

TECHNICAL REPORT

No. TR TRHP - 02

Composition of
Semantic Web Services

-

Σύνθεση υπηρεσιών ιστού
στο σημασιολογικό διαδίκτυο

Παρασκευή Τσούτσα

tsoutsa@teilar.gr

Πάτρα, 2011

Πανεπιστήμιο Πατρών

Τμήμα Μαθηματικών

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΡΙΕΧΟΜΕΝΑ	2
ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ.....	3
ΑΛΦΑΒΗΤΙΚΟΣ ΠΙΝΑΚΑΣ ΣΥΝΤΟΜΟΓΡΑΦΙΩΝ ΚΑΙ ΕΚΦΡΑΣΕΩΝ	3
ΕΙΣΑΓΩΓΗ	5
1 ΣΗΜΑΣΙΟΛΟΓΙΚΟ ΔΙΑΔΙΚΤΥΟ ΚΑΙ ΥΠΗΡΕΣΙΕΣ ΙΣΤΟΥ	6
1.1 ΣΗΜΑΣΙΟΛΟΓΙΚΟ ΔΙΑΔΙΚΤΥΟ.....	6
1.2 ΑΠΟ ΤΙΣ ΥΠΗΡΕΣΙΕΣ ΙΣΤΟΥ ΣΤΙΣ ΣΗΜΑΣΙΟΛΟΓΙΚΕΣ ΥΠΗΡΕΣΙΕΣ ΙΣΤΟΥ	6
1.3 ΟΝΤΟΛΟΓΙΕΣ ΚΑΙ ΣΗΜΑΣΙΟΛΟΓΙΚΟΣ ΙΣΤΟΣ.....	7
1.4 ΓΛΩΣΣΕΣ ΑΝΑΠΑΡΑΣΤΑΣΗΣ ΟΝΤΟΛΟΓΙΩΝ	8
1.5 Η WEB ONTOLOGY LANGUAGE (OWL).....	8
1.6 OWL-S – SEMANTIC SPECIFICATION OF SERVICES	10
ΣΧΟΛΙΑΣΜΟΣ	12
1.7 ΑΛΛΕΣ ΠΡΟΤΑΣΕΙΣ ΓΙΑ ΠΕΡΙΓΡΑΦΗ ΣΗΜΑΣΙΟΛΟΓΙΚΩΝ ΥΠΗΡΕΣΙΩΝ ΙΣΤΟΥ.....	13
2. ΣΥΝΘΕΣΗ ΥΠΗΡΕΣΙΩΝ ΣΗΜΑΣΙΟΛΟΓΙΚΟΥ ΙΣΤΟΥ – STATE OF THE ART.....	14
2.1 ΣΥΝΘΕΣΗ ΜΕ PLANNING ΜΕΘΟΔΟΥΣ.....	14
2.2 ΣΥΝΘΕΣΗ ΜΕ ΤΗ ΧΡΗΣΗ ΤΗΣ SITUATION CALCULUS	15
2.3 ΠΕΡΙΓΡΑΦΗ ΠΡΟΒΛΗΜΑΤΟΣ ΣΕ PLANNING DOMAIN DEFINITION LANGUAGE (PDDL) ΚΑΙ ΣΥΝΘΕΣΗ ΜΕΣΩ ΚΛΑΣΙΚΟΥ ΣΧΕΔΙΑΣΜΟΥ ΕΝΕΡΓΕΙΩΝ.....	16
2.4 ΣΥΝΘΕΣΗ ΜΕ ΤΗ ΧΡΗΣΗ ΤΗΣ ΓΡΑΜΜΙΚΗΣ ΛΟΓΙΚΗΣ (LINEAR LOGIC).....	17
2.5 HTN - PLANNING ΜΕ ΤΗ ΧΡΗΣΗ ΤΟΥ SHOP2	17
2.6 ΣΥΝΘΕΣΗ ΜΕ ΤΗΝ OWL-S	19
2.5 ΣΧΟΛΙΑΣΜΟΣ	20
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	21

ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ

Εικόνα 1: Πως δημιουργήθηκε η OWL.....	8
ΕΙΚΟΝΑ 2 : ΘΕΣΗ ΤΗΣ OWL ΣΤΗ ΣΤΟΙΒΑ ΤΕΧΝΟΛΟΓΙΩΝ.....	10
ΕΙΚΟΝΑ 3: ΑΝΩΤΕΡΗ ΟΝΤΟΛΟΓΙΑ ΤΗΣ ΥΠΗΡΕΣΙΑΣ ΙΣΤΟΥ.....	12

ΑΛΦΑΒΗΤΙΚΟΣ ΠΙΝΑΚΑΣ ΣΥΝΤΟΜΟΓΡΑΦΙΩΝ ΚΑΙ ΕΚΦΡΑΣΕΩΝ

Web Services (Υπηρεσίες ιστού): Είναι μια νέα γενιά εφαρμογών Web. Είναι αυτό-περιγραφόμενες, ανεξάρτητες, αρθρωτές εφαρμογές που μπορούν να δημοσιευτούν, να εντοπιστούν και να κληθούν από το web.

UDDI: Το Universal Description Discovery and Integration, ένας χώρος αποθήκευσης περιγραφών υπηρεσιών διαδικτύου για ανεύρεση αυτών από μια επιχείρηση.

SOAP: Το Simple Object Access Protocol, ένα πρωτόκολλο επικοινωνίας για την πρόσβαση σε μία υπηρεσία.

WSDL: Η Web Services Description Language, η γλώσσα που χρησιμοποιείται για την περιγραφή των υπηρεσιών και βασίζεται στην XML.

BPEL: Η Business Process Execution Language for Web Services είναι μια γλώσσα σε XML που χρησιμοποιείται για την περιγραφή μιας συντιθέμενης υπηρεσίας, και έχει αναπτυχθεί από τις BEA, IBM, MICROSOFT, SAP, SIEBEL και OASIS.

WS-CDL: Η Web Services Choreography Description Language είναι μια γλώσσα περιγραφής σε XML που περιγράφει peer-to-peer συνεργασίες των συμμετεχόντων υπηρεσιών ιστού από μια γενική άποψη. Περιγράφει τη συμπεριφορά τους και τα μηνύματα που ανταλλάσσουν με σκοπό τη δημιουργία ενός κοινού επιχειρηματικού στόχου.

Orchestration (Ενορχήστρωση): Μοντελοποίηση της σύνθεσης των υπηρεσιών ιστού που βασίζεται σε ένα κεντρικό διοργανωτή ο οποίος ενορχηστρώνει την ανταλλαγή μηνυμάτων μεταξύ των υπηρεσιών ιστού που συμμετέχουν στη σύνθεση.

Choreography (Χορογραφία): Μοντελοποίηση της σύνθεσης των υπηρεσιών ιστού που βασίζεται στον κάθε συμμετέχοντα. Η κάθε υπηρεσία έχει συνολική άποψη των άλλων συμμετεχόντων υπηρεσιών ιστού στη σύνθεση.

Service providers: Παρέχουν υπηρεσίες, και διατηρούν έναν κατάλογο/μητρώο (directory/registry) μέσω του οποίου γίνονται διαθέσιμες αυτές οι υπηρεσίες.

Service requestors: Σε συνεργασία με τους service brokers ανακαλύπτουν τις διαθέσιμες υπηρεσίες, και στη συνέχεια με κλήση αυτών εκτελούν κάποια συγκεκριμένη εργασία.

Service brokers: Οι service brokers λειτουργούν ως ένα είδος μεσάζοντα μεταξύ service providers και service requestors.

TCP/IP: Το Transmission Control Program/Internet Protocol είναι το παγκόσμιο πρωτόκολλο δικτύωσης μέσω του οποίου συσκευές όπως προσωπικοί και κεντρικοί υπολογιστές, pagers, κινητά τηλέφωνα κ.α. μπορούν να επικοινωνήσουν μεταξύ τους.

HTML: Η Hypertext Markup Language είναι η επικρατούσα γλώσσα για να γράφουμε σελίδες στο Web. Η HTML δεν είναι μια γλώσσα προγραμματισμού αλλά μια γλώσσα σήμανσης και παρέχει τρόπους για να δημιουργούμε δομημένα κείμενα με επικεφαλίδες παραγράφους και λίστες που περιλαμβάνουν εικόνες, links σε άλλες σελίδες και scripts από άλλες γλώσσες. Ένα τέτοιο κείμενο αποτελείται από ένα σύνολο στοιχείων που εισάγονται με ετικέτες (tags).

XML: Η Extensible Markup Language είναι μια ευέλικτη γλώσσα για τη περιγραφή και τη μεταφορά δομημένων δεδομένων διαμέσου του Web.

ΕΙΣΑΓΩΓΗ

Στις μέρες μας χιλιάδες επιχειρήσεις κάνουν τη δουλειά τους χρησιμοποιώντας το Internet. Παρέχουν πληροφορίες για αεροπορικά εισιτήρια, τιμές μετοχών, αγορές, πωλήσεις, συναλλαγές μέσω πιστωτικών καρτών, διανομές κ.α. Οι εφαρμογές που χρησιμοποιούν, πολύ συχνά χρειάζεται να επικοινωνήσουν με άλλες εφαρμογές μέσω του Internet. Όσο ο αριθμός αυτών των εφαρμογών αυξάνεται τόσο μεγαλώνει και η ανάγκη για δυνατότητα επαναχρησιμοποίησης και σύνθεσης αυτών των εφαρμογών.

Η αρχική ιδέα του Internet δεν ήταν η προοπτική να μοιραστούμε μόνο δεδομένα αλλά και διαθέσιμες πηγές και υπηρεσίες, πράγμα που σημαίνει ότι πρέπει να μπορούμε να συνθέτουμε συναρτήσεις αλλά και ολόκληρες εφαρμογές που θα μπορούν να εκτελούνται σε απομακρυσμένα μηχανήματα.

Στην προηγούμενη εργασία προσεγγίσαμε τις έννοιες και την τεχνολογία γύρω από τις υπηρεσίες ιστού. Σε αυτή την εργασία θα δούμε τη διαφορά με τις σημασιολογικές υπηρεσίες ιστού και τρόπους περιγραφής αυτών. Θα ακολουθήσει η παρουσίαση του προβλήματος της σύνθεσης στις σημασιολογικές υπηρεσίες ιστού, όπως αυτό υπάρχει σήμερα. Στη συνέχεια θα παρουσιασθούν οι μέθοδοι που έχουν προταθεί για να λύσουν το πρόβλημα της σύνθεσης υπηρεσιών.

1 ΣΗΜΑΣΙΟΛΟΓΙΚΟ ΔΙΑΔΙΚΤΥΟ ΚΑΙ ΥΠΗΡΕΣΙΕΣ ΙΣΤΟΥ

1.1 ΣΗΜΑΣΙΟΛΟΓΙΚΟ ΔΙΑΔΙΚΤΥΟ

Εάν το HTML και το Web κάνουν όλα τα online έγγραφα να μοιάζουν σαν ένα τεράστιο βιβλίο, το σημασιολογικό διαδίκτυο θα κάνει όλα τα δεδομένα του κόσμου να μοιάζουν σαν μία τεράστια βάση δεδομένων.[Tim Berners Lee]

Τι είναι όμως το Σημασιολογικό Διαδίκτυο; Πρόκειται για μια επέκταση του σημερινού διαδικτύου με τη διαφορά ότι η πληροφορία αποθηκεύεται σε μορφή που μπορεί να διαβαστεί από τις μηχανές, επιτρέποντας στους υπολογιστές να διαχειριστούν τα δεδομένα με πιο χρήσιμους τρόπους, επεξεργαζόμενοι τις έννοιες μέσα στα κείμενα. Ο υπολογιστής θα μπορεί να αντιληφθεί την πληροφορία που περιέχει ένα ηλεκτρονικό κείμενο, αλλά και να ξεχωρίσει τη σημασία που αποδίδει ο χρήστης, ταξινομώντας τις πληροφορίες με τρόπους που δεν έχουμε γνωρίσει ως σήμερα. Με άλλα λόγια στόχος του Σημασιολογικού Διαδικτύου είναι το διαδίκτυο να γίνει μια παγκόσμια πλατφόρμα ανταλλαγής και επεξεργασίας δεδομένων από ετερογενείς πηγές πληροφορίας. Το Σημασιολογικό Διαδίκτυο θα προσφέρει «λογική δόμηση», οργάνωση και σημασιολογία στα δεδομένα ώστε να είναι κατανοητά σε επίπεδο μηχανής.

Το να γίνουν τα δεδομένα αναγνώσιμα από τις μηχανές είναι μόνο ένα μέρος από τη δουλειά που πρέπει να γίνει για να εκμεταλλευτούμε το σημασιολογικό διαδίκτυο. Ακόμη και εάν είναι τα δεδομένα αναγνώσιμα από τις μηχανές, πως θα διασφαλίσουμε ότι παρουσιάζονται με τον σωστό τρόπο; Για παράδειγμα, μια υπηρεσία μπορεί να ζητά ως input την θερμοκρασία σε βαθμούς Celsius ενώ η υπηρεσία που καταγράφει τη θερμοκρασία να την υπολογίζει σε Fahrenheit. Οι οντολογίες, με τις οποίες θα ασχοληθούμε συνοπτικά στην επόμενη παράγραφο, τυπικά ορίζουν τη δομή και την έννοια των δεδομένων που είναι αναγνώσιμα από τις μηχανές και είναι αυτές που θα καλύψουν το χάσμα που υπάρχει.

1.2 ΑΠΟ ΤΙΣ ΥΠΗΡΕΣΙΕΣ ΙΣΤΟΥ ΣΤΙΣ ΣΗΜΑΣΙΟΛΟΓΙΚΕΣ ΥΠΗΡΕΣΙΕΣ ΙΣΤΟΥ

Σε προηγούμενη έκθεση παρουσιάσαμε αρκετές από τις τελευταίες γενιάς τεχνολογίες ιστού καθώς και το όραμα της εξέλιξης του διαδικτύου σε σημασιολογικό διαδίκτυο. Ποιο είναι το επόμενο βήμα; Πως θα σταθούν οι υπηρεσίες ιστού στο Σημασιολογικό Διαδίκτυο εάν και οι ίδιες δεν υποστηρίζουν σημασιολογία; Οι σημασιολογικές υπηρεσίες ιστού θα πρέπει να υποστηρίζουν εργασίες όπως αναζήτηση, επιλογή και σύνθεση των κατάλληλων υπηρεσιών. Όλα αυτά όμως απαιτούν επίσης να είναι οι ίδιες οι υπηρεσίες επεξεργάσιμες από τους υπολογιστές. Ακριβώς όπως το σημασιολογικό διαδίκτυο υπόσχεται να κάνει το υπάρχον στατικό περιεχόμενο να είναι επεξεργάσιμο από τις μηχανές μέσω της σημασιολογίας, έτσι θα ήταν ιδανικό να εφαρμοσθούν οι ίδιες τεχνικές και στις υπηρεσίες ιστού.

Το όραμα για τις σημασιολογικές υπηρεσίες ιστού είναι να περιγράψουν και να σχολιάσουν ορθά τις διάφορες πλευρές μιας υπηρεσίας ιστού χρησιμοποιώντας μια γλώσσα αναγνωρίσιμη από τη μηχανή, επιτρέποντας έτσι στις μηχανές να τις εντοπίσουν αυτόματα, να τις συνδυάσουν με κάποιες άλλες και να τις χρησιμοποιήσουν. Οι οντολογίες που θα δούμε στην επόμενη παράγραφο εμπλουτίζουν την περιγραφή μιας υπηρεσίας με έννοιες αλλά και σχέσεις μεταξύ των εννοιών αυτών.

1.3 ΟΝΤΟΛΟΓΙΕΣ ΚΑΙ ΣΗΜΑΣΙΟΛΟΓΙΚΟΣ ΙΣΤΟΣ

Ο όρος οντολογία πηγάζει από την φιλοσοφία και έχει προσαρμοσθεί στο πεδίο της Επιστήμης των Υπολογιστών με ελάχιστα διαφορετική έννοια σε σχέση με όσα περιέγραφε ο Αριστοτέλης. Ο Αριστοτέλης έλεγε «Οντολογία είναι η έρευνα σχετικά με το τι όντα υπάρχουν στον κόσμο και ποιες είναι οι μεταξύ τους σχέσεις».

Είναι λογικό και φανερό ότι η φιλοσοφική έννοια του όρου επηρέασε τους ερευνητές που βρέθηκαν αντιμέτωποι με το πρόβλημα της αναπαράστασης γνώσης, του χαρακτηρισμού και της ταξινόμησης του υπαρκτού κόσμου στις εφαρμογές.

Στην Επιστήμη των Υπολογιστών οι ορισμοί των οντολογιών διατηρούν την αφηρημένη γενικότητα του φιλοσοφικού ορισμού. Μια οντολογία πρέπει να καταγράφει κοινά αποδεκτή γνώση και να έχει καλά ορισμένη σημασιολογία και συντακτικό. Αναπαριστά εννοιολογική μορφοποίηση σε μία μορφή «κατανοητή» και επεξεργάσιμη από τους υπολογιστές (π.χ. RDFS/OWL). Ένας αρκετά διαδεδομένος ορισμός για την οντολογία είναι αυτός του Gruber. «Οντολογία είναι ο προσδιορισμός μιας εννοιολογικής σύλληψης» [52].

Ο κύριος λόγος για το πρόσφατα αυξανόμενο ενδιαφέρον για τις οντολογίες είναι διότι κατέχουν εξέχουσα θέση στην ανάπτυξη του Σημασιολογικού Διαδικτύου

Με τη χρήση των οντολογιών για την καταχώριση δεδομένων στο διαδίκτυο επιλύονται προβλήματα που υπάρχουν λόγω του κενού μεταξύ της καταχωρημένης πληροφορίας και της σημασίας της. Π.χ. ο όρος «book» σημαίνει “βιβλίο” αλλά και “κράτηση”; Η ενοποίηση ετερογενούς πληροφορίας π.χ. «book» και “reservation” θα μπορούσε να χρησιμοποιηθεί για την έννοια της «κράτησης» και να τη διαχωρίσει ρητά (disjointness) από την άλλη σημασία του «book» που περιγράφεται από την έννοια “βιβλίο”.

Μια οντολογία ορίζει τους όρους που χρησιμοποιούνται για την περιγραφή και την αναπαράσταση ενός πεδίου γνώσης. Παρέχει μεταπληροφορίες που περιγράφουν τη σήμανση των δεδομένων [12]. Οι οντολογίες αποτελούνται από ορισμούς βασικών εννοιών και σχέσεων μεταξύ τους, όπως Student και PhdStudent. Σχέσεις σαν και αυτές είναι πολύ εύκολο να τις καταλάβει ένας άνθρωπος. Εάν το νόημα των σχέσεων ορισθεί με τυπικό τρόπο θα μπορούσε και μια μηχανή να τις αναγνωρίσει, να τις καταλάβει, να τις επεξεργαστεί και βάση αυτών των σχέσεων να βγάλει τα ίδια συμπεράσματα που θα έβγαζε και ένας άνθρωπος. Δεν αποκτά

φυσικά η μηχανή νόηση, αυτό που γίνεται είναι πως η γνώση του ανθρώπου κωδικοποιείται με τέτοιο τρόπο που η μηχανή μπορεί να τον επεξεργαστεί και να «συμπεράνει» μέσα από κάποιες λογικές διαδικασίες.

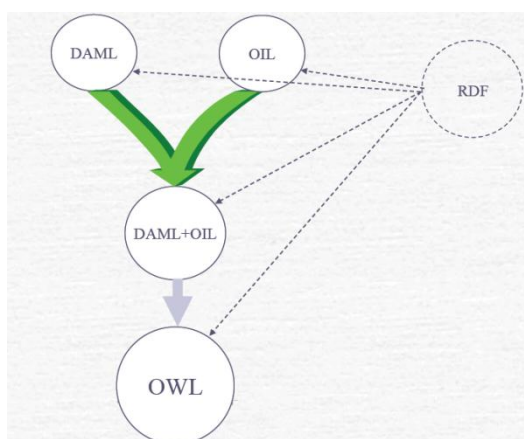
Η διαδικασία ανάπτυξης μιας οντολογίας είναι παρόμοια με το χτίσιμο ενός προγράμματος σε μια αντικειμενοστραφή γλώσσα προγραμματισμού. Στις αναφορές [9], [10] μπορούμε να βρούμε διάφορες προσεγγίσεις για τον τρόπο που θα χτίσουμε μια οντολογία.

1.4 ΓΛΩΣΣΕΣ ΑΝΑΠΑΡΑΣΤΑΣΗΣ ΟΝΤΟΛΟΓΙΩΝ

Η επιλογή της γλώσσας για την ανάπτυξη μιας οντολογίας είναι σημαντική. Η εκφραστικότητα της γλώσσας παίζει σημαντικό ρόλο για το είδος των σχέσεων και των εννοιών που θα αναπαρασταθούν. Κάποιες τις γλώσσες που χρησιμοποιούνται για αυτόν τον σκοπό είναι οι KIF, OKBC, SHOE, XOL, OML, RDF, OIL, DAML+OIL, WSMO και OWL. Στην επόμενη παράγραφο θα δώσουμε μια σύντομη περιγραφή για τη γλώσσα OWL που αποτελεί την πιο πρόσφατη εξέλιξη στις γλώσσες περιγραφής οντολογιών.

1.5 Η WEB ONTOLOGY LANGUAGE (OWL)

Η OWL είναι η πιο πρόσφατη εξέλιξη στις γλώσσες ανάπτυξης οντολογιών για το Σημαιολογικό Ιστό και προτείνεται από την W3C ως η κατάλληλη γλώσσα για την αναπαράσταση των οντολογιών. Στηρίχθηκε στην DAML+OIL για να εξελιχθεί και ακολουθεί το συντακτικό της RDF(S), εμπλουτίζοντας όμως το λεξιλόγιο της για την περιγραφή των κλάσεων και των ιδιοτήτων. Έχει σκοπό να μεταφέρει την εκφραστικότητα και την συλλογιστική δύναμη της περιγραφικής λογικής στον Σημαιολογικό Ιστό.



ΕΙΚΟΝΑ 1: ΠΩΣ ΔΗΜΙΟΥΡΓΗΘΗΚΕ Η OWL

Η OWL λόγω της αυξανόμενης πολυπλοκότητας κατά τη διεξαγωγή συμπερασμάτων χωρίζεται σε τρεις διαφορετικές και αυξανόμενα περιγραφικές υπογλώσσες που η κάθε μια στοχεύει να ικανοποιήσει διαφορετικές πλευρές των απαιτήσεων.

OWL lite: Η γλώσσα αυτή απευθύνεται σε χρήστες οι οποίοι επιθυμούν να χρησιμοποιήσουν την OWL για την περιγραφή γνώσης σε εφαρμογές που δεν έχουν μεγάλες απαιτήσεις σε εκφραστικές δυνατότητες. Αντιστοιχεί στην περιγραφική λογική SHIF. Μπορεί να εκφράσει ιεραρχίες και απλούς περιορισμούς.

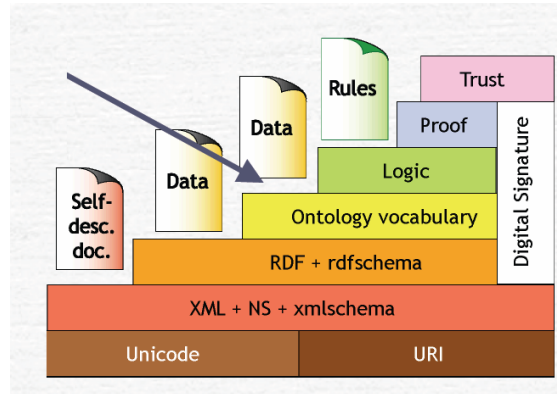
OWL description logic: Απευθύνεται σε εκείνους τους χρήστες που θέλουν τη μέγιστη δυνατή εκφραστικότητα χωρίς απώλεια της υπολογιστικής πληρότητας (computational completeness) και της αποφασισιμότητας (decidability) και την διαθεσιμότητα πρακτικών αλγορίθμων συλλογισμού. Διατηρεί την υπολογιστική πολυπλοκότητα της συλλογιστικής σε χαμηλά επίπεδα αλλά υστερεί στην έλλειψη συμβατότητας με την RDF. Αντιστοιχεί στην περιγραφική λογική SHOIN

OWL full: Βασίζεται σε διαφορετική σημασιολογία από αυτή των προηγούμενων. Δεν έχει εκφραστικούς περιορισμούς αλλά δεν εγγυάται και κάποιες υπολογιστικές ιδιότητες. Είναι απολύτως συμβατή με την RDF τόσο συντακτικά όσο και σημασιολογικά, οπότε οποιαδήποτε έγγραφο της RDF είναι και έγκυρο έγγραφο της OWL full και το αντίστροφο. Προορίζεται για χρήστες που επιθυμούν τη μέγιστη εκφραστικότητα και τη συντακτική ελευθερία της RDF χωρίς να ενδιαφέρονται για το υπολογιστικό κόστος.

Τα OWL έγγραφα ονομάζονται OWL οντολογίες. Μια οντολογία OWL είναι χτισμένη σε RDF και RDF Schema και χρησιμοποιεί XML based syntax. Στο ορισμό μιας οντολογίας μπορούμε με την εντολή *import* να εισάγουμε μια άλλη οντολογία που θα θεωρηθεί ότι είναι μέρος της αρχικής και θα παρέχει και νέους ορισμούς που θα μπορούν να χρησιμοποιηθούν. Να σημειωθεί ότι η OWL *:import* έχει την μεταβατική ιδιότητα.

Οι κλάσεις συνήθως ορίζονται με την χρήση της εντολής *owl:Class*. Υπάρχουν δύο κλάσεις που είναι by default ορισμένες, οι *owl:thing* και η *owl:nothing*. Κάθε κλάση είναι υποκλάση της *thing* και υπερκλάση της *nothing*. Οι κλάσεις μπορούν να οριστούν είτε με τον standard τρόπο *owl:Class* είτε με την εντολή *Owl:restriction* με την οποία ορίζονται νέες ανώνυμες κλάσεις που ικανοποιούν συγκεκριμένους περιορισμούς χωρίς id, ή είναι συνδυασμός άλλων κλάσεων.

Στην OWL υπάρχουν δύο ειδών ιδιότητες. Οι object properties που συνδέουν αντικείμενα μεταξύ τους, όπως *supervises*, *isTaughtBy*, και οι data type properties που συνδέουν αντικείμενα με data type values. Π.χ. *phone*, *title*, *age*. Η OWL δεν έχει προκαθορισμένους τύπους δεδομένων και για τον ορισμό τους επιτρέπεται να χρησιμοποιηθεί η XML Schema.



ΕΙΚΟΝΑ 2 : ΘΕΣΗ ΤΗΣ OWL ΣΤΗ ΣΤΟΙΒΑ ΤΕΧΝΟΛΟΓΙΩΝ

Πιο κάτω παραθέτουμε κάποια εργαλεία διαχείρισης οντολογιών που έχουν αναπτυχθεί μέχρι σήμερα.

- OilEd, University of Manchester <http://oiled.man.ac.uk/>
- KAON , AIFB and FZI, University of Karlsruhe <http://kaon.semanticweb.org/>
- OntoSaurus, ISI (USA) <http://www.isi.edu/isd/ontosaurus.html>
- OntoEdit , Univ. of Karlsruhe <http://ontoserver.aifb.unikarlsruhe.de/ontoedit/>
- WebOnto, KMI (Open University) <http://kmi.open.ac.uk/projects/webonto/>
- WebODE, UPM <http://webode.dia.fi.upm.es/webODE/>
- Ontolingua , KSL (Stanford University) <http://wwwksl.stanford.edu>
- Protégé 2000, SMI (Stanford University) <http://protege.stanford.edu/>

1.6 OWL-S – SEMANTIC SPECIFICATION OF SERVICES

Η OWL-S είναι μια γλώσσα περιγραφής οντολογιών που επεκτείνει την OWL και σκοπό έχει να περιγράψει τις σημασιολογικές υπηρεσίες. Η OWL-S χρησιμοποιείται για να ενισχύσει την εκφραστικότητα της γλώσσας WSDL με την προσθήκη σημασιολογικών πληροφοριών στις περιγραφές των υπηρεσιών και είναι το de facto standard για την περιγραφή σημασιολογικών υπηρεσιών ιστού.

Για να χρησιμοποιήσει μια υπηρεσία ιστού, ένας software agent χρειάζεται μια περιγραφή της υπηρεσίας αναγνωρίσιμη από τη μηχανή και τον τρόπο που θα έχει πρόσβαση σε αυτή με σκοπό την αυτόματη υλοποίηση των παρακάτω ενεργειών.

Discovery Εντοπισμός της υπηρεσίας ιστού

Invocation Εκτέλεση της υπηρεσίας από έναν πράκτορα ή μια άλλη υπηρεσία

Interoperation Αυτόματη εισαγωγή μηνυμάτων για μετατροπή παραμέτρων

Composition Δημιουργία νέας υπηρεσίας μετά από αυτόματη επιλογή και σύνθεση υπαρχουσών υπηρεσιών

Αξιολόγηση	Αξιολόγηση των χαρακτηριστικών μιας υπηρεσίας
Monitoring	Παρακολούθηση των χαρακτηριστικών και ιδιοτήτων της υπηρεσίας ενώ βρίσκεται σε λειτουργία.

Η γλώσσα OWL-S χρησιμοποιεί μια Ανώτερη Οντολογία (Upper Ontology) για τη σημασιολογική περιγραφή υπηρεσιών ιστού. Ο σκοπός της οντολογίας είναι να προσφέρει ένα τρόπο περιγραφής μιας υπηρεσίας ώστε να απαντώνται τα ερωτήματα: (α) Τι παρέχει η συγκεκριμένη υπηρεσία και τι απαιτεί από έναν πράκτορα, (β) Πως λειτουργεί και (γ) Πως μπορούμε να έχουμε πρόσβαση στην υπηρεσία;

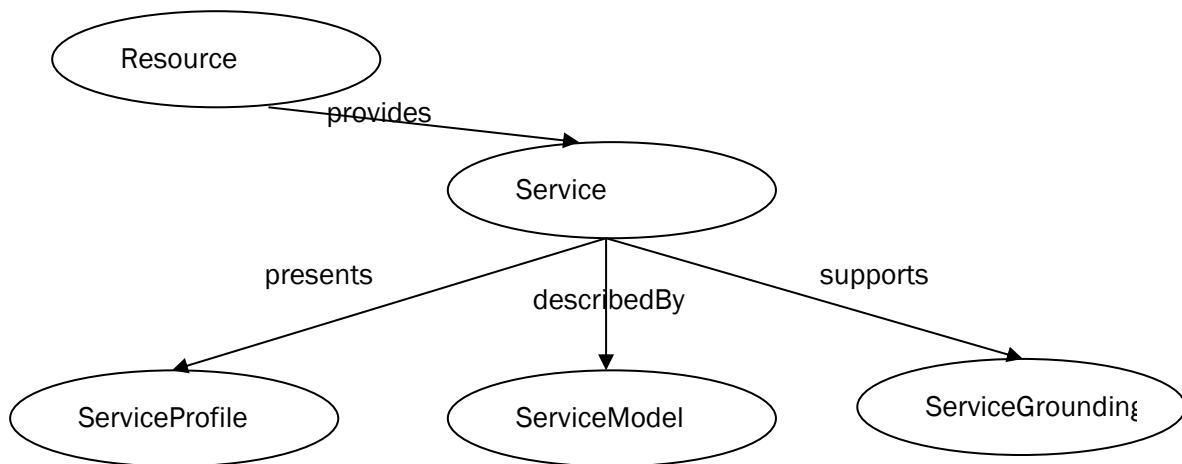
Όπως φαίνεται και στην Εικόνα 17, η οντολογία προβλέπει αρχικά την κλάση Service η οποία αναπαριστά μια υπηρεσία ιστού. Για κάθε στιγμιότυπο της κλάσης Service υπάρχει και μια υλοποίηση της αντίστοιχης υπηρεσίας. Η κλάση Service σχετίζεται με τρεις άλλες:

ServiceProfile «Τι παρέχει η συγκεκριμένη υπηρεσία και τι απαιτεί από έναν πράκτορα;». Το ServiceProfile χρησιμοποιείται για να «διαφημίσει» την υπηρεσία παρέχοντας πληροφορίες που μπορούν να χρησιμοποιηθούν από έναν πράκτορα για να συμπεράνει εάν αυτή πληροί τις προϋποθέσεις και εξασφαλίζει την ποιότητα και ασφάλεια που απαιτείται. Κάθε εμφάνιση της κλάσης Service παρουσιάζει ένα ServiceProfile.

ServiceModel “Πως λειτουργεί;”. Το ServiceModel επιτρέπει σε έναν πράκτορα να εκτελέσει μια σε βάθος ανάλυση εάν η υπηρεσία πληροί τις πιο πάνω προϋποθέσεις, να συνδυάσει περιγραφές από πολλές υπηρεσίες για να εκτελέσει έναν συγκεκριμένο στόχο, να συγχρονίσει τις διεργασίες σε διαφορετικούς πράκτορες, και παρακολουθεί την εκτέλεση της υπηρεσίας.

ServiceGrounding “Πως έχω πρόσβαση στην υπηρεσία;”. Το ServiceGrounding διευκρινίζει τις λεπτομέρειες στην διαδικασία της προσπέλασης της υπηρεσίας όπως ανταλλαγή μηνυμάτων, μεταφορά και διευθυνσιοδότηση. Το grounding (σύνδεση) είναι η αντιστοίχιση από τη αφαιρετική (abstract) προδιαγραφή στη συγκεκριμένη προδιαγραφή, εκείνων των στοιχείων της υπηρεσίας που απαιτούνται για την επικοινωνία με την υπηρεσία.

Σε γενικές γραμμές το ServiceProfile παρέχει τις απαραίτητες πληροφορίες σε έναν πράκτορα για να ανακαλύψει μια υπηρεσία ενώ το ServiceModel και το ServiceGrounding παρέχουν πληροφορίες σε έναν πράκτορα για να χρησιμοποιήσει μια υπηρεσία.



ΕΙΚΟΝΑ 3: ΑΝΩΤΕΡΗ ΟΝΤΟΛΟΓΙΑ ΤΗΣ ΥΠΗΡΕΣΙΑΣ ΙΣΤΟΥ

ΣΧΟΛΙΑΣΜΟΣ

Η OWL-S χαρακτηρίζεται ως μια από τις γλώσσες με την μεγαλύτερη εκφραστικότητα έναντι των άλλων υπαρχόντων και με καλή ορισμένη σημασιολογία σε περιγραφική λογική (Description Logic). Ποια είναι όμως τα μειονεκτήματα της OWL-S; Από τη στιγμή που η OWL-S βρίσκεται «πάνω» από την OWL, «κουβαλά» και τα αρνητικά της OWL [3].

- Το μεγαλύτερο μειονέκτημα είναι η έλλειψη μεταβλητών μέσα στην OWL [3]. Χωρίς τη δυνατότητα να χρησιμοποιηθούν συγκεκριμένες παράμετροι για να περιγράψουν τη ροή των δεδομένων με τον ορισμό μεταβλητών, τα inputs και outputs των ατομικών processes δεν μπορούν να συσχετιστούν μεταξύ τους.
- Στην OWL επιτρέπεται μόνο η εισαγωγή ολόκληρης της οντολογίας και όχι μέρος αυτής.
- Η σημασιολογία της OWL υιοθετεί σαν βάση της λογικής της την υπόθεση του «ανοικτού κόσμου» (open world assumption) που λέει ότι «αν μια πρόταση δεν είναι γνωστό ή δεν μπορεί να αποδειχθεί με βάση την τρέχουσα γνώση ότι είναι αληθής, δεν μπορούμε να συμπεράνουμε ότι είναι ψευδής». Παρόλα αυτά η αντίθετη προσέγγιση του «κλειστού κόσμου» («μια πρόταση είναι ψευδής αν δεν είναι γνωστό ή δεν μπορεί να αποδειχθεί επί του παρόντος ότι είναι αληθής») είναι επίσης χρήσιμη πολλές φορές.
- Υπόθεση μοναδικών ονομάτων. Εάν δύο διαφορετικές κλάσεις ή ιδιότητες έχουν διαφορετικό όνομα δεν είναι απαραίτητο να είναι διαφορετικές. Μπορεί να δηλωθεί πως είναι ίδιες. Σε αντιστοιχία όμως με το προηγούμενο, θα ήταν επίσης χρήσιμο να μπορούσε κανείς να δηλώσει σε ποιο μέρος της οντολογίας θέλουμε να ισχύει αυτό ή όχι.

- Δεν είναι ξεκάθαρο πώς διαχωρίζονται τα αντικείμενα από τα events που βρίσκονται και συμβαίνουν μέσα στο σύστημα από τα εξωτερικά αντικείμενα και events. Επιπλέον μια υπηρεσία μπορεί να συσχετισθεί με ένα μόνο service model, που σημαίνει πως δεν μπορεί να υπάρξει κάποιο εναλλακτικό service model για αυτή την υπηρεσία.
- Μια άλλη περιοχή που δεν καλύπτεται από την OWL-S είναι η συμπεριφορά της υπηρεσίας κατά την εκτέλεση της. Μπορεί βέβαια να περιγραφεί η αρχική και η τελική κατάσταση της υπηρεσίας, αλλά αυτό δεν αρκεί για να περιγραφεί και αξιολογηθεί η συμπεριφορά της κατά την εκτέλεση της σύνθεσης.

1.7 ΑΛΛΕΣ ΠΡΟΤΑΣΕΙΣ ΓΙΑ ΠΕΡΙΓΡΑΦΗ ΣΗΜΑΣΙΟΛΟΓΙΚΩΝ ΥΠΗΡΕΣΙΩΝ ΙΣΤΟΥ

Εκτός από την OWL-S και την WSMO έχουν προταθεί και άλλα πλαίσια για τη σημασιολογική περιγραφή υπηρεσιών ιστού.

Μια από αυτές της προτάσεις αποτελεί η WSDL-S (Web Services Semantics), ήταν αποτέλεσμα του ερευνητικού προγράμματος METEOR-S (University of Georgia και IBM) και έχει προταθεί στον οργανισμό W3C για προτυποποίηση. Είναι μια επέκταση της standard WSDL που περιλαμβάνει στοιχεία σημασιολογίας που θα μπορούσαν να βελτιώσουν την επαναχρησιμοποίηση των Web Services διευκολύνοντας την σύνθεσή τους, βελτιώνοντας τη δυνατότητα ανακάλυψής τους και επιτρέποντας την ολοκλήρωση υπάρχοντος software σε ένα Web Service.

Μία άλλη πρόταση είναι αυτή για το πλαίσιο SWSF (Semantic Web Services Framework) όπως έχει υποβληθεί στον οργανισμό W3C για προτυποποίηση). Το πλαίσιο αυτό περιλαμβάνει τη γλώσσα σημασιολογικής περιγραφής υπηρεσιών ιστού SWSL (Semantic Web Services Language) και την οντολογία SWSO (Semantic Web Services Ontology).

Η γλώσσα SWSL αποτελείται από δύο υπο-γλώσσες, την SWSL-FOL (First-Order Logic) και την SWSL-Rules. Η πρώτη υποστηρίζει πλήρως τη Λογική Πρώτης Τάξης και κυρίως χρησιμοποιείται για την περιγραφή οντολογιών. Παρέχει διαλειτουργικότητα με άλλες οντολογίες υπηρεσιών και διαδικασίες που είναι επίσης βασισμένες στη Λογική Πρώτης Τάξης. Η δεύτερη βασίζεται στο παράδειγμα του Λογικού Προγραμματισμού και χρησιμοποιείται για να υποστηρίξει την χρήση των οντολογιών στον συμπερασμό και σε περιβάλλοντα εκτέλεσης που βασίζονται σ' αυτό το παράδειγμα. Η SWSL είναι μία γενικού σκοπού γλώσσα η οποία όμως έχει σχεδιαστεί λαμβάνοντας υπόψη τις ανάγκες για την περιγραφή σημασιολογικών υπηρεσιών ιστού χρησιμοποιώντας URI (Uniform Resource Identifiers, δηλαδή strings που χρησιμοποιούνται για να ταυτοποιούμε ονόματα ή πηγές στο Internet), namespaces και τύπους δεδομένων σε XML. Η οντολογία SWSO προσφέρει το μοντέλο για την περιγραφή υπηρεσιών ιστού και περιλαμβάνει τον ορισμό αξιωμάτων για τις υπηρεσίες αυτές σε Λογική Πρώτης Τάξης [14].

Είναι βέβαιο πως στο άμεσο μέλλον θα δούμε και άλλες προτάσεις για την περιγραφή των σημασιολογικών υπηρεσιών ιστού που θα λύνουν τα προβλήματα των υπάρχουσών μεθόδων αλλά και θα δίνουν λύση στο μεγάλο κενό που υπάρχει αυτή τη στιγμή μεταξύ των θεωρητικών υλοποιήσεων και των βιομηχανικών υλοποιήσεων.

2. ΣΥΝΘΕΣΗ ΥΠΗΡΕΣΙΩΝ ΣΗΜΑΣΙΟΛΟΓΙΚΟΥ ΙΣΤΟΥ – STATE OF THE ART

Όσο οι υπηρεσίες ιστού αυξάνουν τόσο γίνεται πιο δύσκολο να βρούμε άμεσα την κατάλληλη υπηρεσία που θα εκτελέσει την συγκεκριμένη λειτουργία που ψάχνουμε. Και η αναζήτηση γίνεται ακόμη δυσκολότερη όταν δεν υπάρχει μια υπηρεσία να εκτελέσει αυτή την λειτουργία αλλά θα πρέπει να συνδυαστούν περισσότερες από μια υπηρεσίες για να επιτύχουμε το στόχο μας. Εάν η περιγραφή των υπηρεσιών ιστού ήταν ξεκάθαρη και κατανοητή στις μηχανές θα ήταν εφικτό να δημιουργηθεί μια καινούργια υπηρεσία (πρωτότυπη) από το συνδυασμό των υπάρχοντων υπηρεσιών χωρίς καμιά παρέμβαση από τον άνθρωπο. Οι σημασιολογικές περιγραφές διευκολύνουν στην αυτόματη αναγνώριση υπηρεσιών που έχουν συμβατές εισόδους και αποτελέσματα. Οι γλώσσες περιγραφής υπηρεσιών ιστού με σημασιολογία που αναλύσαμε στο προηγούμενο κεφάλαιο έβαλαν τα θεμέλια για να ξεκινήσει η έρευνα στον τομέα αυτό, στη αυτόματη σύνθεση των σημασιολογικών υπηρεσιών ιστού.

Αρκετές μέθοδοι έχουν αναπτυχθεί τα τελευταία χρόνια από ερευνητές και οργανισμούς με σκοπό να προσδιορίσουν ποιος είναι ο κατάλληλος τρόπος να μοντελοποιηθεί το πρόβλημα της αυτόματης σύνθεσης των υπηρεσιών ιστού. Σε αυτό το κεφάλαιο θα δούμε συνοπτικά κάποιες από αυτές τις προσεγγίσεις.

2.1 ΣΥΝΘΕΣΗ ΜΕ PLANNING ΜΕΘΟΔΟΥΣ

Κάποιες από τις μεθόδους που έχουν προταθεί για τη σύνθεση υπηρεσιών βασίζονται σε AI planning [27,45,46,28,38]. Ένα πρόβλημα σχεδιασμού ενεργειών (Planning Problem) είναι ένα πρόβλημα όπου ζητάμε την εύρεση μιας ακολουθίας ενεργειών, οι οποίες εφαρμοζόμενες σε μια αρχική κατάσταση, έχουν ως αποτέλεσμα την επίτευξη συγκεκριμένων στόχων π.χ. μεταφορά φορτίων, πλοήγηση οχημάτων κ.α. Η ακολουθία των ενεργειών που αποτελεί τη λύση ενός προβλήματος σχεδιασμού ονομάζεται πλάνο (plan) ενώ το πρόγραμμα που την παράγει ονομάζεται σχεδιαστής (planner).

Ένα πρόβλημα σχεδιασμού ενεργειών ορίζεται από τρεις περιγραφές : Της αρχικής κατάστασης (initial state), των στόχων (goals) και των διαθέσιμων ενεργειών (actions). Η σύνθεση των υπηρεσιών είναι η χρησιμοποίηση ενός αριθμού υπηρεσιών με την κατάλληλη σειρά για να

επιτύχουμε ένα συγκεκριμένο στόχο. Αυτό το πρόβλημα θα μπορούσε να είναι ένα πρόβλημα σχεδιασμού P που περιγράφεται από την τριάδα (s, g, a) όπου s είναι η αρχική κατάσταση πριν τη σύνθεση, g είναι ο στόχος και a το σύνολο των ενεργειών που πρέπει να γίνουν. Οι ενέργειες στο πρόβλημα μας είναι οι λειτουργίες των υπηρεσιών. Ο planner θα κληθεί να λύσει το πρόβλημα και να βρει κάποιο πλάνο. Το πρόβλημα θα λυθεί εάν η εκτέλεση των συγκεκριμένων ενεργειών του πλάνου μετατρέπουν την αρχική κατάσταση σε μια κατάσταση που περιγράφεται από κάποιον στόχο. Το πλάνο αποτελείται από το σύνολο των ενεργειών που θα εκτελεστούν και από τη σειρά εκτέλεσης αυτών.

2.2 ΣΥΝΘΕΣΗ ΜΕ ΤΗ ΧΡΗΣΗ ΤΗΣ SITUATION CALCULUS

Οι McIlraith and Son για πρώτη φορά παρουσιάζουν μια μέθοδο για σύνθεση υπηρεσιών στο [5] εφαρμόζοντας τεχνικές λογικής παραγωγής. Η γενική ιδέα αυτής της μεθόδου είναι ότι οι software agents μπορούν να εφαρμόσουν τεχνικές λογικής παραγωγής για να εκτελέσουν αυτόματη αναζήτηση, εκτέλεση και σύνθεση υπηρεσιών ιστού. Τα αιτούμενα του χρήστη αναπαριστούνται σε Situation Calculus. Το Situation Calculus είναι ένα υποσύστημα της Λογικής Πρώτης Τάξης (First Order Logic) και είναι μια λογική προσέγγιση για να μοντελοποιήσουμε δυναμικά συστήματα. Στο Situation Calculus η κατάσταση του κόσμου περιγράφεται από συναρτήσεις και σχέσεις που συσχετίζονται με μια κατάσταση. Η ιστορία του κόσμου είναι στην ουσία μια σειρά από ενέργειες που έγιναν σε μια αρχική κατάσταση. Μια κατάσταση είναι το ιστορικό των πρωτογενών ενεργειών που έλαβαν μέρος. Για παράδειγμα, η συνάρτηση $do(a,s)$ αντιστοιχεί την κατάσταση s μέσω μιας ενέργειας a σε μια νέα κατάσταση.

Στη μέθοδο αυτή κάθε υπηρεσία εκλαμβάνεται ως μια ενέργεια (action), είτε PrimitiveAction είτε ComplexAction. Τα ComplexActions αποτελούν συνθέσεις των PrimitiveActions. Τα PrimitiveActions όταν εκτελούνται ή θα αλλάξουν την κατάσταση του κόσμου ή θα αλλάξουν την γνώση του agent μέσω κάποιας πληροφορίας που του δόθηκε.

Η Golog είναι μια γλώσσα προγραμματισμού που βασίζεται στο Situation Calculus και χρησιμοποιείται για την περιγραφή και reasoning του προβλήματος της σύνθεσης. Η Golog υποστηρίζει την μοντελοποίηση συστημάτων που εξελίσσονται δυναμικά με βάση κάποια αξιώματα για τις προϋποθέσεις και τα αποτελέσματα των ενεργειών και την αρχική κατάσταση των συστημάτων. Αυτό επιτρέπει στα προγράμματα να βγάζουν συμπεράσματα για την κατάσταση των συστημάτων και να εξετάζουν τις επιδράσεις των πιθανών σειρών από ενέργειες πριν καταλήξουν σε μια συγκεκριμένη συμπεριφορά. Οι McIlraith and Son στην πρότασή τους επεκτείνουν την Golog και δίνουν τη δυνατότητα να χρησιμοποιηθούν user constraints. Επίσης εισάγουν την έννοια της «ακολουθίας» (sequence) επιτρέποντας την παρεμβολή κάποιων ενεργειών ώστε να επιτευχθούν οι προϋποθέσεις (preconditions) για την επόμενη ενέργεια που θα εκτελεσθεί από το πρόγραμμα. Οι λειτουργίες των υπηρεσιών έχουν καταχωρηθεί σε DAML-

S/RDF και στη συνέχεια χειροκίνητα μεταφράζονται σε Golog. Στη συνέχεια, δεδομένου ενός συγκεκριμένου στόχου, η γλώσσα Golog χρησιμοποιείται για να αρχικοποιήσει το κατάλληλο σχέδιο για την σύνθεση των υπηρεσιών ιστού.

Οι επεκτάσεις που φέρνει η μέθοδος αυτή είναι η δυνατότητα περιγραφής των προτιμήσεων και περιορισμών που επιβάλλει ο χρήστης του σχεδιασμού όπως και η περιγραφή απαιτήσεων για επιπλέον βήματα σχεδιασμού ενεργειών που μπορεί να είναι αναγκαία αν, για παράδειγμα, ένα επιλεγμένο σχέδιο συνδυασμού ενεργειών παρουσιάσει προβλήματα κατά την εκτέλεσή του.

Με βάση τις προϋποθέσεις και την επιθυμητή κατάσταση ένας σχεδιαστής ενεργειών μπορεί να επιλέξει τις κατάλληλες ενέργειες και τη σειρά με την οποία θα πρέπει να εκτελεσθούν προκειμένου να φθάσουμε στα επιθυμητά αποτελέσματα. Κάθε τέτοια ενέργεια μπορεί να είναι απλή ή σύνθετη. Οι απλές ενέργειες μπορούν να εκτελούνται από κάποια υπηρεσία ιστού και είτε να αποτελούν ενέργειες συλλογής περαιτέρω πληροφορίας είτε ενέργειες καταστάσεων που μπορεί να μας φέρουν πιο κοντά στο επιθυμητό αποτέλεσμα του σχεδιασμού.

2.3 ΠΕΡΙΓΡΑΦΗ ΠΡΟΒΛΗΜΑΤΟΣ ΣΕ PLANNING DOMAIN DEFINITION LANGUAGE (PDDL) ΚΑΙ ΣΥΝΘΕΣΗ ΜΕΣΩ ΚΛΑΣΙΚΟΥ ΣΧΕΔΙΑΣΜΟΥ ΕΝΕΡΓΕΙΩΝ

Η γενική ιδέα πίσω από την εφαρμογή αυτής της μεθόδου [15][16] στη σύνθεση υπηρεσιών ιστού είναι ότι η κάθε υπηρεσία μπορεί να μετασχηματισθεί και να αναπαρασταθεί με όρους σχεδιασμού ενεργειών, αντιστοιχίζοντας απευθείας τμήματα της OWL-S περιγραφής της υπηρεσίας σε στοιχεία ενός αντίστοιχου προβλήματος σχεδιασμού ενεργειών ή σε στοιχεία PDDL γενικότερα. Η γλώσσα PDDL σχεδιάστηκε για να αποτελέσει ένα πρότυπο μέσο κωδικοποίησης πεδίων ορισμού και προβλημάτων σχεδιασμού ενεργειών. Κάθε περιγραφή πεδίου ορισμού σε PDDL αποτελείται από δηλώσεις που περιλαμβάνουν τις απαιτήσεις του χρήστη, καθώς και τύπους οντοτήτων (types), μεταβλητές (variables), σταθερές (constants), συσχετίσεις (relationships), δηλώσεις τελεστών (operators, actions), αξιωμάτων (axioms) και άλλων περιορισμών (safety constraints). Κατά τον ορισμό του προβλήματος σε PDDL ισχύει η υπόθεση του κλειστού σύμπαντος, δηλαδή οτιδήποτε δεν αναφέρεται ρητά ως αληθές είναι ψευδές.

Η λύση της απευθείας αναπαράστασης περιγραφών υπηρεσιών ιστού από OWL-S σε PDDL[13] και η επίλυση του προβλήματος της σύνθεσης με συστήματα σχεδιασμού ενεργειών συμβατά με την PDDL, ανεξαρτήτως εσωτερικής τους υλοποίησης, είναι εφικτή λόγω των ομοιοτήτων μεταξύ των OWL-S και PDDL περιγραφών των υπηρεσιών ιστού.

2.4 ΣΥΝΘΕΣΗ ΜΕ ΤΗ ΧΡΗΣΗ ΤΗΣ ΓΡΑΜΜΙΚΗΣ ΛΟΓΙΚΗΣ (LINEAR LOGIC)

Ο Rao στην αναφορά [6] προτείνει μια μέθοδο για αυτοματοποιημένη σύνθεση υπηρεσιών ιστού (ΥΙ) με τη χρήση της γραμμικής λογικής (Linear Logic). Η μέθοδος χρησιμοποιεί την WSDL γλώσσα περιγραφής των ατομικών υπηρεσιών, ή την BPEL4WS για τις σύνθετες υπηρεσίες και μεταφράζει τις λειτουργίες των υπηρεσιών στη γλώσσα της γραμμικής λογικής. Δοσμένου ενός σετ υπάρχοντων ΥΙ και ενός σετ από ιδιότητες που χαρακτηρίζονται είτε ως «functionalities» είτε ως «non-functional» η μέθοδος βρίσκει εάν υπάρχει μια σύνθεση μεταξύ των υπάρχοντων υπηρεσιών που ικανοποιεί τις απαιτήσεις που έχουν τεθεί από τον χρήστη. Ως functionalities χαρακτηρίζονται τα inputs, outputs, precondition και effects ενώ κάποιες «non-functional» ιδιότητες είναι το κόστος, η ποιότητα, ο χρόνος απόκρισης κ.λπ. Η γενική ιδέα της μεθόδου για τη σύνθεση υπηρεσιών ιστού είναι η ακόλουθη. Η περιγραφή της ΥΙ μεταφράζεται σε extra logical axioms στην LL και τα απαιτούμενα χαρακτηριστικά της συντιθέμενης ΥΙ προσδιορίζονται σε ένα LL sequent. Με τη χρήση ενός LL theorem prover και με βάση τα αξιώματα του συστήματος που προαναφέρθηκαν ελέγχεται αυτό το sequent εάν μπορεί να αποδειχθεί. Το μοντέλο διεργασίας της σύνθετης διαδικτυακής υπηρεσίας δημιουργείται απευθείας από την απόδειξη και η συντιθέμενη υπηρεσία μπορεί να μοντελοποιηθεί ως ένα σετ υπηρεσιών που είναι συσχετισμένες και αλληλοεξαρτώμενες.

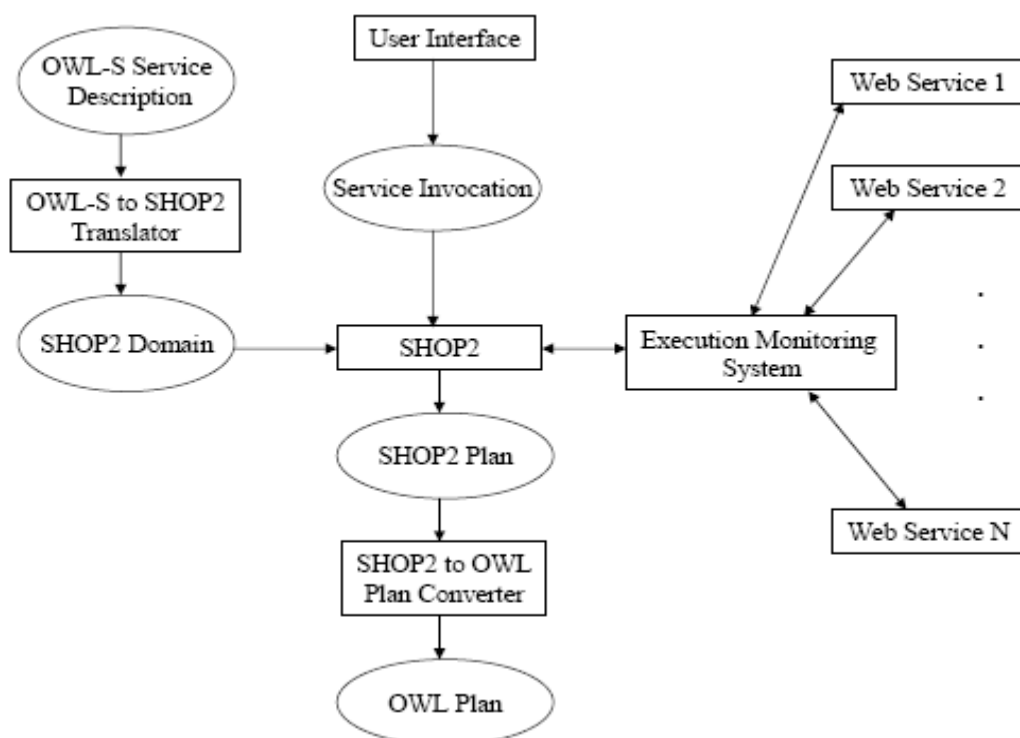
2.5 HTN - PLANNING ΜΕ ΤΗ ΧΡΗΣΗ ΤΟΥ SHOP2

Ενδιαφέρον επικεντρώνει και ο τρόπος σχεδιασμού ενεργειών που είναι γνωστός ως Hierarchical Task Network (HTN) [11]. Ο HTN σχεδιασμός δεν ξεκινάει με μια αρχική κατάσταση αλλά με ένα καθήκον (task) το οποίο πρέπει να εκτελεστεί και αποτελεί τον τελικό στόχο, και μεθόδους που μπορούν να χρησιμοποιηθούν για τον επιμερισμό του. Έτσι, ένα καθήκον επιμερίζεται σε υπο-καθήκοντα (sub-tasks) τα οποία με τη σειρά τους μπορούν να διαιρούνται σε περαιτέρω υπο-καθήκοντα ώσπου να φθάσουμε σε ένα επίπεδο καθηκόντων που θα δώσουν μία ταξινομημένη λίστα ενεργειών που πρέπει να σχεδιαστούν.

Λαμβάνοντας υπόψη τις υπηρεσίες ως καθήκοντα οι ερευνητές στο [8] ανάγουν το πρόβλημα της αυτόματης σύνθεσης σε HTN Planning πρόβλημα. Αντί να έχει ένα σύνολο από στόχους να επιτύχει έχει έναν συγκεκριμένο αριθμό καθηκόντων να εκτελέσει. Για να επιλύσει planning προβλήματα σε ένα σύστημα, το SHOP 2 χρειάζεται συγκεκριμένη γνώση που την αντλεί από τις μεθόδους. Η κάθε μέθοδος παρέχει τον τρόπο με τον οποίο θα μπορούσε να αναλυθεί ένα καθήκον σε μια δομημένη σειρά υπο-καθηκόντων. Για να προκύψει η ακολουθία καθηκόντων του τελικού πλάνου το SHOP2 ενόσω υπάρχουν σύνθετα καθήκοντα τα αποσυνθέτει σε υπο-

καθήκοντα, έως ότου επιτύχει το στόχο των πρωτογενών καθηκόντων που μπορούν να επιτευχθούν απ'ευθείας από τον planning operator.

Έχουν δημιουργηθεί αρκετοί HTN planners για την επίλυση του προβλήματος της σύνθεσης υπηρεσιών. Η διαφορά του SHOP2 με τους υπόλοιπους είναι ότι το SHOP2 σχεδιάζει πλάνα για καθήκοντα (tasks) με την ίδια σειρά που τα καθήκοντα αυτά θα εκτελεστούν στη συνέχεια. Αυτό έχει ως αποτέλεσμα να γνωρίζει ο planner σε κάθε βήμα του σχεδίου την τρέχουσα κατάσταση που βρίσκεται.



Αρχιτεκτονική του συστήματος [4]

Για τη χρήση του SHOP2 απαιτείται η κάθε ατομική υπηρεσία είτε να παράγει output είτε να έχει κάποια effects, όχι όμως και τα δύο. Για να χρησιμοποιηθεί λοιπόν το SHOP2 για σύνθεση θα πρέπει οι υπηρεσίες να επανασχεδιασθούν και να πληρούν τον παραπάνω κανόνα. Επιπλέον όπως αναφέραμε δεν μπορεί να συμμετέχει σε ένα πλάνο ένα σύνθετο καθήκον (σύνθετη process). Θα πρέπει λοιπόν να υπάρχει πάντα τρόπος ανάλυσης των σύνθετων καθηκόντων.

Έχουν δημιουργηθεί αρκετοί HTN planners για την επίλυση του προβλήματος της σύνθεσης υπηρεσιών. Η διαφορά του SHOP2 με τους υπόλοιπους είναι ότι το SHOP2 σχεδιάζει σχέδια για tasks με την ίδια σειρά που τα έργα αυτά θα εκτελεστούν στη συνέχεια. Αυτό έχει ως αποτέλεσμα να γνωρίζει ο planner σε κάθε βήμα του σχεδίου την τρέχουσα κατάσταση που βρίσκεται.

Για να μπορέσει να σχεδιάσει ένα σχέδιο σε ένα δεδομένο πεδίο ορισμού, το SHOP2 χρειάζεται να κατέχει τη «γνώση» για αυτό το σύστημα. Η «γνώση» στο SHOP2 τροφοδοτείται από τους

operators και τις μεθόδους . Ο κάθε operator είναι μια περιγραφή για το τι πρέπει να γίνει για να επιτευχθεί κάποιος πρωτογενής στόχος και η κάθε μέθοδος περιγράφει πως θα αναλυθεί ένα σύνθετο task σε ένα σύνολο από ordered υπο task.

Οι operators, οι μέθοδοι και το planning πρόβλημα ορίζονται ως μια τριάδα (S,T,D) με S να δηλώνει την αρχική κατάσταση, T να είναι μια λίστα από tasks, και D να είναι το σύστημα. Με τα (S,T,D) ως input to SHOP2 θα επιστρέψει ένα σχέδιο $P=(p_1,p_2,\dots,p_n)$ που είναι η ακολουθία των operators που πρέπει να χρησιμοποιηθούν ώστε να επιτύχουμε το T από την αρχική κατάσταση S στο πεδίο ορισμού D.

Χρησιμοποιώντας την αντιστοίχιση της σημασιολογίας που κάνουν οι [1] και οι [5] ορίζεται η σύνθεση υπηρεσιών OWL-S ως

$K = \{K_1, K_2, \dots, K_m\}$ μια συλλογή OWL-S Process models

C μια πιθανή σύνθετη process ορισμένη στο K

S_0 η αρχική κατάσταση

$P = \{P_1, P_2, \dots, P_n\}$ μια ακολουθία ατομικών processes ορισμένων στο K.

Αποδεικνύεται στη συνέχεια πως η P είναι η σύνθεση για το C στο K από την κατάσταση S_0

2.6 ΣΥΝΘΕΣΗ ΜΕ ΤΗΝ OWL-S

Η **OWL-S** έχει τη δυνατότητα να περιγράψει το περιεχόμενο και τις δυνατότητες μιας υπηρεσίας ιστού με σημασιολογία, κάτι που χρησιμοποιείται ως βάση στην αυτοματοποιημένη σύνθεση των υπηρεσιών ιστού. Πολλοί ερευνητές που μελετούν το πρόβλημα της σύνθεσης επιλέγουν την OWL-S στα παραδείγματα τους [5],[7].

Στην πρόταση που γίνεται από τους Narayanan et al. [1] και McIlraith et al. [5], οι περιγραφές των υπηρεσιών αντιστοιχίζονται πρώτα σε Situation Calculus και στη συνέχεια σε Petri nets. Ο στόχος της σύνθεσης των υπηρεσιών ιστού εδώ περιγράφεται με βάση τον τελικό στόχο που έχει επιτευχθεί δεδομένης της αρχικής κατάστασης που υπήρχε [2].

Το Situation Calculus είναι, όπως προαναφέρθηκε, μια γλώσσα λογικής πρώτης τάξης που παρουσιάζει δυναμικά συστήματα όπου όλες οι αλλαγές είναι αποτέλεσμα των ενεργειών που έχουν συμβεί από κάποιους πράκτορες. Οι καταστάσεις είναι αποτέλεσμα μιας σειράς ενεργειών που επηρέασαν την αρχική κατάσταση. Οι ατομικές processes στην OWL-S είναι ενέργειες στο Situation Calculus. Οι υποθέσεις αντιπροσωπεύουν αξιώματα (θετικά ή αρνητικά) και τα outputs knowledge effects. Τα προαπαιτούμενα για μια ατομική process παρουσιάζονται ως well-formed formulae και τα inputs σαν knowledge preconditions.

Η παραπάνω ερευνητική ομάδα έχει επιπλέον κατασκευάσει έναν interpreter που διαβάζει OWL-S αρχεία και επιστρέφει μια δικτυακή απεικόνιση των όσων έχουν υλοποιηθεί. Επιπλέον μια σειρά μεθόδων ανάλυσης μπορούν να εφαρμοσθούν για να διασφαλισθεί η αποφυγή των deadlocks καθώς και η ορθότητα του τερματισμού των διεργασιών ώστε να επιτευχθεί σωστή σύνθεση υπηρεσιών.

2.5 ΣΧΟΛΙΑΣΜΟΣ

Υπάρχουν πολλές ακόμη προτάσεις για τη σύνθεση σημασιολογικών υπηρεσιών ιστού και αυτό οφείλεται στο γεγονός ότι η ερευνητική κοινότητα ασχολείται όλο και περισσότερο με το θέμα αυτό. Ταυτόχρονα όμως υπάρχουν αρκετά ζητήματα που παραμένουν ανοιχτά όσον αφορά τη σύνθεση υπηρεσιών ιστού. Τα ζητήματα αυτά έχουν να κάνουν με τον βαθμό αυτοματοποίησης της διαδικασίας αυτής, με θέματα ασφάλειας, με θέματα προτυποποίησης της σημασιολογικής περιγραφής, την κλίμακα υλοποίησης κ.α. Δεδομένων των παραπάνω ο τομέας της ανακάλυψης και αυτόματης σύνθεσης υπηρεσιών ιστού, λαμβάνοντας υπόψη ότι οι ενέργειες αυτές πρέπει να γίνονται με ακρίβεια όσον αφορά το μετασχηματισμό δεδομένων και καταστάσεων και σε ευρεία κλίμακα, δεν έχει καλυφθεί επαρκώς παρά τις προσπάθειες που έχουν γίνει.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Narayanan, S. and McIlraith, S., Simulation, verification and automated composition of web service, In proceedings of the 11th International World Wide Web Conference, May 2002.
- [2] R.Hull, J.Su. Tools for Composite Web Services: A short Overview.
- [3] S. Balzer, T. Liebig, M. Wagner, Pitfalls of OWL-S – A practical Semantic Web Use Case, 2nd International Conference on Service Oriented Computing ICSOC 2004, New York, USA, November 15-18, 2004.
- [4] Sheshagiri M.,desJardins M.Finin T., A Planner for Composing Services on Planning for Web Services, Lecture Notes in Computer Science, 2005, Volume 3399/2005, 536-547,
- [5] S. McIlraith and T. Son, “Adapting Golog for Composition of Semantic Web Services”, Proc. 8th Int. Conf. on Knowledge Representation and Reasoning (KR2002).
- [6] J.Rao, Application of Linear Logic to Web Service Composition, First International Conference on Web Services, Las Vegas, 2003
- [7] D. Martin, M. Burstein, J. Hobbs, and et al. OWL-S: Semantic Markup for Web Services, Version 1.1. <http://www.daml.org/services/owl-s>, 2004.
- [8] Evren Sirin, B.Parsia, Dan Wu, J.Hendler, Dana Nau, HTN Planning for web service composition using shop2, Elsevier, 2004
- [9] Adam Farquhar, Richard Fikes, James Rice, The Ontolingua Server: a Tool for Collaborative Ontology Construction, 1997
- [10] Natalya F. Noy and Deborah L. McGuinness., A Guide to Creating Your First Ontology, Stanford University,
- [11] Erol et al, HTN Planning: Complexity and Expressivity, 1994
- [12] Dieter Fensel, Enabling Semantic Web Services, Springer, 2007
- [13] Levesque, H., Reiter, R., Lesperance, Y., Lin, F., and Scherl, R, GOLOG: A logic programming language for dynamic domains. Journal of Logic Programming, 31, 59-84., 1997.
- [14] w3c, Semantic Web Services Language (SWSL) , <http://www.w3.org/Submission/SWSF-SWSL/>, 2005
- [15] McDermott, D.: Estimated-regression planning for interactions with Web services. In: Proceedings of the 6th International Conference on AI Planning and Scheduling, Toulouse, France. AAAI Press, Menlo Park (2002)
- [16] J. Peer, «A PDDL Based Tool for Automatic Web Service Composition».Principles and Practice of Semantic Web Reasoning, 2004.