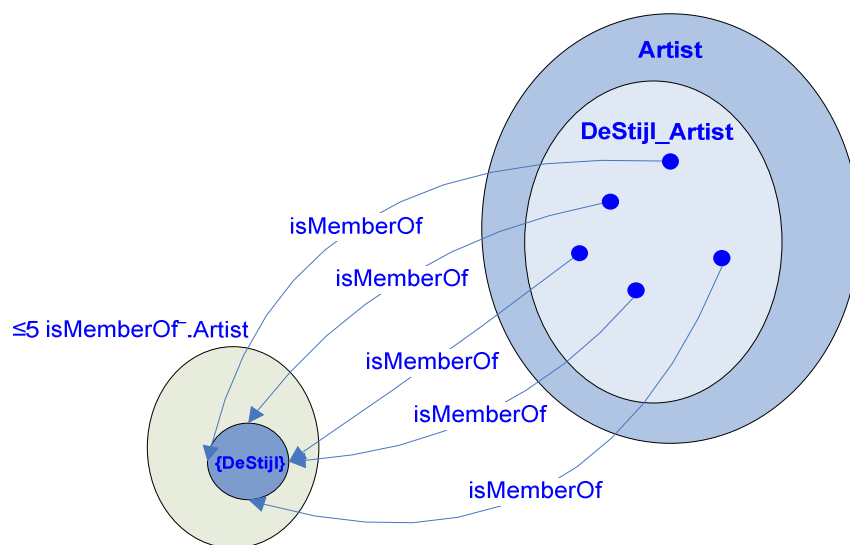
Είδαμε, στην ενότητα 4.6.1, ότι ο μη ντετερμινισμός που εισάγεται από τις διαδικασίες απόφασης ταμπλό είναι κυρίως υπεύθυνος για την αυξημένη πολυπλοκότητα του

συλλογισμού σε εκφραστικές Λογικές Περιγραφές και άρα στην OWL. Στην περίπτωση της OWL DL, τα ονοματικά και ιδιαίτερα ο συνδυασμός τους με αντίστροφους ρόλους και περιορισμούς πληθικότητας (αριθμού) είναι αυτά που δημιουργούν τη μεγαλύτερη επιβάρυνση, μέσω της εφαρμογής του *NN*-κανόνα.

Η τεχνική της σημασιακής προσαρμογής, εφόσον επιτυγχάνει την αύξηση της εκφραστικότητας ενός μοντέλου, είναι αναπόφευκτο ότι θα επιφέρει και αντίστοιχη επιβάρυνση της πολυπλοκότητας. Ωστόσο, η μελέτη των διαδικασιών απόφασης ταμπλό αποκάλυψε ότι η επιβάρυνση αυτή μπορεί να έρθει *κλιμακωτά*, ανάλογα με το πραγματικό επίπεδο εκφραστικότητας που αξιοποιείται από την οντολογία. Κατά την εφαρμογή της τεχνικής στην περίπτωση του CIDOC-CRM επομένως, έχει καταβληθεί προσπάθεια, από τη μία, να αναβαθμιστεί η εκφραστικότητα σε επίπεδο OWL DL, μέσω της αναπαράστασης ονοματικών, αλλά από την άλλη όμως η χρήση τους να γίνεται στοχευμένα, ώστε να αποφεύγεται η αλληλεπίδραση των ονοματικών με άλλες εκφραστικές κατασκευές που οδηγούν στην εφαρμογή του *NN*-κανόνα.

Πράγματι, στη φάση της εντασιακής εμβάθυνσης του μοντέλου καθ' εαυτού, παρατηρούμε ότι τα ονοματικά αποφεύγονται εντελώς, καθότι ούτε αναγκαία είναι ούτε υπαγορεύονται από τις προδιαγραφές του μοντέλου, έστω και σε επίπεδο σύστασης ή σχολιασμού. Στην περίπτωση όμως της εκλέπτυνσης για τη συγκεκριμένη εφαρμογή των καλλιτεχνών και των ζωγράφων, τα ονοματικά αξιοποιούνται, μέσω της χρήσης του τελεστή `owl:hasValue`, για την έκφραση κλάσεων που ταιριάζουν φυσικά στην εν λόγω περιοχή, όπως π.χ. η έννοια του τι είναι πίνακας (κλάση 'Painting'). Αποφεύγεται εντούτοις ο περαιτέρω συνδυασμός των ονοματικών αυτών με περιορισμούς αριθμού και αντίστροφους ρόλους, «γλιτώνοντας» έτσι από μία σοβαρή πηγή μη ντετερμινισμού.

Επειδή είναι διάχυτος ο ισχυρισμός ότι ο παραπάνω συνδυασμός είναι «αφύσικος» σε ρεαλιστικές οντολογίες (βλ. π.χ. [Tsarkon, Horrocks, & Patel-Schneider, 2007]) στη συνέχεια δίνουμε ένα αντιπαράδειγμα, που αποδεικνύει πώς μια τέτοια αλληλεπίδραση μπορεί να είναι επιθυμητή και ακόμα, αναγκαία για την έκφραση συνηθισμένων εννοιών στον καλλιτεχνικό πεδίο.



Εικόνα 5.9: Αλληλεπίδραση ονοματικών, περιορισμών αριθμού και αντίστροφων ρόλων.

Το «De Stijl» («το στυλ») υπήρξε μια καλλιτεχνική κίνηση στην Ολλανδία, στις αρχές του προηγούμενου αιώνα, που φιλοσοφικά θεμελιωνόταν στον λεγόμενο *νεοπλαστικισμό*, πρεσβεύοντας μεταξύ άλλων, την αφαίρεση στην τέχνη. Γνωστότερο μέλος του De Stijl υπήρξε ο ζωγράφος Piet Mondrian, καθώς και άλλοι σχεδιαστές, ζωγράφοι και αρχιτέκτονες. Στην ενότητα 5.5.3 είδαμε ότι το De Stijl έχει οριστεί ως στιγμιότυπο και ο Mondrian συνδέεται με αυτό μέσω της σχέσης P107B.is_current_or_former_member_of. Ας υποθέσουμε λοιπόν ότι για το De Stijl γνωρίζουμε επιπλέον ότι έχει άλλα τέσσερα μέλη, αλλά όχι παραπάνω. Αυτό μπορεί να εκφραστεί σε σύνταξη Λογικών Περιγραφής ως εξής:

$$\{DeStijl\} \sqsubseteq \leq 5 \text{ P107F.has_current_or_former_member.Artist}$$

όπου {De Stijl} υποδηλώνει την *ονοματική* έννοια που περιλαμβάνει μόνο το στιγμιότυπο De Stijl και Artist την αφηρημένη κλάση όλων των καλλιτεχνών. Η κλάση όλων των καλλιτεχνών που ανήκουν σε αυτό το καλλιτεχνικό ρεύμα μπορεί επομένως να εκφραστεί ως:

$$DeStijl_Artist \sqsubseteq Artist \sqcap \exists \text{ P107B.is_current_or_former_member_of.}\{DeStijl\}.$$

Οι σχέσεις που υπονοούν τα παραπάνω αξιώματα μπορούν να εκφραστούν σχηματικά στην εικόνα 5.9. Οι σχέσεις P107F και P107B είναι φυσικά αντίστροφες, ισχύει δηλαδή ότι:

$$\text{P107F.has_current_or_former_member} \equiv \text{P107B.is_current_or_former_member_of}^{-1}.$$

Επειδή όμως $DeStijl_Artist \sqsubseteq Artist$, τα παραπάνω αξιώματα περιορίζουν έμμεσα τον αριθμό των στιγμιότυπων που μπορεί να έχει η κλάση DeStijl_Artist σε πέντε το πολύ, γεγονός που μπορεί να οδηγήσει σε μη δενδρικές δομές στο γράφημα πλήρωσης.

Έχουμε λοιπόν δημιουργήσει μια κατάσταση, με την έκφραση εννοιών συνηθισμένων στο πεδίο αναφοράς, που αναπαράγει αυτήν που αναφέρεται στην ενότητα 3.7.1 και οδηγεί νομοτελειακά στην εφαρμογή του *NN*-κανόνα. Μάλιστα, αξίζει να αναφερθεί ότι αν τα μέλη του De Stijl δεν ήταν το πολύ πέντε, αλλά π.χ. 100, η συμπεριφορά του αλγορίθμου ταμπλό θα ήταν ακόμα χειρότερη, αφού ο αριθμός αυτός εμφανίζεται στην βάση του (διπλού) εκθέτη της πολυπλοκότητας (στην χειρότερη περίπτωση θα έπρεπε να δοκιμαστούν 100 διαφορετικά μονοπάτια, με χρήση υπαναχώρησης).

6. ΣΗΜΑΣΙΑΚΗ ΠΡΟΣΑΡΜΟΓΗ ΚΑΙ ΗΜΙΔΟΜΗΜΕΝΑ ΠΕΔΙΑ ΓΝΩΣΗΣ

6.1. Εισαγωγή

Τα μεταδεδομένα αποτελούν σήμερα έναν από τους πιο διαδεδομένους τρόπους για τη διευκόλυνση της περιγραφής, της ολοκλήρωσης, της ανακάλυψης και της διατήρησης πληροφοριών και πόρων που είναι αποθηκευμένοι σε απομακρυσμένες βάσεις δεδομένων ή φιλοξενούνται σε πύλες Διαδικτύου και ψηφιακές βιβλιοθήκες. Ένας από τους λόγους που το σχήμα Dublin Core (DC) υιοθετείται ευρέως σε τέτοιες περιπτώσεις είναι πιθανώς η απλότητα και η γενική του εφαρμοσιμότητα, που το καθιστά κατάλληλο για ένα πλήθος διαφορετικών, εξαρτώμενων από μεταδεδομένα εφαρμογών.

Σε πολλές τυπικές προσεγγίσεις (όπως για παράδειγμα, στο λογισμικό ψηφιακών αποθεμάτων DSpace), το Σύνολο Πεδίων Μεταδεδομένων DC (DC Metadata Element Set, DCMES) υλοποιείται ως μια επίπεδη συνάθροιση πεδίων. Αυτό είναι επίσης αληθές και για τις εξειδικεύσεις (qualifiers) που δεν υλοποιούνται πάντα ως υπο-ιδιότητες των βασικών πεδίων. Αντίθετα, εμφανίζονται συχνά στο ίδιο επίπεδο με τα πεδία-γονείς και η σχέση υπο-πεδίου/εξειδίκευσης διατηρείται μόνο στις ετικέτες. Η κατάσταση αυτή, η οποία είναι επίσης εμφανής στην περίπτωση του, βασισμένου στο DSpace, ιδρυματικού αποθέματος του Πανεπιστημίου Πατρών, απεικονίζεται στην εικόνα 6.1. Με άλλα λόγια, και σε αντίθεση με την εργασία πάνω στο CIDOC-CRM που εξετάστηκε στο προηγούμενο κεφάλαιο, στη συγκεκριμένη περίπτωση αντιμετωπίζεται ένα πεδίο που είναι στην πραγματικότητα μερικώς δομημένο σημασιακά, αποτελεί δηλαδή ένα ημιδομημένο πεδίο γνώσης, χαρακτηριστικό των εφαρμογών πάσης φύσεως μεταδεδομένων.

DC Field	Value	Language
contributor.author	Κουτσομητρόπουλος, Δημήτριος	el
contributor.author	Koutsomitropoulos, Dimitrios	en
date.accessioned	2006-12-15T13:13:49Z	-
date.available	2006-12-15T13:13:49Z	-
date.issued	2006-12-15T13:13:49Z	-
identifier.uri	http://hdl.handle.net/1987/104	-
description	Εργαστήριο Πληροφοριακών Συστημάτων Υψηλών Επιδώσεων	el
description	HPCLab	en
description.abstract	Στην παρούσα παρουσίαση περιγράφονται ενέργειες προετοιμασίας πρωτοτύπων που στόχο έχουν τη διατήρηση και τη διάσωση της ποιότητας του περιεχομένου. Παράμετροι που αναφέρονται και αναλύονται είναι η καταλληλότητα του εξοπλισμού και των συνθηκών περιβάλλοντος και η μεταχείριση πρωτοτύπων.	el
description.abstract	The presentation above describes preparation actions that intent to the preservation and the rescue of the content quality. Factors like equipment and environment appropriateness as well as prototype usage are analytically adverted.	en

Εικόνα 6.1: Λεπτομερής προβολή στοιχείου στο DSpace.

Η σημασιακή ερμηνεία του μοντέλου DC, που, από ό,τι φαίνεται, δεν αναπαρίσταται πάντα στις εφαρμογές, τυποποιείται μέσα από την προδιαγραφή του *Αφηρημένου Μοντέλου DCMI* (DCMI Abstract Model, DCAM) [Powell, Nilsson, Naev, Johnston, & Baker,

2007], καθώς επίσης και από την πολύ πρόσφατη σύσταση για την έκφραση του DC σε RDF [Nilsson, Powell, Johnston, & Naeve, 2008]. Οι προδιαγραφές αυτές, που συζητούνται στην επόμενη ενότητα, προτείνουν ουσιαστικά μια *οντολογία* για το DC, εκφρασμένη σε RDF(S).

Μια τέτοια οντολογία DC φέρει τη δική της σημασιολογική δομή, η οποία μπορεί να αξιοποιηθεί για την ενεργοποίηση πιο εκλεπτυσμένων περιγραφών των πόρων. Αυτό βέβαια θυμίζει το γνωστό πρόβλημα *αυτοδύναμης εκκίνησης* (bootstrapping) για το Σημαντικό Ιστό (βλ. π.χ. [Dill, et al., 2003; Hendler J., 2008]): Η ύπαρξη υψηλού επιπέδου, σύνθετων και συνδεδεμένων μεταξύ τους περιγραφών πόρων είναι βασικής σημασίας για να έχει κάποια αξία ο Σημαντικός Ιστός. Από τη άλλη, το βάρος της εκ νέου δημιουργίας πλούσιων σχολιασμών είναι πολύ μεγάλο, τόσο από εννοιολογική σκοπιά (δύσκολο να συλληφθούν), όσο και από την άποψη της μεγάλης προσπάθειας που θα πρέπει να καταβληθεί (πολύς χρόνος). Είναι απίθανο ότι υπάρχουσες εφαρμογές, που ήδη εφαρμόζουν επίπεδες περιγραφές για έναν μεγάλο αριθμό πόρων, θα επενδύσουν στην αναδιοργάνωση από το μηδέν του υποκείμενου μοντέλου δεδομένων που χρησιμοποιούν. Ακόμα κι αν αναληφθεί ένα τέτοιο εγχείρημα, το κόστος για την ευθυγράμμιση (alignment) και τον εμπλουτισμό των υπάρχουσών περιγραφών μπορεί να είναι απαγορευτικό.

Έχοντας υπόψη τα παραπάνω, έχει ιδιαίτερο ενδιαφέρον να διερευνηθεί η ορθότητα της υπόθεσης:

Υπόθεση 1^α: *Η SWKD μπορεί να εφαρμοστεί αποδοτικά και εκφραστικά στην περίπτωση σημασιακά ημιδομημένων μοντέλων γνώσης.*

Έτσι, στο κεφάλαιο αυτό, προτείνεται η υλοποίηση μιας οντολογίας για το DC που διεξάγεται ακολουθώντας μια, κατά το δυνατόν, κεντρικοποιημένη προσέγγιση. Για να επιτευχθεί αυτό, χρησιμοποιείται η τεχνική της σημασιακής προσαρμογής, η οποία εισήχθη στο προηγούμενο κεφάλαιο και εφαρμόστηκε με επιτυχία σε πλήρως δομημένα πεδία γνώσης, όπως το CIDOC-CRM. Με τη χρήση της τεχνικής αυτής επιχειρείται η καλύτερη αποτύπωση της επιδιωκόμενης σημασιολογίας του πεδίου μεταδεδομένων DC, έχοντας ως βάση το RDF(S) σχήμα του DC. Στόχος είναι η αναβάθμιση της οντολογίας σε επίπεδο OWL και OWL 1.1, υιοθετώντας τους νέους κατασκευαστές και τις εκλεπτύνσεις που είναι διαθέσιμες μόνο σε αυτές τις γλώσσες. Ταυτόχρονα, η τεχνική «χτίζει» πάνω στο αρχικό μοντέλο, με συνέπεια να μην απαιτούνται οποιεσδήποτε αλλαγές στις αρχικές προδιαγραφές. Με άλλα λόγια, η παραπάνω υπόθεση γίνεται ισοδύναμη με την:

Υπόθεση 2^α: *Η τεχνική της σημασιακής προσαρμογής επιτυγχάνει την εντασιακή αποτύπωση και επέκταση ενός σημασιακά ημιδομημένου μοντέλου, χωρίς να βλάπτεται η γενική του εφαρμοσιμότητα.*

Στη συνέχεια η προκύπτουσα οντολογία, μαζί με τις εκλεπτύνσεις, ενημερώνεται, με αυτοματοποιημένο τρόπο, από μεταδεδομένα DC που είναι ήδη διαθέσιμα στην εγκατάσταση DSpace του ιδρυματικού αποθέματος του Πανεπιστημίου Πατρών (<http://repository.upatras.gr/dspace/>).

Ως αποτέλεσμα, μπορούν να υποστηριχθούν επερωτήσεις στο οντολογικό μοντέλο (μαζί με τα στιγμιότυπά του) οι οποίες:

- α. επιδεικνύουν ίδιες δυνατότητες ανάκτησης με το αρχικό μοντέλο, είναι δυνατόν δηλαδή να υποβληθούν ανάλογα ερωτήματα και να ανακτηθούν ταυτόσημα αποτελέσματα και
- β. αποκτούν έμμεση πληροφορία, η οποία είναι αδύνατον να ανακτηθεί χρησιμοποιώντας την υπάρχουσα διαμόρφωση, αξιοποιώντας τις πλουσιότερες σημασιακές κατασκευές της OWL 1.1 (όπως για παράδειγμα, οι αλυσίδες ρόλων).

Επιπρόσθετα, καθιστώντας διαθέσιμη την προκύπτουσα οντολογία ως δικτυακό έγγραφο, ανοίγει ο δρόμος για τη *σημασιακή διαλειτουργικότητα*, για παράδειγμα, τη συμπερίληψη της οντολογίας αυτής στα οντολογικά έγγραφα άλλων αποθεμάτων και την υλοποίηση επερωτήσεων με σημασιακά κατανεμημένο τρόπο [Solomou, Koutsomitropoulos, & Paratheodorou, 2008].

Στη συνέχεια, γίνεται μια επισκόπηση της τρέχουσας υλοποίησης του DC σε RDF και της υποκείμενης σημασιολογίας του, όπως επίσης και ορισμένων προσεγγίσεων για την έκφραση του DC σε OWL. Κατόπιν, περιγράφεται η διαδικασία για την δημιουργία μιας οντολογίας Ιστού DC: συζητούνται οι επιλογές για την αντιμετώπιση των προβλημάτων που προκύπτουν, παρουσιάζονται τα βήματα της υλοποίησης και παρέχονται ενδεικτικά παραδείγματα των παρεμβάσεων που έγιναν. Μετά, παρουσιάζεται επιγραμματικά η τεχνική διαμόρφωση για την αυτόματη ενημέρωση της οντολογίας με πραγματικά δεδομένα και σχολιάζονται οι νέες δυνατότητες που καθίστανται εφικτές, μέσω της παράθεσης ορισμένων βασισμένων σε συλλογισμό επερωτήσεων, οι οποίες διεξήχθησαν μέσα από το περιβάλλον Protégé. Ακολουθεί το αποτέλεσμα για την αναγωγή και κλασικών προφίλ μεταδεδομένων, που μπορεί να βασίζονται στο DC, σε οντολογίες και Λογικές Περιγραφές.

Το κεφάλαιο αυτό βασίζεται και επεκτείνει την εργασία [Koutsomitropoulos, Solomou, & Paratheodorou, submitted].

6.2. RDF υλοποιήσεις και σημασιολογία του Dublin Core

Στην ενότητα αυτή επιχειρείται να γίνει μια επισκόπηση των υλοποιήσεων του DC και της σημασιολογίας που φέρει. Πρώτα εξετάζεται η σχέση του με το DCAM, και μέσα από μια ιστορική αναδρομή, επισημαίνεται η αναγκαιότητα της υλοποίησης σε RDF(S). Στη συνέχεια εξετάζονται οι ήδη διαθέσιμες υλοποιήσεις σε OWL και σχολιάζονται οι τρόποι προσέγγισης που ακολουθήθηκαν σε αυτές. Τέλος παρουσιάζονται συνοπτικά οι σχετικές εργασίες για την αντιστοίχιση του DC στο CIDOC-CRM και για τη δημιουργία παραδοσιακών προφίλ μεταδεδομένων.

6.2.1. DC και RDF

Οι προσπάθειες για την αποτύπωση της εγγενούς σημασιολογίας του DC ξεκίνησαν το 2005 με την προδιαγραφή του DCAM. Το DCAM, που στην πραγματικότητα αποτελεί μια αφηρημένη μετα-μοντελοποίηση που φέρει τη σημασιολογία του RDF(S), ενημερώθηκε τελευταία φορά τον Ιούνιο του 2007. Η ισοδυναμία μεταξύ των εννοιών του DCAM και των

αντίστοιχων εννοιών του RDF(S) δίνεται στο [Powell, Nilsson, Naeve, Johnston, & Baker, 2007] και παρουσιάζεται στον ακόλουθο πίνακα.

DCMI Abstract Model	RDF/RDFS
<i>resource</i>	Κλάση rdfs:Resource
<i>property</i> ή <i>element</i>	Κλάση rdf:Property
<i>class</i>	Κλάση rdfs:Class
<i>syntax encoding scheme</i>	Κλάση rdfs:Datatype
σχέση <i>has domain</i>	Ιδιότητα rdfs:domain
σχέση <i>has range</i>	Ιδιότητα rdfs:range
σχέση <i>sub-property of</i>	Ιδιότητα rdfs:subPropertyOf
σχέση <i>sub-class of</i>	Ιδιότητα rdfs:subClassOf
<i>plain value string</i>	Απλό λεκτικό
<i>typed value string</i>	Λεκτικό με τύπο

Πίνακας 6.1: Αντιστοιχία μεταξύ DCAM και RDF(S).

Αν και το DC είχε ήδη οριστεί μέσω των XML και RDF συντακτικών, χρειάστηκαν σχεδόν έξι χρόνια για να ενσωματωθεί πλήρως και να υλοποιηθεί το DCAM σε αυτές τις προδιαγραφές [Nilsson, Powell, Johnston, & Naeve, 2008]. Η καινούργια αυτή σύσταση για την έκφραση του DC σε RDF, που λαμβάνει υπόψη τις κατασκευές του DCAM, έρχεται να αντιμετωπίσει μια σειρά προβλημάτων που είχαν οι παλιότερες προδιαγραφές, συμπεριλαμβανομένων [Nilsson & Baker, 2008]:

- Της ύπαρξης δύο παλιών προδιαγραφών έκφρασης του DC σε RDF ([Beckett, Miller, & Brickley, 2002] και [Kokkelink & Schwänzl, 2002]).
- Αντικρουόμενων συστάσεων στα παραπάνω έγγραφα αναφορικά με τη χρήση συμβολοσειρών λεκτικών στα μεταδεδομένα DC.
- Της πολυπλοκότητας υλοποίησης λόγω της χρήσης εξειδικευμένων (idiosyncratic) κατασκευών.
- Των δυσκολιών στην παροχή τυπικών πεδίων ορισμού και τιμών για τις ιδιότητες του DC.

Η καινούργια επίσημη προδιαγραφή επιχειρεί να αντιμετωπίσει τα παραπάνω θέματα μέσω:

- Του ορισμού των πεδίων DC ως ιδιοτήτων.
- Της παροχής τυπικών περιορισμών πεδίου ορισμού και τιμών για τις ιδιότητες [Powell, Johnston, & Baker, 2008].
- Του ορισμού RDF κλάσεων για τα πεδία ορισμού και τιμών και της οργάνωσής τους σε ιεραρχίες.
- Της προδιαγραφής ιδιοτήτων τύπων δεδομένων, αλλά και ιδιοτήτων αντικειμένων (σε όρους OWL), βάσει του καθορισμένου πεδίου τιμών.

Επομένως, είναι φανερό ότι το DCAM, σε συνδυασμό με τη νέα υλοποίησή του επιχειρούν να φέρουν το DC πιο κοντά στην πραγματικότητα του Σημαντικού Ιστού.

6.2.2. Υλοποιήσεις σε OWL

Το κενό αυτό, ανάμεσα σε ένα δικτυακά προσανατολισμένο πρότυπο περιγραφής πόρων και στο Σημαντικό Ιστό έχει αναγνωρισθεί και προηγουμένα. Το ινστιτούτο AIFB του πανεπιστημίου της Καρλσρούης διατηρεί μια υλοποίηση του DC σε OWL (<http://www.aifb.uni-karlsruhe.de/WBS/rvo/ontologies/>). Σε αυτήν, ακολουθείται μια εξελικτική προσέγγιση όπου τελικά φιλοξενείται η πλήρης σημασιολογία του DC, καθιστώντας δυστυχώς την οντολογία μη αποφασίσιμη. Συγκεκριμένα:

- Ορίζονται ιδιότητες που χρησιμοποιούνται στη συνέχεια για την περιγραφή ιδιοτήτων και κλάσεων.
- Οι ιδιότητες αυτές ορίζονται ως ιδιότητες σχολιασμού (annotation properties).
- Τελικά, στην πλήρη εκδοχή, οι ιδιότητες σχολιασμού οργανώνονται σε μια ιεραρχία και τα πεδία λεξιλογίων ορίζονται επίσης και ως κλάσεις, καθιστώντας την οντολογία OWL Full.

Το πρόβλημα με την ανωτέρω τεχνική είναι ότι τα άτομα, οι ιδιότητες και οι κλάσεις δεν είναι πλέον ξένα μεταξύ τους, κάτι που αποτελεί μείζονα απαίτηση για την αποφασισιμότητα των οντολογιών OWL 1.0. Κρατώντας τις ιδιότητες αυτές μόνο ως σχολιασμούς είναι επίσης προβληματικό, γιατί, ενώ κάτι τέτοιο μπορεί να είναι επιθυμητό στις αυτοπεριγραφές (π.χ. ιδιότητες που περιγράφουν ιδιότητες), έχει φτωχή σημασιακή διερμήνευση και θα συνεισέφερε ελάχιστα, αν όχι καθόλου, στη διαδικασία του συλλογισμού.

Σε μια παρεμφερή προσπάθεια, η εταιρεία DRC Inc. προτείνει μια OWL οντολογία του DC (<http://orlando.drc.com/semanticweb/Topics/Ontology/Ontologies.asp>), όπου οι βασικές ιδιότητες του DCMES (δηλαδή χωρίς τις εξειδικεύσεις) υλοποιούνται ως ιδιότητες τύπων δεδομένων. Με τον τρόπο αυτό αποτυπώνεται καλύτερα η σημασιολογία του μοντέλου, από ότι αν υλοποιούνταν ως ιδιότητες σχολιασμού, ωστόσο οι πληρωτές των ιδιοτήτων θα πρέπει αναγκαστικά να είναι πάντα απλά λεκτικά.

Είναι φανερό ότι καμία από τις παραπάνω υλοποιήσεις δεν προσφέρει σημασιολογία πλουσιότερη από αυτή που εκφράζει το RDF(S). Επίσης η DRC στην πραγματικότητα υλοποιεί ένα νέο μοντέλο για το DC, οριζόμενο υπό διαφορετικό χώρο ονομάτων (dces-ont). Στην προσέγγιση που ακολουθείται στη διατριβή, επιχειρείται η διατήρηση και η επαναχρησιμοποίηση του αρχικού σχήματος με την προσθήκη (εισαγωγή) ενός προαιρετικού *ενδιάμεσου* (mediating) σημασιακού προφίλ, σε μια προσπάθεια να αποτυπωθεί πιστά η υπονοούμενη σημασιολογία του DC.

6.2.3. Άλλες προσεγγίσεις

Εκτός από τις προσεγγίσεις για δημιουργία οντολογιών DC που συζητήθηκαν παραπάνω, στο [Kakali, et al., 2007] προτείνεται μια μεθοδολογία αντιστοίχισης των πεδίων του DC σε έννοιες και σχέσεις του μοντέλου CIDOC-CRM. Οι συγγραφείς αναγνωρίζουν τις πλούσιες σημασιακές συσχετίσεις που ενυπάρχουν στο DC, καθώς και το σημασιακό χάσμα για την αποτελεσματική έκφρασή τους, ενώ εστιάζουν κυρίως στα εννοιολογικά προβλήματα που μια τέτοια αντιστοίχιση θέτει. Έτσι, κατά μία έννοια, επιτυγχάνεται η αναγωγή του DC σε επίπεδο οντολογίας, περιορισμένης όμως από τη συγκεκριμένη εννοιολογική σύλληψη που

αποτυπώνεται στο CIDOC-CRM και χωρίς να αξιοποιείται η υψηλότερη εκφραστικότητα (π.χ. χαρακτηριστικά ιδιοτήτων) που παρέχει η OWL.

Σχετικές επίσης είναι και οι ενέργειες που γίνονται μέσα από το DCMI για την τυποποίηση μιας διαδικασίας ορισμού και αναπαράστασης προφίλ μεταδεδομένων εφαρμογής, γνωστής ως «το πλαίσιο της Σιγκαπούρης» (the Singapore Framework) [Nilsson, Baker, & Johnston, 2008]. Στο πλαίσιο αυτό προτείνεται η έννοια των Προφίλ Συνόλων Περιγραφών (Description Set Profiles, DSP), ως βασικής δομικής μονάδας ενός προφίλ εφαρμογής. Μια πρώτη προσπάθεια για την προδιαγραφή μιας γλώσσας αναπαράστασης τέτοιων συνόλων, βασισμένης σε XML Schema, γίνεται στο [Nilsson M. , 2008]. Οπωσδήποτε, η γλώσσα αυτή δεν συνεισφέρει στην αύξηση της εκφραστικότητας ενός, ακόμα και ημιδομημένου, μοντέλου, αλλά ορίζει έναν τυπικό τρόπο για την αναπαράσταση κλασσικών προφίλ μεταδεδομένων. Στην ενότητα 6.5 εξετάζονται η σχέση και οι δυνατότητες έκφρασης της έννοιας των DSP με χρήση της OWL και δίνεται το (παράδοξο) αποτέλεσμα, ότι ακόμα και η OWL 1.1 δεν επαρκεί για το σκοπό αυτό.

6.3. Δημιουργία οντολογίας OWL για το Dublin Core

Προκειμένου να υλοποιηθεί η OWL οντολογία του DC, χρησιμοποιείται ως σημείο έναρξης η συνιστώμενη από το DCMI υλοποίηση σε RDF. Στη συνέχεια δημιουργείται ένα σημασιακό προφίλ εφαρμογής για το προκείμενο απόθεμα μεταδεδομένων, ακολουθώντας την τεχνική που παρουσιάστηκε στο προηγούμενο κεφάλαιο και των φάσεων της (ενότητα 5.4.3). Ακολουθώντας αυτήν την προσέγγιση επιτυγχάνεται:

- Ο σεβασμός και η επαναχρησιμοποίηση της αρχικής προδιαγραφής DC, παρέχοντας ένα σημασιακό προφίλ εφαρμογής κατάλληλο για τις ανάγκες του συγκεκριμένου πεδίου.
- Διατήρηση του εσωτερικού μοντέλου δεδομένων του αποθέματος ανέπαφου, εφαρμόζοντας τις υπηρεσίες διαλειτουργικότητας που παρέχει (OAI-PMH).
- Επαναχρησιμοποίηση των υπαρχουσών περιγραφών με αυτοματοποιημένο τρόπο, διαφανώς προς το χρήστη, χωρίς απαίτηση για πρόσθετους σχολιασμούς.

Στη συνέχεια παρουσιάζονται τα βήματα της υλοποίησης που ακολουθήθηκαν για τη δημιουργία της οντολογίας DC. Πρώτα όμως επιχειρείται να δοθεί απάντηση στο πρόβλημα της αποφασισιμότητας, όπως τέθηκε στην προηγούμενη ενότητα και συζητείται πώς η παρονομασία συνδέεται με την αντιμετώπισή του.

6.3.1. Ο αγώνας για αποφασισιμότητα

Όπως αναφέρθηκε, είναι κυρίως οι αυτοαναφορικές ιδιότητες αυτές που προκαλούν την κατάταξη μιας οντολογίας DC σε επίπεδο OWL Full. Ωστόσο, ακόμα και διαισθητικά, μοιάζει πολύ υψηλό ένα τέτοιο τίμημα, σε σχέση με τα πλεονεκτήματα που παρέχονται, δηλαδή απλοί κειμενικοί σχολιασμοί σε ιδιότητες και κλάσεις:

- Ιδιότητες που περιγράφουν ιδιότητες.
- Κλάσεις που αποτελούν στιγμιότυπα άλλων κλάσεων.

Στην πρώτη περίπτωση είναι ευνόητο πως δεν ενδιαφέρει τόσο τον χρήστη η ημερομηνία, για παράδειγμα, που κάποια ιδιότητα του DC άλλαξε κατάσταση σύστασης. Επομένως, είναι πιο φυσικό τέτοια χαρακτηριστικά να ειδωθούν ως σχόλια κειμένου. Σε πολλές υλοποιήσεις, οι οντολογίες OWL μπορεί να ενσωματώνουν πεδία DC (π.χ. `dc:contributor`) τα οποία θεωρούνται τέτοια σχόλια, χωρίς περαιτέρω σημασία.

Η δεύτερη περίπτωση αποτελεί ξεκάθαρη απαίτηση μετα-μοντελοποίησης και σπανίως η διπλή φύση μιας κλάσης (ταυτόχρονα ως στιγμιότυπο και ως κλάση) θα είναι απαραίτητη για την εξαγωγή χρήσιμων συμπερασμάτων. Και στις δύο περιπτώσεις, η ιδιότητα της παρονομασίας που φέρει η OWL 1.1 φαίνεται να αποτελεί πιο κομψή λύση στο πρόβλημα, διατηρώντας ταυτόχρονα την πολυπλοκότητα σε χαμηλά επίπεδα.

Η παρονομασία (βλ. ενότητα 2.5) είναι μια συντακτική σύμβαση βάσει της οποίας ένα όνομα που ορίζεται σε μια OWL 1.1 οντολογία μπορεί να διερμηνευτεί εναλλακτικά ως κλάση, ιδιότητα ή στιγμιότυπο, ανάλογα με τη χρήση του. Για παράδειγμα, η κλάση `Agent` που ορίζεται στο DCAM ως *στιγμιότυπο* της `AgentClass`, οδηγεί στην εμφάνιση, μέσα σε συμμορφούμενες με την παρονομασία υλοποιήσεις, μιας κλάσης και ενός ατόμου με το ίδιο όνομα 'Agent'. Η παρονομασία δεν έχει άλλες σημασιακές προεκτάσεις: για παράδειγμα, ένας συμβατός μηχανισμός συλλογισμού δεν είναι υποχρεωτικό ότι θα μπορεί να συμπεράνει πως το `Agent` είναι επίσης και *υποκλάση* της `AgentClass`.

Προκειμένου να αντιμετωπιστεί αυτό το εμπόδιο, φαίνεται να είναι διαθέσιμες οι ακόλουθες τέσσερις επιλογές:

1. Διπλασιασμός του ορισμού των ιδιοτήτων (πεδίων) του DC υπό χωριστούς χώρους ονομάτων.
2. Αποκλειστική χρήση παρονομασίας.
3. Ορισμός των ιδιοτήτων ως σχολιασμών.
4. Μίξη των παραπάνω τεχνικών.

Η επιλογή (4) σημαίνει να αναγνωριστούν οι πιο προβληματικές ιδιότητες (δηλαδή αυτές που χρησιμοποιούνται περισσότερο σε αυτοαναφορές, αλλά αναθέτουν μόνο λεκτικές τιμές) και να οριστούν ως σχολιασμοί ή να επαναοριστούν κάτω από διαφορετικούς χώρους ονομάτων. Έτσι οι ιδιότητες αυτές θα χρησιμοποιούνται απλώς ως σχόλια και δεν θα έχουν οντολογική ερμηνεία. Για τις υπόλοιπες, αυτές ορίζονται ως ιδιότητες αντικειμένου και τύπων δεδομένων. Οι συγκρούσεις που απομένουν αφήνεται να επιλυθούν από την παρονομασία.

Η λύση (2) σημαίνει να αφεθεί στην παρονομασία να αποφασίζει για όλες τις αμφισημίες, αλλά ο πειραματισμός μας έδειξε ότι αυτό έχει ασταθή συμπεριφορά στις υποστηρίζουσες υλοποιήσεις (π.χ. στο `Protégé 4`, που διαθέτει ενσωματωμένες ιδιότητες DC ως σχολιασμούς).

Η λύση (1) μπορεί να φαίνεται οριστική, αλλά έχει μεγάλο ρίσκο, γιατί ο ορισμός νέων χώρων ονομάτων θέτει σε κίνδυνο την εφαρμοσιμότητα του σχήματος και είναι ενάντια στην όλη φιλοσοφία της προσαρμογής. Επίσης γίνεται με κόστος προβλήματα προς τα πίσω συμβατότητας. Αυτό για παράδειγμα αντανακλά σε μεγάλο βαθμό στη φάση του συντακτικού μετασχηματισμού, βήμα (ε) (βλ. ενότητα 6.3.2), όπου χρειάστηκε να τροποποιήσουμε τις ρυθμίσεις συγκομιδής μεταδεδομένων (*harvesting*) του `DSpace` για το εξειδικευμένο DC: Αντικαταστήσαμε τον χώρο ονομάτων `dc:` με `dc:terms:` ακόμα και για

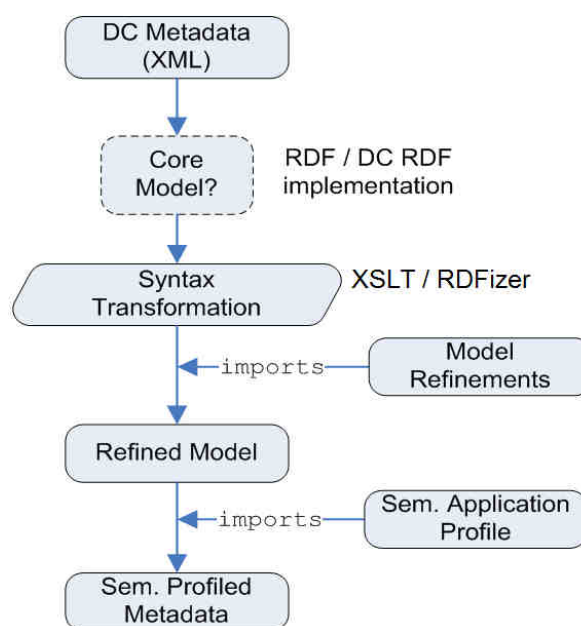
τα παραδοσιακά πεδία DC (για παράδειγμα `dc:contributor` → `dcterms:contributor`).

Παρά τους περιορισμούς αυτούς, αυτή είναι η λύση που ακολουθείται στη νέα σύσταση για το DC, τουλάχιστον για τους ορισμούς ιδιοτήτων. Συγκεκριμένα, παραδοσιακά πεδία `dc:`, όπως τα `contributor` και `creator` διπλασιάζονται, υπό τον υπάρχοντα χώρο ονομάτων `dcterms:`. Τα πεδία `dc:` μπορούν να δέχονται λεκτικές τιμές, ενώ οι ιδιότητες `dcterms:` δέχονται μη λεκτικά αντικείμενα (άτομα).

Κατά τη δημιουργία και προσαρμογή της οντολογίας DC υιοθετήθηκε αναπόφευκτα η παραπάνω λύση. Ωστόσο, φαίνεται ότι η παρονομασία, όπως υποστηρίζεται αυτή τη στιγμή, μπορεί να χειριστεί ικανοποιητικά αμφίσημες κλάσεις όπως η `Agent`. Ακόμη, το `rdf:Property`, βάσει του οποίου ορίζονται οι ιδιότητες `dcterms:`, μπορεί να διερμηνευθεί είτε ως ιδιότητα τύπου δεδομένων, είτε ως ιδιότητα αντικείμενου ανάλογα με τη χρήση του. Επιβεβαιώνεται επίσης ότι μπορεί να διερμηνευθεί και ως άτομο, κάτι που εξυπηρετεί όταν χρησιμοποιούνται άλλες ιδιότητες για την περιγραφή μιας ιδιότητας (αυτοαναφορές).

6.3.2. Βήματα υλοποίησης

Μια βασική διαφοροποίηση από την υλοποίηση του κεφαλαίου 5 είναι ότι εδώ, δεν υπάρχει ένα καλώς δομημένο πεδίο γνώσης, όπως αυτό που εκφράζεται από το CIDOC-CRM, αλλά τα διαθέσιμα στιγμιότυπα είναι ουσιαστικά μια επίπεδη συρραφή πεδίων. Η υλοποίηση του DC σε RDF μπορεί να αποτελεί θετικό σημείο αφετηρίας, αλλά παραμένει ένα χάσμα που πρέπει να γεφυρωθεί: απαιτείται δηλαδή η μετάβαση από την απουσία ενός σημασιακού μοντέλου σε μια κατάσταση *υποκείμενο-κατηγορημα-αντικείμενο* (subject-predicate-object), με στόχο την επίκληση της σημασιολογίας του RDF. Η μετάβαση αυτή παρέχεται από ένα XSLT (βλ. ενότητα 6.4.1) το οποίο μπορεί να θεωρηθεί μέρος της φάσης συντακτικού μετασχηματισμού της διαδικασίας σημασιακής προσαρμογής (εικόνα 5.4). Τα βήματα της υλοποίησης απεικονίζονται στην εικόνα 6.2.



Εικόνα 6.2: Βήματα υλοποίησης της οντολογίας DC.

Συντακτικός μετασχηματισμός

Ο μετασχηματισμός αυτός αναλύεται στα κάτωθι:

- α) Ένας ελάχιστος συντακτικός μετασχηματισμός για την «εκκαθάριση» των πλεονάζουσων δηλώσεων του OAI. Για παράδειγμα τα στοιχεία XML `<responseDate>` και `<request>` είναι συγκεκριμένα ως προς το OAI και δεν χρειάζονται. Το ίδιο ισχύει και για τις κεφαλίδες των εγγραφών.
- β) Ανάθεση τύπων (δεδομένων) στις λεκτικές τιμές. Το DC παρέχει μια αφαίρεση για τους τύπους δεδομένων, που ονομάζεται *σχήματα κωδικοποίησης σύνταξης* (syntax encoding schemes). Στο [DCMI Usage Board, 2008] η χρήση αυτών των σχημάτων είναι *προαιρετική*, δηλαδή συστήνεται ως καλή πρακτική και τα πεδία τιμών των αντίστοιχων ιδιοτήτων ορίζονται απλώς ως `rdfs:Literal`. Εντούτοις, τα σχήματα αυτά είναι στην πραγματικότητα ισοδύναμα με τους τύπους δεδομένων του XML Schema, οι οποίοι είναι πιο πιθανό να υποστηρίζονται από τις εφαρμογές OWL, αφού στο [Motik, Patel-Schneider, & Horrocks, 2008] ορίζεται ότι μια εφαρμογή μπορεί να δώσει λάθος, αν αντιμετωπίσει μη υποστηριζόμενο (δηλ. όχι XML Schema) τύπο δεδομένων. Για παράδειγμα, το `dcterms:W3CDTF` αντιστοιχεί στο `xsd:date`, το `dcterms:RFC3066` στο `xsd:language` και ούτω καθεξής. Αυτοί οι τύποι δεδομένων θα πρέπει να ανατίθενται με ρητό τρόπο στις λεκτικές τιμές, αφού η OWL απαιτεί τη ρητή ανάθεση τύπων στους πόρους, ακόμα κι αν το πεδίο τιμών μιας ιδιότητας έχει οριστεί ως τύπος `xsd:`.
- γ) Υποστασιοποίηση (reification) λεκτικών (ως αντικείμενα) που προέρχονται από συγκεκριμένα (ελεγχόμενα) λεξιλόγια, όπως οι τύποι MIME. Συγκεκριμένα, το DSspace συσχετίζει στοιχεία με λεκτικά που αντιστοιχούν σε τύπους MIME, μέσω της ιδιότητας `dcterms:format`. Η χρήση τύπων MIME ως πληρωτών του `dcterms:format` είναι επίσης συνιστώμενη πρακτική του DCMI [DCMI Usage Board, 2008]. Για παράδειγμα, η περίπτωση ενός εγγράφου MS Word μπορεί να οριστεί:

```
<dcterms:format rdf:resource="&dspace-ont;msword"/>
```

- δ) Υποστασιοποίηση άλλων λεκτικών για να μπορέσουν να αναγνωριστούν και να εκφραστούν πιθανές σημασιακές σχέσεις. Για παράδειγμα, τα ονόματα των συγγραφέων στο DSspace ορίζονται ως αντικείμενα και τα, διαχωρισμένα με κόμμα, μικρά ονόματα και επίθετά τους (όπως τα αποθηκεύει το DSspace) αναλύονται συντακτικώς και ανατίθενται σε αντίστοιχες ιδιότητες από τη δημοφιλή οντολογία FOAF (<http://xmlns.com/foaf/spec/>). Η γλώσσα των κειμενικών τιμών διατηρείται, με χρήση της οδηγίας `xml:lang`. Δεν μπορούν όμως να συνδεθούν οι μεταφράσεις -για παράδειγμα, οι μεταφράσεις του τίτλου ενός στοιχείου ή των ονομάτων των συγγραφέων- αφού η σχέση αυτή χάνεται την ίδια στιγμή που ο χρήστης εισάγει αυτήν την πληροφορία στο DSspace (το υποκείμενο σχήμα δεν υποστηρίζει το στόχο αυτό).

Υπάρχει επίσης το πρόβλημα του πώς να αναφερόμαστε σε όλα αυτά τα νέα στιγμιότυπα. Μια πιθανή και κομψή λύση είναι να υποστασιοποιηθούν τα σχετικά λεκτικά χρησιμοποιώντας ανώνυμα άτομα, δηλαδή έναν κενό κόμβο (b-node) για κάθε ένα από αυτά:


```
<dcterms:contributor rdf:parseType="Resource">
  <foaf:name rdf:datatype="&xsd:string">XXX</foaf:name>
  <foaf:surname rdf:datatype="&xsd:string">YYY</foaf:surname>
</dcterms:contributor>
```

Το θέμα της υποστήριξης τέτοιων ανώνυμων ατόμων στην OWL 1.1 δεν έχει ακόμα επιλυθεί πλήρως [Motik, Patel-Schneider, & Horrocks, 2008]. Επιπλέον, το Protégé 4 εμφανίζει προβληματική συμπεριφορά στο χειρισμό τους, αφού δημιουργεί αυθαίρετα στιγμιότυπα (*genid*) και αποτυγχάνει να συσχετίσει τους κόμβους αυτούς με τις ιδιότητές τους (όνομα, επίθετο). Επομένως, έχει υιοθετηθεί η λιγότερο ευέλικτη λύση της ανάθεσης ενιαίων ονομάτων της μορφής [XXX, YYY] σε κάθε στιγμιότυπο που προκύπτει από μια λεκτική τιμή. Ανάλογη διαδικασία ακολουθείται και για τους πληρωτές της ιδιότητας *sponsorship*.

- ε) Εισαγωγή και επανανάθεση χώρων ονομάτων. Αυτό σημαίνει ότι το *dc:* αντικαθίσταται από το *dcterms:* και επίσης εισάγεται το *dSPACE-ont:* για πεδία συγκεκριμένα ως προς το *DSPACE*. Αυτό μπορεί επίσης να θεωρηθεί και ως αυτόνομη φάση (φάση 0: *φάση συμβατότητας* ή φάση 5: *διατήρηση εφαρμοσιμότητας* στην τεχνική) που είναι αναπόφευκτη προκειμένου να επιτευχθεί συμβατότητα με την RDF υλοποίηση του DC.

Το αποτέλεσμα της φάσης αυτής είναι η δημιουργία ενός εγγράφου RDF, το οποίο περιέχει τριπλέτες που περιγράφουν τους πόρους του αποθέματος, αναμειγμένες με τους χώρους ονομάτων του FOAF και του XML Schema, έστω 'instances.rdf'.

Εντασιακή εμβάθυνση

Στη συνέχεια, μπορούμε να προχωρήσουμε στην εντασιακή εμβάθυνση του μοντέλου DC. Προς το σκοπό αυτό, χρειάζεται πρώτα να βρεθεί (ή να επινοηθεί) ένα μοντέλο-πυρήνας. Το μοντέλο αυτό:

- παρέχεται από τη σημασιολογία του RDF, που είναι εγγενής στην RDF υλοποίηση του DC,
- προσαρμόζεται περαιτέρω από τη νέα υλοποίηση του DC που ενσωματώνει το DCAM, καθώς επίσης και περιορισμούς πεδίου ορισμού και τιμών,
- προσαρμόζεται περαιτέρω από *μετα-RDF* χαρακτηρισμούς των ιδιοτήτων, όπως αντιστροφή, συμμετρία, μεταβατικότητα, συναρτησιακότητα. Για παράδειγμα, ορίζουμε την ιδιότητα *dcterms:relation* ως συμμετρική και την ιδιότητα *dcterms:hasPart* ως αντίστροφη της *dcterms:isPartOf*.

Το αποτέλεσμα της φάσης αυτής είναι η δημιουργία ενός εγγράφου OWL που εισάγει το έγγραφο της αρχικής RDF υλοποίησης του *dcterms* ('*dcterms.rdf*') και περιέχει τις παραπάνω εμβαθύνσεις (έστω '*dc-ont.owl*').

Σημασιακή εκλέπτυνση

Μπορούν τώρα να προστεθούν σημασιακές εκλεπτύνσεις για το συγκεκριμένο σενάριο εφαρμογής, δηλαδή το ιδρυματικό απόθεμα του Πανεπιστημίου Πατρών. Αυτό μπορεί να γίνει ως εξής:

α) Μοντελοποίηση των λεξιλογίων σε ιεραρχίες υπαγωγής. Για παράδειγμα, το λεξιλόγιο των τύπων MIME που υποστηρίζει το DSpace υλοποιείται ως μια διαμέριση κλάσεων, με στιγμιότυπα, και συσχετίζεται με το `dcterms:FileFormat`.

β) Αναγνώριση και αναπαράσταση των εννοιών του DSpace *στοιχείο* (item), *συλλογή* (collection) και *κοινότητα* (community) ως κλάσεις. Αναφερόμαστε στα άτομα που ανήκουν σε αυτές τις κλάσεις μέσω των μοναδικών τους *handles*, τα οποία ανατίθενται αυτόματα από το DSpace. Για παράδειγμα:

```
<dspace-ont:collection rdf:about="&dspace-ont;hdl_1987_42"/>
```

Επιπλέον, το collection ορίζεται ως υποκλάση του community, εκφράζοντας έτσι το γεγονός ότι στο DSpace οι συλλογές εκλεπτούν τις κοινότητες. Τα στοιχεία συνδέονται με τις συλλογές μέσω της ιδιότητας `dcterms:hasPart` (αντίστροφη της `dcterms:isPartOf`):

```
<dspace-ont:item rdf:about="&dspace-ont;hdl_1987_57">
```

```
<dcterms:isPartOf rdf:resource="&dspace-ont;hdl_1987_42"/>
```

```
</dspace-ont:item>
```

γ) Αναγνώριση και αναπαράσταση των εννοιών, που δεν υπάρχουν στο DC, 'author' και 'sponsorship' (συγγραφέας και χρηματοδότηση) και συσχετισμός τους με το αρχικό μοντέλο: ορίζονται, υπό διαφορετικό χώρο ονομάτων, τα `author` και `sponsorship` ως OWL ιδιότητες αντικειμένου και καθίστανται υπο-ιδιότητες των `dcterms:contributor` και `dcterms:description` αντίστοιχα.

δ) Μοντελοποίηση σύνθετων σχέσεων χρησιμοποιώντας τελεστές σχηματισμού ρόλων και περιορισμούς (υπαρξιακούς και καθολικούς): Για παράδειγμα, ορίζουμε την έννοια του «συν-συγγραφέα» (co-author) ως:

$$\text{author}^- \circ \text{author} \sqsubseteq \text{co_author},$$

και εκλεπτόνουμε το `sponsorship` ως:

$$\text{dcterms:contributor}^- \circ \text{sponsorship} \sqsubseteq \text{sponsorship},$$

που σημαίνει ότι και οι συγγραφείς ενός στοιχείου λαμβάνουν χρηματοδότηση από τον ίδιο φορέα.

Δηλώνουμε επίσης ότι:

$$\neg \text{item} \sqcap \exists \text{dcterms:hasPart.item} \sqsubseteq \text{collection},$$

που σημαίνει πως στιδήποτε 'hasPart' ένα item τότε είναι συλλογή, εκτός αν είναι item το ίδιο. Επίσης οι συλλογές μπορούν να έχουν μόνο στοιχεία ως μέρη:

$$\text{collection} \sqsubseteq \forall \text{dcterms:hasPart.item}.$$

Η φάση αυτή καταλήγει στη δημιουργία ενός νέου OWL 1.1 εγγράφου ('dspace-ont.owl'), το οποίο περιέχει κατασκευές που εκλεπτούν το μοντέλο δεδομένων του DSpace με σημασιακό τρόπο.

6.4. Αποτελέσματα

Η ενότητα αυτή παρουσιάζει ορισμένα πειραματικά αποτελέσματα, υπό μορφή ευφυών ερωτημάτων, που προκύπτουν από τη δημιουργία και ενημέρωση της οντολογίας DC.

Πρώτα δίνεται η τεχνική διαμόρφωση που ακολουθήθηκε για την ανάκτηση των μεταδεδομένων του αποθέματος και την αντιστοίχισή τους στην οντολογία. Κατόπιν δίνονται τα αποτελέσματα των συμπερασμών.

6.4.1. Τεχνική διαμόρφωση

Προκειμένου για την εφαρμογή και εκτίμηση της τεχνικής, η οντολογία ενημερώνεται με περιγραφές που προέρχονται από το απόθεμα του Πανεπιστημίου Πατρών και το οποίο βασίζεται στο DSpace. Το DSpace παρέχει τη δυνατότητα συγκομιδής των μεταδεδομένων του μέσω μιας διεπαφής OAI-PMH. Εσωτερικά, ακολουθεί ένα ορισμένο προφίλ εφαρμογής, το οποίο έχει πολλά κοινά στοιχεία με το *Προφίλ Εφαρμογής για Βιβλιοθήκες* (DC-Lib, [DCMI-Libraries Working Group, 2004]). Μαζί με τις εξειδικεύσεις, το DSpace περιλαμβάνει συνολικά 66 πεδία, ορισμένα από τα οποία δεν είναι ορατά στο χρήστη.

Η υπηρεσία OAI-PMH μπορεί να παραμετροποιηθεί σχετικά με το ποια πεδία θα μπορούν να εξαχθούν (των χώρων ονομάτων και των ετικετών τους συμπεριλαμβανομένων) και επίσης υποστηρίζει τόσο το απλό DC (oai_dc), όσο και τις εξειδικεύσεις του (qdc). Επιλέγουμε το δεύτερο, ως πλουσιότερο από σημασιακή άποψη και εφαρμόζουμε το εργαλείο RDFizer (http://simile.mit.edu/wiki/OAI-PMH_RDFizer) που αναπτύσσεται στα πλαίσια του προγράμματος Simile, το οποίο συλλέγει τα OAI μεταδεδομένα και τα αποθηκεύει σε μορφή RDF. Κατά τη διαδικασία όμως αυτή χρησιμοποιούμε το δικό μας XSLT έγγραφο, που επιχειρεί την καλύτερη αποτύπωση των έμμεσων σημασιακών σχέσεων και την δημιουργία των συγκεκριμένων ως προς OWL στιγμιοτύπων. Το έγγραφο αυτό είναι ακριβώς εκείνο που υλοποιεί τη φάση του συντακτικού μετασχηματισμού.

Το σημασιακό προφίλ εφαρμογής, που είναι αποθηκευμένο ως `dspace-ont.owl`, εισάγεται κατόπιν από το μετασχηματισμένο RDF έγγραφο. Το `dspace-ont.owl` με τη σειρά του, εισάγει το `dc-ont.owl`, το οποίο εισάγει το `dcterms.rdf`. Ο χειρισμός της προκύπτουσας οντολογίας γίνεται μέσω του περιβάλλοντος Protégé 4.

6.4.2. Διεξαγωγή επερωτήσεων

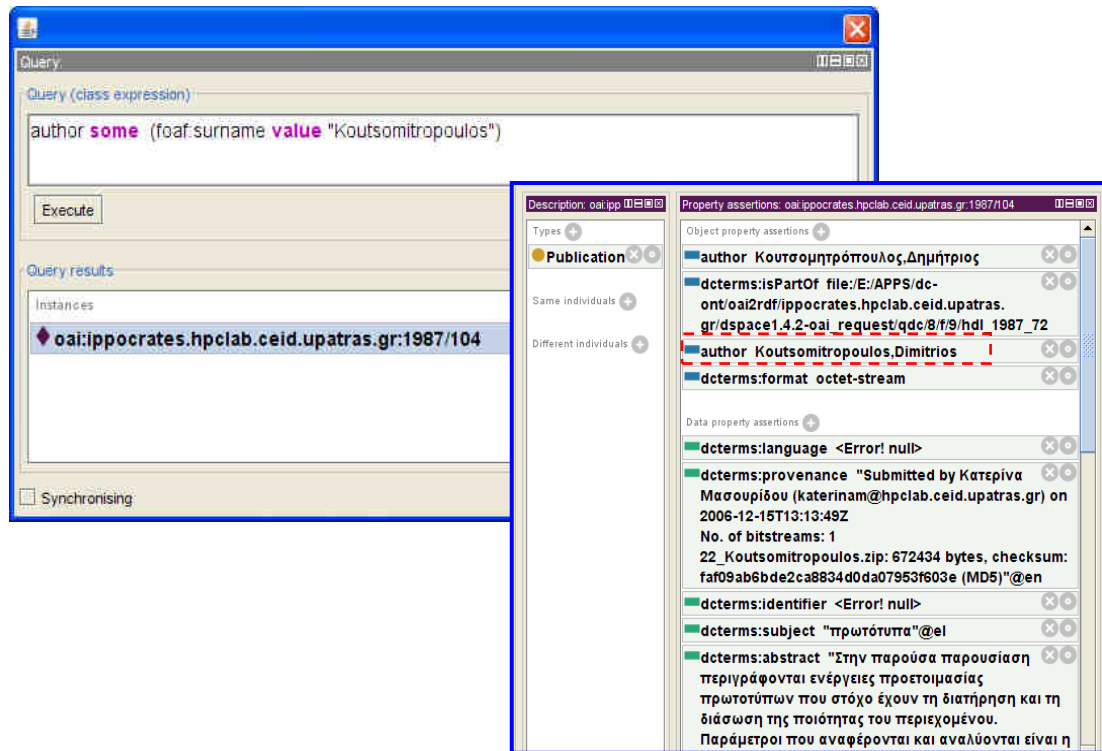
Στη συνέχεια, επιχειρείται να τεθούν ορισμένα ευφυή ερωτήματα στην παραγόμενη OWL (1.1) οντολογία. Προς το σκοπό αυτό, χρησιμοποιείται η Καρτέλα Επερωτήσεων Λογικών Περιγραφής (DL Query Tab) του Protégé 4 alpha, και δοκιμάζονται και οι δύο ενσωματωμένοι μηχανισμοί συλλογισμού, FaCT++ και Pellet.

Αρχικά, διεξάγεται ένα ερώτημα απλού συντακτικού ταιριάσματος (string-based) που θα ήταν εφικτό και μέσω της διεπαφής επερωτήσεων του ίδιου του DSpace. Στο ερώτημα που φαίνεται στην εικόνα 6.3, ζητάμε τα στοιχεία των οποίων ο συγγραφέας έχει ένα ορισμένο επίθετο. Με τη διεξαγωγή ορισμένων παρόμοιων συμπερασμών, που περιλαμβάνουν και συζευγμένα ερωτήματα περιορισμένης, αναγκαστικά, μορφής, επιβεβαιώνεται ότι είναι δυνατό να παραχθούν ταυτόσημα αποτελέσματα και συμπεραίνουμε ότι η οντολογία μας είναι τουλάχιστον σημασιακά ισοδύναμη προς το υπάρχον μοντέλο. Ωστόσο, αναζητήσεις τύπου Google (αποκλεισμός λέξεων, αναζήτηση ως προς ρίζα-stemming) και ταίριασμα γενικών χαρακτήρων (ο ειδικός χαρακτήρας *), συνήθεις στο DSpace, δεν υποστηρίζονται από την παρούσα διαμόρφωση, αφού κανένας από τους ενσωματωμένους μηχανισμούς συλλογισμού δεν φαίνεται να υποστηρίζει την όψη (facet) *pattern* του XML Schema.

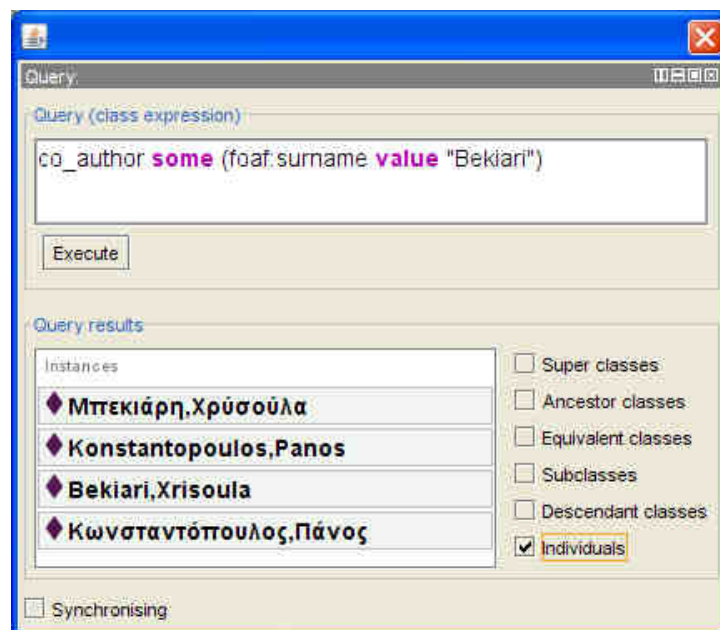
Να σημειωθεί επίσης ότι δεν μπορούν να πραγματοποιηθούν συντακτικά ερωτήματα σε παρονομασμένες ιδιότητες, γιατί το DL Query Tab δεν τις διερμηνεύει σωστά. Για παράδειγμα η ερώτηση, σε σύνταξη Μάντσεστερ (βλ. ενότητα 2.5):

```
dcterms:subject some string
```

δεν επιτρέπεται να εκτελεστεί καν, παρόλο που το `dcterms:subject` παρονομάζεται σωστά ως ιδιότητα αντικειμένου και ως ιδιότητα τύπων δεδομένων.



Εικόνα 6.3: Ερώτημα συντακτικού ταιριάσματος και αποτέλεσμα μέσω του Protégé 4.0.

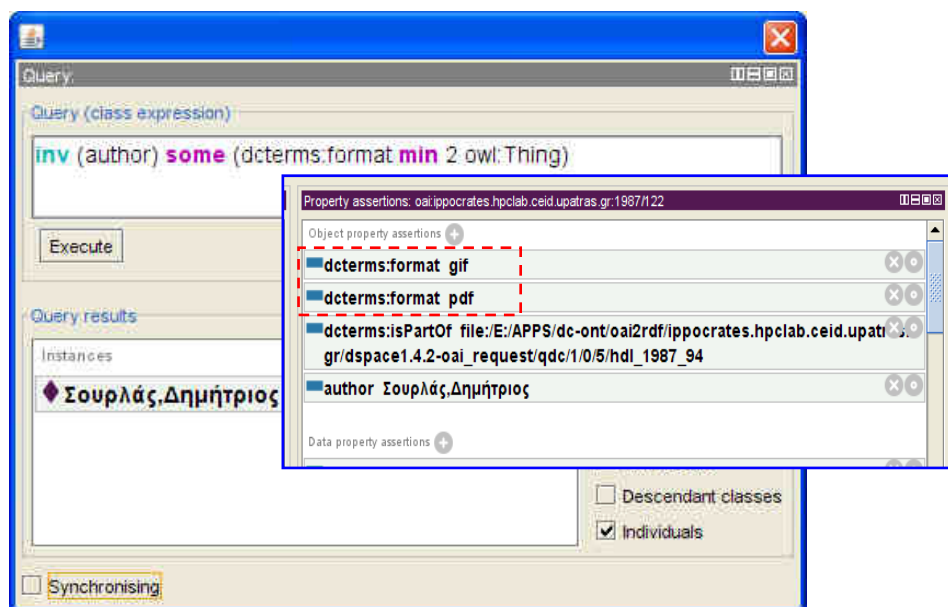


Εικόνα 6.4: Ερώτημα Λογικών Περιγραφής και αποτέλεσμα, με χρήση αλυσίδων ρόλων.

Στη συνέχεια, διεξάγουμε ορισμένους συμπερασμούς που βασίζονται στις νέες σημασιακές δομές που έχουν προστεθεί στο προφίλ. Τα αποτελέσματα μπορεί να υπονοούνται από το τρέχον μοντέλο δεδομένων, αλλά δεν υπάρχει τρόπος να ανακτηθούν με χρήση της δεδομένης διαμόρφωσης.

Για παράδειγμα μπορούμε να ζητήσουμε τους συν-συγγραφείς ενός συγκεκριμένου συγγραφέα ή να ρωτήσουμε από πού ένας δημιουργός αντλεί χρηματοδότηση. Στην εικόνα 6.4 ζητάμε τους συν-συγγραφείς ενός συγγραφέα, βάσει του επιθέτου του. Λόγω του ορισμού της ιδιότητας `co_author`, το αποτέλεσμα είναι άμεσο.

Προχωρώντας, στην εικόνα 6.5 ρωτάμε «ποιος έχει συγγράψει στοιχείο που να αποτελείται από τουλάχιστον 2 τύπους αρχείου;» χρησιμοποιώντας έναν περιορισμό πληθικότητας στο `dcterms:format`. Επειδή το στοιχείο 1987/122 σχετίζεται με δύο διακριτούς τύπους MIME, συγκεκριμένα gif και pdf, προκύπτει το αποτέλεσμα.



Εικόνα 6.5: Ερώτημα Λογικών Περιγραφής και αποτέλεσμα, με χρήση περιορισμών πληθικότητας.

Μπορούμε επίσης να πραγματοποιήσουμε επερωτήσεις που εμπλέκουν και τις υπόλοιπες σημασιακές εκλεπτύνσεις που έχουν εισαχθεί. Τα αποτελέσματα οδηγούν στο συμπέρασμα ότι η προτεινόμενη προσέγγιση καθιστά δυνατό ένα υψηλότερο επίπεδο ανακάλυψης πληροφορίας με ελάχιστη χειρωνακτική παρέμβαση.

6.5. Αναγωγή προφίλ μεταδεδομένων Dublin Core σε Λογικές Περιγραφές

Στην ενότητα αυτή εξηγείται πώς σχετίζονται τα παραδοσιακά προφίλ μεταδεδομένων εφαρμογής με τις οντολογίες Ιστού και τις Λογικές Περιγραφές. Αρχικά, γίνεται μια σύντομη περιγραφή της πρόσφατης πρότασης για αναπαράσταση προφίλ μεταδεδομένων μέσω της έννοιας των Προφίλ Συνόλων Περιγραφών (DSP) και βάσει του XML Schema, όπου τονίζεται η σκοπιμότητα μιας αντίστοιχης αναπαράστασης και με χρήση οντολογικών γλωσσών. Στη συνέχεια εισάγεται η αναγωγή των DSP σε Λογικές Περιγραφές και συζητούνται τα

εκφραστικά χαρακτηριστικά που απαιτούνται, καθώς και η επιβάρυνση που επιφέρουν στην πολυπλοκότητα του συλλογισμού.

6.5.1. Σκοπός και αναγκαιότητα

Η ανάγκη της ύπαρξης ενός συνεπούς πλαισίου για την ανάπτυξη προφίλ εφαρμογής έχει επίσης αναγνωριστεί και από το ίδιο το DCMI. Ύστερα από το συνέδριο DC 2007, το επονομαζόμενο «Πλαίσιο της Σιγκαπούρης» για την ανάπτυξη προφίλ μεταδεδομένων εφαρμογής που βασίζονται στο DC εισήχθη στο [Nilsson, Baker, & Johnston, 2008]. Βασισμένο σε έννοιες του DCAM, το πλαίσιο της Σιγκαπούρης έχει ως κύρια προτεραιότητα την ανάπτυξη μιας γλώσσας περιορισμών, όπως ακριβώς ένα XML Schema, που θα καθορίζει και τυπικά τι είδους ιδιότητες (πιθανώς προερχόμενες από διαφορετικά πρότυπα) συμπεριλαμβάνονται στο εν λόγω προφίλ και τι είδους τιμές είναι κατάλληλες για αυτές τις ιδιότητες, που πιθανώς να περιορίζονται από συγκεκριμένα συντακτικά και λεξιλογικά σχήματα κωδικοποίησης. Ο στόχος ενός τέτοιου Προφίλ Συνόλων Περιγραφών (DSP) [Nilsson M. , 2008] είναι να προσδιορίζει εκείνες τις εγγραφές μεταδεδομένων που ταιριάζουν (ή συμμορφώνονται) προς το συγκεκριμένο DSP. Αυτό σημαίνει ότι η γλώσσα για την έκφραση των DSP μπορεί πρωτίστως να χρησιμοποιηθεί για την έκφραση των δομικών και συντακτικών περιορισμών που επιβάλλει ένα προφίλ εφαρμογής, αφήνοντας εκτός εμβέλειας τη σημασιακή διαλειτουργικότητα, που αποτελεί σημαντικό στόχο της προσέγγισης της διατριβής. Εξάλλου, όπως αναφέρεται στο Πλαίσιο της Σιγκαπούρης:

«Είναι σημαντικό να αντιληφθεί κανείς ότι η σημασιολογία των όρων που χρησιμοποιούνται στα προφίλ εφαρμογής φέρεται από τους ορισμούς τους, που είναι ανεξάρτητοι από το προφίλ εφαρμογής [...]. Η σημασιακή διαλειτουργικότητα παρέχεται από την ορθή χρήση των όρων που ορίζονται σε ένα ή περισσότερα λεξιλόγια και τα προφίλ εφαρμογής υπάρχουν για να παρέχουν υψηλού επιπέδου συντακτική ή δομική διαλειτουργικότητα...»

Ήδη δείξαμε πως ο παραπάνω ισχυρισμός δεν ισχύει στην περίπτωση των σημασιακών προφίλ και επίσης ότι η σημασιακή προσαρμογή εκλεπτύνει τη σημασιολογία των όρων και οδηγεί προς τη σημασιακή διαλειτουργικότητα. Περαιτέρω, και παρά την υλοποίηση της γλώσσας DSP σε ένα XML Schema [Nilsson M. , 2008], μπορεί να θεωρηθεί ότι η έκφρασή της σε γλώσσα συμβατή με το RDF είναι καταλληλότερη, δεδομένων και των πρόσφατων υλοποιήσεων του DC σε RDF(S), αλλά και της οντολογίας DC σε OWL που δείξαμε στο παρόν κεφάλαιο.

6.5.2. Η διαδικασία της αναγωγής

Αρχικά παρατηρούμε ότι οι δομικοί περιορισμοί, όπως οι επιτρεπόμενες τιμές και οι τύποι των πόρων, έχουν την αντιστοιχία τους στους περιορισμούς πεδίου ορισμού και πεδίου τιμών του RDFS. Επίσης η έννοια των επιτρεπόμενων ιδιοτήτων (allowed properties) μπορεί να αναπαρασταθεί ως εξής:

Βασικό δομικό στοιχείο ενός DSP είναι το πρότυπο περιγραφής (description template). Ένα DSP μπορεί να περιλαμβάνει πολλά τέτοια πρότυπα. Ένα πρότυπο περιγραφής

αντιστοιχεί στην περιγραφή πόρων ενός συγκεκριμένου τύπου (π.χ. items, persons...) και ορίζει περιορισμούς πάνω στο σύνολο των ιδιοτήτων που αφορούν το συγκεκριμένο τύπο πόρων (είναι δηλαδή στο πεδίο ορισμού τους). Οι περιορισμοί σε μια ιδιότητα εκφράζονται μέσω του *προτύπου δήλωσης* (statement template) και άρα ένα πρότυπο περιγραφής μπορεί να έχει πολλά πρότυπα δήλωσης.

Ένα πρότυπο περιγραφής μπορεί επομένως να ειπωθεί ως μία μόνο ιδιότητα, που διαμερίζεται στο σύνολο των επιτρεπόμενων ιδιοτήτων (δηλαδή μια *ν-αδική* ιδιότητα). Ένας τρόπος για τον ορισμό τέτοιων *ν-αδικών* ιδιοτήτων δίνεται στο [Noy & Rector, 2006]. Για κάθε πρότυπο περιγραφής μπορεί να οριστεί μια κλάση *Description_ID*. Στη συνέχεια, για κάθε επιτρεπόμενη ιδιότητα P_1, \dots, P_n :

$$Description_ID \sqsubseteq \exists P_1.range_1 \sqcap \dots \sqcap \exists P_n.range_n \quad (1)$$

Το παραπάνω εκφράζει τον περιορισμό ότι κάθε στιγμιότυπο του *Description_ID* έχει τουλάχιστον μία σχέση, διαμέσου των P_1, \dots, P_n με ένα στιγμιότυπο από το κατάλληλο πεδίο τιμών (*range*). Για να εκφραστεί ο περιορισμός ότι οι P_1, \dots, P_n μπορούν να συσχετίζονται τα στιγμιότυπα του *Description_ID* μόνο με τα κατάλληλα πεδία τιμών, μπορεί να χρησιμοποιηθεί καθολική ποσοτικοποίηση:

$$Description_ID \sqsubseteq \forall P_1.range_1 \sqcap \dots \sqcap \forall P_n.range_n \quad (2)$$

Περιορισμοί που αφορούν το πλήθος των επιτρεπόμενων πληρωτών για κάθε ιδιότητα μπορούν να αναπαρασταθούν με ανάλογο τρόπο, χρησιμοποιώντας προσδιορισμένους περιορισμούς αριθμού, αντικαθιστώντας έτσι τους (πιο γενικούς) υπαρξιακούς περιορισμούς.

Για να φανεί ότι οι παραπάνω εκφράσεις είναι επίσης και ικανές (όχι μόνο αναγκαίες), έστω ότι $x \in Description_ID$ και $P_k(x, y)$, όπου $1 \leq k \leq n$ και το y δεν ανήκει στο επιτρεπόμενο πεδίο τιμών. Τότε, λόγω (2), η οντολογία καθίσταται ασυνεπής, αφού το y πρέπει να ανήκει στο κατάλληλο για την P_k πεδίο τιμών. Επίσης, λόγω (1), το x πρέπει να έχει (αν και μπορεί να μην έχουν δηλωθεί ακόμα) άλλες $n-1$ σχέσεις, μέσα από τις επιτρεπόμενες ιδιότητες.

Η έκφραση του γεγονότος ότι οι P_1, \dots, P_n είναι οι *μόνες* ιδιότητες που επιτρέπονται είναι κάπως πιο περίτεχνη. Αυτό που χρειάζεται είναι η δυνατότητα έκφρασης του αξιώματος *τομής και συμπληρώματος ρόλων*. Η OWL 1.1 επιτρέπει ξένους ρόλους, αλλά όχι και τομές ρόλων εν γένει. Έστω U ο καθολικός ρόλος, δηλαδή ο γονέας όλων των ρόλων (βλ. ενότητα 3.4.4). Ισχύει ότι:

$$\exists P_1.range_1 \sqcup \dots \sqcup \exists P_n.range_n \sqsubseteq \exists U.T$$

Θέλουμε να εκφράσουμε ότι οι κλάσεις $\exists P_m.range_m$, για κάθε m διαφορετικό από $1 \dots n$, δεν επιτρέπονται μέσα στο *Description_ID*. Αν έχουμε στη διάθεσή μας τομή και συμπλήρωμα ρόλων, το σύνολο των μη επιτρεπόμενων ιδιοτήτων μπορεί να συμβολιστεί ως:

$$U \sqcap \neg(P_1 \sqcup \dots \sqcup P_n)$$

Το *Description_ID* θα πρέπει να μην περιλαμβάνει στιγμιότυπα τα οποία να σχετίζονται με άλλα μέσω των μη επιτρεπόμενων ιδιοτήτων. Δηλαδή:

$$Description_ID \sqcap \exists(U \sqcap \neg(P_1 \sqcup \dots \sqcup P_n)). T \equiv \emptyset$$

Καταλήγουμε λοιπόν ότι για την έκφραση των DSP απαιτείται η διαθεσιμότητα των τριών λογικών τελεστών (ένωση, τομή, συμπλήρωμα) και σε εκφράσεις ή τουλάχιστον μόνο σε ονόματα ρόλων, κανέναν από τους οποίους δεν είναι διαθέσιμος στην OWL 1.1. Επίσης απαιτείται η ύπαρξη προσδιορισμένων περιορισμών αριθμού (το Q στην ονοματολογία) που δεν υπάρχει στην OWL DL, ενώ ο καθολικός ρόλος μπορεί και να απαλειφθεί.

Οι Lutz και Sattler [Lutz & Sattler, 2000] δείχνουν ότι η Λογική Περιγραφής ALC (\sqcup , \sqcap , \neg) δηλαδή η βασική λογική ALC με λογικούς τελεστές στους ρόλους ανήκει στην κλάση πολυπλοκότητας NEXP. Επίσης, στο [Tobies, 2001], δείχνεται ότι η Λογική Περιγραφής $ALCIQ$ (\sqcup , \sqcap , \neg), δηλαδή η προηγούμενη λογική εκτεταμένη με προσδιορισμένους περιορισμούς αριθμού και αντίστροφους ρόλους, η οποία επαρκεί για την έκφραση DSP, αλλά δεν αντιστοιχεί σε κάποια οντολογική γλώσσα, είναι NEXP-complete. Με τον περιορισμό ότι οι εκφράσεις ρόλων, αν μετασχηματιστούν σε *Διαζευγμένη Κανονική Μορφή* (Disjunctive Normal Form, DNF, δηλαδή σε ένωση τομών), θα πρέπει σε κάθε στοιχείο της διάζευξης (ένωσης) να έχουν τουλάχιστον ένα μη αρνητικό σκέλος, η λογική αυτή γίνεται PSPACE-complete. Για παράδειγμα, η έκφραση $\neg(P_1 \sqcup \dots \sqcup P_n)$ σε DNF γράφεται $\neg P_1 \sqcap \dots \sqcap \neg P_n$ που αποτελείται από ένα μόνο στοιχείο διάζευξης το οποίο δεν έχει κανένα μη αρνητικό σκέλος, εκτός αν χρησιμοποιηθεί ο καθολικός ρόλος. Ακόμη, οι Schmidt και Tishkovsky [Schmidt & Tishkovsky, 2007] δείχνουν ότι η Λογική Περιγραφής $ALBO$, δηλαδή η βασική λογική ALC , εκτεταμένη με ένωση και συμπλήρωμα ρόλων, αντίστροφους ρόλους και ονοματικά είναι NEXP-complete (η τομή δεν αναφέρεται γιατί $P \sqcap R \equiv \neg(\neg P \sqcup \neg R)$, όπου P, R ονόματα ρόλων) και υλοποιούν αντίστοιχο αλγόριθμο ταμπλό. Αναφέρουν επίσης ότι η εφαρμογή συμπληρώματος πάνω σε αλυσίδες ρόλων οδηγεί σε μη αποφασισιμότητα, κάτι που ισχύει επίσης και για την εφαρμογή της τομής [Calvanese & Giacomo, 2007].

Συμπερασματικά, η σημασιολογία του RDF(S) και της OWL δεν επαρκεί για την έκφραση των δομικών και συντακτικών περιορισμών των προφίλ εφαρμογής. Επιπρόσθετα, αν και ρητός στόχος των DSP είναι η ενασχόληση με συντακτικούς μόνο περιορισμούς, αποδεικνύεται ότι έχουν σοβαρές σημασιακές προεκτάσεις.

7. ΣΥΜΠΕΡΑΣΜΑΤΑ-ΜΕΛΛΟΝΤΙΚΕΣ ΚΑΤΕΥΘΥΝΣΕΙΣ

7.1. Επισκόπηση και συνεισφορά

Μια γενική διαπίστωση από την έρευνα που διεξήχθη στη διατριβή αυτή είναι ότι ο Σημαντικός Ιστός είναι τόσο επιθυμητός, όσο και το εύρος των συμπερασμών που επιτρέπει. Πράγματι, οι δυνατότητες ανακάλυψης γνώσης είναι στην πραγματικότητα το ουσιαστικό κίνητρο για τη δύσκολη, από εννοιολογικής άποψης, δομική και σημασιολογική οργάνωση της πληροφορίας σε μια γλώσσα αναπαράστασης, όπως η OWL. Δείξαμε ότι τέτοιες δυνατότητες είναι εφικτές και ότι μπορούν να εφαρμοστούν, με αξιοποιήσιμα αποτελέσματα, σε ρεαλιστικά σενάρια εφαρμογών. Εντούτοις, έχουν υψηλό τίμημα:

- α. Η πολυπλοκότητα του συλλογισμού για την εξαγωγή συμπερασμών και άρα για την ανακάλυψη γνώσης στο Σημαντικό Ιστό είναι αρκετά υψηλή.
- β. Εκτός από την επακριβή εννοιολογική μοντελοποίηση του πεδίου υπάρχει ανάγκη για την παραγωγή πλούσιων περιγραφών, οι οποίες συνήθως επιβαρύνουν το χρήστη.

Στη συνέχεια εστιάζουμε στα συμπεράσματα στα οποία οδηγεί η διατριβή και στη συνεισφορά της, αναφορικά με τα δύο παραπάνω σημεία, αλλά και ως προς τις ερευνητικές υποθέσεις που τέθηκαν στην εισαγωγή της διατριβής. Τέλος, στην επόμενη ενότητα, εξετάζονται οι πιο σχετικές ερευνητικές κατευθύνσεις που μπορούν να προκύψουν ως ευδοκίμηση της διεξαχθείσας έρευνας.

7.1.1. Αποδοτικότητα και εκφραστικότητα

Για το (α) παρατηρούμε ότι τα θεωρητικά αποτελέσματα είναι, στις περισσότερες περιπτώσεις, αποτελέσματα πληρότητας (completeness). Αυτό σημαίνει ότι οι κλάσεις πολυπλοκότητας στις οποίες κατατάσσονται τα αντίστοιχα προβλήματα είναι και οι καλύτερες δυνατές (άνω όρια). Το γεγονός αυτό, παρά το ότι υλοποιημένοι αλγόριθμοι, που βασίζονται σε ταμπλό, έχουν συνήθως συμπεριφορά χειρότερη κατά μία τάξη μεγέθους, δεν αφήνει περιθώρια για βελτιώσεις.

Η σοβαρότητα της κατάστασης αυτής επιδεινώνεται περαιτέρω, αν κανείς αναλογιστεί ότι οι συμπερασμοί διεξάγονται όχι σε ένα κλειστό περιβάλλον, ούτε απευθύνονται σε εξειδικευμένους ερευνητές, αλλά φιλοδοξούν να προσθέσουν αξία στην καθημερινή πλοήγηση του χρήστη στο Διαδίκτυο· κι εκεί δεν υπάρχει η πολυτέλεια της επένδυσης σε τόσο μεγάλους χρόνους απόκρισης, αφού η εμπειρία και η μελέτη της αλληλεπίδρασης έχουν δείξει ότι το ενδιαφέρον σύντομα εξανεμίζεται.

Η μελέτη διαφόρων καλώς ορισμένων φορμαλισμών από τον χώρο της Τεχνητής Νοημοσύνης, οι οποίοι σχετίζονται με την Λογική Πρώτης Τάξης, έδειξε ότι έχουν κοινά σημεία με τον θεμελιώδη φορμαλισμό του Σημαντικού Ιστού, δηλαδή τις Λογικές Περιγραφές, και ότι υπάρχουν τρόποι μερικής αναγωγής από τον ένα στον άλλο. Παρόλο

που η πολυπλοκότητα των προβλημάτων είναι ανεξάρτητη της αναπαράστασης, αυτό σημαίνει ότι υλοποιημένα συστήματα και αλγόριθμοι που έχουν τις ρίζες τους σε άλλους φορμαλισμούς, μελετημένους για χρόνια και με καλή συμπεριφορά (για παράδειγμα ο αποδοτικός αλγόριθμος RETE), μπορούν να χρησιμοποιηθούν για εκφραστικά υποσύνολα της OWL ή επαυξημένα υπο-τμήματα.

Για αυτά τα τμήματα, που μπορούν να οριοθετηθούν σαφώς ή να μεταφραστούν σε ισοδύναμα προβλήματα άλλων φορμαλισμών, π.χ. σε σύνολα κανόνων ή λογικά προγράμματα, η πολυπλοκότητα χειρότερης περίπτωσης είναι σαφώς μικρότερη. Αυτό βέβαια ισχύει και για τις Λογικές Περιγραφές· αλλά η συμπεριφορά των αλγορίθμων ταμπλό, όπως έχουν υλοποιηθεί για πολύ εκφραστικές λογικές και όπως απέχουν από τη βέλτιστη πολυπλοκότητα ακόμα και για απλές λογικές, θα είναι πιο συχνά κοντά στη χειρότερη περίπτωση, σε σχέση με τους άλλους αλγορίθμους, για αυτά τα υπο-τμήματα. Χαρακτηριστικό παράδειγμα αποτελεί η περίπτωση όπου η οντολογία έχει μικρό TBox και μεγάλο ABox, οπότε η συμπεριφορά ενός βασισμένου σε ανάλυση αλγορίθμου (KAON2) είναι καλύτερη σε σχέση με τον αντίστοιχο ταμπλό. Μάλιστα, υπάρχουν περιπτώσεις που εναλλακτικοί αλγόριθμοι μπορεί να έχουν επιδόσεις συγκρίσιμες με αυτές των ΣΔΒΔ (DL-Lite). Υπό το πρίσμα αυτό είναι λοιπόν σκόπιμη και αναγκαία η μελέτη και η τυποποίηση των βατών τεμαχίων της OWL, των εκφραστικών δηλαδή υποσυνόλων της γλώσσας που απολαμβάνουν καλύτερες υπολογιστικές ιδιότητες.

Όσον αφορά τους αλγορίθμους ταμπλό καθ' εαυτούς, είδαμε επίσης ότι συνοδεύονται από μια σειρά βελτιστοποιήσεων που μπορούν να βελτιώσουν σημαντικά τις επιδόσεις τους, στη μέση περίπτωση. Ακόμα, αξιολογικό χαρακτηριστικό των ταμπλό προσεγγίσεων είναι ότι εμφανίζουν διαβαθμίσιμη συμπεριφορά: οι κοστοβόροι κανόνες καλούνται μόνο όταν η εκφραστικότητα της προς επεξεργασία οντολογίας έχει ή συνδυάζει συγκεκριμένα (πιο εκφραστικά) χαρακτηριστικά, ειδάλλως προσαρμόζονται σε λιγότερο εκφραστικά επίπεδα, με μικρότερη πολυπλοκότητα.

Εξάλλου, από τη συζήτηση της μεθοδολογίας SWKD διαφαίνεται, μεταξύ άλλων, ένα σοβαρό έλλειμμα στον τρόπο με τον οποίο ο τελικός χρήστης ή αυτοματοποιημένοι πράκτορες μπορούν να αλληλεπιδρούν με τις οντολογίες και να τις αξιοποιούν συνθέτοντας ερωτήματα και λαμβάνοντας απαντήσεις σε ορισμένη μορφή. Η καθιέρωση ενός προτύπου για τη διαδικασία αυτή ήδη απασχολεί την ερευνητική κοινότητα και θα μπορούσε να θεωρηθεί επιθυμητή σε ένα παραγωγικό σύστημα. Η προδιαγραφή της SPARQL-DL αναγνωρίζεται ως ένα σημαντικό βήμα προς μια τέτοια κατεύθυνση.

Η KDI, παρά τις ελλείψεις στον τομέα της εκφραστικότητας, λόγω του υποκείμενου μηχανισμού συμπερασμού που χρησιμοποιεί, υλοποιεί μια κατανεμημένη αρχιτεκτονική και αποκαλύπτει ότι δεν αρκούν οι βασικές υπηρεσίες συλλογισμού που παρέχουν οι κλασικοί μηχανισμοί συμπερασμού Λογικών Περιγραφής. Αντίθετα, απαιτείται ο μη τετριμμένος συνδυασμός τους σε πιο υψηλού επιπέδου υπηρεσίες, οι οποίες θα πρέπει κατ' ελάχιστο να δίνουν τη δυνατότητα στο χρήστη να θέτει επερωτήσεις που περιλαμβάνουν και στιγμιότυπα (ABox) και να οδηγείται εντέλει, μέσα από ένα διαφανές μονοπάτι συμπερασμών, στην ανακάλυψη και ανάκτηση αυτών. Ο συνδυασμός της ταξινομικής πληροφορίας της οντολογίας με τα γεγονότα που αφορούν στα στιγμιότυπα είναι κρίσιμης σημασίας προς το σκοπό αυτό και θα πρέπει τελικά να μπορεί να οδηγεί, με εξίσου διάφανο τρόπο, και στην απάντηση συζευγμένων ερωτημάτων.

Ως αποτέλεσμα η υπόθεση:

Υπόθεση 1: *Οι τεχνολογίες και η λογική θεωρία του Σημαντικού Ιστού μπορούν να συνεισφέρουν αποδοτικά και εκφραστικά στο πρόβλημα της SWKD,*

δεν επαληθεύεται άμεσα, αλλά στο βαθμό που επιτρέπεται από συγκεκριμένους θεωρητικούς, οργανωτικούς και τεχνικούς παράγοντες, που εντοπίζονται και αναδεικνύονται στην παρούσα διατριβή.

7.1.2. Μοντελοποίηση πεδίου και περιγραφές

Το πρόβλημα που συνοψίζεται στο (β) είναι και η συνήθης κριτική που ασκείται για το Σημαντικό Ιστό και είναι γνωστό με διάφορες ονομασίες (chicken-egg problem, bootstrapping problem...). Προς αυτή την κατεύθυνση, διακρίνουμε μεταξύ:

- 1) της ανάγκης για εκφραστική μοντελοποίηση του πεδίου και
- 2) της ανάγκης για ύπαρξη πλούσιων περιγραφών και στιγμιοτύπων.

Για το (1), δείξαμε πώς η τεχνική της σημασιακής προσαρμογής μπορεί να επαυξήσει την εκφραστικότητα ενός εννοιολογικού μοντέλου και τεκμηριώσαμε, και πειραματικά, πώς αυτό ενεργοποιεί τη διεξαγωγή ισχυρών συμπερασμών. Ταυτόχρονα, οι παρεμβάσεις στο αρχικό μοντέλο έγιναν με τέτοιο τρόπο, ώστε να διατηρείται η διαλειτουργικότητα. Συνάγουμε άρα ότι επιβεβαιώνεται η ορθότητα της υπόθεσης:

Υπόθεση 2: *Η τεχνική της σημασιακής προσαρμογής επιτυγχάνει την εντασιακή επέκταση (εμβάθυνση) ενός μοντέλου, χωρίς όμως να βλάπτεται η γενική του εφαρμοσιμότητα.*

Όμως, η πρακτική αξιοποίηση αυτής της τεχνικής έχει νόημα μόνο όταν πληροφορίες, οργανωμένες σύμφωνα με το οντολογικό προφίλ, είναι διαθέσιμες. Στην περίπτωση του CIDOC-CRM, λόγω και της εγγενούς πολυπλοκότητας του μοντέλου, οι πρακτικές υλοποιήσεις του είναι σπάνιες και δεν υπάρχουν συμβατά (μετα-) δεδομένα που τουλάχιστον να είναι ευρέως διαθέσιμα. Στην περίπτωση όμως πιο απλών εννοιολογικών μοντέλων, όπως το σχήμα μεταδεδομένων Dublin Core, και ακριβώς λόγω της απλότητας και της γενικότητάς του, υπάρχουν διαθέσιμες και διαδεδομένες υλοποιήσεις.

Δείξαμε ότι, και σε αυτήν την περίπτωση, που το πεδίο γνώσης είναι, ουσιαστικά, αδόμητο από σημασιακής άποψης, μπορεί να αξιοποιηθεί η υποδομή του Σημαντικού Ιστού και να διεξαχθούν χρήσιμοι συμπερασμοί, με αποτέλεσμα την επαλήθευση της υπόθεσης:

Υπόθεση 1^α: *Η SWKD μπορεί να εφαρμοστεί αποδοτικά και εκφραστικά στην περίπτωση σημασιακά ημιδομημένων μοντέλων γνώσης.*

Τούτο κατέστη εφικτό με την εφαρμογή της τεχνικής της σημασιακής προσαρμογής στην περίπτωση του DC, καταλήγοντας σε μια εκφραστική οντολογία για το μοντέλο αυτό, καθώς και σε ένα σημασιακά ισχυρό (επαυξημένο) προφίλ για τη συγκεκριμένη εφαρμογή των ψηφιακών αποθεμάτων. Ταυτόχρονα, δείξαμε πως αυτές οι ενέργειες προσαρμογής έχουν κεντρικό χαρακτήρα, μπορούν δηλαδή να γίνουν σε συγκεκριμένα ενδιάμεσα σημεία και δεν επηρεάζουν ούτε επιβαρύνουν τις ενέργειες χαρακτηρισμού που πραγματοποιεί ο

τελικός χρήστης (δηλαδή τη συμπλήρωση των μεταδεδομένων, που παραμένει απλή). Το σημασιακό χάσμα που προκύπτει και που δεν είναι τόσο έντονο στην περίπτωση πλήρως σημασιακά δομημένων μοντέλων -όπως το CIDOC-CRM, όπου οι περιγραφές των χρηστών θα είναι εκ των προτέρων οντολογικά συμβατές- δείξαμε ότι μπορεί να γεφυρωθεί. Προς το σκοπό αυτό γίνεται χρήση μιας αυτοματοποιημένης διαδικασίας μετασχηματισμού των επίπεδων μεταδεδομένων σε οντολογικά στιγμιότυπα, η οποία βασίζεται κυρίως στην υποστασιοποίηση των κειμενικών τιμών ως αντικείμενα και, εν μέρει, στο χαρακτηριστικό της παρονομασίας. Με τον τρόπο αυτό μπορούμε με ασφάλεια να ισχυριστούμε ότι η παρακάτω υπόθεση, στην οποία ανάγεται η 1^α, επίσης επαληθεύεται:

Υπόθεση 2^α: *Η τεχνική της σημασιακής προσαρμογής επιτυγχάνει την εντασιακή αποτύπωση και επέκταση ενός σημασιακά ημιδομημένου μοντέλου, χωρίς να βλάπτεται η γενική του εφαρμοσιμότητα.*

Μάλιστα δίνεται έτσι απάντηση στο (2), δηλαδή στο πρόβλημα της έλλειψης πλούσιων περιγραφών για το Σημαντικό Ιστό.

Παραδόξως, η έκφραση κλασικών προφίλ εφαρμογής μεταδεδομένων, όπως αυτά επιχειρείται πρόσφατα να τυποποιηθούν, γύρω από το DC, με το πλαίσιο της Σγκαπούρης, δεν μπορεί να γίνει με την τρέχουσα εκφραστικότητα της OWL. Δείξαμε ακριβώς ποια εκφραστικά χαρακτηριστικά απουσιάζουν από την τελευταία προδιαγραφή της γλώσσας και απαιτούνται για τον προσδιορισμό τέτοιων προφίλ. Αποδείξαμε παρ' όλα αυτά ότι οι συντακτικοί περιορισμοί που επιβάλλει ένα κλασικό προφίλ μπορούν να αναχθούν σε σημασιακούς περιορισμούς μέσα σε μία οντολογία. Η δυσκολία -έως και μη αποφασισιμότητα- που θα έχει ο συλλογισμός με την προσθήκη και αυτών των χαρακτηριστικών δεν θα πρέπει όμως να ειδωθεί ως αδυναμία: Αφενός, το XML Schema μπορεί πολύ εύκολα να χρησιμοποιηθεί για την επιβολή και επαλήθευση τέτοιων συντακτικών περιορισμών και, αφετέρου, το ισοδύναμό τους σε Λογικές Περιγραφές δεν υπάρχει ανάγκη να συνοδεύεται από την ταυτόχρονη ύπαρξη και των υπολοίπων εκφραστικών χαρακτηριστικών της OWL 1.1, εκτός μόνο από λίγων.

Ενώ λοιπόν τα παραδοσιακά προφίλ μεταδεδομένων, όπως μπορούν να εκφραστούν μέσα από το XML Schema, ασχολούνται με το να διασφαλίσουν τη *συντακτική* διαλειτουργικότητα, τα προτεινόμενα επαυξημένα σημασιακά προφίλ αποτελούν τη βάση για τη *σημασιακή* διαλειτουργικότητα. Με άλλα λόγια, επιτρέπουν τη συνδιαλλαγή και διαπραγμάτευση πληροφοριών μεταξύ συστημάτων, τις οποίες, ενώ μπορεί να είναι κατά βάση ετερογενείς, προερχόμενες από πολλαπλές πηγές και αδόμητες σημασιακά, μπορούν να τις συσχετίζουν και, βάσει συμπερασμών, να τις συνδυάζουν με στόχο την ανακάλυψη νέας γνώσης.

7.2. Μελλοντικές γραμμές έρευνας και εφαρμογές

Από την έρευνα που διεξήχθη στα πλαίσια της διατριβής και τα πορίσματά της, καθώς και από τα προβλήματα που αναδείχθηκαν και αντιμετωπίστηκαν σε ορισμένο βαθμό, μπορούν να αναδειχθούν ορισμένες κατευθύνσεις έρευνας σε εφαρμοσμένο και θεωρητικό επίπεδο. Οι κατευθύνσεις αυτές μπορούν να ειδωθούν ως συνέχιση των παρόντων αποτελεσμάτων

και ταυτόχρονα σκιαγραφούν μέρος του ελλείμματος που πρέπει να καλυφθεί, προκειμένου να ευοδωθούν οι στόχοι και οι προσδοκίες από έναν Σημαντικό Ιστό.

7.2.1. Βελτίωση της επίδοσης

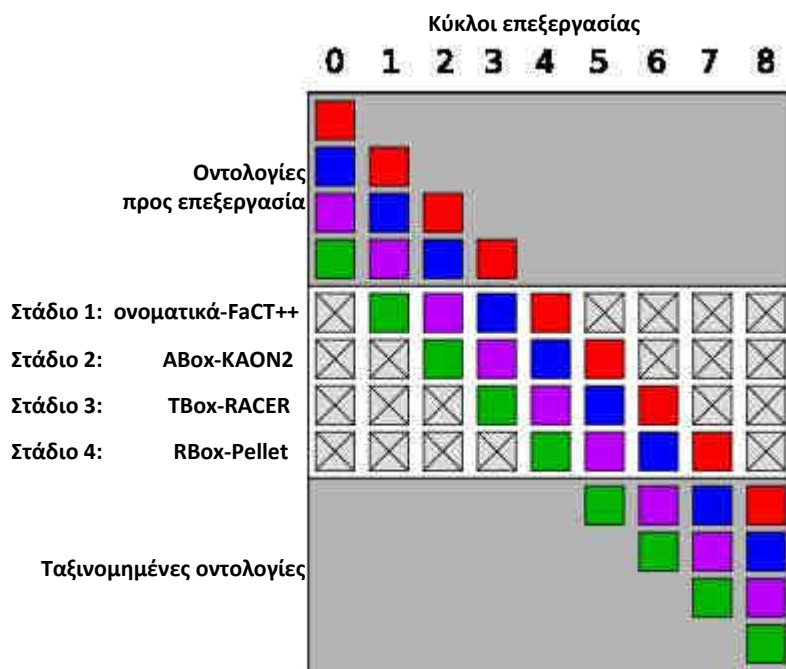
Αρχικά, έχει φανεί ότι επιβεβαιώνεται η διαπίστωση πως η ανακάλυψη γνώσης στο Σημαντικό Ιστό υποφέρει από μεγάλο βαθμό δυσεπιλυσιμότητας, γεγονός το οποίο επιδεινώνεται όσο αυξάνεται η εκφραστικότητα. Ενώ τα θεωρητικά αποτελέσματα είναι βέλτιστα και για αυτό απογοητευτικά, ξεχωρίζουν από τη διατριβή ορισμένες καινοτομικές προσεγγίσεις που θα μπορούσαν να συνεισφέρουν στη βελτίωση της επίδοσης του συλλογισμού και άρα της ανακάλυψης γνώσης:

- Έλεγχος υβριδικών τεχνικών, με συνδυασμό των βέλτιστων στοιχείων κάθε εναλλακτικής προσέγγισης.
- Σημασιακή κατανομή φόρτου, ώστε ο συλλογισμός διακριτών τμημάτων μιας οντολογίας να ανατίθεται σε διαφορετικό μηχανισμό.
- Παραλληλισμός και σημασιακή σωλήνωση, με στόχο την ακολουθιακή επεξεργασία των οντολογικών τμημάτων.

Εκτός από τους στενά συνδεδεμένους με τις Λογικές Περιγραφής αλγορίθμους ταμπλό, υπάρχει μια σειρά εναλλακτικών προσεγγίσεων που μπορούν να χρησιμοποιηθούν για συλλογισμό, πολλές φορές με καλύτερα αποτελέσματα, ειδικά σε συγκεκριμένες περιπτώσεις που οι αλγόριθμοι ταμπλό υστερούν. Ταυτόχρονα, το αντίστροφο επίσης ισχύει, δηλαδή υπάρχουν καταστάσεις όπου οι ταμπλό αλγόριθμοι υπερτερούν έναντι των εναλλακτικών προσεγγίσεων. Ο υβριδισμός προκύπτει επομένως φυσικά και έχει το νόημα του συνδυασμού των καλύτερων χαρακτηριστικών από όλες τις τεχνικές. Προς την κατεύθυνση αυτή, χαρακτηριστική είναι η γραμμή έρευνας που ακολουθείται στους λεγόμενους αλγορίθμους υπερ-ταμπλό [Motik, Shearer, & Horrocks, 2007] οι οποίοι συνδυάζουν ανάλυση πρώτης τάξης με κανόνες ταμπλό στοχεύοντας στην μείωση του βαθμού του μη ντετερμινισμού που εισάγεται ακόμα και για απλές σχετικά λογικές. Επίσης, το σύστημα SHER της IBM [Dolby, et al., 2007], ενώ είναι χτισμένο γύρω από τον μηχανισμό συλλογισμού Pellet, επιτυγχάνει τη βελτίωση της επίδοσης για μεγάλα ABox, αρχικά με το να τα συνοψίζει (summarization) σε ομάδες αποθηκευμένες σε μια βάση δεδομένων και στη συνέχεια να εκλεπτύνει (refine) βαθμηδόν και επιλεκτικά τις ομάδες αυτές. Στόχος είναι να βρεθούν το δυνατόν νωρίτερα τα στιγμιότυπα εκείνα που ικανοποιούν το τιθέμενο συζευγμένο ερώτημα. Κανένας όμως από τους παραπάνω αλγορίθμους δεν μπορεί να χειριστεί αυτή τη στιγμή ονοματικά και, αναγκαστικά, η πολυπλοκότητα θα βελτιώνεται μόνο στην εκάστοτε μέση περίπτωση.

Οι διαφορετικές όμως προσεγγίσεις για συλλογισμό με οντολογίες δεν προτείνονται, γιατί κάποιες από αυτές μπορεί να είναι βέλτιστες αυτόνομα (γιατί τότε θα μπορούσαν να χρησιμοποιηθούν μόνο αυτές οι βέλτιστες), αλλά εμφανίζουν καλύτερες υπολογιστικές ιδιότητες σε οντολογίες με συγκεκριμένα χαρακτηριστικά. Αντιστρέφοντας τον παραπάνω ισχυρισμό προκύπτει ότι διαφορετικές τεχνικές θα μπορούσαν να χρησιμοποιηθούν για διαφορετικά χαρακτηριστικά μιας (της ίδιας) οντολογίας. Με άλλα λόγια, αν μια οντολογία μπορεί να διαιρεθεί σε εκφραστικά τμήματα, κάθε ένα από αυτά μπορεί να ανατεθεί σε διαφορετικό μηχανισμό συμπερασμού. Ήδη μια τέτοια προσέγγιση ακολουθείται στην

περίπτωση του συστήματος QuOnto (βλ. ενότητα 2.6.2), όπου, για την περίπτωση του υποσυνόλου DL-Lite, ο συλλογισμός του TBox και του ABox ανατίθεται χωριστά σε ένα σύστημα ταμπλό και μια βάση δεδομένων αντίστοιχα. Μια τέτοια *σημασιακή κατανομή φόρτου* (semantic load-balancing) βέβαια είναι εφικτή, γιατί έχει βρεθεί τυπικά ότι, για μια τόσο απλή γλώσσα, το ένα τμήμα δεν επηρεάζει το άλλο ή, αλλιώς, ότι ο συλλογισμός στο TBox δεν δημιουργεί ανεξέλεγκτες επιπλοκές στο συλλογισμό με το ABox και αντίστροφα. Στη γενική περίπτωση, μένει να διερευνηθεί πόσο συνεπής μπορεί να παραμείνει ένας τέτοιος διαχωρισμός, ο οποίος σίγουρα δεν θα είναι τετριμμένος (υπενθυμίζεται ότι ένας αλγόριθμος ταμπλό πραγματοποιεί συλλογισμό στο ABox ανάγοντάς το σε ταξινομικά αξιώματα). Ενδεχομένως μια προσέγγιση θα μπορούσε να εξετάσει την κατανομή σε λειτουργικό (functional) επίπεδο: Ο διαχωρισμός να επιχειρείται αρχικά ήδη κατά τη φάση της συντακτικής ανάλυσης μιας οντολογίας, όπως είναι διατυπωμένη σε RDF/XML σύνταξη, και βάσει της τυπικής σημασιολογίας της γλώσσας, η οντολογία να ομαδοποιείται σε (πιθανώς αλληλεπικαλυπτόμενα συντακτικώς) υποσύνολα. Οι σημασιακές προεκτάσεις των αξιωμάτων και γεγονότων που περιλαμβάνονται σε κάθε υποσύνολο αφορούν μόνο αυτό και δεν επηρεάζουν τα άλλα υποσύνολα.



Εικόνα 7.1: Σχηματική απεικόνιση της σημασιακής σωλήνωσης.

Η παραπάνω ιδέα της σημασιακής τμηματοποίησης, που στην ουσία προσβλέπει στα οφέλη της παράλληλης επεξεργασίας, θα μπορούσε να βρει έκφραση και στην τεχνική της *σωλήνωσης* (pipelining). Εφόσον θεωρηθεί εφικτό ότι μια οντολογία μπορεί να χωριστεί σε -σημασιακώς διακριτά- τμήματα, το πρόβλημα π.χ. της ταξινόμησης για κάθε ένα από αυτά θα είναι ανεξάρτητο από τα υπόλοιπα. Κατά συνέπεια, ο χρόνος απόκρισης ενός αλγορίθμου θα είναι μικρότερος για ένα υπο-τμήμα, από ότι για ολόκληρη την οντολογία. Σύμφωνα με την τεχνική της σωλήνωσης, ένα σύνολο διαφορετικών οντολογιών, των οποίων επιθυμούμε για παράδειγμα την ταξινόμηση, μπορεί να τροφοδοτηθεί σειριακά σε

έναν αριθμό μηχανισμών συμπερασμού, κάθε ένας από τους οποίους καταλαμβάνει ένα διαφορετικό στάδιο της σωλήνωσης και ασχολείται με διαφορετικό τμήμα μιας οντολογίας. Σε κάθε τέτοιο στάδιο μπορεί να βρίσκεται ο ίδιος αλγόριθμος, π.χ. το ταμπλό για την *SROIQ*. Αν προκύψει ότι η διάσπαση μιας οντολογίας μπορεί να γίνει σε τμήματα με ελεγχόμενα χαρακτηριστικά, για παράδειγμα, χωριστά οι σχέσεις υπαγωγής των κλάσεων, χωριστά τα χαρακτηριστικά ιδιοτήτων, χωριστά τα ονοματικά και χωριστά τα στιγμιότυπα, σε κάθε στάδιο της σωλήνωσης μπορεί να χρησιμοποιηθεί διαφορετικός, εξειδικευμένος κατά περίπτωση μηχανισμός (εικόνα 7.1). Η σωλήνωση γενικά δεν επιταχύνει τον ίδιο τον αλγόριθμο, αλλά αυξάνει την *ρυθμαπόδοση* (throughput) του συλλογισμού με μεγάλο αριθμό οντολογιών. Στο παραπάνω παράδειγμα της εικόνας 7.1, μετά την παρέλευση 4 κύκλων αρχικοποίησης, αρχίζουμε να λαμβάνουμε στην έξοδο μια ταξινομημένη οντολογία σε χρόνο υποπολλαπλάσιο (ιδανικά, στο ¼) του χρόνου που κανονικά θα απαιτούνταν για την ταξινόμηση ολόκληρης της οντολογίας. Αυτό συμβαίνει γιατί κάθε στάδιο της σωλήνωσης ασχολείται με διαφορετικό κομμάτι κάθε οντολογίας, το οποίο, υποθετικά, μπορεί να διεκπεραιωθεί σε μικρότερο χρόνο.

7.2.2. Εξελικτικότητα σημασιακών προφίλ

Η τεχνική της σημασιακής προσαρμογής εισάγει την έννοια των σημασιακών προφίλ προσαρμογής και επιτυγχάνει την ισχυροποίηση της εκφραστικότητάς τους με διαλειτουργικό τρόπο. Παρά την έμφαση που δίνεται στην ενίσχυση της εκφραστικότητας ενός μοντέλου και των σχέσεων μεταξύ των στιγμιότυπων του για την ενεργοποίηση ισχυρών συμπερασμών, η τεχνική υπονοεί και ένα είδος *σημασιακής εξέλιξης*: Ξεκινώντας από το αρχικό μοντέλο - OWL έγγραφο το οποίο έχει φτωχά εκφραστικά χαρακτηριστικά, παράγονται άλλες οντολογίες που το εμβαθύνουν και το εκλεπτύνουν σημασιακά, ενώ η μια συμπεριλαμβάνει την άλλη. Η κάθε μια από αυτές μπορεί να χρησιμοποιεί κατασκευαστές που την κατατάσσουν σε ολοένα υψηλότερο βαθμό εκφραστικότητας, προτείνοντας έτσι κλιμακωτά επίπεδα σημασιακής περιγραφής.

Έστω για παράδειγμα το ακόλουθο απόσπασμα από το CIDOC-CRM:

Επίπεδο 1: *SH*

```
<owl:ObjectProperty rdf:ID="P96F.by_mother">
  <rdfs:domain rdf:resource="#E67.Birth"/>
  <rdfs:range rdf:resource="#E21.Person"/>
  <rdfs:subPropertyOf rdf:resource="#P11F.had_participant"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="P96B.gave_birth">
  <rdfs:domain rdf:resource="#E21.Person"/>
  <rdfs:range rdf:resource="#E67.Birth"/>
  <rdfs:subPropertyOf rdf:resource="#P11B.participated_in"/>
</owl:ObjectProperty>
```

Ο ορισμός των δύο παραπάνω ιδιοτήτων δεν ξεπερνά το επίπεδο της Λογικής Περιγραφής *SH*. Αν προστεθεί το γεγονός ότι οι δύο αυτές ιδιότητες είναι μεταξύ τους αντίστροφες, δηλαδή:

Επίπεδο 2: SHI

```
<owl:imports rdf:resource="Επίπεδο1.owl"/>

<owl:ObjectProperty rdf:about="&crm;P96F.by_mother">
  <owl:inverseOf rdf:resource="&crm;P96B.gave_birth">
</owl:ObjectProperty>
```

η λογική γίνεται *SHI* και ο συλλογισμός κάπως δυσκολεύει. Αν τώρα προσθέσουμε το γεγονός ότι κάθε άτομο έχει ακριβώς μία μητέρα:

Επίπεδο 3: SHIF

```
<owl:imports rdf:resource="Επίπεδο2.owl"/>
<owl:FunctionalProperty rdf:about="&crm;P96F.by_mother" />
```

η οντολογία γίνεται *SHIF* και βρίσκεται στο επίπεδο της OWL-Lite.

Με ανάλογο τρόπο, θα μπορούσαν να προστεθούν ονοματικά, που θα ανήγαγαν την οντολογία σε επίπεδο OWL DL και στη συνέχεια π.χ. αλυσίδες ρόλων, κατατάσσοντάς την στην εκφραστικότητα της OWL 1.1. Επομένως δεν ενδιαφέρει τόσο τα οντολογικά έγγραφα να αποτελούν μέρος ενός ενιαίου σημασιακού προφίλ, το κάθε ένα από τα οποία επικεντρώνει σε διαφορετική πτυχή ενός μοντέλου (στον εντασιακό πυρήνα και στην εκλέπτυνση εφαρμογής), αλλά το κάθε έγγραφο αποτελεί αυτόνομο προφίλ, που αποτυπώνει πιο επακριβώς και λεπτομερέστερα το προηγούμενο.

Στο παραπάνω παράδειγμα, ό,τι συμπερασμός μπορεί να διεξαχθεί στα επίπεδα 1 και 2 μπορεί να διεξαχθεί και στο επίπεδο 3. Το αντίστροφο όμως δεν ισχύει: οι συμπερασμοί του κάθε επιπέδου δεν είναι γενικά αναπαράξιμοι στα προηγούμενα επίπεδα. Έτσι η τεχνική της σημασιακής προσαρμογής μπορεί να επιτύχει ένα είδος *αυξητικού συλλογισμού*, όπου, ενώ περιμένουμε για τα αποτελέσματα του υψηλότερου επιπέδου, μπορούν κατά το χρόνο αυτό να προκύψουν ορθά -αλλά όχι πλήρη- αποτελέσματα από τα προηγούμενα στάδια, που διεκπεραιώνονται κατά κανόνα ταχύτερα.

Να σημειωθεί επίσης ότι η προσέγγιση αυτή είναι διαφορετική από την ιδέα της σημασιακής σωλήνωσης ή της κατανομής φόρτου που εισήχθησαν νωρίτερα: Πρώτον, η τεχνική αυτή είναι εγγυημένο ότι θα δώσει αποτελέσματα, αφού δεν εξαρτάται από το διαχωρισμό της οντολογίας σε ανεξάρτητα σημασιακά υποσύνολα, το οποίο μπορεί να μην είναι γενικά εφικτό. Αντίθετα βασίζεται σε πολλαπλές διαφορετικές οντολογίες που δεν αποτελούν μέρος ενός συνόλου, αλλά εκλεπτούνουν τεχνητά η μια την άλλη. Δεύτερον, παρόλο που αναμένεται η επεξεργασία των διαφορετικών επιπέδων να γίνεται ταυτόχρονα (πιθανώς από διαφορετικούς μηχανισμούς σε χωριστά υπολογιστικά συστήματα) δεν χρειάζεται να συνδυαστούν τα αποτελέσματα της επεξεργασίας για την παραγωγή της τελικής απάντησης: Αντιθέτως, είμαστε ικανοποιημένοι και με τα ενδιάμεσα, μη πλήρη, αλλά οπωσδήποτε σωστά αποτελέσματα από τα χαμηλότερα στάδια. Κάτι τέτοιο μπορεί να είναι επιθυμητό σε χρονικά κρίσιμα σενάρια, όπως αμυντικές εφαρμογές, ιατρικές διαγνώσεις και διαχείριση βιομηχανικών συστημάτων, όπου η μερική γνώση που ανακτάται μπορεί να αξιοποιηθεί, εφόσον εγγυημένα είναι ορθή και, σταδιακά, να λαμβάνονται οι πιο αναλυτικές απαντήσεις.

BIBΛΙΟΓΡΑΦΙΑ

- Alani, H., Kim, S., Millard, D. E., Weal, M. J., Hall, W., Lewis, P. H., et al. (2003). Automated Ontology-Based Knowledge Extraction from Web Documents. *IEEE Intelligent Systems*, 18 (1), pp. 14-21.
- Antoniou, G., & Harmelen, F. v. (2004). *A Semantic Web Primer*. MIT Press.
- Baader, F., & Nutt, W. (2007). Basic Description Logics. In F. Baader, D. Calvanese, D. McGuinness, D. Nardi, & P. F. Patel-Schneider (Eds.), *The Description Logics Handbook* (2nd ed.). Cambridge.
- Baader, F., Brandt, S., & Lutz, C. (2005). Pushing the EL Envelope. *Proc. of the 19th Joint Int. Conf. on Artificial Intelligence (IJCAI 2005)*.
- Baader, F., Küsters, R., & Wolter, F. (2007). Extensions to description logics. In F. Baader, D. Calvanese, D. McGuinness, D. Nardi, & P. F. Patel-Schneider (Eds.), *The Description Logic Handbook* (2nd ed.). Cambridge.
- Balzer, D. (2006). *Partial Definition of the CIDOC Conceptual Reference Model version 3.4.9 in OWL*. Retrieved from <http://cidoc.ics.forth.gr/OWL/crm3.4.9/index.html>
- Bechhofer, S. (2006). DIG 2.0: An Interface for Description Logics Systems. *Poster in 5th Int. Semantic Web Conference (ISWC 2006)*. The Description Logic Implementation Group.
- Bechhofer, S., Horrocks, I., Patel-Schneider, P. F., & Tessaris, S. (1999). A proposal for a description logic interface. *Proc. of International Workshop on Description Logics (DL'99)*, (pp. 33-36).
- Beckett, D., Miller, E., & Brickley, D. (2002). Expressing Simple Dublin Core in RDF/XML. DCMI Recommendation. Retrieved from <http://dublincore.org/documents/dcmes-xml/>
- Biron, P. V., & Malhotra, A. (2004). XML Schema Part 2: Datatypes Second Edition. W3C Recommendation. Retrieved from <http://www.w3.org/TR/xmlschema-2/>
- Brachman, R. J., & Levesque, H. J. (2004). *Knowledge Representation and Reasoning*. Morgan Kaufmann.
- Brickley, D., & Guha, R. V. (2004). RDF Vocabulary Description Language 1.0: RDF Schema. W3C Recommendation. Retrieved from <http://www.w3.org/TR/rdf-schema/>
- Broekstra, J., Kampman, A., & Harmelen, F. v. (2002). Sesame: A Generic Architecture for Storing and Querying RDF and RDF Schema. *Proc. of the 1st International Semantic Web Conference (ISWC 2002)* (pp. 54-68). Springer.
- Calvanese, D., & Giacomo, G. d. (2007). Expressive Description Logics. In F. Baader, D. Calvanese, D. McGuinness, D. Nardi, & P. F. Patel-Schneider (Eds.), *The Description Logic Handbook* (2nd ed.). Cambridge.
- Calvanese, D., Giacomo, G. d., & Lenzerini, M. (1998). On the decidability of query containment under constraints. *Proc. of the 17th ACM Symp. on Principles of Database Systems (PODS 1998)* (pp. 149-158). ACM Press & Addison Wesley.
- Calvanese, D., Giacomo, G. d., Lembo, D., Lenzerini, M., & Rosati, R. (2005). Tailoring OWL for Data Intensive Ontologies. *Proc. of the 1st OWL Experiences and Directions Workshop (OWLED 2005)*.
- Chaudhri, V. K., Farquhar, A., Fikes, R., Karp, P. D., & Rice, J. P. (1998). *Open Knowledge Base Connectivity 2.0.3 Proposed Specification*. Retrieved from <http://www.ai.sri.com/~okbc/spec/okbc2/okbc2.html>
- Crofts, N., Doerr, M., & Gill, T. (2003). The CIDOC Conceptual Reference Model: A standard for communicating cultural contents. *Cultivate Interactive* (9), pp. <http://www.cultivate-int.org/issue9/chios/>.
- DCMI Usage Board. (2008). DCMI Metadata Terms. DCMI Recommendation. Retrieved from <http://dublincore.org/documents/dcmi-terms/>

- DCMI-Libraries Working Group. (2004). Library Application Profile. DCMI Working Draft. Retrieved from <http://dublincore.org/documents/2004/09/10/library-application-profile/>
- Dickinson, I. (2004). *Implementation experience with the DIG 1.1 specification*. Hewlett-Packard.
- Dill, S., Eiron, D., Gibson, D., Gruhl, D., Guha, R., Jhingran, A., et al. (2003). SemTag and Seeker: Bootstrapping the Semantic Web via Automated Semantic Annotation. *Proc. of the 12th International Conference on World Wide Web*, (pp. 178-186).
- Doerr, M. (2003). The CIDOC conceptual reference model: an ontological approach to semantic interoperability of metadata. *AI Magazine*, 24 (3), pp. 75-92.
- Doerr, M., Hunter, J., & Lagoze, C. (2003). Towards a Core Ontology for Information Integration. *Journal of Digital Information*, 4 (1), p. 169.
- Dolby, J., Fokoue, A., Kalyanpur, A., Kershenbaum, A., Ma, L., Schonberg, E., et al. (2007). Scalable Semantic Retrieval Through Summarization and Refinement. *Proc. of the 21st Conference on Artificial Intelligence (AAAI 2007)*, (pp. 299-304).
- Duval, E., Hodgins, W., Sutton, S., & Weibel, S. (2002). Metadata Principles and Practicalities. *D-Lib Magazine*, 8 (4), p. <http://www.dlib.org/dlib/april02/weibel/04weibel.html>.
- Fensel, D., Horrocks, I., Harmelen, F. v., Decker, S., Erdmann, M., & Klein, M. (2000). OIL in a nutshell. *Proc. of the 12th European Workshop on Knowledge Acquisition, Modeling, and Management (EKAW '00)* (pp. 1-16). Springer.
- Fikes, M. R., & Genesereth, R. E. (1994). *Knowledge Interchange Format (KIF) Version 3.0 Reference Manual*. Retrieved from <http://logic.stanford.edu/kif/Hypertext/kif-manual.html>
- Fikes, R., Hayes, P., & Horrocks, I. (2002). *DQL - A Query Language for the Semantic Web*. KSL Technical Report 02-05.
- Fikes, R., Hayes, P., & Horrocks, I. (2003). *OWL-QL: A Language for Deductive Query Answering on the Semantic Web*. KSL Technical Report 03-14.
- Gasse, F., Sattler, U., & Haarslev, V. (2008). Rewriting Rules into SROIQ Axioms. *Proceedings of the 2008 International Workshop on Description Logics (DL-2008)*. to appear.
- Georgantīs, N. P., Koutsomitropoulos, D. A., Zafiris, P. A., & Papatheodorou, T. (2002). A Review and Evaluation of Platforms and Tools for building e-Catalogs. *Proc. of 35th Annual Hawaii International Conference on System Sciences (HICSS'02)*. IEEE.
- Glimm, B., Horrocks, I., & Sattler, U. (2006). *Conjunctive Query Answering for the Description Logic SHOIQ*. Technical Report, University of Manchester, School of Computer Science.
- Glimm, B., Horrocks, I., Lutz, C., & Sattler, U. (2008). Conjunctive Query Answering for the Description Logic SHIQ. *Journal of Artificial Intelligence Research* (31), pp. 151-198.
- González-Castillo, J., Trastour, D., & Bartolini, C. (2001). Description Logics for Matchmaking of Services. *Proc. of KI-2001 Workshop on Applications of Description Logics*.
- Grau, B. C., & Motik, B. (2008). OWL 2 Web Ontology Language: Model-Theoretic Semantics. W3C Working Draft. Retrieved from <http://www.w3.org/TR/owl2-semantics/>
- Grau, B. C., Horrocks, I., Parsia, B., Patel-Schneider, P., & Sattler, U. (2006). Next Steps for OWL. *Proc. of OWL Experiences and Directions Workshop (OWLED '06)*.
- Grau, B. C., Motik, B., & Patel-Schneider, P. (2008). OWL 2 Web Ontology Language: XML Serialization. W3C Working Draft. Retrieved from http://www.w3.org/2007/OWL/wiki/XML_Serialization#ref-OWL_2_Specification
- Grau, B. C., Motik, B., Wu, Z., Fokoue, A., & Lutz, C. (2008). OWL 2 Web Ontology Language: Profiles. W3C Working Draft. Retrieved from <http://www.w3.org/TR/owl2-profiles/>
- Grosz, B. N., Horrocks, I., Volz, R., & Decker, S. (2003). Description Logic Programs: Combining Logic Programs with Description Logic. *Proc. of the Twelfth International World Wide Web Conference (WWW 2003)* (pp. 48-57). ACM.

- Gruber, T. R. (1993a). A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5 (2), pp. 199-220.
- Gruber, T. R. (1993b). Toward Principles for the Design of Ontologies Used for Knowledge Sharing. In N. Guarino, & R. Poli (Eds.), *Formal Ontology in Conceptual Analysis and Knowledge Representation*. Kluwer.
- Guarino, N. (1998). Formal Ontology and Information Systems. In N. Guarino (Ed.), *Formal Ontology in Information Systems. Proceedings of FOIS'98* (pp. 3-15). IOS Press.
- Guha, R., & McCool, R. (2003). TAP: A Semantic Web Platform. *Computer Networks*, 42 (5), pp. 557-577.
- Haarslev, V., & Möller, R. (2003). Racer: A Core Inference Engine for the Semantic Web. *Proc. of the 2nd International Workshop on Evaluation of Ontology-based Tools (EON2003)*, (pp. 27-36).
- Heery, R., & Patel, M. (2000). Mixing and Matching Metadata Schemas. *Ariadne* (25), pp. <http://www.ariadne.ac.uk/issue25/app-profiles/intro.html>.
- Heflin, J., Hendler, J., & Luke, S. (1999). SHOE: A Knowledge Representation Language for Internet Applications. Retrieved from <http://www.cs.umd.edu/projects/plus/SHOE/pubs/#tr99>
- Hendler, J. (2008). Web 3.0: Chicken Farms on the Semantic Web. *Computer*, 41 (1), pp. 106-108.
- Hendler, J., & Heflin, J. (2000). Searching the Web with SHOE. In *Artificial Intelligence for Web Search: Papers from the AAAI Workshop* (pp. 35-40). AAAI Press.
- Hodges, W. (1993). *Model Theory*. Cambridge.
- Horridge, M., Bechhofer, S., & Noppens, O. (2007). Igniting the OWL 1.1 Touch Paper: The OWL API. *Proc. of the OWL Experiences and Directions Workshop (OWLED'07)*.
- Horridge, M., Drummond, N., Goodwin, J., Rector, A., Stevens, R., & Wang, H. H. (2006). The Manchester Syntax. *Proc. of the OWL Experiences and Directions Workshop (OWLED '06)*.
- Horridge, M., Tsarkov, D., & Redmond, T. (2006). Supporting Early Adoption of OWL 1.1 with Protégé-OWL and FaCT++. *Proc. of the OWL Experiences and Directions Workshop (OWLED '06)*.
- Horrocks, I. (2007). Implementation and Optimization Techniques. In F. Baader, D. Calvanese, D. McGuinness, D. Nardi, & P. F. Patel-Schneider (Eds.), *The Description Logic Handbook* (2nd ed.). Cambridge.
- Horrocks, I., & Patel-Schneider, P. F. (2003). Reducing OWL entailment to description logic satisfiability. In D. Fensel, K. Sycara, & J. Mylopoulos (Ed.), *Proc. of the 2003 International Semantic Web Conference (ISWC 2003). Number 2870 of LNCS*, pp. 17-29. Springer.
- Horrocks, I., & Sattler, U. (2005). A Tableaux Decision Procedure for SHOIQ. *Proc. of the 19th Int. Joint Conf. on Artificial Intelligence (IJCAI 2005)*, (pp. 448-453).
- Horrocks, I., & Sattler, U. (1999). A Description Logic with Transitive and Inverse Roles and Role Hierarchies. *Journal of Logic and Computation*, 9 (3), pp. 385-410.
- Horrocks, I., & Sattler, U. (2007). A Tableaux Decision Procedure for SHOIQ. *Journal of Automated Reasoning*, 39 (3), pp. 245-429.
- Horrocks, I., & Sattler, U. (2004). Decidability of SHIQ with complex role inclusion axioms. *Artificial Intelligence*, 120 (1-2), pp. 79-104.
- Horrocks, I., & Sattler, U. (2001). Ontology Reasoning in the SHOQ(D) Description Logic. *Proc of the Int. Joint Conf. on Artificial Intelligence (IJCAI 2001)* (pp. 199-204). Morgan Kaufmann.
- Horrocks, I., & Sattler, U. (2002). Optimised reasoning for SHIQ. *Proc. of the 15th Eur. Conf. on Artificial Intelligence (ECAI 2002)*, (pp. 277-281).
- Horrocks, I., & Tessaris, S. (2002). Querying the Semantic Web: a Formal Approach. *Proc. of the 1th Int. Semantic Web Conference (ISWC '02)*, (pp. 177-191).
- Horrocks, I., & Voronkov, A. (2006). Reasoning Support for Expressive Ontology Languages Using a Theorem Prover. *Proc of the 4th International Symposium on Foundations of Information and Knowledge Systems (FoIKS)* (pp. 201-218). Springer.

- Horrocks, I., Kutz, O., & Sattler, U. (2005). *The Even More Irresistible SROIQ*. Technical Report, University of Manchester.
- Horrocks, I., Kutz, O., & Sattler, U. (2006). The Even More Irresistible SROIQ. *Proc. of the 10th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR 2006)* (pp. 57-67). AAAI Press.
- Horrocks, I., Parsia, B., Patel-Schneider, P. F., & Hendler, J. (2005). Semantic web architecture: Stack or two towers? *Third Workshop on Principles and Practice of Semantic Web Reasoning*.
- Horrocks, I., Patel-Schneider, P. F., & Harmelen, F. v. (2003). From SHIQ and RDF to OWL: The making of a web ontology language. *Journal of Web Semantics*, 1 (1), pp. 7-26.
- Horrocks, I., Patel-Schneider, P. F., Bechhofer, S., & Tsarkov, D. (2005). OWL Rules: A Proposal and Prototype Implementation. *Journal of Web Semantics*, 3 (1), pp. 23-40.
- Hsu, E., & McGuinness, D. (2003). Wine Agent: Semantic Web Testbed Application. *Proc. of the 2003 Workshop on Description Logics*.
- Hustadt, U., & Schmidt, R. A. (2000). Modal reasoning by translation and first-order resolution. *Proc. of 2000 Int. Conference on Automated Reasoning with Analytic Tableaux and Related Methods (TABLEAUX 2000)* (pp. 67-71). Springer.
- Hustadt, U., Motik, B., & Sattler, U. (2004). Reducing SHIQ Description Logic to Disjunctive Datalog Programs. *Proc. of the 9th International Conference on Knowledge Representation and Reasoning (KR2004)*, (pp. 152-162).
- Hutt, K. (2005). A Comparison of RDF Query Languages. *Rensselaer at Hartford Computer Science Seminar*. http://www.rh.edu/~rhb/cs_seminar_2005/SessionE1/hutt.pdf.
- Kakali, C., Lourdi, I., Stasinopoulou, T., Bountouri, L., Papatheodorou, C., Doerr, M., et al. (2007). Integrating Dublin Core Metadata for Cultural Heritage Collections Using Ontologies. *Proc. of the 7th International Conference on Dublin Core and Metadata Application (DC 2007)*, (pp. 128-139).
- Karp, P. D., Chaudhri, V. K., & Thomere, J. (1999). *XOL: An XML-Based Ontology Exchange Language*. Retrieved from <http://www.ai.sri.com/pkarp/xol/xol.html>
- Kazakov, Y. (2008). SRIQ and SROIQ are Harder than SHOIQ. *Proc. of the 21st Int. Workshop on Description Logics (DL2008)*. to appear.
- Kazakov, Y., & Motik, B. (2006). A Resolution-Based Decision Procedure for SHOIQ. In U. Furbach, & N. Shankar (Ed.), *Proc. of Third International Joint Conference on Automated Reasoning (IJCAR 2006)*. Number 4130 of LNCS, pp. 662-677. Springer.
- Kazakov, Y., Sattler, U., & Zolin, E. (2007). How many legs do I have? Non-simple roles in number restrictions revisited. *Proc. of the 14th Int. Conf. on Logic for Programming Artificial Intelligence and Reasoning (LPAR 2007)*. Springer.
- Kifer, M., De Bruijn, J., Boley, H., & Fensel, D. (2005). A Realistic Architecture for the Semantic Web. *Proc. of 2005 Int. Conf. on Rules and Rule Markup Languages (RuleML 2005)*. Number 3791 of LNCS, pp. 17-29. Springer.
- Kiryakov, A., Ognyanov, D., & Manov, D. (2005). OWLIM – a Pragmatic Semantic Repository for OWL. *Proc. of Int. Workshop on Scalable Semantic Web Knowledge Base Systems (SSWS 2005), WISE 2005* (pp. 182-192). Springer.
- Kokkeliink, S., & Schwänzl, R. (2002). *Expressing Qualified Dublin Core in RDF / XML*. Retrieved from <http://dublincore.org/documents/2002/07/31/dcq-rdf-xml/>
- Koutsomitropoulos, D. A., & Papatheodorou, T. S. (2007). Expressive Reasoning about Cultural Heritage Knowledge Using Web Ontologies. *Proc. of 3d International Conference on Web Information Systems and Technologies (WEBIST 07), WIA track*, pp. 276-281.
- Koutsomitropoulos, D. A., Fragakis, M. F., & Papatheodorou, T. S. (2005). A Methodology for Conducting Knowledge Discovery on the Semantic Web. In *Proc. of 16th ACM Conference on Hypertext and Hypermedia (Hypertext 2005)-International Workshop on Adaptive and Personalized Semantic Web*.

- Koutsomitropoulos, D. A., Fragakis, M. F., & Papatheodorou, T. S. (2006a). A Methodology for Conducting Knowledge Discovery on the Semantic Web. In S. Sirmakessis (Ed.), *Adaptive and Personalized Semantic Web* (Vol. 14 of Studies In Computational Intelligence, pp. 95-105). Springer.
- Koutsomitropoulos, D. A., Fragakis, M. F., & Papatheodorou, T. S. (2006b). Discovering Knowledge in Web Ontologies: A Methodology and Prototype Implementation. *Proc. of SEMANTICS 2006 International Conference* (pp. 151-164). Österreichische Computer Gesellschaft.
- Koutsomitropoulos, D. A., Meidanis, D. P., Kandili, A. N., & Papatheodorou, T. S. (2008). Establishing the Semantic Web Reasoning Infrastructure on Description Logic Inference Engines. In Y. Manolopoulos, J. Filipe, P. Constantopoulos, & J. Cordeiro (Eds.), *Enterprise Information Systems* (Vol. 3 of Lecture Notes in Business Information Processing, pp. 351-362). Springer.
- Koutsomitropoulos, D. A., Meidanis, D. P., Kandili, A. N., & Papatheodorou, T. S. (2006). OWL-based Knowledge Discovery Using Description Logics Reasoners. In *Proc. of 8th International Conference on Enterprise Information Systems (ICEIS 2006), SAIC track*, (pp. 43-50).
- Koutsomitropoulos, D. A., Paloukis, G. E., & Papatheodorou, T. S. (2007b). From Metadata Application Profiles to Semantic Profiling: Ontology Refinement and Profiling to Strengthen Inference-based Queries on the Semantic Web. *Int. Journal on Metadata, Semantics and Ontologies*, 2 (4), pp. 268-280.
- Koutsomitropoulos, D. A., Paloukis, G. E., & Papatheodorou, T. S. (2007a). Ontology-based Knowledge Acquisition through Semantic Profiling. *Proc. of 2nd International Conference on Metadata and Semantics Research (MISR 2007)*.
- Koutsomitropoulos, D. A., Paloukis, G. E., & Papatheodorou, T. S. (to appear). Semantic Application Profiles: A Means to Enhance Knowledge Discovery in Domain Metadata Models. In M. A. Sicilia, & M. Lytras (Eds.), *Metadata and Semantics*. Springer.
- Koutsomitropoulos, D. A., Solomou, G. D., & Papatheodorou, T. S. (submitted). Towards a Practical Dublin Core Ontology for the Semantic Web. *7th Semantic Web Int. Conference (ISWC 2008)*.
- Kremen, P., & Sirin, E. (2008). SPARQL-DL Implementation Experience. *Proc. of the OWL Experiences and Directions Workshop (OWLED-2008)*.
- Krötzsch, M., Rudolph, S., & Hitzler, P. (2008). Description Logic Rules. *Proceedings of the 18th European Conference on Artificial Intelligence (ECAI-08)*. to appear.
- Levy, A., & Rousset, M. (1998). CARIN: A Representation Language Combining Horn rules and Description Logics. *Artificial Intelligence*, 104 (1-2), pp. 165-209.
- Luke, S., & Heflin, J. (2000). *SHOE 1.01 Proposed Specification*. Retrieved from <http://www.cs.umd.edu/projects/plus/SHOE/spec.html>
- Luke, S., Spector, L., & Rager, D. (1996). Ontology-Based Knowledge Discovery on the World-Wide Web. In A. Franz, & H. Kitano (Eds.), *Working Notes of the Workshop on Internet-Based Information Systems at the 13th National Conference on Artificial Intelligence (AAAI '96)* (pp. 96-102). AAAI Press.
- Lutz, C., & Sattler, U. (2000). Mary likes all Cats. *Proc. of the 2000 Int. Workshop in Description Logics (DL 2000), Vol 33 in CEUR-WS*, pp. 213-226.
- Manola, F., & Miller, E. (2004). RDF Primer. W3C Recommendation. Retrieved from <http://www.w3.org/TR/rdf-primer/>
- McBride, B. (2002). Jena: A Semantic Web Toolkit. *IEEE Internet Computing*, 6 (6), pp. 55-59.
- McGuinness, D. L., & Harmelen, F. v. (2004). OWL Web Ontology Language Overview. W3C Recommendation. Retrieved from <http://www.w3.org/TR/owl-features/>
- McGuinness, D., Fikes, R., Hendler, J., & Stein, L. (2002). DAML+OIL: an ontology language for the Semantic Web. *IEEE Intelligent Systems*, 17 (5), pp. 72-80.
- Motik, B., & Sattler, U. (2006). A Comparison of Reasoning Techniques for Querying Large Description Logic ABoxes. *Proc. of the 13th International Conference on Logic for Programming Artificial Intelligence and Reasoning (LPAR 2006)*.

- Motik, B., Patel-Schneider, P. F., & Horrocks, I. (2008). OWL 2 Web Ontology Language: Structural Specification and Functional-Style Syntax. W3C Working Draft. Retrieved from <http://www.w3.org/TR/owl2-syntax/>
- Motik, B., Sattler, U., & Struder, R. (2005). Query Answering for OWL-DL with Rules. *Journal of Web Semantics*, 3 (1), pp. 41-60.
- Motik, B., Shearer, R., & Horrocks, I. (2007). Optimized Reasoning in Description Logics using Hypertableaux. *Proc. of the 21st Int. Conf. on Automated Deduction (CADE-21)* (pp. 67-83). Springer.
- Nardi, D., & Brachman, R. J. (2007). An Introduction to Description Logics. In F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, & P. Patel-Schneider (Eds.), *The Description Logic Handbook* (2nd ed.). Cambridge.
- Network Inference. (2003). Description Logics. White paper. Retrieved from <http://www.networkinference.com>
- Nilsson, M. (2008). Description Set Profiles: A constraint language for Dublin Core Application Profiles. DCMI Working Draft. Retrieved from <http://dublincore.org/documents/dc-dsp/>
- Nilsson, M., & Baker, T. (2008). Notes on DCMI specifications for Dublin Core metadata in RDF. DCMI Note. Retrieved from <http://dublincore.org/documents/dc-rdf-notes/>
- Nilsson, M., Baker, T., & Johnston, P. (2008). The Singapore Framework for Dublin Core Application Profiles. Retrieved from <http://www.dublincore.org/documents/singapore-framework/>
- Nilsson, M., Powell, A., Johnston, P., & Naeve, A. (2008). Expressing Dublin Core metadata using the Resource Description Framework (RDF). DCMI Recommendation. Retrieved from <http://dublincore.org/documents/dc-rdf/>
- Noy, N., & Rector, A. (Eds.). (2006). Defining N-ary Relations on the Semantic Web. Semantic Web Best Practices and Deployment Working Group Note. Retrieved from <http://www.w3.org/TR/swbp-n-aryRelations/>
- Oischinger, M., Schiemann, B., & Goerz, G. (2008). Short Documentation of the CIDOC CRM (4.2.4) Implementation in OWL-DL. C.S.D., University of Erlangen-Nuremberg. Retrieved from <http://www8.informatik.uni-erlangen.de/inf8/en/dienste.html>
- Paolucci, M., Kawamura, T., Payne, T. R., & Sycara, K. (2002). Semantic Matching of Web Services Capabilities. *Proc. of the 1st International Semantic Web Conference (ISWC '02)*.
- Papadimitriou, C. H. (1995). *Computational Complexity*. Addison-Wesley.
- Parsia, B., & Patel-Schneider, P. F. (2008). OWL 2 Web Ontology Language: Primer. W3C Working Draft. Retrieved from <http://www.w3.org/TR/owl2-primer/>
- Patel-Schneider, P. F., & Horrocks, I. (2006). OWL 1.1 Web Ontology Language Overview. W3C Member Submission. Retrieved from <http://www.w3.org/Submission/owl11-overview/>
- Patel-Schneider, P. F., Hayes, P., & Horrocks, I. (2004). *OWL Web Ontology Language: Semantics and Abstract Syntax*.
- Powell, A., & Johnston, P. (2003). *Guidelines for implementing Dublin Core in XML*. DCMI Recommendation, <http://dublincore.org/documents/dc-xml-guidelines/>.
- Powell, A., Johnston, P., & Baker, T. (2008). Domains and Ranges for DCMI Properties. Retrieved from <http://dublincore.org/documents/domain-range/>
- Powell, A., Nilsson, M., Naeve, A., Johnston, P., & Baker, T. (2007). DCMI Abstract Model. DCMI Recommendation. Retrieved from <http://dublincore.org/documents/abstract-model/>
- Prud'hommeaux, E., & Seaborne, A. (2008). SPARQL Query Language for RDF. W3C Recommendation. Retrieved from <http://www.w3.org/TR/rdf-sparql-query/>
- Ratanajaipan, P., Nantajeewarawat, E., & Wuwongse, V. (2006). OWL/XDD: A Formal Language for Application Profiles. *Proc. of 2006 IEEE Conf. on Cybernetics and Intelligent Systems*.

- Ribière, M., & Charlton, P. (2000). *Ontology Overview*. Retrieved from <http://www.fipa.org/docs/input/f-in-00045/f-in-00045.pdf>
- Russell, S., & Norvig, P. (2005). *Τεχνητή Νοημοσύνη: Μια σύγχρονη Προσέγγιση* (2η εκδ.). (Γ. Ρεφανίδης, Φ. Σκουλαρίκης, & Τ. Άλβας, Μεταφρ.) Κλειδάριθμος.
- Schmidt, R. A., & Tishkovsky, D. (2007). Using Tableau to Decide Expressive Description Logics with Role Negation. *Proc. of the 6th Int. Semantic Web Conference (ISWC 2007)* (pp. 438-451). Springer.
- Shah, U., Finin, T., Joshi, A., Cost, R. S., & Mayfield, J. (2002). Information Retrieval on the Semantic Web. *Proc. of the 11th International Conference on Information and Knowledge Management*, (pp. 461-468).
- Sirin, E., & Parsia, B. (2007). SPARQL-DL: SPARQL Query for OWL-DL. *3rd OWL Experiences and Directions Workshop (OWLED-2007)*.
- Sirin, E., Parsia, B., Grau, B. C., Kalyanpur, A., & Katz, Y. (2007). Pellet: A practical OWL-DL reasoner. *Journal of Web Semantics*, 5 (2).
- Smith, B., & Welty, C. (2001). Ontology: Towards a new synthesis. In C. Welty, & B. Smith (Eds.), *Formal Ontology in Information Systems* (pp. iii-x). ACM.
- Solomou, G., Koutsomitropoulos, D., & Papatheodorou, T. (2008). Semantics-Aware Querying of Web-Distributed RDF(S) Repositories. *Proc. of 1st Workshop on Semantic Interoperability in the European Digital Library (SIEDL 08), 5th European Semantic Web Conference (ESWC 08)*, (pp. 39-50).
- Stoilos, G., Simou, N., Stamou, G., & Kollias, S. (2006). Uncertainty and the Semantic Web. *IEEE Intelligent Systems*, 21 (5), pp. 84-87.
- Stoilos, G., Stamou, G., Pan, J., Tzouvaras, V., & Horrocks, I. (2007). Reasoning with Very Expressive Fuzzy Description Logics. *Journal of Artificial Intelligence Research*, 30 (8), pp. 273-320.
- ter Horst, H. J. (2005). Combining RDF and Part of OWL with Rules: Semantics, Decidability, Complexity. *Proc. of 2005 Int. Semantic Web Conference (ISWC 2005)*, (pp. 668-684).
- Tobies, S. (2001). *Complexity results and practical algorithms for logics in Knowledge Representation*. PhD Thesis, LuFG Theoretical Computer Science, RWTH-Aachen.
- Tsarkov, D., & Horrocks, I. (2003). DL Reasoner vs. First-Order Prover. *Proc. of the 2003 Description Logic Workshop (DL 2003)*, (pp. 152-159).
- Tsarkov, D., & Horrocks, I. (2006). FaCT++ description logic reasoner: System description. *Proc. of the 3rd Int. Joint Conf. on Automated Reasoning (IJCAR '06)*.
- Tsarkov, D., Horrocks, I., & Patel-Schneider, P. F. (2007). Optimising Terminological Reasoning for Expressive Description Logics. *Journal of Automated Reasoning*, 39 (3), pp. 277-316.
- Tsarkov, D., Riazanov, A., Bechhofer, S., & Horrocks, I. (2004). Using Vampire to Reason with OWL. *Proc. of the 3rd International Semantic Web Conference (ISWC 2004)* (pp. 471-485). Springer.
- Villanueva-Rosales, N., & Dumontier, M. (2007). Describing chemical functional groups in OWL-DL for the classification of chemical compounds. *Proc. of OWL Experiences and Directions Workshop (OWLED '07)*.
- Βλαχάβας, Ι., Κεφαλάς, Π., Βασιλειάδης, Ν., Κόκκορας, Φ., & Σακελλαρίου, Η. (2006). *Τεχνητή Νοημοσύνη* (3η εκδ.). Γκιούρδας.
- Κουτσομητρόπουλος, Δ. (2004). *Αξιοποίηση Οντολογιών και Χρήση Τεχνολογιών του Σημαντικού Ιστού για Αναπαράσταση, Διαχείριση και Ανακάλυψη Γνώσης μέσω του Διαδικτύου*. Μεταπτυχιακή Διπλωματική Εργασία, Πανεπιστήμιο Πατρών, Τμήμα Μηχανικών Η/Υ και Πληροφορικής, Πάτρα.

- ονοματικά, 47
- ακολουθία εκτέλεσης
 - προς τα εμπρός, 29, 39, 89
 - προς τα πίσω, 39
- αλγόριθμος
 - RETE, 39
 - ταμπλό, 52
- αλυσίδες ρόλων, 28
- ανάλυση, 38
 - SLD, 39
- αποδοτικότητα, 91
- Άρνηση ως αποτυχία, 40
- αρνητικά γεγονότα, 26
- αυτοπαθείς έννοιες, 26
- βατά τεμάχια, 29
- γνώση
 - εκτασιακή, 45
 - εντασιακή, 45
- διαδικασία απόφασης ταμπλό. *Βλέπε*
 - αλγόριθμος ταμπλό
- διαλειτουργικότητα
 - σημασιακή, 125
 - συντακτική, 137
- διαχρονικότητα, 90
- εκφραστικότητα, 90
- Έλεγχος στιγμιότυπου, 30
- εννοιολογική σύλληψη, 10
- επιλύσιμα τεμάχια, 10
- ιδιότητες
 - (μη) ανακλαστικές, 25
 - (μη) συμμετρικές, 25
 - ξένες, 25
- Ικανοποιησιμότητα έννοιας, 30
- καθολικός ρόλος, 21, 50
- Λογικές Περιγραφής, 39
- μηχανισμοί απόδειξης θεωρημάτων, 39
- μοντέλο
 - έννοιας, 53
 - επιδιωκόμενο, 11
 - λογικής γλώσσας, 11
 - οντολογίας, 25
- ονοματικά, 63
- οντολογία, 9, 11
- παρονομασία, 20, 28, 129
- πεδίο
 - απτό, 21, 55
 - γνώσης, 102
 - γνώσης, ημιδομημένο, 123
 - τιμών δεδομένων, 27
- περιορισμοί αριθμού, 26
 - απροσδιόριστοι, 44
 - προσδιορισμένοι, 44
- περιορισμοί πληθικότητας. *Βλέπε*
 - περιορισμοί αριθμού
- Περιορισμοί πληθικότητας, 18
- περιορισμοί ρόλων
 - καθολικοί, 44
 - υπαρξιακοί, 44
- πολυπλοκότητα
 - επερώτησης, 30
 - συνδυασμένη, 30
 - ταξινομική, 30
- Πολυπλοκότητα
 - δεδομένων, 30
- προκαθορισμένο γεγονός, 37
- προσαρμογή
 - μεταδεδομένων εφαρμογής, 104
 - σημασιακή, 103, 113
- σύγκρουση, 54
- συζευγμένα ερωτήματα, 30, 75
- συλλογισμός
 - ασαφής, 38
 - μη μονοτονικός, 37
 - μονοτονικός, 1
- σύμπαν αναφοράς, 10
- συμπερασμός, 37
 - βασισμένος σε στιγμιότυπα, 75
 - ταξινομικός, 74
- Συνέπεια οντολογίας, 30
- ταμπλό, 53
- Ταξινόμηση οντολογίας, 30
- Ταξινομική πολυπλοκότητα, 30
- Υπαγωγή έννοιας, 30
- υπαναχώρηση
 - κατευθυνόμενη από εξάρτηση, 60
 - με άλμα, 60

assertional	δηλωτικός
backjumping	υπαναχώρηση με άλμα
cardinality	πληθικότητα
conceptualization	εννοιολογική σύλληψη
concrete domain	απτό πεδίο
consistency	συνέπεια
dependency-directed backtracking	υπαναχώρηση κατευθυνόμενη από εξάρτηση
domain	πεδίο, πεδίο ορισμού
entailment	συνεπαγωγή
existential quantification	υπαρξιακή ποσοτικοποίηση
fuzzy reasoning	ασαφής συλλογισμός
inference	συμπερασμός
inference engine	μηχανισμός συμπερασμού
internalization	εσωτερίκευση
interoperability	διαλειτουργικότητα
interpretation	διερμήνευση
local reflexivity	τοπική ανακλαστικότητα, αυτοπάθεια
nominal	ονοματικό
profiling	προσαρμογή
pruning	παρονομασία
qualified number restriction	προσδιορισμένοι περιορισμοί αριθμού
range	πεδίο τιμών
reasoning	συλλογισμός
reflexive	ανακλαστικός
refutation	διάψευση
resolution	ανάλυση
role filler	πληρωτής ρόλου
tableau decision procedure	διαδικασία απόφασης ταμπλό
theorem prover	μηχανισμός απόδειξης θεωρημάτων
unfolding	ξεδίπλωμα
universal quantification	καθολική ποσοτικοποίηση
universal role	καθολικός ρόλος
unqualified number restriction	απροσδιόριστοι περιορισμοί αριθμού

