



ΑΝΩΤΑΤΟ ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΠΑΤΡΑΣ  
ΣΧΟΛΗ ΔΙΟΙΚΗΣΗΣ & ΟΙΚΟΝΟΜΙΑΣ  
ΤΜΗΜΑ ΕΠΙΧΕΙΡΗΜΑΤΙΚΟΥ ΣΧΕΔΙΑΣΜΟΥ  
& ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

## **ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ**

### **Θέμα:**

*Semantic Web: Τεχνολογίες υλοποίησης του σημασιολογικού Ιστού.*

Επιμέλεια Εργασίας :	Ελευθερίου Σταματούλα	A. M. 296
	Ιωαννίδου Θεοφανή	A.M. 226
	Λελη Χαρίκλεια	A.M. 353

Υπεύθυνος Καθηγητής: κ<sup>ος</sup> Γρηγόρης Μπεληγιάννης

ΠΑΤΡΑ 2005



# ΠΕΡΙΕΧΟΜΕΝΑ

Πρόλογος.....	10
---------------	----

## ΚΕΦΑΛΑΙΟ 1

1.1 Το Web σήμερα.....	12
1.2 Από το Web στο Semantic Web. –Παραδείγματα.....	14
1.2.1 Η διαχείριση διοικητικής γνώσης .....	15
1.2.2, Ηλεκτρονικό εμπόριο, business-to-consumer (B2C).....	16
1.2.3 Το ενδοεπιχειρησιακό ηλεκτρονικό εμπόριο.....	18
1.2.4 Προσωπικοί διαμεσολαβητές: Ένα μελλοντικό σενάριο.....	19
1.3 Οι τεχνολογίες του Semantic Web.....	20
1.3.1Μεταδεδομένα.....	20
1.3.2 Οντολογίες.....	23
1.3.3 Η λογική.....	26
1.3.4 Διαμεσολαβητές .....	28

## ΚΕΦΑΛΑΙΟ 2

2.1 Περίληψη.....	30
2.1.1 Εισαγωγή: Οι αυξανόμενες ανάγκες του Web .....	31
2.1.2 Οι οντολογίες.....	33
2.1.2.1 Απλές οντολογίες και οι χρήσεις τους.....	39
2.1.2.2 Δομημένες οντολογίες και οι χρήσεις τους.....	41
2.1.2.3 Απόκτηση οντολογίας.....	46

2.1..2.4 Συνέπειες και ανάγκες σχετικές με οντολογίες.....	47
2.2 Οι Λογικές Περιγραφές(Description Logics) σαν Γλώσσες	
Οντολογιών για το Semantic Web.....	52
2.2.1 Περιγραφή.....	52
2.2.1.1 Το Semantic Web και οι οντολογίες.....	52
2.2.1.2 Λογικές Περιγραφές(Description Logic).....	53
2.2.2 Η Εκφραστική Περιγραφική Λογική SHIQ.....	57
2.2.3 Περιγράφοντας τις Οντολογίες σε SHIQ.....	60
2.2.4 SHIQ και DAML+OIL.....	62
2.2.5 Αιτιολόγηση στην SHIQ.....	66
2.2.6 Περιορισμοί Και Μεταβλητές Στην SHIQ.....	68
2.3 Από SHIQ και RDF στην OWL: Η παραγωγή μιας γλώσσας	
οντολογίας Web.....	70
2.3.1 Εισαγωγή.....	70
2.3.2 Επισκόπηση της OWL.....	71
2.3.3 Οι επιρροές στην OWL.....	74
2.3.3.1 Περιγραφές Λογικής.....	74
2.3.3.2 Κατασκευαστές Γλωσσών.....	77
2.3.3.3 Το παράδειγμα πλαισίων.....	79
2.3.3.4 Η RDF σύνταξη.....	81
2.3.4 Τα προηγούμενα συστήματα της OWL.....	81
2.3.4.1 SHOE.....	82
2.3.4.2 DAML- ONT.....	82
2.3.4.3 OIL.....	83
2.3.4.4 DAML+OIL.....	84
2.3.5 Τα προβλήματα κατά μήκος της πορείας.....	84
2.3.5.1 Τα συντακτικά προβλήματα.....	84
2.3.5.2 Τα σημασιολογικά προβλήματα.....	86
2.3. 5.3 Εκφραστική δύναμη.....	89
2.3.5.4 Τα υπολογιστικά προβλήματα.....	90

2.3.6	Οι λύσεις.....	90
2.3.6.1	Η αναγνωσιμότητα.....	91
2.3.6.2	Διαχειρίζοντας τις μη καλά σχεδιασμένες γραφικές παραστάσεις.....	92
2.3.6.3	Η παροχή μιας σημασιολογικής θεωρίας για την OWL.....	93
2.3.6.4	Αποφεύγοντας τα παράδοξα.....	94
2.3.6.5	Η διατήρηση της αποφασιστικότητας.....	95
2.3.7	OWL.....	96
2.3.7.1	Η OWL ως περιγραφική λογική.....	98
2.3.7.2	Σημασιολογία για την OWL DL.....	98
2.3.7.3	Μια ευκολότερη OWL: Η OWL Lite.....	101
2.3.7.4	OWL Full όπως η επέκταση RDF.....	102
2.3.7.5	Σημασιολογία για την OWL Full.....	102
2.3.8	Οι μελλοντικές επεκτάσεις.....	103

### ΚΕΦΑΛΑΙΟ 3

3.1	Σημασιολογικές υπηρεσίες Web.....	106
3.2	OWL-S.....	109
3.2.1	Υποστήριξη για την ανακάλυψη.....	110
3.2.2	SWRL.....	112
3.3	WSMO.....	115
3.4	METEOR-S.....	120
3.4.1	IRS-III.....	121
3.5	Μια περίπτωση χρήσης κίνητρου.....	122

3.5.1 Περιγραφή.....	122
3.5.2 Πεδίο δράσης.....	123
3.5.3 Δράστες, ρόλοι και στόχοι.....	124
3.5.4 Σενάρια χρήσης παραδείγματος.....	125

## ΚΕΦΑΛΑΙΟ 4

4.1 Ένα εννοιολογικό πρότυπο για την ανακάλυψη.....	126
4.1.1 Καθορισμός της υπηρεσίας.....	126
4.1.2 Επίπεδα αφαίρεσης.....	127
4.1.3 Πεδίο.....	128
4.1.4 Υποθέσεις.....	128
4.1.5 Εννοιολογικό πρότυπο.....	130
4.2 Κατάσταση προόδου στην ανάκτηση τμημάτων λογισμικού και ανακάλυψη υπηρεσιών Web.....	133
4.2.1 Κατάσταση προόδου στην ανάκτηση τμημάτων λογισμικού.....	133
4.2.2 Κατάσταση προόδου στην ανακάλυψη υπηρεσιών Web.....	142
4.3 Περιγραφή των υπηρεσιών και των στόχων του Web.....	150
4.4 Ανακάλυψη των αυτόματων υπηρεσιών του Web.....	152
4.4.1 Ανακάλυψη βασισμένη στη λέξη.....	152
4.4.2 Σημασιολογικός χαρακτηρισμός των αποτελεσμάτων.....	154
4.5 Χρησιμοποιώντας τη DL για το χαρακτηρισμό των αποτελεσμάτων.....	175
4.5 Συνενώσεις υπηρεσιών στον αυτοματοποιημένο Ιστό.....	188
4.7 Σχέση μεταξύ της ανακάλυψης και της μεσολάβησης.....	199
4.7.1 Υποθέσεις στη μεσολάβηση.....	199
4.7.2 Απαιτήσεις μεσολάβησης .....	200

<b>4.8 Επιτεύγματα .....</b>	<b>203</b>
<b>4.9 Ανοικτά σημεία .....</b>	<b>204</b>
<b>4.10 Μελλοντική εργασία .....</b>	<b>206</b>

## **ΚΕΦΑΛΑΙΟ 5**

<b>5.1 Σημασιολογία για τη σύνθεση υπηρεσιών στο Web.....</b>	<b>207</b>
<b>5.2 Λειτουργικού επιπέδου Σύνθεση.....</b>	<b>209</b>
<b>5.2.1 Ιστορικό και κατάσταση προόδου.....</b>	<b>210</b>
<b>5.2.2 Φορμαλισμός και Σημασιολογία.....</b>	<b>212</b>
<b>5.2.3 Αυτοματοποιημένη σύνθεση υπηρεσιών.....</b>	<b>217</b>
<b>5.2.4 τεχνικές εφαρμογής (υποστήριξη καταλόγου για την αυτοματοποιημένη σύνθεση υπηρεσιών).....</b>	<b>224</b>
<b>5.2.5 Αξιολόγηση.....</b>	<b>234</b>
<b>5.3 Σύνθεση ειδικού επιπέδου επεξεργασίας .....</b>	<b>242</b>
<b>5.3.1 Ιστορικό και κατάσταση προόδου.....</b>	<b>247</b>
<b>5.3.2 Επίσημος καθορισμός του προβλήματος .....</b>	<b>249</b>
<b>5.3.3 Αυτοματοποιημένη υπηρεσία σύνθεσης ειδικού επιπέδου επεξεργασίας.....</b>	<b>259</b>
<b>5.3.4 Αξιολόγηση και εκτίμηση.....</b>	<b>267</b>

## **ΚΕΦΑΛΑΙΟ 6**

<b>6.1</b>	<b>Ενοποίηση της υπηρεσίας ανακάλυψης και της σύνθεσης.....</b>	<b>273</b>
<b>6.2</b>	<b>Ανακάλυψη μέσα στη σύνθεση.....</b>	<b>273</b>
<b>6.3</b>	<b>Επαυξητική προσέγγιση.....</b>	<b>274</b>
<b>6.4</b>	<b>Επαναληπτική προσέγγιση.....</b>	<b>275</b>
<b>6.5</b>	<b>Σύνθεση δύο επιπέδων.....</b>	<b>276</b>
<b>6.6</b>	<b>Ανακάλυψη ειδικού επιπέδου επεξεργασίας.....</b>	<b>276</b>
	<b>Βιβλιογραφία – Προτεινόμενα βιβλία για ανάγνωση.....</b>	<b>278</b>



## Ευχαριστίες

Αισθανόμαστε την υποχρέωση να ευχαριστήσουμε όλους εκείνους που στα διάφορα στάδια της συγγραφής έκαναν διορθώσεις και πρότειναν βελτιώσεις. Ιδιαίτερα τον καθηγητή του τμήματος Επιχειρηματικού Σχεδιασμού και Πληροφοριακών Συστημάτων του Ανώτατου Τεχνολογικού Εκπαιδευτικού Ιδρύματος Πατρών, κ. Μπεληγιάννη Γρηγόριο, υπεύθυνο καθηγητή της παρούσας πτυχιακής εργασίας, για την πολύτιμη βοήθεια του καθώς και για τις εποικοδομητικές παρατηρήσεις του.

## ΠΡΟΛΟΓΟΣ

Τα τελευταία δύο χρόνια γίνεται πολύς λόγος και για το Semantic Web, ενός διαδικτύου δηλαδή στο οποίο η πληροφορία είναι καλύτερα ορισμένη επιτρέποντας με αυτόν τον τρόπο την ολοκλήρωση, αυτοματοποίηση και επαναχρησιμοποίηση των δεδομένων.

Ο Ιστός (Web) του μέλλοντος προβλέπεται να αποτελεί μια παγκόσμια βάση δεδομένων και γνώσης με πληροφορίες οι οποίες θα είναι "κατανοητές" από μηχανές (machine-understandable information). Οι κύριες τεχνολογίες για την υλοποίηση του Σημασιολογικού Ιστού είναι τα μεταδεδομένα, οι οντολογίες, οι διαμεσολαβητές και η λογική.

Η λέξη "Σημασιολογία" έχει ρίζα τις Ελληνικές λέξεις "σημάδι", "σημαίνω" και "σημαντικός" και σήμερα αναφέρεται στο νόημα, συχνά, σε επίπεδο γλώσσας. Μπορούμε να πούμε ότι ο Σημασιολογικός Ιστός αποτελεί το μεγαλύτερο σε παγκόσμιο επίπεδο έργο ευφυής ενσωμάτωσης συστημάτων ώστε να συνεργάζονται δια-λειτουργικά. Ο Tim Berners-Lee, που επινόησε τον Παγκόσμιο Ιστό το 1989, είχε το όραμα, που τώρα συμμερίζονται πολλοί - ενός ιστού δεδομένων που μπορούν να επεξεργαστούν από μηχανές.

Το παρόν σύγγραμμα ασχολείται με τις τεχνικές υλοποίησης του σημασιολογικού Ιστού. Τα μεταδεδομένα σχετίζονται με αντικείμενα, ιδέες, διαδικασίες καθώς και των σχέσεων που τις διέπουν. Αυτά τα δεδομένα έχουν σημασιολογικό νόημα το οποίο με την σειρά του είναι δυνατόν να περιγραφεί από φορμαλισμούς ή λεξιλόγια που καλούνται οντολογίες. Οι οντολογίες επιτρέπουν την αναπαράσταση του σημασιολογικού Ιστού κάνοντας χρήση γλωσσών όπως η SHOE, η DAML, η OIL και η υβριδική DAML+OIL που αποτελούν επεκτάσεις της RDF δανειζόμενες χαρακτηριστικά από αντίστοιχες γλώσσες αναπαράστασης του πεδίου της τεχνητής νοημοσύνης. Η διαδικασία της μετατροπής της πληροφορίας σε σημασιολογική είναι δυνατή μέσω ενός πλαισίου

περιγραφής, του RDF (Resource Description Framework) και των RDF schemas. Το World Wide Web Consortium (W3C) έχει δημιουργήσει μία ομάδα για την δημιουργία της γλώσσας OWL η οποία θα περιγράφει τις οντολογίες που υπάρχουν στο διαδύκτιο. Το παρόν έγγραφο εξετάζει τα προβλήματα της αυτόματης ανακάλυψης και της σύνθεσης Web υπηρεσιών που χρησιμοποιούν σημασιολογικά σχόλια. Το πρόβλημα της αυτόματης ανακάλυψης υπηρεσιών μπορεί να αντιμετωπιστεί ως πρόβλημα εντοπισμού μιας υπηρεσίας που μπορεί να συμπληρώσει αυτόματα κάποια απαιτούμενα στοιχεία. Η σύνθεση διακρίνεται σε δύο επίπεδα: τη λειτουργικού επιπέδου σύνθεση και την ειδικού επιπέδου επεξεργασίας σύνθεση. Η λειτουργικού επιπέδου σύνθεση επεκτείνει το πρόβλημα ανακαλύψεων επιλέγοντας, σε περίπτωση που μια μοναδική υπηρεσία που μπορεί να εκπληρώσει το στόχο δεν μπορεί να βρεθεί, ένα συνδυασμό υπηρεσιών που μπορούν να τον εκπληρώσουν. Η ειδικού επιπέδου επεξεργασίας σύνθεση καλύπτει μια πιο πρόσφατη φάση του γενικού στόχου σύνθεσης. Ο στόχος είναι να εξασφαλιστεί ο εκτελέσιμος κώδικας που εφαρμόζει τη σύνθεση.

Τέλος σε αυτό το σύγγραμμα εισάγουμε το πρόβλημα της ανακάλυψης και της σύνθεσης, καθώς και τους τρόπους επίλυσης τους. Ένα εννοιολογικό μοντέλο περιλαμβάνει τη περιγραφή των στόχων, την ανακάλυψη των πιθανών υποψήφιων υπηρεσιών και παρέχεται η συνένωση των υπηρεσιών που ανακαλύφθηκαν. Με βάση αυτό το πρότυπο αναλύουμε ποιες τρέχουσες προτάσεις καλύπτουν τις ανάγκες για την ανακάλυψη υπηρεσιών. Επίσης αναφέρουμε πως η ανακάλυψη και οι δύο τύποι συνθέσεων μπορούν να ενσωματωθούν

# ΚΕΦΑΛΑΙΟ 1

## 1.1 Το Web σήμερα

Το World Wide Web έχει αλλάξει τον τρόπο επικοινωνίας των ανθρώπων και τον τρόπο διεύθυνσης των επιχειρήσεων και αποτελεί τον πυρήνα μιας επανάστασης που προσπαθεί να στρέψει τον αναπτυγμένο κόσμο προς μια οικονομία γνώσης και γενικότερα να δημιουργήσει μια κοινωνία γνώσης.

Αυτή η ανάπτυξη έχει αλλάξει και τον τρόπο που χρησιμοποιούμε τους υπολογιστές. Έτσι ενώ αρχικά χρησιμοποιήθηκαν για αριθμητικούς υπολογισμούς, αυτήν την περίοδο η κυριότερη χρήση τους είναι η επεξεργασία πληροφοριών και οι πιο χαρακτηριστικές εφαρμογές τους είναι οι βάσεις δεδομένων, η επεξεργασία κειμένων, και τα παιχνίδια.

Το μεγαλύτερο μέρος του περιεχομένου του Web σήμερα, είναι απόλυτα προσβάσιμο απ' τους ανθρώπους. Ακόμη και το περιεχόμενο αυτό που παράγεται αυτόματα από τις βάσεις δεδομένων παρουσιάζεται συνήθως χωρίς τις αρχικές δομικές πληροφορίες που βρίσκονται στις βάσεις δεδομένων και έτσι είναι κατανοητό. Οι χαρακτηριστικότερες δραστηριότητες του Web, σήμερα, περιλαμβάνουν ανθρώπους που ψάχνουν για πληροφορίες, ανθρώπους που κάνουν έρευνα για να έρθουν σε επαφή με άλλους ανθρώπους, την αναθεώρηση των καταλόγων των on line καταστημάτων και την παραγγελία προϊόντων με τη συμπλήρωση φορμών καθώς επίσης και την επιθεώρηση του υλικού που αναφέρεται σε ενήλικες.

Αυτές οι δραστηριότητες όμως, δεν υποστηρίζονται καλά από τα εργαλεία του λογισμικού. Εκτός από την ύπαρξη των Links, οι οποίες δημιουργούν τις συνδέσεις μεταξύ των διαφόρων στοιχείων, τα πιο πολύτιμα, και απολύτως απαραίτητα, εργαλεία είναι οι μηχανές αναζήτησης.

Οι βασισμένες σε λέξεις, μηχανές αναζήτησης, όπως η AltaVista, το Yahoo, και το Google, είναι τα κύρια εργαλεία που χρησιμοποιούνται σήμερα. Είναι σαφές ότι το Web δεν θα είχε τόσο μεγάλη επιτυχία χωρίς τις μηχανές αναζήτησης. Εντούτοις, υπάρχουν σοβαρά προβλήματα που προκύπτουν απ' τη χρήση τους. Τέτοια προβλήματα είναι τα :

Ø Υψηλή δυνατότητα ανάκλησης, χαμηλή ακρίβεια. Ακόμα κι αν οι κύριες σελίδες ανακτώνται, έχουν μικρή χρησιμότητα γιατί μπορεί να έχουν ανακτηθεί ταυτόχρονα και άλλες 28.758 σχετικές ή άσχετες σελίδες. Έτσι κάποιες φορές η μεγάλη πληθώρα δεδομένων μπορεί να λειτουργήσει και τόσο αρνητικά όσο και η έλλειψη δεδομένων.

Ø Χαμηλή δυνατότητα ή και έλλειψη ανάκλησης. Πολλές φορές τυχαίνει να μην παίρνουμε καμία απάντηση για το αίτημά μας, ή ακόμη και να μην έχουν ανακτηθεί κάποιες σημαντικές, ή έστω σχετικές, σελίδες. Αν και η χαμηλή δυνατότητα ανάκλησης δεν εμφανίζεται συχνά ως πρόβλημα με τις υπάρχουσες μηχανές αναζήτησης, κάποιες φορές συμβαίνει

Ø Τα αποτελέσματα είναι ιδιαίτερα ευαίσθητα στο λεξιλόγιο. Συχνά οι αρχικές μας λέξεις κλειδιά, δεν φέρνουν τις πληροφορίες που θα θέλαμε . Αυτό συμβαίνει γιατί κάποια έγγραφα χρησιμοποιούν διαφορετική ορολογία από αυτή που χρησιμοποιήσαμε εμείς στο ερώτημά που θέσαμε στην αρχή. Αυτό βέβαια δεν μας ικανοποιεί γιατί ερωτήματα με την ίδια σημασιολογία θα έπρεπε να φέρνουν τα ίδια αποτελέσματα..

Ø Τα αποτελέσματα είναι σε διαφορετικές ιστοσελίδες. Εάν χρειαζόμαστε πληροφορίες οι οποίες συλλέγονται από διάφορες πηγές, πρέπει να θέσουμε ερωτήσεις για να βρούμε τα σχετικά έγγραφα, και έπειτα να εξαγάγουμε τις τμηματικές πληροφορίες που μας παρέχονται και να τις ενώσουμε..

Κατά έναν περίεργο τρόπο, παρά την τεχνολογική ανάπτυξη των μηχανών αναζήτησης, οι δυσκολίες παραμένουν ουσιαστικά οι ίδιες. Φαίνεται ότι η ποσότητα των πληροφοριών που παρέχονται από το Web δεν συμβαδίζει με την τεχνολογική πρόοδο.

Αλλά ακόμα κι αν μια αναζήτηση είναι επιτυχής, ο άνθρωπος είναι αυτός που πρέπει να διαβάσει τα επιλεγμένα έγγραφα για να βρει τις πληροφορίες που χρειάζεται. Αυτό σημαίνει ότι δεν υπάρχει μεγάλη υποστήριξη για την ανάκτηση των πληροφοριών και επομένως είναι μία πολύ χρονοβόρα διαδικασία. Έτσι, ο όρος «ανάκτηση πληροφοριών»

, που χρησιμοποιείται σε συνδυασμό με τις μηχανές αναζήτησης, είναι κάπως παραπλανητικός . Ένας πιο κατάλληλος όρος θα μπορούσε να είναι ο όρος «εύρεση θέσης». Επίσης, τα αποτελέσματα των αναζητήσεων στο Web δεν είναι εύκολα προσιτά από άλλα εργαλεία λογισμικού διότι οι μηχανές αναζήτησης είναι συχνά απομονωμένες εφαρμογές.

Το κυριότερο εμπόδιο για την παροχή καλύτερης υποστήριξης στους χρήστες του Web είναι ότι, αυτή τη στιγμή, η έννοια του περιεχομένου του Web δεν είναι προσιτή έννοια από τις μηχανή. Φυσικά, υπάρχουν εργαλεία που μπορούν να ανακτήσουν τα κείμενα, να τα χωρίσουν σε μέρη, να ελέγξουν την ορθογραφία, να μετρήσουν τις λέξεις τους, κ.τ.λ. αλλά στην ερμηνεία των προτάσεων και την εξαγωγή των χρήσιμων πληροφοριών για τους χρήστες, οι ικανότητες του τρέχοντος λογισμικού είναι ακόμα περιορισμένες. Πολύ απλά, είναι δύσκολο να διακρίνει τις έννοιες :

«εγώ είμαι καθηγητής της πληροφορικής»

Και

«Εγώ είμαι καθηγητής της πληροφορικής, μπορεί να σκεφτείτε. Καλά. ...»

## **1.2 Από το Web στο Semantic Web-Παραδείγματα**

Πώς μπορεί να βελτιωθεί η τρέχουσα κατάσταση χρησιμοποιώντας επεξεργαστή κειμένων; Μια λύση είναι να χρησιμοποιηθεί το περιεχόμενο του Web έτσι όπως είναι διαμορφωμένο σήμερα και να αναπτυχθούν, οι όλο και περιπλοκότερες, τεχνικές βασισμένες στη τεχνητή νοημοσύνη και την υπολογιστική γλωσσολογία. Αυτή η προσέγγιση αναπτύσσεται εδώ και κάποιο καιρό, αλλά παρά από μια μικρή πρόοδο που έχει σημειωθεί ο στόχος φαντάζει ακόμα πάρα πολύ μακρινός .

Είναι σημαντικό να γίνει κατανοητό ότι το Semantic Web δεν θα είναι μια νέα σφαιρική λεωφόρος της πληροφορίας που θα αναπτύσσεται παράλληλα με το υπάρχον World Wide Web αλλά θα εξελιχθεί πολύ περισσότερο από το ήδη υπάρχον Web. Το Semantic Web διαδίδεται μέσα από την κοινοπραξία World Wide Web (W3C), ως ένα διεθνές σώμα τυποποίησης για το Web. Η κατευθυντήρια δύναμη της πρωτοβουλίας του Semantic Web είναι του Tim Berners-Lee, το ίδιο το πρόσωπο που εφηύρε το WWW προς το τέλος της δεκαετίας του '80 και αναμένει με αυτήν την πρωτοβουλία την πραγματοποίηση του αρχικού οράματός του Web, ένα όραμα όπου η έννοια των

πληροφοριών διαδραμάτιζε πολύ σημαντικότερο ρόλο από αυτόν που έχει στο σημερινό Web.

Η ανάπτυξη του Semantic Web έχει την δύναμη βιομηχανίας, και οι κυβερνήσεις επενδύουν σε αυτό. Η Αμερικανική κυβέρνηση έχει καθιερώσει την DAML ως γλωσσικό έργο σήμανσης διαμεσολαβητών DARPA, και το Semantic Web είναι μεταξύ των γραμμών της κεντρικής δράσης στο πλαίσιο του έκτου προγράμματος της Ευρωπαϊκής Ένωσης.

### **1.2.1 Η διαχείριση διοικητικής γνώσης**

Η διαχείριση διοικητικής γνώσης ασχολείται με την απόκτηση, την πρόσβαση, και τη διατήρηση της γνώσης μέσα σε μια οργάνωση. Αυτό έχει προκύψει ως μια βασική δραστηριότητα των μεγάλων επιχειρήσεων επειδή βλέπει την εσωτερική γνώση ως διανοητικό προτέρημα από όπου μπορεί να υπάρξει μεγάλη παραγωγικότητα, δημιουργώντας νέα αξία, και αυξάνοντας την ανταγωνιστικότητα. Η διαχείριση γνώσης είναι ιδιαίτερα σημαντική για τους διεθνείς οργανισμούς με τα γεωγραφικά διασκορπισμένα τμήματα.

Οι περισσότερες πληροφορίες είναι διαθέσιμες σήμερα με μια όχι καλά δομημένη μορφή, όπως π.χ. το κείμενο, ο ήχος, και το βίντεο. Από τη διοικητική προοπτική της γνώσης, η τρέχουσα τεχνολογία πάσχει από περιορισμούς στις ακόλουθες περιοχές:

- Ø Έρευνα των πληροφοριών. Οι επιχειρήσεις εξαρτώνται συνήθως από μηχανές αναζήτησης βασισμένες στις λέξεις, οι περιορισμοί των οποίων έχουμε ήδη περιγραφεί.

- Ø Εξαγωγή πληροφοριών. Ο χρόνος και η ανθρώπινη προσπάθεια είναι απαραίτητα στοιχεία για τον έλεγχο των ανακτημένων εγγράφων με τις σχετικές πληροφορίες, γιατί οι υπάρχοντες ευφυείς διαμεσολαβητές είναι ανίκανοι να εκτελέσουν αυτόν τον στόχο.

- Ø Διατήρηση των πληροφοριών. Αυτήν την περίοδο υπάρχουν προβλήματα, όπως οι ασυνέπειες στην ορολογία και η αποτυχία στην προσπάθεια να αφαιρεθούν οι “ξεπερασμένες” πληροφορίες

- Ø Αποκάλυψη των πληροφοριών. Η νέα γνώση που υπάρχει σιωπηρά στις εταιρικές βάσεις δεδομένων εξάγεται χρησιμοποιώντας την ανάλυση

δεδομένων. Εντούτοις, αυτός ο στόχος είναι ακόμα πολύ δύσκολο να πραγματοποιηθεί λόγω των μη καλά δομημένων συλλογών των δεδομένων. .

Ø Πρόσβαση σε πληροφορίες: Συχνά είναι επιθυμητό να περιοριστεί η πρόσβαση σε ορισμένες πληροφορίες, σε ορισμένες ομάδες υπαλλήλων. Αυτά που κρύβουν ορισμένες πληροφορίες, είναι γνωστά από την περιοχή των βάσεων δεδομένων αλλά είναι δύσκολο να προστατευτούν στο ενδοδίκτυο (ή στο Web).

Ο στόχος του Semantic Web είναι να χρησιμοποιήσουμε τα πιο προηγμένα συστήματα διαχείρισης γνώσης:

Ø Η γνώση θα οργανωθεί στα εννοιολογικά διαστήματα .

Ø Τα αυτοματοποιημένα εργαλεία θα υποστηρίξουν τη συντήρηση και τον έλεγχο για τις ασυνέπειες.

Ø Η βασισμένη στη λέξη κλειδί, αναζήτηση, θα αντικατασταθεί απαντώντας στην ερώτηση: η ζητούμενη γνώση θα ανακτηθεί, θα εξαχθεί και θα παρουσιασθεί με φιλικό τρόπο στον άνθρωπο?

Ø Θα υποστηρίζονται ερωτήσεις που οι απαντήσεις τους βρίσκονται σε διάφορα έγγραφα.

Ø Θα είναι δυνατό να καθορίζεται ποιος θα έχει πρόσβαση σε ορισμένα μέρη των πληροφοριών (ακόμη και μέρη των εγγράφων) .

### 1.2.2 Ηλεκτρονικό εμπόριο, Business-to-Consumer (B2C)

Το ηλεκτρονικό εμπόριο , Business-to-Consumer, κυριαρχεί στο Web. Ένα χαρακτηριστικό παράδειγμα περιλαμβάνει ένα χρήστη που επισκέπτεται ένα ή περισσότερα on-line καταστήματα, κοιτάζοντας βιαστικά τις προσφορές τους, επιλέγοντας και παραγγέλλοντας τα προϊόντα.

Το ιδανικό θα ήταν ο χρήστης να συνέλεγε τις πληροφορίες για τις τιμές, τους όρους, τις καταστάσεις (όπως η διαθεσιμότητα των προϊόντων ) όλων, ή τουλάχιστον όλων των σημαντικών, on-line καταστημάτων και έπειτα να επιλέξει την καλύτερη προσφορά. Το εγχειρίδιο, όμως, που κοιτάζει βιαστικά ο χρήστης είναι πάρα πολύ χρονοβόρο και έτσι στο τέλος θα επισκεφτεί ένα ή πολύ λίγα on-line καταστήματα πριν λάβει μια απόφαση.



Για να βελτιωθεί αυτή η κατάσταση υπάρχουν διάφορα εργαλεία στο Web για διευκολύνουν τους καταναλωτές να αποφασίσουν για κάποια προϊόντα. Αυτά τα εργαλεία είναι ουσιαστικά διαμεσολαβητές του λογισμικού που επισκέπτονται διάφορα καταστήματα, εξάγουν τις πληροφορίες και τις τιμές για τα διάφορα προϊόντα και συντάσσουν μια επισκόπηση της αγοράς. Η λειτουργία αυτή παρέχεται από προγράμματα που εξάγουν πληροφορίες από ένα on-line κατάστημα και πρέπει να αναπτύσσεται ένα πρόγραμμα ανά κατάσταση. Αυτή η “προσέγγιση” όμως, πάσχει από διάφορα μειονεκτήματα.

Οι πληροφορίες αυτές εξάγονται από τα on-line καταστήματα μέσω της βασισμένης, σε λέξη κλειδί, αναζήτησης και άλλων μέσων της ανάλυσης κειμένου. Αυτή η διαδικασία χρησιμοποιεί τις υποθέσεις για την εγγύτητα ορισμένων κομματιών των πληροφοριών (παραδείγματος χάριν, η τιμή υποδεικνύεται από την λέξη “τιμή” που ακολουθείται από το σύμβολο \$ και από έναν θετικό αριθμό). Αυτή η ευρετική προσέγγιση είναι επιρρεπής σε λάθη και γι’ αυτό δεν είναι πάντα σίγουρο ότι θα λειτουργήσει σωστά. Εξαιτίας λοιπόν των δυσκολιών αυτών εξάγονται περιορισμένες πληροφορίες. Παραδείγματος χάριν, πληροφορίες όπως οι δαπάνες ναυτιλίας, οι χρόνοι παράδοσης, οι περιορισμοί στη χώρα προορισμού, το επίπεδο ασφάλειας, και οι πολιτικές μυστικότητας ουσιαστικά δεν εξάγονται. Όλοι αυτοί οι παράγοντες όμως μπορούν να βοηθήσουν πολύ τον χρήστη να πάρει την τελική απόφασή του. Επιπλέον, τα προγράμματα είναι χρονοβόρα και οι διάφορες αλλαγές που γίνονται στην εμφάνιση των on-line καταστημάτων απαιτούν δαπανηρό επαναπρογραμματισμό.

Το Semantic Web επιτρέπει την ανάπτυξη διαμεσολαβητών λογισμικού που μπορούν να ερμηνεύσουν πληροφορίες για τα προϊόντα και τους όρους της τιμολόγησης των υπηρεσιών.

- Ø Οι πληροφορίες για τα προϊόντα θα εξαχθούν σωστά και οι πολιτικές παράδοσης και μυστικότητας θα ερμηνευθούν και θα συγκριθούν με τις απαιτήσεις των χρηστών.
- Ø Οι πληροφορίες για τη φήμη των on-line καταστημάτων θα ανακτηθούν από άλλες πηγές, παραδείγματος χάριν, από ανεξάρτητες αντιπροσωπείες ή καταναλωτικούς οργανισμούς.

- Ø Ο προγραμματισμός χαμηλού επιπέδου θα ξεπεραστεί.
- Ø Πιο περίπλοκοι διαμεσολαβητές θα είναι σε θέση να διεξάγουν τις διαπραγματεύσεις, εκ μέρους του αγοραστή, με τους διαμεσολαβητές των καταστημάτων.

### **1.2.3 Το ενδοεπιχειρησιακό ηλεκτρονικό εμπόριο**

Οι περισσότεροι χρήστες συνδέουν το εμπορικό μέρος του Web με το B2C ηλεκτρονικό εμπόριο, αλλά η μεγαλύτερη οικονομική υπόσχεση όλων των on-line τεχνολογιών βρίσκεται στον τομέα του ενδοεπιχειρησιακού (B2B) ηλεκτρονικού εμπορίου.

Παραδοσιακά οι επιχειρήσεις έχουν ανταλλάξει τα στοιχεία τους χρησιμοποιώντας την προσέγγιση ηλεκτρονικής ανταλλαγής δεδομένων (EDI). Εντούτοις αυτή η τεχνολογία είναι περίπλοκη και κατανοητή μόνο από τους εμπειρογνώμονες. Είναι δύσκολο κάποιος να προγραμματίζει και να διατηρεί ταυτόχρονα τις πληροφορίες, και είναι επιρρεπές σε λάθη. Κάθε B2B επικοινωνία απαιτεί χωριστό προγραμματισμό και γι' αυτό τέτοιες επικοινωνίες είναι δαπανηρές. Τέλος, το EDI είναι μια απομονωμένη τεχνολογία.

Τα στοιχεία που ανταλλάσσονται δεν μπορούν να ενσωματωθούν εύκολα σε άλλες επιχειρησιακές εφαρμογές. Το Διαδίκτυο εμφανίζεται να είναι μια ιδανική υποδομή για την ενδοεπιχειρησιακή επικοινωνία. Οι επιχειρήσεις εξετάζουν λύσεις όλο και περισσότερο βασισμένες στο Διαδίκτυο, και νέα επιχειρησιακά πρότυπα έχουν προκύψει, όπως π.χ. οι πύλες B2B. Ακόμα και τώρα όμως το B2B ηλεκτρονικό εμπόριο αντιμετωπίζει έλλειψη προτύπων. Η HTML (γλώσσα σήμανσης υπερκειμένων) είναι πάρα πολύ δύσκολο να υποστηρίξει τις δραστηριότητες με αποτελεσματικό τρόπο: δεν παρέχει ούτε σωστή δομή αλλά ούτε και σημασιολογία για τις πληροφορίες. Τα νέα πρότυπα XML είναι μια μεγάλη βελτίωση αλλά μπορούν προς το παρόν να υποστηρίξουν τις επικοινωνίες μόνο σε περιπτώσεις όπου υπάρχει μία εκ των προτέρων συμφωνία για το λεξιλόγιο που χρησιμοποιείται και για την έννοιά του.

Η πραγματοποίηση του Semantic Web θα επιτρέψει στις επιχειρήσεις να αναπτύξουν συνεργασίες χωρίς πολλά χρήματα. Οι διαφορές στην ορολογία θα επιλυθούν χρησιμοποιώντας τα τυποποιημένα αφηρημένα πρότυπα των πεδίων, και τα στοιχεία θα ανταλλάσσονται χρησιμοποιώντας τις μεταφραστικές υπηρεσίες. Η δημοπρασία, οι

διαπραγματεύσεις, και η σύνταξη των συμβάσεων θα πραγματοποιούνται αυτόματα (ή σχεδόν αυτόματα) από τους διαμεσολαβητές λογισμικού.

#### **1.2.4 Προσωπικοί διαμεσολαβητές: Ένα μελλοντικό σενάριο**

Ο Michael είχε ένα ατύχημα με το αυτοκίνητο και αισθανόταν κάποιο πόνο στο λαιμό. Ο πρώτος παθολόγος που τον παρακολούθησε του πρότεινε μια σειρά από φυσιοθεραπείες. Ο Michael ζήτησε από το διαμεσολαβητή του Semantic Web να εργαστεί με κάποιες πιθανότητες.

Ο διαμεσολαβητής ανάκτησε τις λεπτομέρειες της συνιστώμενης θεραπείας από τον διαμεσολαβητή του γιατρού και ανέτρεξε στον κατάλογο των θεραπόντων γιατρών που διατηρούσε η εταιρεία που του παρείχε την ασφάλεια υγείας. Ο πράκτορας έλεγξε εκείνους που βρέθηκαν μέσα σε μια ακτίνα 10 χλμ από το γραφείο του Michael ή το σπίτι, και ανέτρεξε τη φήμη τους από κάποιες υπηρεσίες που εμπιστευόταν. Κατόπιν προσπάθησε να ταιριάξει τα διαθέσιμα ραντεβού τους με το πρόγραμμα του Michael. Σε λίγα λεπτά ο διαμεσολαβητής του εμφάνισε δύο προτάσεις. Δυστυχώς, ο Michael δεν ήταν ευχαριστημένος από κανέναν τους. Ο ένας ήταν διαθέσιμος σε δύο εβδομάδες και για τον δεύτερο θα έπρεπε να οδηγήσει σε ώρες αιχμής. Επομένως, ο Michael αποφάσισε να θέσει τους ακριβέστερους χρονικούς περιορισμούς και ζήτησε από το διαμεσολαβητή να προσπαθήσει πάλι.

Λίγα λεπτά αργότερα ο διαμεσολαβητής επέστρεψε με μια εναλλακτική λύση: Ένας θεράπων γιατρός με άριστη φήμη ήταν διαθέσιμος σε δύο ημέρες. Εντούτοις, υπήρξαν μερικά δευτερεύοντα προβλήματα. Μερικά από τα λιγότερο σημαντικά ραντεβού εργασίας του Michael θα έπρεπε να αλλάξουν. Ο διαμεσολαβητής προσφέρθηκε να το ρυθμίσει αυτό το θέμα εάν τελικά το αποφάσιζε ο Michael. Ο θεράπων γιατρός δεν υπήρχε όμως στον κατάλογο του ασφαλιστή επειδή χρέωνε περισσότερο από τη μέγιστη κάλυψη του ασφαλιστή. Ο διαμεσολαβητής είχε βρει το όνομά του από έναν ανεξάρτητο κατάλογο θεραπόντων γιατρών και ήδη είχε ελέγξει εάν ο Michael είχε το δικαίωμα μέγιστης κάλυψης του ασφαλιστή. Είχε διαπραγματευτεί επίσης με τον διαμεσολαβητή του θεράποντος γιατρού για μια ειδική έκπτωση. Ο θεράπων γιατρός είχε πρόσφατα αποφασίσει να χρεώσει περισσότερο από το μέσο όρο και ήθελε πολύ να βρει νέους ασθενείς.

Ο Michael ήταν ευχαριστημένος από τη σύσταση επειδή θα έπρεπε να πληρώσει μόνο μερικά δολάρια επιπλέον. Εντούτοις, επειδή είχε εγκαταστήσει το διαμεσολαβητή του Semantic Web μόνο μερικές ημέρες πριν, του ζήτησε να του εξηγήσει μερικούς από τους ισχυρισμούς του, όπως π.χ. για το πώς καθιερώθηκε η φήμη του θεράποντος γιατρού, γιατί ήταν απαραίτητο να αλλάξει μερικά από τα ραντεβού εργασίας του ο Michael, πώς έγινε η διαπραγμάτευση για την τιμή, κ.τ.λ. Ο διαμεσολαβητής παρείχε τις σωστές πληροφορίες.

Ο Michael ήταν ικανοποιημένος. Ο νέος διαμεσολαβητής του Semantic Web επρόκειτο να κάνει την ζωή του ευκολότερη και έτσι του ζήτησε να λάβει όλα τα απαραίτητα μέτρα για να πραγματοποιηθεί ο στόχος.

### **1.3 Οι τεχνολογίες του Semantic Web**

Τα σενάρια που περιγράφονται στην προηγούμενη παράγραφο δεν είναι επιστημονική φαντασία και δεν απαιτούν επαναστατική πρόοδο στην επιστήμη για να επιτευχθούν. Μπορούμε εύλογα να υποστηρίξουμε ότι η πρόκληση είναι να υιοθετηθεί η τεχνολογία παρά κάτι επιστημονικό: μερικές λύσεις σε όλα τα σημαντικά μέρη του προβλήματος υπάρχουν. Αυτή τη στιγμή, οι μέγιστες ανάγκες είναι στους τομείς της ολοκλήρωσης, της τυποποίησης, της ανάπτυξης και της υιοθέτησης των εργαλείων από τους χρήστες. Η περαιτέρω τεχνολογική πρόοδος θα οδηγήσει σε ένα πιο προηγμένο Semantic Web που μπορεί, σε γενικές γραμμές, να επιτευχθεί σήμερα.

#### **1.3.1 Τα μεταδεδομένα**

Αυτήν την περίοδο, το περιεχόμενο του Web είναι σχηματισμένο περισσότερο για τους ανθρώπινους αναγνώστες παρά για τα προγράμματα. Η HTML είναι η κυρίαρχη γλώσσα στις σελίδες του Web και γράφεται άμεσα ή χρησιμοποιώντας τα εργαλεία. Ένα μέρος μιας τυπικής σελίδας του Web ενός φυσιοθεραπευτή μπορεί να μοιάζει με αυτό:

```
<h1>Agilitas κέντρο φυσιοθεραπείας</h1>
```

Καλωσήρθατε στην αρχική σελίδα του κέντρου φυσιοθεραπείας Agilitas. Αισθάνεστε πόνο; Είχατε κάποιο τραυματισμό; Αφήστε το προσωπικό μας, την Lisa Davenport, την Kelly Townsend (την καλή μας γραμματέα) και τον Steve Matthews να φροντίσουν το σώμα και την ψυχή σας.

## < h2>Consultation hours</h2>

Mon	11am	-	7pm 
Tue	11am	-	7pm 
Wed	3pm	-	7pm 
Thu	11am	-	7pm 
Fri 11am - 3pm			

Σημειώστε όμως ότι δεν προσφέρουμε διαβουλεύσεις κατά τη διάρκεια των εβδομάδων ενός

< a href = ".. " >State Origin </a> παιχνιδιού.

Για τους ανθρώπους οι πληροφορίες παρουσιάζονται με έναν ικανοποιητικό τρόπο, αλλά οι μηχανές θα έχουν προβλήματα. Η βασισμένη σε λέξεις κλειδιά, αναζήτηση θα αναγνωρίσει τις λέξεις κλειδιά ώρες φυσιοθεραπείας και ώρες ραντεβού. Ένας ευφυής διαμεσολαβητής μπορεί να είναι σε θέση να προσδιορίσει το προσωπικό του κέντρου, αλλά και αυτός θα έχει πρόβλημα να διακρίνει τον θεράποντα γιατρό από τη γραμματέα ή ακόμη και με την εύρεση της ακριβής ώρας ραντεβού.

Η προσέγγιση του Semantic Web για την επίλυση αυτών των προβλημάτων δεν είναι η ανάπτυξη ακόμη πιο ευφυών διαμεσολαβητών. Αντ' αυτού προτείνει να λυθεί το πρόβλημα απ τις ιστοσελίδες. Εάν η HTML αντικατασταθεί από καταλληλότερες γλώσσες, οι ιστοσελίδες θα μπορούν να εμφανίσουν το περιεχόμενό τους πολύ πιο γρήγορα. Επιπλέον εκτός από τον περιορισμό των πληροφοριών μορφοποίησης που στοχεύουν στην παραγωγή ενός εγγράφου για τους ανθρώπινους αναγνώστες, θα μπορούσαν να περιέχουν πληροφορίες και για το περιεχόμενό τους. Στο παράδειγμά μας θα παρέχονταν πληροφορίες όπως:

```
<company>
<treatmentOffered>Physiotherapy</treatmentOffered>
<companyName>Agilitas Physiotherapy Centre</companyName>
<staff>
<therapist>Lisa Davenport</therapist>
<therapist>Steve Matthews</therapist>
<secretary>Kelly Townsend</secretary>
</staff>
</company>
```

Ο όρος μεταδεδομένα αναφέρεται σε τέτοιου είδους πληροφορίες: δεδομένα για τα δεδομένα. Τα μεταδεδομένα συλλαμβάνουν μέρος της έννοιας των δεδομένων, κατά συνέπεια και ο όρος Semantic στο Semantic Web(Σημασιολογικό Ιστό).

Στο σενάριο του παραδείγματός στην παράγραφο 1.2 δεν φάνηκε να υπάρχει κανένα εμπόδιο στην πρόσβαση στις πληροφορίες της ιστοσελίδας: οι λεπτομέρειες θεραπείας, τα ραντεβού, οι τιμές και οι περιγραφές προϊόντων, φάνηκαν όπως όλες αυτές οι πληροφορίες που θα μπορούσαν να ανακτηθούν άμεσα από το υπάρχον περιεχόμενο του Web. Αυτό δεν συμβαίνει όμως χρησιμοποιώντας το κείμενο το οποίο βασίζεται στον χειρισμό των πληροφοριών αλλά μάλλον μόνο όταν εκμεταλλευθεί η μηχανή τα επεξεργάσιμα μεταδεδομένα.

Όπως με την τρέχουσα ανάπτυξη των ιστοσελίδων, οι χρήστες δεν θα είναι απαραίτητο να είναι εμπειρογνώμονες της πληροφορικής για να αναπτύξουν ιστοσελίδες γιατί θα είναι σε θέση να χρησιμοποιήσουν τα εργαλεία για αυτόν το λόγο. Ακόμα όμως παραμένει η ερώτηση γιατί οι χρήστες πρέπει να νοιάζονται για το εάν πρέπει να εγκαταλείψουν την HTML για τις γλώσσες του Semantic Web. Ίσως μπορούμε να δώσουμε μια αισιόδοξη απάντηση εάν συγκρίνουμε την κατάσταση σήμερα με τον πρώτο καιρό που άρχισε να αναπτύσσεται το Web. Οι χρήστες αποφάσισαν να χρησιμοποιούν την HTML επειδή ήταν υιοθετημένη ως πρότυπο και περίμεναν να δουν τα πλεονεκτήματά της από τους πρώτους που την χρησιμοποίησαν. Άλλοι πάλι ακολούθησαν όταν διατέθηκαν περισσότερα και καλύτερα εργαλεία του Web και έτσι σύντομα η HTML ήταν παγκοσμίως αποδεκτό πρότυπο.

Ομοίως, παρατηρούμε αυτήν την περίοδο την έγκαιρη υιοθέτηση της XML. Ενώ δεν αρκεί για την πραγματοποίηση του οράματος του Semantic Web, η XML είναι ένα σημαντικό πρώτο βήμα . Οι πρώτοι χρήστες και ίσως μερικές μεγάλες οργανώσεις που ενδιαφέρονται για την διαχείριση γνώσης και το B2B ηλεκτρονικό εμπόριο, θα υιοθετήσουν την XML και την RDF, τα τρέχοντα πρότυπα του Semantic Web W3C. Αυτή η ορμή θα οδηγήσει σε όλο και περισσότερους προμηθευτές εργαλείων και τελικών χρηστών που θα υιοθετήσουν την τεχνολογία.

Αυτό θα είναι ένα αποφασιστικό βήμα στη επιχείρηση του Semantic Web, αλλά είναι επίσης και μια πρόκληση. Όπως αναφέραμε, η μέγιστη πρόκληση αυτή την στιγμή δεν είναι επιστημονική αλλά είναι μια πρόκληση υιοθέτησης της τεχνολογίας.

### 1.3.2 Οντολογίες

Ο όρος οντολογία προέρχεται από τη φιλοσοφία. Εκεί, χρησιμοποιείται ως όνομα ενός πεδίου, δηλαδή, τη μελέτη της φύσης της ύπαρξης (η κυριολεκτική μετάφραση της ελληνικής λέξης Οντολογία), τον κλάδο της μεταφυσικής που έχει στρέψει το ενδιαφέρον του στον προσδιορισμό και στους γενικότερους όρους των ειδών των πραγμάτων που υπάρχουν πραγματικά, και τον τρόπο περιγραφής τους. Παραδείγματος χάριν, η παρατήρηση ότι ο κόσμος αποτελείται από τα αντικείμενα που μπορούν να ομαδοποιηθούν στις αφηρημένες κατηγορίες που εδρεύουν στις κοινές ιδιότητες είναι μια χαρακτηριστική οντολογική υποχρέωση.

Εντούτοις, στα πιο πρόσφατα έτη, η οντολογία έχει γίνει μια από τις πολλές λέξεις που χρησιμοποιούνται “πειρατικά” από την πληροφορική και της δίνουν ένα συγκεκριμένο τεχνικό νόημα που είναι μάλλον διαφορετικό από το αρχικό. Αντί "της οντολογίας" μιλάμε τώρα για "μια οντολογία". Εμείς θα χρησιμοποιήσουμε τον ορισμό που έδωσε ο T.R.Gruber και που αργότερα βελτιώθηκε από τον P. Studer, δηλαδή: “Μια οντολογία είναι ένα συγκεκριμένο ρητό και τυπικά επίσημο μιας σύλληψης”.

Γενικά, μια οντολογία περιγράφει τυπικά μια περιοχή της ομιλίας. Χαρακτηριστικά, μια οντολογία αποτελείται από έναν κατάλογο όρων και των σχέσεων μεταξύ αυτών των όρων. Οι όροι δείχνουν τις σημαντικές έννοιες (κατηγορίες αντικειμένων) της περιοχής. Παραδείγματος χάριν, σε μια πανεπιστημιακή ρύθμιση, τα μέλη του προσωπικού, οι σπουδαστές, οι σειρές μαθημάτων, τα θέατρα διάλεξης, και οι επιστήμες είναι μερικές σημαντικές έννοιες.

Οι σχέσεις περιλαμβάνουν χαρακτηριστικά τις ιεραρχίες των κατηγοριών. Μια συγκεκριμένη ιεραρχία ορίζει ότι μια κατηγορία C θα είναι υποκατηγορία μιας άλλης κατηγορίας C’ εάν κάθε αντικείμενο στο C συμπεριλαμβάνεται επίσης στο C’’. Παραδείγματος χάριν, όλα τα μέλη του προσωπικού είναι ικανά. Το σχήμα 1.1 παρουσιάζει μια ιεραρχία για την πανεπιστημιακή περιοχή.

Εκτός από τις σχέσεις των υποκατηγοριών, οι οντολογίες μπορούν να περιλάβουν πληροφορίες όπως :

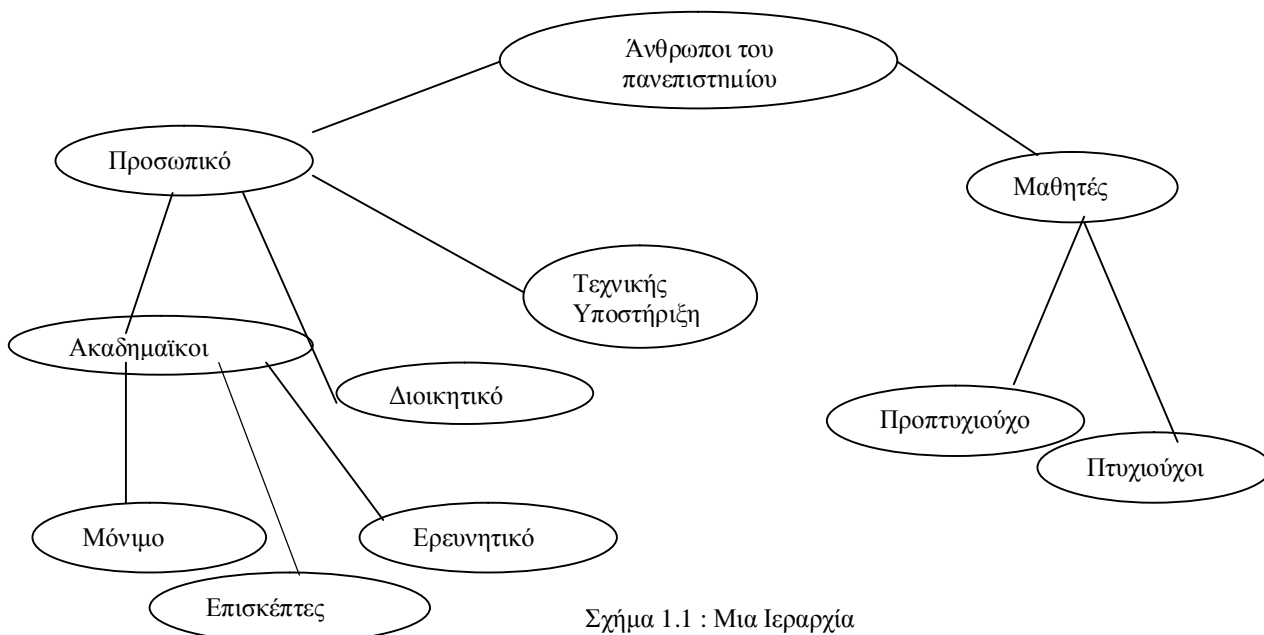
- ∅ Ιδιότητες (το X διδάσκει στο Y) .

∅ Περιορισμούς αξίας (μόνο τα μέλη σχολής μπορούν να διδάξουν τις σειρές μαθημάτων) .

∅ Δηλώσεις όχι καλά συνδεδεμένες μεταξύ τους (π.χ. η σχολή και το γενικό προσωπικό δεν μπορούν να συνδυαστούν)

∅ Συγκεκριμενοποίηση των λογικών σχέσεων μεταξύ των αντικειμένων (κάθε τμήμα πρέπει να περιλαμβάνει τουλάχιστον δέκα μέλη της σχολής)

Στα πλαίσια του Web, οι οντολογίες παρέχουν μια κοινή κατανόηση μιας περιοχής. Μια τέτοια κοινή κατανόηση είναι απαραίτητη για να υπερνικήσει τις διαφορές στην ορολογία. Ένας κώδικας μιας εφαρμογής μπορεί να είναι ο ίδιος με τον τομέα ενός κώδικα μιας άλλης εφαρμογής. Ένα άλλο πρόβλημα είναι ότι δύο εφαρμογές μπορούν να χρησιμοποιήσουν τους ίδιους όρους με διαφορετικές έννοιες. Για παράδειγμα στο πανεπιστήμιο A, μια σειρά μαθημάτων μπορεί να αναφέρεται σε ένα θέμα όπως την πληροφορική, ενώ στο πανεπιστημιακό B μπορεί να αναφέρεται σε ένα ενιαίο θέμα . Τέτοιες διαφορές μπορούν να αποφευχθούν με τη χαρτογράφηση της ιδιαίτερης ορολογίας σε μια κοινή οντολογία ή από τις άμεσες χαρτογραφήσεις μεταξύ των οντολογιών. Σε καθεμία περίπτωση, είναι εύκολο να δει κάποιος ότι οι οντολογίες υποστηρίζουν τη σημασιολογική διαλειτουργικότητα.



Σχήμα 1.1 : Μια Ιεραρχία



Οι οντολογίες είναι χρήσιμες για την οργάνωση και τη ναυσιπλοΐα των ιστοχώρων. Πολλές ιστοσελίδες εκθέτουν σήμερα στην αριστερή πλευρά της σελίδας τα κορυφαία επίπεδα μιας ιεραρχικής έννοιας των όρων. Ο χρήστης μπορεί να κάνει κλικ σε έναν από αυτούς για να επεκτείνει τις υποκατηγορίες.

Επίσης, οι οντολογίες είναι χρήσιμες για την ακρίβεια των αναζητήσεων στο Web. Οι μηχανές αναζήτησης μπορούν να ψάξουν τις σελίδες που αναφέρονται σε μια ακριβή έννοια μιας οντολογίας αντί της συλλογής όλων των σελίδων στις οποίες είναι ορισμένες, γενικά και διαφορούμενες και εμφανίζονται οι λέξεις κλειδιά.

Επιπλέον, οι αναζητήσεις στο Web μπορούν να εκμεταλλευτούν τη γενίκευση /τις πληροφορίες ειδίκευσης. Εάν μια ερώτηση αποτυγχάνει στο να βρεθούν οποιαδήποτε σχετικά έγγραφα, η μηχανή αναζήτησης μπορεί να προτείνει στο χρήστη μια γενικότερη ερώτηση. Αυτό είναι κατανοητό και για τη μηχανή να τρέχουν τέτοιες ερωτήσεις για να μειώσει το χρόνο αντίδρασης σε περίπτωση που ο χρήστης υιοθετήσει μια πρόταση. Ακόμη κι εάν ανακτώνται πάρα πολλές απαντήσεις για ένα ερώτημα, η μηχανή αναζήτησης μπορεί να προτείνει στο χρήστη μερικές πιο ειδικευμένες απαντήσεις.

Στη Τεχνητή Νοημοσύνη (AI) υπάρχει μια μακροχρόνια παράδοση ανάπτυξης και χρησιμοποίησης των γλωσσών οντολογίας. Αυτή τη στιγμή, οι σημαντικότερες γλώσσες οντολογίας για τον Ιστό είναι οι ακόλουθες:

- Ø Η XML παρέχει μια επιφάνεια σύνταξης για τα δομημένα έγγραφα αλλά δεν επιβάλλει κανέναν σημασιολογικό περιορισμό στην έννοια αυτών των εγγράφων.
- Ø Η XML Schema είναι μια γλώσσα για τον περιορισμό της δομής των XML των εγγράφων.
- Ø Η RDF είναι ένα πρότυπο στοιχείων για τα αντικείμενα ("πόροι") και τις σχέσεις μεταξύ τους, παρέχει μια απλή σημασιολογία για το πρότυπο στοιχείων και αυτά τα πρότυπα στοιχείων μπορούν να αντιπροσωπευθούν με XML σύνταξη.
- Ø Η RDF Schema είναι μια γλώσσα περιγραφής λεξιλογίου για την περιγραφή των ιδιοτήτων και τις κατηγορίες των πόρων της RDF, με μια σημασιολογία για τις ιεραρχίες γενίκευσης όπως οι ιδιότητες και οι κατηγορίες.
- Ø Η OWL είναι μια πλουσιότερη γλώσσα περιγραφής λεξιλογίου για την περιγραφή των ιδιοτήτων και των κατηγοριών, όπως οι σχέσεις μεταξύ των κατηγοριών (π.χ. δεν ταιριάζουν), του αριθμού στοιχείων συνόλου (π.χ. "ακριβώς ένας"), της

ισότητας, της πλουσιότερης δακτυλογράφησης των ιδιοτήτων, των χαρακτηριστικών των ιδιοτήτων (π.χ., συμμετρία), και των απαριθμημένων κατηγοριών.

### 1.3.3 Η λογική

Η λογική είναι η επιστήμη που μελετά τις αρχές. Γενικά, η λογική προσφέρει, πρώτον, επίσημες γλώσσες για την έκφραση της γνώσης και δεύτερον, μας παρέχει την κατανοητή επίσημη σημασιολογία: στις περισσότερες λογικές, η έννοια των προτάσεων είναι κατανοητή χωρίς να είναι απαραίτητη η λειτουργική γνώση. Συχνά μιλάμε για τη δηλωτική γνώση: περιγράφουμε τι κρατιέται χωρίς φροντίδα το πώς μπορεί να συναχθεί.

Και τρίτον, οι αυτοματοποιημένοι αιτιολογητές μπορούν να συναγάγουν (να συμπεράνουν) τα συμπεράσματα από τη δεδομένη γνώση, καθιστώντας κατά συνέπεια την υπονοούμενη γνώση ρητά. Τέτοιοι αιτιολογητές έχουν μελετηθεί εκτενώς στην Τεχνητή Νοημοσύνη. Πιο κάτω αναφέρουμε ένα παράδειγμα ενός συμπεράσματος. Υποθέτουμε ότι ξέρουμε ότι όλοι οι καθηγητές είναι ικανά μέλη, και επιπλέον και μέλη προσωπικού, και ότι ο Michael είναι καθηγητής. Στη κατηγορηματική λογική οι πληροφορίες εκφράζονται ως εξής:

καθ. (X)  $\rightarrow$  ικανότητα(X)

ικανότητα (X)  $\rightarrow$  προσωπικό(X)

καθ. (michael)

Κατόπιν μπορούμε να συναγάγουμε τα εξής:

καθ. (X)

προσωπικό (michael)

καθ. (michael)  $\rightarrow$  προσωπικό (X)

Σημειώστε ότι αυτό το παράδειγμα περιλαμβάνει τη γνώση που βρίσκεται χαρακτηριστικά στις οντολογίες. Κατά συνέπεια η λογική μπορεί να χρησιμοποιηθεί για να αποκαλύψει την οντολογική γνώση που δίνεται σιωπηρά. Με αυτόν τον τρόπο, μπορεί επίσης να βοηθήσει να αποκαλύψει τις απροσδόκητες σχέσεις και τις ασυνέπειες.

Η λογική είναι γενικότερη από τις οντολογίες. Μπορεί επίσης να χρησιμοποιηθεί από τους ευφυείς διαμεσολαβητές για τη λήψη των αποφάσεων και την επιλογή των σχεδίων

δράσης. Παραδείγματος χάριν, ένας διαμεσολαβητής κάποιου καταστήματος μπορεί να αποφασίσει να χορηγήσει μια έκπτωση σε έναν πελάτη βασισμένη στον κανόνα

Υποστήριξη Πελάτη (X) -> Έκπτωση(5%)

όπου η πίστη των πελατών καθορίζεται από τα στοιχεία που αποθηκεύονται στην εταιρική βάση δεδομένων. Γενικά υπάρχει ανταλλαγή μεταξύ της εκφραστικής δύναμης και υπολογιστικών ικανοτήτων. Όσο πιο εκφραστική είναι μια λογική, τόσο πιο ακριβή υπολογιστικά είναι τα συμπεράσματα. Ορισμένα συμπεράσματα μπορούν να γίνουν ακόμη και αδύνατα εάν τα εμπόδια δεν αντιμετωπίζονται. Ευτυχώς, η γνώση που σχετίζεται περισσότερο με το Semantic Web φαίνεται να είναι μιας σχετικά περιορισμένης μορφής. Παραδείγματος χάριν, τα προηγούμενα παραδείγματά μας περιέλαβαν κανόνες της μορφής, "εάν οι όροι, κατόπιν συμπέρασμα," και τελικά μόνο ορισμένα αντικείμενα έπρεπε να εξεταστούν. Αυτό το υποσύνολο της λογικής είναι εύκολο να αντιμετωπιστεί και υποστηρίζεται από τα εργαλεία συλλογισμού.

Ένα σημαντικό πλεονέκτημα της λογικής είναι ότι μπορεί να παρέχει και εξηγήσεις για τα συμπεράσματα: η σειρά βημάτων συμπεράσματος μπορεί να ανιχνευθεί εκ νέου. Επιπλέον οι ερευνητές της Τεχνητής νοημοσύνης έχουν αναπτύξει τρόπους για να εξηγούν με έναν φιλικό προς τον άνθρωπο, τρόπο, με την οργάνωση μιας απόδειξης ως φυσική αφαίρεση και την ομαδοποίηση διάφορων χαμηλού επιπέδου βημάτων συμπεράσματος. Στα μεταβήματα αυτά ένα πρόσωπο εξετάζει χαρακτηριστικά ένα ενιαίο βήμα απόδειξης. Τελικά μια εξήγηση θα επιστημάνει μια απάντηση πίσω σε ένα δεδομένο σύνολο γεγονότων και των κανόνων συμπεράσματος που χρησιμοποιούνται.

Οι εξηγήσεις είναι σημαντικές για το Semantic Web επειδή αυξάνουν την εμπιστοσύνη των χρηστών στους διαμεσολαβητές του Semantic Web.

Οι εξηγήσεις είναι επίσης απαραίτητες για τις δραστηριότητες μεταξύ των διαμεσολαβητών. Ενώ μερικοί διαμεσολαβητές θα είναι σε θέση να συναγάγουν τα λογικά συμπεράσματα, άλλοι θα έχουν μόνο την ικανότητα να επικυρώσουν τις αποδείξεις, δηλαδή να ελέγξουν εάν μια αξίωση που γίνεται από έναν άλλο διαμεσολαβητή τεκμηριώνεται. Παραθέτουμε το εξής παράδειγμα: υποθέστε ότι ο διαμεσολαβητής 1, που αντιπροσωπεύει το on-line κατάστημα, στέλνει μήνυμα "μου οφείλετε \$80" (όχι στη φυσική γλώσσα, εννοείται, αλλά σε μια επίσημη, επεξεργασίμη

γλώσσα μηχανής) στο διαμεσολαβητή 2, που αντιπροσωπεύει το πρόσωπο. Κατόπιν ο διαμεσολαβητής 2 θα ζητήσει μια εξήγηση, και ο διαμεσολαβητής 1 θα αποκριθεί με μια ακολουθία της μορφής:

Απόδειξη αγοράς από το Web άνω των \$80

Απόδειξη παράδοσης (παραδείγματος χάριν, που ακολουθεί τον αριθμό του UPS)

Τους όρους και τις διατάξεις του καταστήματος:

αγορά(X, στοιχείο) ^ τιμή(στοιχείο, τιμή) ^ παράδοση (στοιχείο, X) ^πληρωμή(X, γεγονότα τιμών)

Έτσι θα επισημανθούν χαρακτηριστικά σε κάποιες διευθύνσεις Web (η εμπιστοσύνη που θα έχει διαπιστωθεί από τους διαμεσολαβητές), και οι κανόνες θα μπορούν να είναι ένα μέρος μιας κοινής οντολογίας εμπορίου ή της πολιτικής του on-line καταστήματος.

Λογικά για να είναι χρήσιμο στο Web πρέπει να είναι χρησιμοποιήσιμο από κοινού με άλλα στοιχεία και επεξεργάσιμο από τις μηχανές. Οι αρχικές προσεγγίσεις λειτουργούν σε επίπεδο XML, αλλά στο μέλλον οι κανόνες και οι αποδείξεις θα πρέπει να αντιπροσωπευθούν σε επίπεδο γλωσσών RDF και οντολογίας, όπως οι DAML+OIL και OWL.

### 1.3.4 Διαμεσολαβητές

Οι διαμεσολαβητές είναι κομμάτια του λογισμικού που λειτουργούν αυτόνομα και φιλενεργά. Εννοιολογικά εξελίχθηκαν από τις έννοιες του αντικειμενοστραφούς προγραμματισμού και της βασισμένης στα συστατικά, ανάπτυξη λογισμικού.

Ένας προσωπικός διαμεσολαβητής στο Semantic Web θα λάβει μερικούς στόχους και προτιμήσεις από το άτομο, θα αναζητήσει τις πληροφορίες από τις πηγές του Web, θα επικοινωνήσει με άλλους διαμεσολαβητές, θα συγκρίνει τις πληροφορίες για τις απαιτήσεις και τις προτιμήσεις των χρηστών, θα διαλέξει ορισμένες επιλογές, και θα δώσει τις απαντήσεις στο χρήστη. Ένα παράδειγμα ενός τέτοιου διαμεσολαβητή είναι ο προσωπικός διαμεσολαβητής του Michael στο παράδειγμα φυσιοθεραπείας της παραγράφου 1.2.4.

Πρέπει να σημειωθεί ότι οι διαμεσολαβητές δεν θα αντικαταστήσουν τους ανθρώπινους χρήστες στο Semantic Web, ούτε αυτοί θα λαμβάνουν απαραίτητως τις

αποφάσεις. Σε πολλές, εάν όχι στις περισσότερες περιπτώσεις ο ρόλος τους θα είναι να συλλέγουν και να οργανώνουν τις πληροφορίες, και να παρουσιάζουν διάφορες επιλογές στους χρήστες.

Οι διαμεσολαβητές Semantic Web θα χρησιμοποιήσουν όλες τις τεχνολογίες που έχουμε περιγράψει:

- Ø Τα μεταδεδομένα θα χρησιμοποιηθούν για να προσδιορίσουν και να εξαγάγουν τις πληροφορίες από τις οντολογίες πηγών του Web.
- Ø Οι οντολογίες θα χρησιμοποιηθούν για να βοηθήσουν στις αναζητήσεις στο Web, να ερμηνεύσουν τις ανακτημένες πληροφορίες, και να επικοινωνούν με άλλους διαμεσολαβητές.
- Ø Η λογική θα χρησιμοποιηθεί για να ερμηνευθούν οι ανακτημένες πληροφορίες και να εξαχθούν συμπεράσματα.

## ΚΕΦΑΛΑΙΟ 2

### 2.1 Περίληψη

Οι οντολογίες έχουν κινηθεί πέρα από τις περιοχές της βιβλιοθήκης των επιστημών, της φιλοσοφίας, και της αντιπροσώπευσης γνώσης. Γι' αυτές ενδιαφέρονται πλέον τα εμπορικά τμήματα, τα CEOs και οι επικρατέστερες επιχειρήσεις. Έρευνες επιχειρήσεων που ασχολούνται με αναλύσεις όπως η έρευνα Forrester υποβάλλουν έκθεση σχετικά με τους κρίσιμους ρόλους των οντολογιών υπέρ της φυλλομέτρησης και της αναζήτησης του ηλεκτρονικού εμπορίου και υπέρ της διαλειτουργικότητας της διευκόλυνσης της διαχείρισης γνώσης και της διαμόρφωσης. Στην πρώτη περίπτωση τώρα, βλέπουμε τις οντολογίες που χρησιμοποιούνται ως κεντρικά ελεγχόμενα λεξιλόγια που είναι ενσωματωμένα στους καταλόγους, στις βάσεις δεδομένων, στις δημοσιεύσεις του internet, στις εφαρμογές διαχείρισης γνώσης, κ.λπ. Οι μεγάλες οντολογίες είναι ουσιαστικά συστατικά σε πολλές online εφαρμογές συμπεριλαμβανομένης και της αναζήτησης (όπως Yahoo και Lycos), του ηλεκτρονικού εμπορίου (όπως το Amazon και το eBay), της διαμόρφωσης (όπως η Dell και η PC-Order), κ.λπ.

Στην άλλη περίπτωση επίσης, βλέπουμε τις οντολογίες που έχουν μακροχρόνια διάρκεια ζωής, μερικές φορές, στα πολλαπλά προγράμματα (όπως UMLS, κώδικες SIC, κ.λπ.). Τέτοια διαφορετική χρήση δημιουργεί πολλές επιπτώσεις στο περιβάλλον της οντολογίας.

Παρακάτω, θα συζητήσουμε για τις οντολογίες και τις σύγχρονες απαιτήσεις τους στο Web σήμερα. Θα περιγράψουμε μερικές επιθυμητές ιδιότητες των οντολογιών. Θα συζητήσουμε επίσης για τις απλές και τις σύνθετες οντολογίες και τον τρόπο που μπορούν να χρησιμοποιηθούν για να υποστηρίξουν τις ποικίλες εφαρμογές. Θα ολοκληρώσουμε με μια συζήτηση των ανερχόμενων τάσεων στις οντολογίες και τα περιβάλλον τους και θα αναφέρουμε εν συντομία στην εξέλιξη της οντολογίας μας.

### **2.1.1 Εισαγωγή: Οι αυξανόμενες ανάγκες του Web**

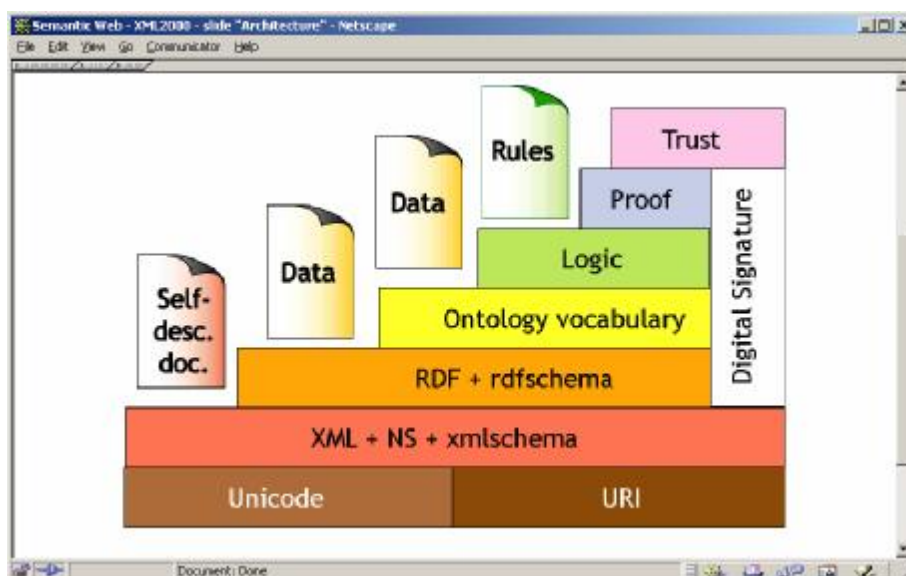
Μπορούμε να είμαστε συγκρατημένοι για την επόμενη σημαντική εξέλιξη του online περιβάλλοντος. Τις πρώτες ημέρες του Web, οι σελίδες HTML δημιουργούνταν με το χέρι. Οι σελίδες περιείχαν τις πληροφορίες για το τον τρόπο παρουσίασης των πληροφοριών σε μια σελίδα. Τα πρόωρα adopters επιδόθηκαν στο Web γρήγορα δεδομένου ότι παρείχαν μια κατάλληλη μέθοδο για τη διανομή πληροφοριών. Αναμφισβήτητα, η παραγωγή των εργαλείων για την παραγωγή μηχανών και τη διαχείριση ιστοσελίδων επέτρεψαν στο Web να απογειωθεί πραγματικά. Οι πλατφόρμες εργαλείων επέτρεψαν στους μη τεχνικούς να παραγάγουν και να δημοσιεύσουν ιστοσελίδες γρήγορα και εύκολα. Οι παραγόμενες σελίδες περιέλαβαν περιεχόμενο και πληροφορίες που στόχευαν σε ανθρώπινους αναγνώστες (παρά στη στοχοθέτηση των προγραμμάτων ή των αυτόματων αναγνωστών). Το Web συνεχίζει να αυξάνεται σε ένα εκπληκτικό ποσοστό με τις ιστοσελίδες να διαδίδονται ολοκληρωμένες και να ενσωματώνονται σε πολλές πτυχές των επιχειρήσεων και της προσωπικής ζωής. Ωστόσο οι ιστοσελίδες διατηρούν ένα μεγάλο μέρος του χαρακτήρα τους που έχει σαν στόχο την ανθρώπινη κατανάλωση. Κατά συνέπεια, εφαρμογές όπως η αναζήτηση απαιτούν ακόμα από τους ανθρώπους να αναθεωρήσουν τις σελίδες αποτελεσμάτων προκειμένου να βρεθεί η σωστή απάντηση στις ερωτήσεις τους.

Ενώ η πρόοδος των μηχανών αναζήτησης όπως το Google βελτιώνει την κατάσταση, οι περισσότεροι άνθρωποι συμφωνούν ότι για να βρουν τις ακριβείς πληροφορίες που αναζητούν, το Web δεν είναι τόσο απλό όσο κάποιος θα ήλπιζε. Ένας λόγος που συμβαίνει αυτό είναι ότι οι απαντήσεις στις ερωτήσεις αναζήτησης μπορεί να βρίσκεται σε μεγάλους καταλόγους γεμάτους με απαντήσεις για το ερώτημα αυτό. Οι απαντήσεις σπάνια είναι ακριβώς στο τμήμα της σελίδας που η μηχανή αναζήτησης "νομίζει" ότι περιλαμβάνει την απάντηση στην ερώτηση. Επιπλέον, τα χαρακτηριστικά των ιστοσελίδων δεν περιέχουν τις επιπλέον πληροφορίες για το περιεχόμενο της σελίδας. Εάν οι σελίδες περιείχαν αυξημένες πληροφορίες σχετικά με το τι περιεχόμενο και τι υπηρεσίες μπορούσαν να ληφθούν (και πως μπορούν να ληφθούν), τότε οι σελίδες θα μπορούσαν να χρησιμοποιηθούν πιο αποτελεσματικά από ένα πρόγραμμα που επιστρέφει ακριβώς στο τμήμα εκείνο της σελίδας που περιλαμβάνει την συγκεκριμένη απάντηση σε μία ερώτηση. Μόλις στοχεύσουν οι ιστοσελίδες για τη κατανάλωση μηχανής ή

κατανάλωση προγράμματος, αντί της ανθρώπινης κατανάλωσης, η επόμενη γενεά του Web μπορεί να πραγματοποιηθεί. Ο πολλαπλασιασμός των γλωσσών σήμανσης που στοχεύουν να αυξήσουν το περιεχόμενο και τις υπηρεσίες αντί ακριβώς της απλής παρουσίασης πληροφοριών μπορεί να αντιμετωπισθεί ως υποστήριξη για αυτήν την θέση. Οι γλώσσες σήμανσης όπως XML, RDF [Lass98], RDFS [BG00], DAML [HMcG00], κ.λπ. γίνονται περισσότερο αποδεκτές από τους χρήστες και οι υπεύθυνοι για την ανάπτυξη εφαρμογών βλέπουν την ανάγκη για την μεγαλύτερη κατανόηση τους.

Η άποψη που παρουσιάζεται σε αυτό το έγγραφο είναι σύμφωνη με το όραμα που υποβάλλεται από τον Tim Berners-Lee της W3C κοινοπραξίας. Σε μια ευρέως αναφερόμενη παρουσίαση [Ber00] στη διάσκεψη XML το 2000 ο Berners-Lee παρουσίασε το όραμά του, το Semantic Web ως μία επεξεργάσιμη μηχανή.

Πιστεύουμε ότι η επόμενη εξέλιξη του Web απαιτεί από τις μηχανές να καταλάβουν το περιεχόμενο των σελίδων, τι είδους πληροφορίες μπορούν να ληφθούν από αυτές και τι σημαίνουν αυτές οι πληροφορίες. Οι γλώσσες σήμανσης επιτρέπουν την προδιαγραφή αυτών των πληροφοριών. Ο Berners-Lee πρόσφερε ένα διάγραμμα αρχιτεκτονικής στην παρουσίασή του που αποτελεί ένα καλό θεμέλιο και παρουσιάζεται στο σχήμα 2.1 :



Σχήμα 2.1 : Μοντέλο Αρχιτεκτονικής του Berners-Lee



Το μοντέλο αυτό παρουσιάζει γλώσσες σήμανσης στη βάση (ακριβώς επάνω από Unicode) για τη χρήση στους λεπτομερείς όρους (ή στο Web προσδιορισμός πηγών γνώσης). Το επόμενο στρώμα και αυτό που θα εξετάσουμε εδώ, είναι το στρώμα οντολογίας. Σε αυτό το στρώμα, μπορούμε να καθορίσουμε τους όρους και τις σχέσεις τους με άλλους όρους. Το επόμενο στρώμα είναι το στρώμα λογικής. Σε αυτό το στρώμα, μπορούμε να συναγάγουμε τις πληροφορίες, με τέτοιο τρόπο ώστε να μας επιτραπεί να συναγάγουμε τις επιπτώσεις των ορισμών και των σχέσεων των όρων. Στο υπόλοιπο αυτού του εγγράφου, θα συζητήσουμε για τα στρώματα οντολογίας και λογικής, για το τι έχουν έρθει να επισημάνουν στο Web, και πώς κάποιος μπορεί να παραγάγει τις οντολογίες και να τις χρησιμοποιήσει στις εφαρμογές.

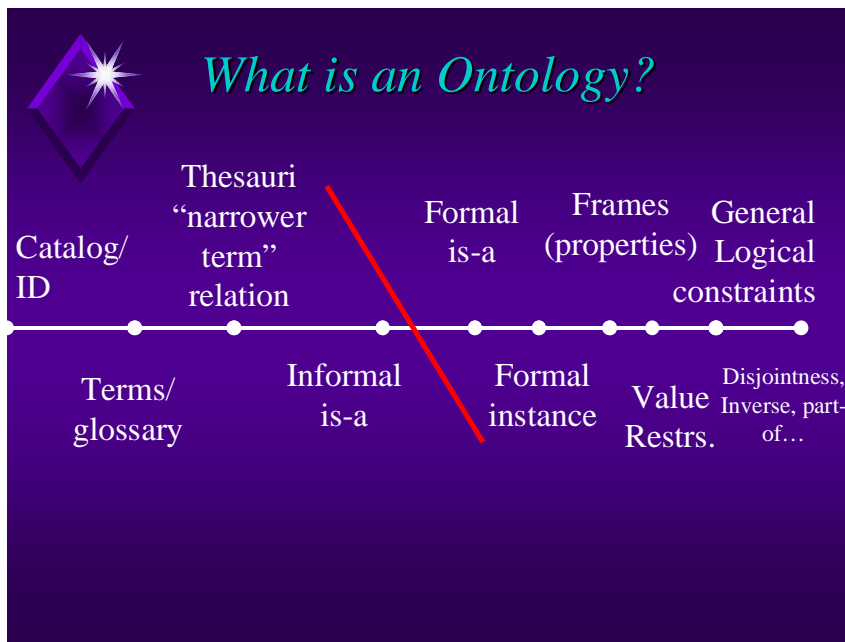
### **2.1.2 Οι οντολογίες**

Ο όρος οντολογία χρησιμοποιείται πολλά χρόνια. Το Merriam Webster, παραδείγματος χάριν, δημιούργησε την οντολογία circa το 1721 και δίνει δύο ορισμούς (1) ένας κλάδος της μεταφυσικής ενδιαφερόμενος για τη φύση και τις σχέσεις της ύπαρξης και (2) μια ιδιαίτερη θεωρία για τη φύση της ύπαρξης ή τα είδη ύπαρξης. Αυτοί οι ορισμοί παρέχουν μια αφηρημένη φιλοσοφική έννοια της οντολογίας. Οι μαθηματικές ή επίσημες οντολογίες έχουν γραφτεί επίσης πολύ καιρό πριν. Ο Smith το 1998 επισημαίνει ότι τουλάχιστον από το 1900, η έννοια μιας επίσημης οντολογίας έχει διακριθεί από την επίσημη λογική από το φιλόσοφο Husserl. Ενώ οι οντολογίες (ακόμη και επίσημες οντολογίες) είχαν μια μακροχρόνια ιστορία, παρέμειναν ως θέμα ακαδημαϊκού ενδιαφέροντος μεταξύ των φιλοσόφων, των γλωσσολόγων, των βιβλιοθηκάρων, και των ερευνητών αντιπροσώπευσης γνώσης μέχρι κάπως πρόσφατα.

Οι οντολογίες έχουν κερδίσει το ενδιαφέρον και την αποδοχή από υπολογιστικά ακροατήρια (εκτός από τα φιλοσοφικά ακροατήρια). Ο Guarino [Gua98] παρέχει μια συμπαθητική συλλογή των τομέων που αγκαλιάζουν τις οντολογίες συμπεριλαμβανομένης της εφαρμοσμένης μηχανικής γνώσης, της αντιπροσώπευσης γνώσης, της ποιοτικής διαμόρφωσης, της γλωσσικής τεχνολογίας, του σχεδίου βάσεων δεδομένων, της ανάκτησης πληροφοριών και της εξαγωγής, της διαχείρισης και της οργάνωσης γνώσης. Εκείνη η συλλογή συνολικά στις αρχές του 1998 δεν περιέλαβε την έμφαση του Web που υπάρχει σήμερα. Περιλάμβανε όμως τους τομείς της επιστήμης

των βιβλιοθηκών, της εμπλουτισμένης οντολογίας αναζήτησης κοινωνικών συναλλαγών (π.χ., eCyc (<http://www.e-Cyc.com/> και FindUR), ενδεχομένως κάποιες από τις μεγαλύτερες ιστοσελίδες του ηλεκτρονικού εμπορίου (π.χ., Amazon.com, Yahoo Shopping, κ.λπ.), και της διαμόρφωσης.

Σε αυτό το έγγραφο, θα περιορίσουμε την αίσθηση μας στις οντολογίες σε εκείνα που βλέπουμε ανερχόμενα στο Web. Η σημερινή χρήση της οντολογίας στον Web έχει ένα διαφορετικό ράπισμα από τις προηγούμενες φιλοσοφικές έννοιες. Ένας ευρέως διαδεδομένος ορισμός για τις οντολογίες είναι του Gruber [ Gru93b ] "ο λεπτομερής όρος μιας αντίληψης". Θα χρησιμοποιήσουμε αυτήν την έννοια και θα επεκταθούμε επάνω στην χρήση του όρου. Οι άνθρωποι (και οι υπολογιστικοί διαμεσολαβητές) τυπικά έχουν κάποια γνώμη ή αντίληψη της έννοιας των όρων. Τα προγράμματα λογισμικού παρέχουν μερικές φορές μια προδιαγραφή των εισαγωγών και των αποτελεσμάτων ενός προγράμματος, το οποίο θα μπορούσε να χρησιμοποιηθεί ως προδιαγραφή του προγράμματος. Ομοίως οι οντολογίες μπορούν να χρησιμοποιηθούν για να παρέχουν μια συγκεκριμένη προδιαγραφή των ονομάτων όρου και των εννοιών όρου. Μέσα στη γραμμή σκέψης όπου μια οντολογία είναι μια προδιαγραφή της σύλληψης ενός όρου, υπάρχουν ακόμα διάφορες πιθανές ερμηνείες. Οι οντολογίες του Web μπορούν να αντιμετωπισθούν ως φάσμα λεπτομέρειας στην προδιαγραφή τους. Ένα απλό (γραμμικό) φάσμα των ορισμών απεικονίζεται στο σχήμα 2.2 παρακάτω.



Σχήμα 2.2: Φάσμα Οντολογιών

Αυτό το φάσμα προέκυψε από μια συνομιλία σε προετοιμασία για μια επιτροπή οντολογίας στο AAAI '99. Η επιτροπή (Lehman, McGuinness, Ushold, και Welty), που επιλέχτηκε λόγω των ετών εμπειρίας τους στις οντολογίες διαπίστωσαν ότι αντιμετώπισαν πολλές μορφές προδιαγραφών από διαφορετικούς ανθρώπους που υπηρέτησαν τις οντολογίες. Ο McGuinness καθόρισε την εικόνα σε αυτή που βλέπουμε παραπάνω.

Μια από τις απλούστερες έννοιες μιας πιθανής οντολογίας μπορεί να είναι ένα ελεγχόμενο λεξιλόγιο - δηλ., ένας πεπερασμένος κατάλογος όρων. Οι κατάλογοι είναι ένα παράδειγμα αυτής της κατηγορίας. Οι κατάλογοι μπορούν να παρέχουν μια σαφή ερμηνεία των όρων - παραδείγματος χάριν, κάθε χρήση ενός όρου, ας πούμε το αυτοκίνητο - θα δείξει ακριβώς το ίδιο προσδιοριστικό - πέστε 25. Μια άλλη πιθανή προδιαγραφή οντολογίας είναι ένα γλωσσάριο (ένας κατάλογος όρων και εννοιών).

Οι έννοιες διευκρινίζονται χαρακτηριστικά ως δηλώσεις φυσικής γλώσσας. Αυτό παρέχει ένα είδος σημασιολογίας ή σημασίας δεδομένου ότι οι άνθρωποι μπορούν να διαβάσουν τις δηλώσεις φυσικής γλώσσας και να τις ερμηνεύσουν. Χαρακτηριστικά, οι ερμηνείες δεν είναι σαφείς και έτσι αυτές οι προδιαγραφές δεν είναι επαρκείς για τους

διαμεσολαβητές των υπολογιστών, κατά συνέπεια αυτό δεν θα ικανοποιούσε τα κριτήρια να είναι επεξεργάσιμη από τις μηχανές.

Οι θησαυροί παρέχουν κάποια πρόσθετη σημασιολογία στις σχέσεις μεταξύ των όρων. Παρέχουν τις πληροφορίες όπως στις σχέσεις συνωνύμου. Σε πολλές περιπτώσεις οι σχέσεις τους μπορούν να ερμηνευθούν με σαφή τρόπο από τους διαμεσολαβητές. Τυπικά οι θησαυροί δεν παρέχουν ρητή ιεραρχία (αν και με το στενότερο και ευρύτερο όρο οι προδιαγραφές, θα μπορούσαν να συναγάγουν μια απλή ιεραρχία). Οι πρόωρες προδιαγραφές του Web των ιεραρχιών όρου, όπως το Yahoo, παρέχουν μια βασική έννοια της γενίκευσης και της ειδίκευσης.

Το Yahoo, παραδείγματος χάριν, παρέχει έναν μικρό αριθμό κορυφαίων κατηγοριών όπως η ενδυμασία και η κατηγορία των φορεμάτων (για τις γυναίκες) ως είδος ενδυμασίας. Ένας μικρός αριθμός ανθρώπων θεωρεί τις προηγούμενες κατηγορίες (καταλόγων, γλωσσαρίων, και θησαυρών) οντολογίες αλλά πολλοί προτιμούν να περιλάβουν μια ρητή ιεραρχία προτού να θεωρηθεί κάτι οντολογία. Το Yahoo, παραδείγματος χάριν, παρέχει μια ρητή ιεραρχία. Η ιεραρχία της δεν είναι μια ακριβή υποκατηγορία ή ιεραρχία "isa". Αυτό το σημείο διακρίθηκε στη φωτογραφική διαφάνεια φάσματος δεδομένου ότι φαίνεται να συλλαμβάνει πολλές από τις φυσικές εμφανιζόμενες ταξινομίες στο Web.

Σε αυτά τα σχέδια οργάνωσης, είναι χαρακτηριστική η περίπτωση μιας πιο συγκεκριμένης κατηγορίας μέσα σε μία μεγαλύτερη κατηγορία. Παραδείγματος χάριν, η κατηγορία γενικής ενδυμασίας περιλαμβάνει την υποκατηγορία γυναικεία (που πρέπει να είναι ακριβέστερα ενδυμασία των με τον τίτλο γυναικών) που περιλαμβάνει έπειτα τις υποκατηγορίες αξεσουάρ και φορέματα. Ενώ συμβαίνει ότι κάθε περίπτωση ενός φορέματος είναι μια περίπτωση ενδυμασίας (και πιθανώς μια περίπτωση γυναικείου φορέματος), δεν είναι η περίπτωση ότι ένα φόρεμα είναι γυναίκα και δεν είναι επίσης περίπτωση ότι ένα άρωμα (μια περίπτωση ενός αξεσουάρ των γυναικών) είναι μια περίπτωση ενδυμασίας. Αυτή η μίξη των κατηγοριών όπως τα εξαρτήματα στα σχέδια ταξινόμησης στο Web δεν είναι μοναδική στο Yahoo - εμφανίζεται σε πολλά σχέδια ταξινόμησης στο Web. Χωρίς αληθινές σχέσεις υποκατηγοριών (ή αληθινό "isa"), θα δούμε ότι ορισμένα είδη παραγωγικών χρήσεων των οντολογιών γίνονται προβληματικά.

Το επόμενο σημείο περιλαμβάνει τις ακριβείς ιεραρχίες υποκατηγοριών. Σε αυτά τα συστήματα εάν το A είναι υποκατηγορία του β, κατόπιν εάν ένα αντικείμενο είναι μια περίπτωση του β απαραίτητως ακολουθεί ότι το αντικείμενο είναι μια περίπτωση του A. Παραδείγματος χάριν, εάν "το φόρεμα" είναι μια υποκατηγορία "της ενδυμασίας" και "το αγαπημένο μου φόρεμα" είναι μια περίπτωση "φορέματος", κατόπιν ακολουθεί ότι "το αγαπημένο μου φόρεμα" είναι μια περίπτωση "ενδυμασίας". Οι ακριβείς ιεραρχίες υποκατηγοριών είναι απαραίτητες για την εκμετάλλευση της κληρονομιάς. Το επόμενο σημείο στο φάσμα οντολογίας περιλαμβάνει τις επίσημες περιπτώσεις σχέσεων. Μερικά σχέδια ταξινόμησης περιλαμβάνουν μόνο τα ονόματα κατηγορίας ενώ άλλοι περιλαμβάνουν τα βασικά χαρακτηριστικά περιεχόμενα. Αυτό το σημείο περιλαμβάνει επίσης περιπτώσεις.

Το επόμενο σημείο περιλαμβάνει τα πλαίσια. Εδώ οι κατηγορίες περιλαμβάνουν τις πληροφορίες ιδιοκτησίας. Παραδείγματος χάριν, η κατηγορία "ενδυμασίας" μπορεί να περιλάβει τις ιδιότητες "της τιμής" και "isMadeFrom". Το συγκεκριμένο φόρεμά μου μπορεί να έχει μια τιμή \$100 και μπορεί να γίνει από το βαμβάκι. Οι ιδιότητες γίνονται πιο χρήσιμες όταν διευκρινίζονται σε γενικό επίπεδο κατηγορίας και έπειτα κληρονομούνται με συνέπεια από τις υποκατηγορίες και τις περιπτώσεις. Σε μια καταναλωτική ιεραρχία, μια γενική κατηγορία όπως το καταναλωτικό προϊόν μπορεί να έχει μια σύνδεση «τιμής» με αυτό. Ενδεχομένως η ενδυμασία θα ήταν η γενική κατηγορία στην οποία η ιδιότητα "isMadeFrom" συνδέεται. Αυτό θα σήμαινε ότι το πεδίο "isMadeFrom" είναι ενδυμασία. Όλες οι υποκατηγορίες αυτών των κατηγοριών θα κληρονομούσαν αυτές τις ιδιότητες.

Ένα πιο εκφραστικό σημείο στο φάσμα οντολογίας περιλαμβάνει τους περιορισμούς αξίας. Εδώ μπορούμε να τοποθετήσουμε τους περιορισμούς σε αυτό που μπορεί να γεμίσει μια ιδιότητα. Παραδείγματος χάριν, μια ιδιότητα "τιμών" να περιοριστεί για να έχει ενδιάμεσο κενό που είναι ένας αριθμός (ή ένας αριθμός σε μια ορισμένη σειρά) και η ιδιότητα "isMadeFrom" μπορεί να περιοριστεί και να έχει ένα ενδιάμεσο κενό που είναι ένα είδος υλικού. Εδώ μπορούμε να δούμε ένα πιθανό πρόβλημα με ένα σχέδιο ταξινόμησης που δεν υποστηρίζει ακριβές "isa" ή τις σχέσεις υποκατηγοριών. Παραδείγματος χάριν, εάν "το άρωμα" ήταν μια υποκατηγορία "της ενδυμασίας", θα

κληρονομούσε την ιδιότητα "isMadeFrom" και τον περιορισμό αξίας του υλικού που δηλώθηκε.

Δεδομένου ότι οι οντολογίες πρέπει να εκφράσουν περισσότερες πληροφορίες, οι εκφραστικές απαιτήσεις τους αυξάνονται. Παραδείγματος χάριν, μπορούμε να θελήσουμε να συμπληρώσουμε την αξία μιας ιδιοκτησίας βασισμένης σε μια μαθηματική εξίσωση χρησιμοποιώντας τις τιμές από άλλες ιδιότητες. Μερικές γλώσσες επιτρέπουν στα οντολόγους να δηλώσουν τις αυθαίρετες λογικές δηλώσεις. Οι πολύ εκφραστικές γλώσσες οντολογίας όπως αυτή που βλέπει σε Ontolingua ή CycL επιτρέπουν στα οντολόγους να διευκρινίσουν τους πρώτη σειρά λογικών περιορισμών μεταξύ των όρων και πιο λεπτομερείς σχέσεις όπως ο διαχωρισμός των κατηγοριών, διαχωρισμός των καλυμμάτων, τις αντίστροφες σχέσεις, τις part-hole σχέσεις, κ.λπ.

Σε αυτό το έγγραφο, θα απαιτήσουμε οι ακόλουθες ιδιότητες να διατηρηθούν προκειμένου να θεωρηθεί κάτι σαν οντολογία. Οι προδιαγραφές που συναντούν αυτές τις ιδιότητες θα αναφερθούν ως απλές οντολογίες

- Ø πεπερασμένο ελεγχόμενο (εκτατό) λεξιλόγιο
- Ø σαφής ερμηνεία των κατηγοριών και των σχέσεων όρου
- Ø ακριβείς ιεραρχικές σχέσεις υποκατηγοριών μεταξύ των κατηγοριών

θεωρούμε τις ακόλουθες ιδιότητες χαρακτηριστικές αλλά μη υποχρεωτικές:

- Ø προδιαγραφή ιδιοκτησίας σε μία βάση ανά-κατηγορία
- Ø μεμονωμένος συνυπολογισμός στην οντολογία
- Ø προδιαγραφή περιορισμού αξίας οντολογίας σε μια βάση ανά-κατηγορία

τέλος, οι ακόλουθες ιδιότητες μπορεί να είναι επιθυμητές αλλά όχι υποχρεωτικές ούτε χαρακτηριστικές

- Ø προδιαγραφή διαχωρισμού των κατηγοριών
- Ø προδιαγραφή των αυθαίρετων λογικών σχέσεων μεταξύ των όρων
- Ø διακεκριμένες σχέσεις όπως αντίστροφες και part-hole

Η γραμμή στο διάγραμμά μας έχει δημιουργηθεί έτσι ώστε όλα όσα είναι δεξιά της να ονομάζονται οντολογίες και να ικανοποιήσουν τουλάχιστον τους πρώτους τρεις όρους που δηλώνονται παραπάνω. Επιπλέον, όλα όσα είναι δεξιά του μπορούν να χρησιμοποιηθούν ως μια βάση για την εξαγωγή συμπεράσματος.

### **2.1.2.1 Απλές οντολογίες και οι χρήσεις τους**

Θα εξετάσουμε τώρα τις οντολογίες και τον αντίκτυπό τους στις εφαρμογές. Χωρίζουμε το τμήμα αυτό σε δύο μέρη: σε χρήσεις των απλών οντολογιών και σε χρήσεις για τις περιπλοκότερες οντολογίες. Το κάνουμε αυτό επειδή γνωρίζουμε ότι η οικοδόμηση των πιο περίπλοκων οντολογιών μπορεί να έχει υπερβολικό κόστος για ορισμένες εφαρμογές.

Οι απλές οντολογίες δεν είναι τόσο δαπανηρές για να οικοδομηθούν και υπάρχουν και πολλές διαθέσιμες. Οι απλές οντολογίες διατίθενται με πολλές μορφές. Πολλές υπάρχουν και σαν δωρεάν λογισμικό στο Web σήμερα, ή ακόμη και ως εσωτερικές δομές οργάνωσης πληροφοριών μέσα στις επιχειρήσεις, τα πανεπιστήμια, κ.λπ. Έχουν παρατηρηθεί κάποιες προσπάθειες όπως οι DMOZ ([www.dmoz.com](http://www.dmoz.com)) οι οποίες παράγουν μεγάλες απλές οντολογίες. Οι DMOZ, παραδείγματος χάριν, έχουν δυνάμεις άνω των 35.000 εθελοντικών συντακτών και στο χρόνο δημοσιεύσεων, είχαν πάνω από 360.000 κατηγορίες σε μια ταξινομία. Επιπλέον, διαθέσιμες σήμερα υπάρχουν και κάποιες λίγο πιο περίπλοκες οντολογίες. Παραδείγματος χάριν, το ενοποιημένο ιατρικό γλωσσικό σύστημα (UMLS - <http://www.nlm.nih.gov/research/umls/kaj>) που αναπτύσσεται από την εθνική βιβλιοθήκη της ιατρικής είναι μια μεγάλη περίπλοκη οντολογία για την ιατρική ορολογία. Μερικές επιχειρήσεις όπως οι Cycorp ([www.cyc.com](http://www.cyc.com)) παρέχουν τμήματα μεγάλων, λεπτομερών οντολογιών. Το ζήτημα της απόκτησης και της συντήρησης οντολογίας θα αναλυθεί παρακάτω αργότερα, αλλά προς το παρόν, θελήσαμε να τονίσουμε ότι πολλές απλές και μερικές περίπλοκες οντολογίες βρίσκονται σε μεγάλη διαθεσιμότητα σήμερα.

Τώρα θα εξετάσουμε μερικούς από τους τρόπους που οι απλές οντολογίες μπορούν να χρησιμοποιηθούν στην πράξη. Κατ' αρχάς, παρέχουν ένα ελεγχόμενο λεξιλόγιο. Αυτό από μόνο του μπορεί να παρέχει μεγάλη δύναμη στους χρήστες, τους συντάκτες και στις βάσεις δεδομένων αφού, μπορούν όλοι να χρησιμοποιήσουν τους όρους από το ίδιο λεξιλόγιο. Επιπλέον τα προγράμματα μπορούν να δημιουργήσουν τις διεπαφές που ενθαρρύνουν τη χρήση των ελεγχόμενων όρων. Το αποτέλεσμα είναι ότι οι άνθρωποι χρησιμοποιούν το ίδιο σύνολο όρων. Φυσικά, μερικοί από τους όρους μπορούν ακόμα να χρησιμοποιηθούν με διαφορετικές ιδιότητες, αλλά η κοινή χρήση του όρου είναι μια έναρξη για τη διαλειτουργικότητα.

Δεύτερον, μια απλή ταξινόμηση μπορεί να χρησιμοποιηθεί για την υποστήριξη οργάνωσης και κυβέρνησης περιοχών. Πολλοί ιστοχώροι (web sites) εκθέτουν σήμερα στην αριστερή πλευρά μιας σελίδας τα κορυφαία επίπεδα μιας γενικευμένης ιεραρχίας των όρων. Οι κατηγορίες είναι χαρακτηριστικά δελεαστικές και ένας χρήστης μπορεί να τις επιλέξει να επεκτείνει τις υποκατηγορίες.

Τρίτον, οι ταξινομήσεις μπορούν να χρησιμοποιηθούν για να υποστηρίξουν το προσδοκώμενο περιβάλλον. Ένα σημαντικό χαρακτηριστικό γνώρισμα των αλληλεπιδράσεων των χρηστών, είναι να είναι σε θέση να έχουν ρεαλιστικές προσδοκίες για ένα site. Εάν μπορούν να ερευνήσουν ακόμη και τις κορυφαίες κατηγορίες επιπέδων της ιεραρχίας, μπορούν γρήγορα να καθορίσουν εάν το περιεχόμενο του site τους ενδιαφέρει ή όχι.

Τέταρτο, οι ταξινομήσεις μπορούν να χρησιμοποιηθούν ως δομή "ομπρελών" από τις οποίες μπορεί να επεκταθεί το περιεχόμενο. Μερικές δωρεάν διαθέσιμες οντολογίες προσπαθούν να παρέχουν μια ταξινομική οργάνωση υψηλού επιπέδου από την οποία πολλές προσπάθειες μπορούν να κληρονομήσουν τους όρους τους. Το UNSPSC (καθολική τυποποιημένη ταξινόμηση προϊόντων και υπηρεσιών [www.unspsc.org](http://www.unspsc.org)) είναι ένα τέτοιο σχήμα κατηγοριοποίησης. Έγινε από κοινού από το πρόγραμμα ανάπτυξης Ηνωμένων Εθνών και το Dun & Bradstreet και στόχευσε στην παροχή της υποδομής για τη διαλειτουργικότητα των όρων στις περιοχές των προϊόντων και των υπηρεσιών. Παρέχει ένα σχέδιο ταξινόμησης (με τους σχετικούς αριθμούς). Παραδείγματος χάριν, η κατηγορία 50 - τα τρόφιμα, το ποτό, και τα προϊόντα καπνού έχουν μια υποκατηγορία, την οικογένεια 5010, που καλείται "φρούτα και λαχανικά και καρύδια και σπόροι". Αυτή στη συνέχεια περιέχει μια υποκατηγορία, την 501015 που καλείται "λαχανικά", η οποία βέβαια περιέχει στη συνέχεια την υποκατηγορία 50101538 που καλείται "φρέσκα λαχανικά". Διάφορες εφαρμογές ηλεκτρονικού εμπορίου ψάχνουν τέτοιες δομές οργάνωσης ομπρελών σήμερα και στην πραγματικότητα ένας αριθμός απ' αυτές έχει επιλέξει να είναι συμμορφωμένος με το UNSPSC. Οι περισσότερες εφαρμογές θα πρέπει να επεκτείνουν αυτές τις οντολογίες, αλλά εάν οι εφαρμογές πρέπει να επικοινωνούν με τους διάφορους ικανοποιημένους προμηθευτές, είναι απαραίτητο να χρησιμοποιηθεί μια κοινή οντολογία ανώτερου επιπέδου.



Πέμπτον, οι ταξινομίες μπορούν να παρέχουν υποστήριξη φυλλομέτρησης. Το περιεχόμενο ενός site μπορεί να υποστηριχθεί με όρους της ταξινόμιας. Αυτό μπορεί να γίνει μηχανικά στο Yahoo ή και αυτόματα, ενδεχομένως, χρησιμοποιώντας μια προσέγγιση συγκέντρωσης. Μόλις μετα-κολληθεί μια σελίδα (ή υπηρεσία) με έναν όρο που επιλέγεται από ένα ελεγχόμενο λεξιλόγιο, οι μηχανές αναζήτησης μπορούν να εκμεταλλευτούν την επικόλληση και να παρέχουν ενισχυμένες ικανότητες αναζήτησης.

Έκτον, οι ταξινομίες μπορούν να χρησιμοποιηθούν για να παρέχουν και υποστήριξη αναζήτησης. Μια μέθοδος επέκτασης ερώτησης μπορεί να χρησιμοποιηθεί προκειμένου να επεκταθεί μια ερώτηση των χρηστών με όρους από πιο συγκεκριμένες κατηγορίες στην ιεραρχία. Δοκιμάσαμε αυτήν την προσέγγιση στην εργασία μας FindUR και διαπιστώσαμε ότι υπό ορισμένες συνθήκες (όπως το σύντομο μήκος εγγράφων και οι περιορισμένες ικανοποιημένες περιοχές), η επέκταση της ερώτησης μπορεί ριζικά να βελτιώσει την αναζήτηση.

Έβδομον, οι ταξινομίες μπορούν να χρησιμοποιηθούν και στην υποστήριξη αποσαφήνισης αίσθησης. Εάν ο ίδιος όρος εμφανίζεται σε πολλές θέσεις σε μια ταξινόμια, μια εφαρμογή μπορεί να κινηθεί προς ένα γενικότερο επίπεδο στην ταξινόμια αυτή προκειμένου να βρεθεί η αίσθηση της λέξης. Παραδείγματος χάριν, εάν μια οντολογία περιέχει τις πληροφορίες ότι ο Jordan είναι μια περίπτωση ενός καλαθοσφαιριστή και επίσης μια περίπτωση μιας χώρας, μια εφαρμογή μπορεί να επιλέξει να ρωτήσει έναν χρήστη που ψάχνει για τον Jordan εάν ενδιαφέρεται για τον καλαθοσφαιριστή ή για την χώρα.

### **2.1.2.2 Δομημένες οντολογίες και οι χρήσεις τους**

Μέχρι αυτό το σημείο, έχουμε εστιάσει την προσοχή μας στις απλές ταξινομίες και στη χρήση αυτών στις εφαρμογές. Μόλις όμως αρχίσει να μεγαλώνει η δομή των οντολογιών, μπορούν εύκολα να προωθήσουν περισσότερη εξουσία στις εφαρμογές. Επιπλέον μόλις αποκτήσουν μεγαλύτερη δομή από τις απλές συνδέσεις γενίκευσης, οι πληροφορίες των ιδιοτήτων μπορούν να χρησιμοποιηθούν με πολλές μορφές. Κατ' αρχάς, μπορούν να χρησιμοποιηθούν για τα απλά είδη ελέγχου συνέπειας. Εάν οι οντολογίες περιέχουν τις πληροφορίες για τις ιδιότητες και τους περιορισμούς αξίας των ιδιοτήτων, τότε ο έλεγχος των τύπων μπορεί να γίνει μέσα στις εφαρμογές.

Παραδείγματος χάριν, έστω ότι έχουμε μια κατηγορία αποκαλούμενη ως "αγαθό" η οποία έχει μια ιδιότητα που αποκαλείται "τιμή" και έναν αριθμητικό περιορισμό αξίας. Εάν κάτι που ανήκει στην κατηγορία αυτή κοστολογηθεί με κάτι που δεν είναι αριθμός τότε μπορεί να θεωρηθεί σφάλμα. Αυτό χρησιμοποιεί τους περιορισμούς των απλών τιμών οι οποίοι είναι τύποι. Ένας περιορισμός τιμής περιλαμβάνει ένα πεδίο ορισμού, παραδείγματος χάριν, έναν αριθμό μεταξύ του 10 και του 100. Κατόπιν εάν η "τιμή" είναι 10.000, είναι εκτός πεδίου ορισμού και μπορεί να καθοριστεί ως ένα λάθος.

Οι δεύτερες οντολογίες μπορούν να χρησιμοποιηθούν για να παρέχουν την ολοκλήρωση. Μια εφαρμογή μπορεί να λάβει ένα μικρό ποσό πληροφοριών από έναν χρήστη, όπως το γεγονός ότι ψάχνει μια υψηλής ευκρίνειας οθόνη σε ένα PC, και έχει έπειτα την οντολογία για να επεκτείνει το ακριβές pixel που πρέπει να αναμένεται. Αυτό μπορεί να γίνει απλά με τον καθορισμό του τι είναι ο όρος "HighResolutionPc" όσον αφορά μια ιδιαίτερη σειρά pixel σε δύο ρόλους - "verticalResolution" και "horizontalResolution". Ομοίως, οι πληροφορίες μπορούν να αλληλεπιδράσουν. Παραδείγματος χάριν, το ιατρικό σύστημα μπορεί να λάβει τις πληροφορίες από μια οντολογία για το εάν ένας ασθενής αρχικά δηλώνεται σαν άτομο, και μετά δηλώνεται το γένος του. Εάν ο ασθενής είναι "αρσενικό" οι πληροφορίες μπορούν να χρησιμοποιηθούν έτσι ώστε η ερώτηση σχετικά με το εάν ο ασθενής είναι ή όχι έγκυος δεν πρέπει να υποβληθεί δεδομένου ότι μπόρεσαν να υπάρξουν πληροφορίες στο σύστημα ότι τα άτομα των οποίων το γένος είναι αρσενικό ξεχωρίζουν από τα άτομα που είναι έγκυρα.

Τρίτον, οι οντολογίες μπορούν να είναι σε θέση να παρέχουν την υποστήριξη διαλειτουργικότητας. Στην απλή περίπτωση η οποία περιλαμβάνει τα ελεγχόμενα λεξιλόγια, υπάρχει ενισχυμένη υποστήριξη διαλειτουργικότητας δεδομένου ότι οι διαφορετικοί χρήστες/εφαρμογές χρησιμοποιούν το ίδιο σύνολο όρων. Στις απλές ταξινομίες, μπορούμε να αναγνωρίσουμε πότε μια εφαρμογή χρησιμοποιεί έναν όρο που είναι γενικότερος ή πιο συγκεκριμένος από έναν άλλο όρο και υπάρχει μεγαλύτερη διευκόλυνση της διαλειτουργικότητας. Στις πιο εκφραστικές οντολογίες, μπορούμε να έχουμε έναν πλήρη λειτουργικό καθορισμό για το πώς ένας όρος αφορά έναν άλλο όρο και έτσι, μπορούμε να χρησιμοποιήσουμε τα αξιώματα ή τις χαρτογραφήσεις ισότητας για να εκφράσουμε έναν όρο ακριβώς από την άποψη του άλλου και με αυτόν τον τρόπο

να υποστηρίξουμε περισσότερη "ευφυή" διαλειτουργικότητα. Παραδείγματος χάριν, μια οντολογία μπορεί να περιλάβει έναν καθορισμό ότι ένας " υπάλληλος του Stanford " είναι ίδιος με ένα "πρόσωπο" του οποίου την ιδιότητα " του εργοδότη" την βρίσκουμε με το "πανεπιστήμιο του Stanford ". Αυτός ο καθορισμός μπορεί να χρησιμοποιηθεί για να επεκτείνει τον όρο " υπάλληλος του Stanford " σε μια εφαρμογή που δεν αναγνωρίζει τους όρους "υπάλληλος του Stanford" ή "υπάλληλος" αλλά καταλαβαίνει τους όρους "πρόσωπο", "εργοδότης", και " Stanford ".

Τέταρτο, οι οντολογίες μπορούν να χρησιμοποιηθούν για να υποστηρίξουν τον έλεγχο επικύρωσης και επαλήθευσης των στοιχείων (και των σχημάτων). Εάν μια οντολογία περιέχει τις περιγραφές κατηγορίας, όπως "υπάλληλος του Stanford", αυτοί οι ορισμοί μπορούν να χρησιμοποιηθούν ως ερωτήσεις στις βάσεις δεδομένων για να ανακαλύψουν ποιο είδος κάλυψης υπάρχει αυτήν την περίοδο στα σύνολα δεδομένων. Παραδείγματος χάριν, εάν κάποιος επρόκειτο να εκθέσει την κατηγορία "υπάλληλος του Stanford" σε μια διεπαφή κάποιας εφαρμογής, θα ήταν χρήσιμο να είναι γνωστό πρώτα εάν το σύνολο των δεδομένων περιέχει οποιεσδήποτε περιπτώσεις "προσώπου" του οποίου η ιδιότητα "εργοδότης" παίρνει την τιμή "πανεπιστήμιο του Stanford ". Επιπλέον, εάν σε ένα απλό πρότυπο στοιχείων, δηλώσουμε ότι ένα "πρόσωπο" είχε το πολύ έναν "εργοδότη", κατόπιν θα μπορούσαμε να χρησιμοποιήσουμε τις πληροφορίες εκείνες που ελέγχουμε για να δούμε εάν οποιεσδήποτε τρέχουσες πληροφορίες για το "πρόσωπο" στο σύνολο δεδομένων περιείχαν περισσότερους από έναν "εργοδότη". Ομοίως, οι έλεγχοι θα μπορούσαν να γίνουν για να δούμε εάν υπήρξαν αυτήν την περίοδο "πρόσωπα" στο σύνολο δεδομένων που ήταν γνωστοί ως "υπάλληλοι" αλλά ακόμα δεν είχαν μια τιμή στην ιδιότητα των "εργοδοτών" (με αυτόν τον τρόπο δείχνουμε ότι το σύνολο δεδομένων δεν είναι πλήρες). Η Chimaera [HMcG00] είναι ένα παράδειγμα εξέλιξης περιβάλλοντος οντολογίας που παρέχει ένα σύνολο διαγνωστικών δοκιμών για τον έλεγχο των οντολογιών και για τα προβλήματα στους ορισμούς οντολογίας καθώς επίσης και για τα προβλήματα με τα στοιχεία των παραδειγμάτων. Ψάχνει τις αποδείξιμες ασυνέπειες καθώς επίσης και τους όρους που "τυπικά" απεικονίζονται σε καταστάσεις όπου μια οντολογία ή κάποια στοιχεία μπορεί να χρειάζονται επίλυση.

Πέμπτον, οι οντολογίες που περιέχουν τις πληροφορίες σήμανσης μπορούν να κωδικοποιήσουν τις δοκιμές ολόκληρων ακολουθιών. Μια οντολογία μπορεί να περιέχει

διάφορους ορισμούς όρων, μερικούς ορισμούς περίπτωσης, και έπειτα να περιλάβει έναν καθορισμό όρου που να θεωρείται ερώτηση - βρείτε όλους τους όρους που ικανοποιούν τους ακόλουθους όρους. Οι πληροφορίες σήμανσης θα μπορούσαν να κωδικοποιηθούν με αυτήν την ερώτηση και να συμπεριλάβουν τι θα πρέπει να είναι η απάντηση, παρέχοντας κατά συνέπεια αρκετές πληροφορίες για την κωδικοποίηση ελεγχόμενων παλλινδρομημένων στοιχείων.

Ένα τέτοιο παράδειγμα οντολογίας μπορούμε να δούμε στη διεύθυνση <http://ksl.stanford.edu/projects/DAML/chimaera-jtp-cardinality-test1.daml>. Η οντολογία περιέχει μια ακολουθία ελέγχου παλινδρόμησης για τον έλεγχο των συμπερασμάτων των αριθμών των στοιχείων του συνόλου (όπως τα πρόσωπα που έχουν δύο εργοδότες που δηλώνονται ακόμα και εάν έχουν το πολύ έναν εργοδότη) σε ένα θεώρημα του Stanford (<http://www.ksl.Stanford.EDU/software/jtp/>).

Έκτον, οι οντολογίες μπορεί να αποτελέσουν τη βάση για την υποστήριξη διαμόρφωσης. Οι κατηγορίες όρων μπορούν να καθοριστούν έτσι ώστε να περιέχουν τις περιγραφές των ειδών που μπορεί να είναι σε ένα σύστημα. Επιπλέον οι αλληλεπιδράσεις μεταξύ των ιδιοτήτων μπορούν να καθοριστούν έτσι ώστε εάν συμπληρώσουμε ένα πεδίο μπορεί να αναγκάσουμε ένα άλλο πεδίο να συμπληρωθεί αυτόματα. Παραδείγματος χάριν, κάποιος μπορεί να παράγει μια οντολογία πληροφοριών για τα προϊόντα των home theatre όπως γίνεται σε ένα μικρό παράδειγμα με τη χρήση ενός απλού συστήματος που βασίζεται στη λογική (logic-based). Οι όροι όπως η τηλεόραση, ο ενισχυτής, ο δέκτης, κ.λπ... καθορίζονται. Επιπλέον, οι πληροφορίες που συνδέουν τους όρους περιλαμβάνονται. Μια κατηγορία υψηλής ποιότητας τηλεοράσεων (HighQualityTelevisions) καθορίζεται έτσι ώστε οι χρήστες να μπορούν να επιλέξουν από αυτήν την κατηγορία και ο διαμορφωτής να συμπληρώνει αυτόματα τα ορισμένα σύνολα κατασκευαστών για να επιλέγει από αυτά, ελάχιστες διαγώνιες τιμές, τις τιμές κ.λπ. Επίσης, οι πληροφορίες που διαδίδουν τους περιορισμούς από ένα συστατικό σε άλλο κωδικοποιούνται. Παραδείγματος χάριν, μερικά από τα προϊόντα αυτού του συστήματος προορίστηκαν για να πωληθούν σε ζευγάρια. Εάν κάποιος αγοράζει ένα ιδιαίτερο καλό ηχείο (που πωλείται μόνο σε ζευγάρια, κατά συνέπεια δύο ηχεία προστίθενται στον κατάλογο), οι περιορισμοί στις ιδιαίτερες στάσεις ηχείων εμφανίζονται στην προδιαγραφή διαμόρφωσης. Υπάρχουν πολλά τέτοια παραδείγματα

διαμόρφωσης που χρησιμοποιούν τις οντολογίες, μερικές από τις οποίες περιγράφονται σε ένα ειδικό ζήτημα διαμόρφωσης της τεχνητής νοημοσύνης για το σχέδιο εφαρμοσμένης μηχανικής, την ανάλυση, και το περιοδικό κατασκευής [Darr98].

Έβδομον, οι οντολογίες μπορεί να υποστηρίξουν τη δομημένη, συγκριτική, και προσαρμοσμένη αναζήτηση. Παραδείγματος χάριν, εάν κάποιος ψάχνει τηλεοράσεις, μπορεί να λάβει μια περιγραφή κατηγορίας για την τηλεόραση από μια οντολογία, να μάθει για τις ιδιότητές της (όπως η διαγώνιος, η τιμή, ο κατασκευαστής, κ.λπ...), και έπειτα να γίνει η σύγκριση των τηλεοράσεων με την παρουσίαση των τιμών και των ιδιοτήτων της κάθε μιας. Οι ιδιότητες αυτές μπορούν επίσης να χρησιμοποιηθούν για να γεμίσουν τα πεδία με τα στοιχεία που έχουμε και να παρουσιάσουν μια λεπτομερή έκθεση με τα στοιχεία που βρήκαν. Αυτό αποτελεί επίσης τη βάση για την παροχή διαφορετικών διεπαφών αναζήτησης - το απλό παράθυρο κειμένου μαζί με την αναζήτηση διεπαφών εκθέτουν σημαντικές ιδιότητες των προϊόντων. Οι περιπλοκότερες οντολογίες μπορούν να παραχθούν από χαρακτηριστικές ιδιότητες που είναι πιο χρήσιμες στο να παρουσιάζουν συγκριτικές αναλύσεις έτσι ώστε οι χρήστες να μπορούν να έχουν συνοπτικές περιγραφές για τα προϊόντα αντί για συγκρίσεις πλήρεις λεπτομερειακές. Κατά συνέπεια, οι οντολογίες με τις πληροφορίες σήμανσης μπορούν επίσης να χρησιμοποιηθούν για να περιορίσουν τις συγκριτικές αναζητήσεις.

Όγδοο, οι οντολογίες μπορούν να χρησιμοποιηθούν για την εκμετάλλευση της γενίκευσης /για την ειδίκευση των πληροφοριών. Εάν μια εφαρμογή αναζήτησης διαπιστώνει ότι η ερώτηση ενός χρήστη παράγει πάρα πολλές απαντήσεις, κάποιος μπορεί να τεμαχίσει την ερώτηση για να δει εάν οποιοδήποτε όροι εμφανίζονται σε μια οντολογία, και σε αυτή την περίπτωση, η εφαρμογή αναζήτησης μπορεί να εστιάσει σε εκείνο τον όρο. Παραδείγματος χάριν, εάν κάποιος έκανε μια αναζήτηση για τις συναυλίες στην περιοχή του κόλπου του Σαν Φρανσίσκο και πήρε πάρα πολλές απαντήσεις, μια μηχανή αναζήτησης μπορεί να ανατρέξει μια συναυλία σε μια οντολογία και να ανακαλύψει ότι υπάρχουν υποκατηγορίες της συναυλίας (και μπορεί επίσης να ανακαλύψει ότι υπάρχουν συγκεκριμένες θέσεις συναυλίας στην περιοχή του κόλπου). Η μηχανή αναζήτησης θα μπορούσε έπειτα να επιλέξει να παρουσιάσει στο χρήστη ένα ιδιαίτερο είδος συναυλίας (έστω μία rock συναυλία). Περαιτέρω η μηχανή αναζήτησης θα μπορούσε φιλενεργά να έχει έτοιμες τις ερωτήσεις στο background περιμένοντας την

εισαγωγή χρηστών ή επίσης τις πληροφορίες από τις πιο δημοφιλείς ερωτήσεις. Κατόπιν η μηχανή αναζήτησης θα μπορούσε επίσης να παρουσιάσει έναν κατάλογο υποκατηγοριών των συναυλιών και να παρέχει στο χρήστη τον κατά προσέγγιση αριθμό ανακτήσεων που ο χρήστης θα έπαιρνε εάν ειδίκευε την ερώτησή του στους διαφορετικούς τρόπους. Αυτοί είναι ακριβώς μερικοί από τους τρόπους με τους οποίους οι οντολογίες μπορούν να χρησιμοποιηθούν για να καθαρίσουν τις ερωτήσεις αναζήτησης.

### **2.1.2.3 Απόκτηση οντολογίας**

Τώρα που τουλάχιστον μπορούν να πειστούν μερικοί αναγνώστες ότι οι οντολογίες είναι χρήσιμα συστατικά στις εφαρμογές, θα εξετάσουμε μερικές πηγές οντολογιών. Κατ' αρχάς, όπως αναφέραμε προηγουμένως, πολλές οντολογίες υπάρχουν στο δημόσιο τομέα. Μπορεί να είναι δυνατό να αρχίσουμε με ένα υπάρχον πρότυπο βιομηχανίας και να το χρησιμοποιήσουμε ως αφετηρία μιας οντολογίας. Οι πλέον πιθανοί υπεύθυνοι για την ανάπτυξη εφαρμογής θα πρέπει να τροποποιήσουν ή/και να επεκτείνουν τις οντολογίες που είναι διαθέσιμες και που αναπτύχθηκαν για άλλες χρήσεις. Ακόμα, μια μεθοδολογία για τη λήψη των οντολογιών πρόκειται να αρχίσει με μια τυποποιημένη οντολογία βιομηχανίας και έπειτα να τη τροποποιήσει ή και να την επεκτείνει. Μια άλλη μεθοδολογία πρόκειται ημι-αυτόματα να παραγάγει μια αφετηρία για μια οντολογία. Πολλές ταξινομικές δομές υπάρχουν στο Web ή στον πίνακα περιεχομένων των εγγράφων. Κάποιος μπορεί να ψάξει σε ορισμένες περιοχές για να λάβει μια αρχική ταξινομική δομή και έπειτα να την αναλύσει, να την τροποποιήσει, και να την επεκτείνει.

Μια καλή ερώτηση είναι πού να ψάξουμε για υπάρχουσες οντολογίες ή για πηγές πληροφοριών. Πολλά ελεγχόμενα λεξιλόγια παρέχονται σήμερα. Μερικές φορές οι οργανώσεις προτύπων, όπως οι NIST (το εθνικό ίδρυμα προτύπων και τεχνολογίας - <http://www.nist.gov/>), υποστηρίζουν τις προσπάθειες στην παραγωγή των ελεγχόμενων λεξιλογίων και των οντολογιών. Μερικές κοινοπραξίες κανονίζουν για να παραγάγουν τις οντολογίες. Δείτε, παραδείγματος χάριν, τη RosettaNet (<http://www.rosettanet.org>) στον τομέα της τεχνολογίας πληροφοριών, της ηλεκτρονικής τεχνολογίας, των ηλεκτρονικών συστατικών, και της κατασκευής ημιαγωγών. Δημιουργούν σε βιομηχανικό επίπεδο ανοικτά πρότυπα ηλεκτρονικού εμπορίου και παρέχουν μια γλώσσα

για τις επιχειρησιακές διαδικασίες. Μερικές φορές οι εμπορικές οργανώσεις παρέχουν κατηγορίες ιεραρχιών στα sites τους που μπορούν επίσης να χρησιμοποιηθούν ως πρότυπη δομή ελεγχόμενου λεξιλογίου. Υπάρχουν επίσης ευρείες πηγές κατηγορίας δομών. Ουσιαστικά κάθε περιοχή ηλεκτρονικού εμπορίου κωδικοποιεί σήμερα τουλάχιστον μια ταξινομική οργάνωση των όρων. Τα sites όπως το Amazon στην οργάνωση των πληροφοριών των βιβλίων και της μουσικής τους παρέχουν μια πολύ ευρεία οργάνωση των πληροφοριών. Μια άλλη ανερχόμενη τάση είναι η χρήση των γλωσσών σήμανσης. Μερικές σελίδες σχολιάζονται χρησιμοποιώντας τις γλώσσες σήμανσης όπως XML, RDF, DAML, κ.λπ. Οι σελίδες συμπεριλαμβανομένων των σχολιασμών μπορεί να χρησιμοποιούν όρους σήμανσης από ελεγχόμενα λεξιλόγια. Μερικές βιβλιοθήκες φέρνουν στην επιφάνεια οντολογίες ενδεχομένως της χρήσης τους στη σήμανση. Παραδείγματος χάριν, το πρόγραμμα DAML διατηρεί μια βιβλιοθήκη των οντολογιών DAML στη διεύθυνση <http://www.daml.org/ontologies/>. Ένα μεγάλο μέρος αυτού του τμήματος έχει εισαγάγει την ιδέα της λήψης είτε μιας απλής είτε μίας σύνθετης οντολογίας ως αφετηρία και έπειτα την αναλύει, την τροποποιεί και την διατηρεί στη διάρκεια του χρόνου. Στο επόμενο τμήμα, θα αντιμετωπίσουμε το ζήτημα των επιπτώσεων και των αναγκών από τις εφαρμογές που είναι βασισμένες σε οντολογίες (ontology-based).

#### **2.1.2.4 Συνέπειες και ανάγκες σχετικές με οντολογίες**

Κατά την έναρξη μιας εφαρμογής βασισμένη σε οντολογίες (ontology-based), τα δύο σημαντικά στοιχεία θα είναι γλώσσα και το περιβάλλον.

**Γλώσσα:** Κατά την εξέταση μιας εφαρμογής σχετική με οντολογίες αναπόφευκτα θα προκύψει το ζήτημα της γλώσσας οντολογίας. Μια οντολογία πρέπει να κωδικοποιηθεί σε κάποια γλώσσα. Εάν κάποιος χρησιμοποιεί μια απλή οντολογία, προκύπτουν λίγα ζητήματα. Εντούτοις, εάν κάποιος εξετάζει μια πιο σύνθετη οντολογία, πρέπει να εξεταστεί η εκφραστική δύναμη μιας γλώσσας αντιπροσώπευσης και συλλογισμού. Όπως με οποιοδήποτε που πρόβλημα προκύπτει όταν επιλέγεται μια γλώσσα, πρέπει να είναι επιστημολογικά ικανοποιητική, δηλαδή να είναι σε θέση να εκφράσει τις έννοιες στα πεδία. Παραδείγματος χάριν, εάν κάποιος θέλει να κάνει έλεγχο διακύμανσης σε μια

εφαρμογή ηλεκτρονικού εμπορίου, τότε θα ήταν παράλογο να επιλεχτεί μια απλή γλώσσα που περιέχει μόνο τις σχέσεις των υποκατηγοριών και των περιπτώσεων και δεν περιλαμβάνει τις ιδιότητες προδιαγραφών με τους περιορισμούς αξίας. Υπάρχουν διάφορες υποψήφιες γλώσσες οντολογίας - στην πραγματικότητα υπάρχουν τόσες πολλές που μερικές ερευνητικές προσπάθειες προέκυψαν την τελευταία δεκαετία προκειμένου να παραχθούν πρότυπα προδιαγεγραμμένων γλωσσών (όπως η προσπάθεια KRSS - η σύστημα προδιαγραφών αντιπροσώπευσης γνώσης ανταλλαγής προδιαγραφών (όπως KIF - το σχήμα ανταλλαγής γνώσης που είναι τώρα προτεινόμενα πρότυπα Ansi [KIF]), και τα κοινά πρότυπα διεπαφών προγραμματισμού αίτησης (όπως OKBC - ανοικτή συνδετικότητα βάσεων γνώσεων).

Κάποιος όχι μόνο θέλει να εξετάσει τα αντιπροσωπευτικά κατασκευάσματα σε μια γλώσσα αλλά επίσης θέλει να εξετάσει και τη λογική στην οποία στηρίζεται η γλώσσα. Μερικοί τομείς όπως οι λογικές περιγραφές ([www.dl.kr.org](http://www.dl.kr.org)), το κάνουν αυτό με μια κεντρική εστίαση στο γλωσσικό σχέδιο. Ψάχνουν τις αλλαγές που διατηρούν την εκφραστική δύναμη που απαιτείται από τις εφαρμογές και εξετάζουν επίσης τι χρειάζεται να παρέχουν οι μηχανές συμπεράσματος που μπορούν να παρέχουν τις αφαιρέσεις που βασίζονται σε κατασκευές που αντιπροσωπεύονται στη γλώσσα. Παραδείγματος χάριν, εάν μια γλώσσα υποστηρίζει την έννοια ότι δύο κατηγορίες είναι ξεχωριστές, κατόπιν πρέπει να είναι σε θέση μια μηχανή συλλογισμού να τις χτιστεί έτσι ώστε να επιβάλλει τον περιορισμό ότι οι κατηγορίες είναι χωριστές. Κατά συνέπεια, μια μηχανή συμπεράσματος πρέπει να είναι σε θέση να προειδοποιήσει τον χρήστη για το εάν θα δημιουργεί μια περίπτωση ή μια υποκατηγορία ή δύο ξεχωριστές κατηγορίες.

Επίσης, μια γλώσσα πρέπει να είναι προσιτή με τις υπάρχουσες πλατφόρμες, δηλαδή να είναι τέτοια ώστε οι μη ειδικοί να μπορούν να τη χρησιμοποιήσουν για να κάνουν την εννοιολογική διαμόρφωσή τους. Το Web είναι σαφώς η σημαντικότερη πλατφόρμα με την οποία είναι συμβατή σήμερα, κατά συνέπεια οποιαδήποτε γλωσσική επιλογή πρέπει να είναι σε θέση να χρησιμοποιεί τις πηγές του Web. Επιπλέον, τα συστήματα που βασίζονται στη μορφοποίηση είχαν μια μακροχρόνια ιστορία μέχρι να θεωρηθούν εννοιολογικά εύχρηστα, κατά συνέπεια ένα παράδειγμα μορφοποίησης μπορεί να αξίζει.

Οι γλωσσικές προσπάθειες που βλέπουμε σήμερα προσπαθούν να πάρουν το καλύτερο της έρευνας για την εκφραστική τους δύναμη μαζί με τη δύναμη του συλλογισμού και να



παρέχουν αντιπροσωπευτικές και ισχυρές γλώσσες που ξέρουν τις ιδιότητες συλλογισμού. Το γλωσσικό πρόγραμμα σήμανσης διαμεσολαβητών DARPA, παραδείγματος χάριν, προσπάθησε να πάρει τις ανερχόμενες γλώσσες Web όπως η XML και η RDF και να δημιουργήσει μια γλώσσα που να είναι συμβατή με το σύστημα Web αλλά να πλησιάζει στην ιστορία των 20 ετών των λογικών περιγραφής στην επιλογή των γλωσσικών κατασκευασμάτων μαζί με τα παραδείγματα συλλογισμού. Η προκύπτουσα γλώσσα - DAML+OIL - προσπαθεί να συγχωνεύσει το καλύτερο των υπάρχουσών γλωσσών Web, των λογικών περιγραφής, και των συστημάτων συλλογισμού πλαισίων. Η OIL προσπαθεί να παρέχει μια βαλμένη σε στρώσεις, προσέγγιση στο γλωσσικό σχέδιο.

**Περιβάλλον:** Μια άλλη εκτίμηση είναι πώς να αναλύσεις, να τροποποιήσεις, και να διατηρήσεις μια οντολογία κατά τη διάρκεια του χρόνου.. Υπάρχουν διάφορα απλά εργαλεία οντολογίας διαθέσιμα στο εμπόριο. Μερικές επιχειρήσεις ανάκτησης πληροφοριών όπως η Verity παρέχουν απλούς συντάκτες για παραγωγή ιεραρχιών γενίκευσης και απλών ξεφυλλισμάτων γενίκευσης. Το Verity, παραδείγματος χάριν, έχει παράγει έναν "συντάκτη θέματος" για χρόνια που θα υποστηρίξει τους χρήστες στην παραγωγή των ταξινομιών και τη χρησιμοποίησή τους στις ερωτήσεις αναζήτησης. Οι ερευνητικές προσπάθειες έχουν υπάρξει για πολλά έτη στην παραγωγή κουτιών εργαλείων οντολογίας. Τα αναφερθέντα εργαλεία του πανεπιστημίου του Stanford, η Ontolingua [Farquhar-et-al 1997] και Chimaera [McGuinness-et-al. 2000] είναι ακριβώς δύο παραδείγματα, εντούτοις τα παραδείγματα αφθονούν συμπεριλαμβανομένου του OilEd (<http://img.cs.man.ac.uk/oil/>) από το πανεπιστήμιο του Manchester και του Protégé από το Stanford Medical Informatics είναι για να ονομάσουμε μερικά. Οι υπεύθυνοι για την ανάπτυξη εφαρμογής μπορούν να επιλέξουν τους εμπορικούς προμηθευτές ως προμηθευτές των κουτιών εργαλείων τους, περίπλοκες ερευνητικές εφαρμογές σαν βάση, ή κάτι ενδιάμεσα. Μερικές επιχειρήσεις με τις εκτενείς ανάγκες οντολογίας όπως η VerticalNet (<http://www.verticalnet.com/>) έχουν αναπτύξει τα εργαλεία οντολογίας τους προκειμένου να χτιστούν οι οντολογίες που ικανοποιούν τις ανάγκες ενός περίπλοκου εμπορικού ontologist. Τα εργαλεία τους χτίστηκαν αφού αναλύθηκαν τα υπάρχοντα ερευνητικά πρωτότυπα και έπειτα σχεδιάστηκαν για να

ανταποκριθούν στα εμπορικά πρότυπα που απαιτήθηκαν στις διαφορετικές, τωρινές εφαρμογές ηλεκτρονικού εμπορίου.

Όταν αποφασίσουμε να χρησιμοποιήσουμε ή να χτίσουμε ένα περιβάλλον οντολογίας, υπάρχουν διάφορα ζητήματα που πρέπει να εξεταστούν συμπεριλαμβανομένων των εξής:

- Ø Συνεργασία και διανεμημένη υποστήριξη εργατικών δυναμικών. Μερικά περιβάλλοντα οντολογίας επιτρέπουν στους χρήστες να μοιραστούν ένα περιβάλλον - δηλ., να δει ο ένας το περιβάλλον εργασίας του άλλου. Αυτό μπορεί να είναι ιδιαίτερα χρήσιμο για το Ontolingua, παραδείγματος χάριν, που υποστηρίζει αυτήν την έννοια. Επιπλέον, όταν διανέμονται οι εργαζόμενοι σε διάφορες θέσεις, είναι σημαντικό να υπάρχει ένα περιβάλλον που να επιτρέπει πρόσβαση από πολλές θέσεις. Αυτό γίνεται πιο χαρακτηριστικό σήμερα με τις αρχιτεκτονικές κεντρικών υπολογιστών. Τέλος, η συνεργασία μπορεί να απαιτήσει τον έλεγχο συμπτώσεων, το κλείδωμα, και ένα είδος άδειας και προσαρμογής των συστημάτων
- Ø Αλληλοσύνδεση πλατφόρμων. Δεδομένου ότι οι εφαρμογές ενσωματώνονται στις πιο σύνθετες πλατφόρμες, είναι σημαντικό για το περιβάλλον να είναι σε θέση να διαβάσουν και να γράψουν συμβατά σχήματα, να είναι σε θέση να ενσωματωθούν με τα πολλαπλάσια περιβάλλον υλικού/λογισμικού, κ.λπ. Οι εφαρμογές που βασίζονται σε Java παρέχουν μια κατάλληλη προσέγγιση σε αυτό το πρόβλημα αλλά άλλα συστήματα που υποστηρίζουν πολλά σχήματα εισαγωγής-εξαγωγής, καταλαβαίνουν τα κοινά πρότυπα, και παρέχουν υπηρεσίες μεταφράσεων και χαρτογράφησης που μπορούν να βοηθήσουν την κλίμακα.
- Ø Κλίμακα. Πολλές εφαρμογές οντολογίας μπορεί σήμερα να πρέπει να μετρήσουν μερικά μεγέθη που θα πρέπει να είναι μεγαλύτερα από τις εφαρμογές του παρελθόντος. Είναι σημαντικό να εξεταστεί η κλίμακα από την άποψη του μεγέθους των οντολογιών καθώς επίσης και των αριθμών ταυτόχρονων χρηστών.
- Ø Προσαρμογή. Δεδομένου ότι οι εφαρμογές γίνονται μακρόβιες και επίσης επεκτείνονται σε διαφορετικά περιβάλλοντα ενδεχομένως διεθνώς, είναι σημαντικό να είναι σε θέση να υποστηρίξει πολλές εκδόσεις των οντολογιών. Στα χαρακτηριστικά περιβάλλοντα τεχνολογίας λογισμικού, υπάρχουν συστήματα ελέγχου κώδικα πηγής και προσαρμογής.

- Ø Ασφάλεια. Μερικές εφαρμογές θα έχουν ανάγκες για διαφορετική πρόσβαση στις μερίδες της οντολογίας. Κατά συνέπεια, είναι σημαντικό να υπάρξει ένα περιβάλλον που να μπορεί να εκθέσει τις μερίδες της οντολογίας βασισμένης σε ένα πρότυπο ασφάλειας. Το πρότυπο ασφάλειας μπορεί να πρέπει να υποστηρίξει να διαβάσει και να γράψει την πρόσβαση
- Ø Ανάλυση. Τα περιβάλλοντα αναμένεται να υποστηρίξουν την απόκτηση, την εξέλιξη, και τη συντήρηση των οντολογιών. Κατά συνέπεια, θα ήταν καλό να αναμένονται οι οντολογίες να έχουν τις περιόδους όταν είναι ελλιπείς και ανακριβείς. Η υποστήριξη ανάλυσης που μπορεί να στρέψει την προσοχή του χρήστη στις περιοχές που είναι πιθανό να χρειαστούν την τροποποίηση μπορεί να είναι αρκετά χρήσιμη. Το *chimaera* περιβάλλον οντολογίας, παραδείγματος χάριν, υποστηρίζει διάφορες διαγνωστικές δοκιμές που στοχεύουν στη βοήθεια των χρηστών για να προσδιοριστούν ευαπόδεικτα οι ανακριβείς οντολογίες καθώς επίσης και τα πιθανά ζητήματα των προβλημάτων
- Ø Κύκλος ζωής. Δεδομένου ότι οι οντολογίες γίνονται μεγαλύτερες και μακροχρόνιες, θα αναμενόταν ότι οι υπεύθυνοι για την ανάπτυξη εφαρμογής των οντολογιών θα τις διατηρούσαν κατά τη διάρκεια πολλών ετών. Επιπλέον, μπορούν συνεχώς τις νέες οντολογίες στο σύστημά τους να τις διασύνδεουν με περισσότερα διαφορετικά συστήματα. Κατά συνέπεια, γίνεται σημαντικό να εξεταστεί η υποστήριξη για τα ζητήματα εξέλιξης οντολογίας όπως οι συγχωνεύσεις όρων, τα πολλαπλά διαστήματα ονόματος, τα συστήματα ελέγχου κώδικα πηγής, τα συστήματα συντήρησης αλήθειας, τα εξεταστικά συστήματα παλινδρόμησης, κ.λ.π.
- Ø Ευκολία στη χρήση. Ακόμα κι αν ένα περιβάλλον έχει όλα όσα χρειάζεται ένας υπεύθυνος για την ανάπτυξη εφαρμογής, αλλά είναι δύσκολο για το χρήστη να αποφασίσει πώς να χρησιμοποιήσει τα μέρη του περιβάλλοντος, μπορεί να μην πάρει χρησιμοποιημένα. Κατά συνέπεια τα υλικά κατάρτισης, τα σεμινάρια, η εννοιολογική υποστήριξη διαμόρφωσης, τα ,γραφικά εργαλεία ξεφυλλίσματος, κ.λπ. όλα μπορούν να είναι σημαντικά

## **2.2 Οι Λογικές Περιγραφές(Description Logics) σαν Γλώσσες Οντολογιών για το Semantic Web**

Το όραμα του Semantic Web έχει όλο και μεγαλύτερη απήχηση στον ακαδημαϊκό κόσμο και τη βιομηχανία. Οι λογικές περιγραφές παρουσιάζονται συχνά ως ένα από τα εργαλεία που μπορούν να υποστηρίξουν το Semantic Web και να βοηθήσουν έτσι το όραμά του να γίνει πραγματικότητα. Παρακάτω περιγράφουμε τις λογικές περιγραφές και το τι ακριβώς προσφέρουν στο Semantic Web. Οι λογικές περιγραφές είναι πολύ χρήσιμες για την διατήρηση και την ενσωμάτωση των οντολογιών, οι οποίες παρέχουν στον Semantic Web την ικανότητα να κατανοήσει τις βασικές σημασιολογικές έννοιες που χρησιμοποιούνται για τον σχολιασμό των ιστοσελίδων. Όλα αυτά έχουν γίνει εφικτά χάριν της μεγάλης έρευνας που έχει γίνει σ' αυτή την περιοχή .

### **2.2.1 Περιγραφή**

Ο στόχος αυτής της εισαγωγής είναι να σκιαγραφήσει, σε ένα άτυπο επίπεδο, ποιος είναι ο Semantic Web, γιατί χρειάζεται τις οντολογίες και τι ρόλο παίζουν οι λογικές περιγραφές. Όσον αφορά το τελευταίο σημείο, θα προχωρήσουμε αρχικά σε μια μικρή ανάπτυξη των λογικών περιγραφές, και έπειτα θα δείξουμε γιατί είναι καλοταίριασμένες ως γλώσσες οντολογίας. Επιπλέον στην συνέχεια θα παραθέσουμε και τεχνικές λεπτομέρειες.

#### **2.2.1.1 Το Semantic Web και οι οντολογίες**

Για πολλούς ανθρώπους, το World Wide Web(www) έχει γίνει αναπόφευκτα το κυριότερο μέσο για την εύρεση πληροφοριών και απαντήσεων σε διάφορα ερωτήματα. Οι περισσότερες όμως, απαντήσεις που δίνονται από τις μηχανές αναζήτησης είναι τελείως άσχετες ή και πολλές ενδιαφέρουσες απαντήσεις δεν βρίσκονται καν. Αυτό συμβαίνει γιατί οι περισσότερες πηγές(resources) είναι κατανοητές μόνο από ανθρώπους.

Το Semantic Web ψάχνει και για πηγές κατανοητές από τις μηχανές, των οποίων όμως οι πληροφορίες μπορούν να είναι επεξεργάσιμες και για τα δύο αυτοματοποιημένα εργαλεία, δηλαδή τους ανθρώπους και τις μηχανές . Στην συνέχεια θα θεωρήσουμε ότι αυτά τα αυτοματοποιημένα εργαλεία, είναι διαμεσολαβητές. Αυτή η διανομή των

πληροφοριών μεταξύ των διαμεσολαβητών απαιτεί σημασιολογική αύξηση, δηλ., σχολιασμό μιας ιστοσελίδας με πληροφορίες για το περιεχόμενό της, ο οποίος είναι κατανοητός από όλους τους διαμεσολαβητές που ψάχνουν τον Ιστό. Ένας τέτοιος σχολιασμός θα δοθεί σε κάποια τυποποιημένη, εκφραστική γλώσσα και θα χρησιμοποιεί καθορισμένους όρους. Για να σιγουρευτούμε όμως ότι οι διαμεσολαβητές έχουν κάποια κοινή λογική πρέπει να βρεθεί μια οντολογία[Gua95] που θα τους δημιουργήσει μια κοινή ορολογία.

Για να γίνει αυτό πρέπει να χρησιμοποιηθεί μια καλά σχεδιασμένη, καθορισμένη με σαφήνεια και συμβατή, με τον Ιστό, γλώσσα της οποίας η σύνταξη πρέπει να είναι και διαισθητική στους ανθρώπινους χρήστες αλλά και συμβατή με τα υπάρχοντα πρότυπα του Web. Η σημασιολογία της πρέπει να είναι τυπική, δεδομένου ότι ειδιάλλως δεν θα μπορέσει να γίνει κατανοητή απ' όλους τους διαμεσολαβητές. Τέλος, η εκφραστική της δύναμη πρέπει να είναι επαρκής, δηλ., η γλώσσα πρέπει να είναι αρκετά εκφραστική και με τις σχετικές της έννοιες αρκετά λεπτομερείς.

### **2.2.1.2 Λογικές Περιγραφής**

Οι λογικές περιγραφής[BS00] είναι μια οικογένεια των γλωσσών αντιπροσώπευσης γνώσης που χρησιμοποιούνται για να αντιπροσωπεύσουν τη γνώση μιας περιοχής εφαρμογής με έναν καλά δομημένο και κατανοητό τρόπο. Οι λογικές περιγραφής ονόματος παρακινούνται από το γεγονός ότι, αφ' ενός, οι σημαντικές έννοιες της περιοχής περιγράφονται από τις περιγραφές έννοιας, δηλ., εκφράσεις που χτίζονται από τις ατομικές έννοιες (μοναδιαία κατηγορήματα) και τους ατομικούς ρόλους (δυναμικά κατηγορήματα) χρησιμοποιώντας τους κατασκευαστές έννοιας και ρόλου που παρέχονται από το ιδιαίτερο DL, και αφ' ετέρου, η διαφορά του DL από τους προκατόχους του, όπως είναι τα σημασιολογικά δίκτυα και τα πλαίσια, δεδομένου ότι είναι εξοπλισμένο με μια επίσημη λογική, βασισμένη στη σημασιολογία.

Ας υποθέσουμε ότι θέλουμε να καθορίσουμε την έννοια «ενός άνδρα που είναι παντρεμένος με μία γιατρό κι έχει τουλάχιστον 5 παιδιά, τα οποία είναι όλα καθηγητές." Αυτή η έννοια μπορεί να δοθεί με την ακόλουθη περιγραφή έννοιας:

**Human  $\sqcap$   $\neg$ Female  $\sqcap$   $\exists$ married.Doctor  $\sqcap$  ( $\geq 5$  hasChild)  $\sqcap$   $\forall$ hasChild.Professor**

Αυτή η περιγραφή υιοθετεί την κλίση κατασκευαστών (constructors conjunction) ( $\sqcap$ ), η οποία ερμηνεύεται όπως μια καθορισμένη διατομή, την άρνηση ( $\neg$ ), η οποία ερμηνεύεται όπως το καθορισμένο συμπλήρωμα, καθώς και τον υπαρξιακό κατασκευαστή περιορισμού ( $\exists R.C$ ), τον κατασκευαστή περιορισμού αξίας ( $\forall R.C$ ), και τον κατασκευαστή περιορισμού αριθμού ( $\geq n R$ ). Ένα άτομο, για παράδειγμα ο Bob, ανήκει στην  $\exists$ married, εάν υπάρχει ένα άτομο που είναι παντρεμένο με τον Bob (δηλ., συσχετίζεται με τον Bob με τον ρόλο του παντρεμένου) και είναι γιατρός (δηλ., ανήκει στην έννοια γιατρός). Ομοίως, ο Bob ανήκει στο ( $\geq 5$  hasChild) εάν έχει τουλάχιστον 5 παιδιά και ανήκει στο  $\forall$ hasChild εάν όλα τα παιδιά του (δηλ., όλα τα άτομα που σχετίζονται μαζί του μέσω του ρόλου hasChild) είναι καθηγητές.

Εκτός από αυτόν τον φορμαλισμό περιγραφής, τα DLs είναι συνήθως εξοπλισμένα με μια ορολογία και έναν υποθετικό φορμαλισμό. Στην απλούστερη μορφή τους, τα ωρολογιακά αξιώματα μπορούν να χρησιμοποιηθούν για να εισάγουν ονόματα (συντμήσεις) για τις σύνθετες περιγραφές. Παραδείγματος χάριν, θα μπορούσαμε να εισάγουμε τη σύντμηση HappyMan για την περιγραφή της παραπάνω έννοιας. Οι πιο εκφραστικοί ωρολογιακοί φορμαλισμοί επιτρέπουν τη δήλωση περιορισμών, όπως :

**$\exists$ hasChild.Human  $\sqsubseteq$  Human,**

ο οποίος λέει ότι μόνο οι άνθρωποι μπορούν να έχουν παιδιά ανθρώπους. Ο υποθετικός φορμαλισμός μπορεί να χρησιμοποιηθεί για να δηλώσει τις ιδιότητες των ατόμων. Παραδείγματος χάριν, οι ισχυρισμοί

**HappyMan(BOB), hasChild(BOB, MARY)**

δηλώνουν ότι ο Bob ανήκει στην έννοια HappyMan και ότι η Mary είναι ένα από τα παιδιά του.

Τα συστήματα λογικής περιγραφής παρέχουν στους χρήστες ικανότητες συμπεράσματος που συνάγουν την υπονοούμενη γνώση, από τη ρητά αντιπροσωπευόμενη γνώση. Ο αλγόριθμος υπαγωγής(subsumption algorithm) καθορίζει τις υποέννοιες και υπερέννοιες των σχέσεων: το C εντάσσεται στο D εάν όλες οι

περιπτώσεις του C είναι απαραίτητες και περιπτώσεις του D. Δηλαδή, η πρώτη περιγραφή ερμηνεύεται πάντα ως υποσύνολο της δεύτερης περιγραφής. Παραδείγματος χάριν, λαμβάνοντας υπόψη τον καθορισμό της έννοιας HappyMan που αναφέρθηκε παραπάνω, το HappyMan εντάσσεται στην περίπτωση **HasChild.Professor**, εφόσον ο HappyMan έχει τουλάχιστον 5 παιδιά όλα εκ των οποίων είναι καθηγητές και έχουν επίσης ένα παιδί που είναι καθηγητής.

Ο αλγόριθμος περίπτωσης(instance algorithm) καθορίζει τις σχέσεις περίπτωσης: το μεμονωμένο *i* είναι μια περίπτωση της περιγραφής C εάν το *i* ερμηνεύεται πάντα ως στοιχείο του C. Παραδείγματος χάριν, λαμβάνοντας υπόψη τους παραπάνω ισχυρισμούς και τον καθορισμό HappyMan, η MARY είναι μια περίπτωση καθηγητή. Ο αλγόριθμος συνέπειας(consistency algorithm) καθορίζει εάν μια βάση γνώσεων (που αποτελείται από ένα σύνολο ισχυρισμών και ένα σύνολο ωρολογιακών αξιωμάτων) είναι μη-αντιφατική. Παραδείγματος χάριν, εάν προσθέσουμε το: Professor (MARY) στους δύο παραπάνω ισχυρισμούς, η βάση γνώσεων που περιέχει αυτούς τους ισχυρισμούς μαζί με τον καθορισμό HappyMan παραπάνω είναι ασυμβίβαστη.

### **Οι λογικές Περιγραφές ως Γλώσσες Οντολογίας**

Οι υψηλής ποιότητας οντολογίες είναι κρίσιμες για το Semantic Web, και η κατασκευή, η ολοκλήρωση, και η εξέλιξή τους εξαρτάται πολύ από τη διαθεσιμότητα μιας καλά καθορισμένης σημασιολογίας και των ισχυρών εργαλείων συλλογισμού. Δεδομένου ότι τα DLs επιτρέπουν και τα δύο, πρέπει να είναι ιδανικοί υποψήφιοι για τις γλώσσες οντολογίας. Αυτό ήταν ήδη σαφές πριν δέκα χρόνια, αλλά εκείνη την περίοδο, υπήρξε ένας θεμελιώδης κακός συνδυασμός μεταξύ της εκφραστικής δύναμης και της αποδοτικότητας του συλλογισμού που παρέχουν τα DL συστήματα και της εκφραστικότητας και των μεγάλων βάσεων γνώσεων που χρειάστηκαν οι οντολογίες. Μέσω της βασικής έρευνας στα DLs των τελευταίων 10-15 ετών που έχουμε συνοψίσει παραπάνω, αυτό το χάσμα μεταξύ των αναγκών των οντολογιών και των συστημάτων που οι ερευνητές των DL παρέχουν έχει γίνει επιτέλους αρκετά στενό έτσι ώστε να χτίσει σταθερές γέφυρες.

Όσον αφορά μια γλώσσα οντολογίας για τον Σημασιολογικό Ιστό, υπάρχει μια κοινή πρωτοβουλία από τις US/EU για ένα W3C πρότυπο οντολογίας, το οποίο για ιστορικούς

λόγους ονομάζεται DAML+OIL. Αυτή η γλώσσα έχει μια σύνταξη που βασίζεται στο σχήμα RDF (και είναι έτσι το συμβατό σύστημα Web), καθώς και στους κοινούς οντολογικούς πρωτόγονους από τις γλώσσες πλαισίων. Η σημασιολογία της μπορεί να είναι από μια μετάφραση στο εκφραστικό DL SHIQ και οι υπεύθυνοι για την ανάπτυξη έχουν δοκιμάσει έναν καλό συμβιβασμό μεταξύ της εκφραστικότητας και της πολυπλοκότητας του συλλογισμού. Αν και ο συλλογισμός σε SHIQ είναι επιδεικτικός απόφασης, έχει μια μάλλον υψηλή πολυπλοκότητα (ExpTime). Εντούτοις, υπάρχει ένα ιδιαίτερα βελτιστοποιημένο SHIQ reasoner (FaCT) διαθέσιμο, το οποίο συμπεριφέρεται αρκετά καλά στην πράξη.

Μερικά από τα χαρακτηριστικά γνωρίσματα του SHIQ που καθορίζουν αυτό το DL αρκετά εκφραστικό, έτσι ώστε να χρησιμοποιηθεί ως γλώσσα οντολογίας είναι τα παρακάτω :

Αρχικά, η SHIQ παρέχει αριθμητικούς περιορισμούς που είναι πιο εκφραστικοί από αυτούς που παρουσιάστηκαν παραπάνω. Με τους διαθέσιμους αριθμητικούς περιορισμούς σε SHIQ, καθώς επίσης και όντας σε θέση να πει ότι ένα πρόσωπο έχει το πολύ δύο παιδιά (χωρίς αναφορά των ιδιοτήτων αυτών των παιδιών):

**$\{ \leq 2 \text{ hasChild} \}$ ,**

κάποιος μπορεί επίσης να διευκρινίσει ότι υπάρχει το πολύ- ένας γιος και το πολύ-πολύ μια κόρη:

**$\{ \leq 1 \text{ hasChild.} \neg \text{Female} \} \sqcap \{ \leq 1 \text{ hasChild.Female} \}$**

Δεύτερον, η SHIQ επιτρέπει την διατύπωση των σύνθετων ωρολογιακών αξιωμάτων όπως «οι άνθρωποι έχουν γονείς ανθρώπους»:

**$\text{Human} \sqsubseteq \exists \text{hasParent.Human.}$**

Τρίτον, η SHIQ επιτρέπει επίσης τους αντίστροφους ρόλους, τους μεταβατικούς ρόλους, και τους υπορόλους. Παραδείγματος χάριν, εκτός από το hasChild κάποιος μπορεί επίσης να χρησιμοποιήσει και το αντίστροφό του, hasParent, κάποιος μπορεί να διευκρινίσει ότι το hasAncestor είναι μεταβατικό, και ότι το hasParent είναι ένας υπορόλος του hasAncestor.



Έχει υποστηριχτεί ότι στο DL και στην κοινότητα οντολογίας ότι αυτά τα χαρακτηριστικά γνωρίσματα διαδραματίζουν έναν κεντρικό ρόλο κατά την περιγραφή των ιδιοτήτων των αθροισμένων αντικειμένων και κατά τον οικοδόμηση των οντολογιών [Sat00]. Η πραγματική χρήση του DLs που παρέχει αυτά τα χαρακτηριστικά γνωρίσματα ως ελλοχεύοντα λογικό φορμαλισμό του γλωσσικού OIL, της οντολογίας Web και του DAML+OIL τεκμηριώνει αυτήν την αξίωση.

### 2.2.2 Η Εκφραστική Περιγραφική Λογική SHIQ

Σε αντίθεση με τα περισσότερα DLs που εξετάζονται στη λογοτεχνία, και τα οποία επικεντρώνονται στους κατασκευαστές για τον καθορισμό εννοιών, το DL SHIQ [HST99] επιτρέπει και αυτό εκφραστικούς ρόλους. Φυσικά, αυτοί οι ρόλοι μπορούν έπειτα να χρησιμοποιηθούν στον καθορισμό των εννοιών. Ας αρχίσουμε με τον καθορισμό των SHIQ -ρόλων, και συνεχίζουμε έπειτα με τον καθορισμό των SHIQ -εννοιών.

**Καθορισμός 1 (σύνταξη και σημασιολογία των SHIQ -ρόλων):** Έστω ότι το  $\mathbf{R}$  να είναι ένα σύνολο ονομάτων ρόλου, το οποίο χωρίζεται σε ένα καθορισμένο  $\mathbf{R}_+$  των μεταβατικών ρόλων και σε ένα καθορισμένου  $\mathbf{R}_p$  των κανονικών ρόλων. Το σύνολο όλων των SHIQ ρόλων είναι  $\mathbf{R} \cup \{r^- \mid r \in \mathbf{R}\}$ , όπου το  $r^-$  καλείται αντίστροφο του ρόλου  $r$ . Ένα αξίωμα συνυπολογισμού ρόλου είναι του εντύπου  $r \sqsubseteq s$ , όπου τα  $r, s$  είναι SHIQ ρόλοι. Μια ιεραρχία ρόλου είναι ένα σύνολο των αξιωμάτων συνυπολογισμού ρόλου.

Μια ερμηνεία  $I = (\Delta^I, \cdot^I)$  αποτελείται από ένα σύνολο  $\Delta^I$  αποκαλούμενο ως η περιοχή του  $I$  και μια λειτουργία  $\cdot^I$  που χαρτογραφεί κάθε ρόλο σε ένα υποσύνολο  $\Delta^I \times \Delta^I$  έτσι ώστε, για όλο το  $p \in \mathbf{R}$  και το  $r \in \mathbf{R}_+$ ,

$$\langle x, y \rangle \in p^I \quad \text{iff} \quad \langle y, x \rangle \in (p^-)^I, \\ \text{if } \langle x, y \rangle \in r^I \text{ and } \langle y, z \rangle \in r^I \text{ then } \langle x, z \rangle \in r^I.$$

Μια ερμηνεία  $I$  ικανοποιεί έναν Δείκτη διάθλασης ιεραρχίας ρόλου  $r^I \subseteq s^I$  για κάθε  $r \subseteq s \in \mathcal{R}$  και μια τέτοια ερμηνεία καλείται πρότυπο του  $R$ .

Η απεριόριστη χρήση αυτών των ρόλων σε όλους τους κατασκευαστές εννοιών SHIQ θα οδηγούσε σε ένα μη επιδεικτικό απόφασης DL [HST99]. Επομένως, πρέπει να καθορίσουμε ένα κατάλληλο υποσύνολο όλων των SHIQ ρόλων. Αυτό απαιτεί περισσότερη ανάλυση:

1. Η αντίστροφη σχέση στις δυαδικές σχέσεις είναι συμμετρική, δηλ., το αντίστροφο του  $r^-$  είναι πάλι το  $r$ . Για να αποφύγουμε τις εκφράσεις ρόλου όπως  $r^-$ ,  $r^{--}$ , κ.λπ., καθορίζουμε μια λειτουργία  $\text{Inv}$ , η οποία επιστρέφει το αντίστροφο ενός ρόλου:

$$\text{Inv}(r) := \begin{cases} r^- & \text{if } r \text{ is a role name,} \\ s & \text{if } r = s^- \text{ for a role name } s. \end{cases}$$

2. Δεδομένου ότι ο καθορισμένος συνυπολογισμός είναι μεταβατικός και μια σχέση συνυπολογισμού μεταξύ δύο μεταφορών ρόλων με τα αντίστροφά τους, μια δεδομένη ιεραρχία ρόλου  $R$  υπονοεί τις πρόσθετες σχέσεις συνυπολογισμού. Για να αποδώσει αυτό το γεγονός, καθορίζουμε ότι :

$$\subseteq_{\mathcal{R}} := \mathcal{R} \cup \{\text{Inv}(r) \subseteq \text{Inv}(s) \mid r \subseteq s \in \mathcal{R}\}$$

Χρησιμοποιούμε το  $\stackrel{\subseteq_{\mathcal{R}}}{\equiv}$  ως σύντμηση για το  $r \equiv_{\mathcal{R}} s$  και  $r \stackrel{\subseteq_{\mathcal{R}}}{\equiv} s$ . Σε αυτήν την περίπτωση, κάθε πρότυπο του  $R$  ερμηνεύει αυτούς τους ρόλους ως την ίδια δυαδική σχέση.

3. Προφανώς, μια δυαδική σχέση είναι μεταβατική εάν το αντίστροφό της είναι μεταβατικό. Κατά συνέπεια, εάν το  $s \stackrel{\subseteq_{\mathcal{R}}}{\equiv} r$  και  $r \equiv_{\mathcal{R}} s$  είναι μεταβατικά, τότε οποιοδήποτε πρότυπο του  $R$  ερμηνεύει το  $s$  ως μεταβατική δυαδική σχέση. Για να αποδείξουμε τους συλλογισμούς αυτούς, καθορίζουμε:

$$\text{Trans}(s, \mathcal{R}) := \begin{cases} \text{true} & \text{if } r \in \mathbf{R}_+ \text{ or } \text{Inv}(r) \in \mathbf{R}_+ \text{ for some } r \text{ with } r \equiv_{\mathcal{R}} s \\ \text{false} & \text{otherwise.} \end{cases}$$

4. Ένας ρόλος  $\mathbf{R}$  καλείται απλός Δείκτης διάθλασης π.χ  $\text{Trans}(s; \mathbf{R}) = \text{ψεύτικος}$  για όλο το  $s \not\equiv_{\mathcal{R}} r$ .

### Καθορισμός 2 (σύνταξη και σημασιολογία των SHIQ -εννοιών):

Ας υποθέσουμε ότι  $\mathbf{Nc}$  είναι ένα σύνολο ονομάτων έννοιας. Το σύνολο SHIQ -εννοιών είναι το μικρότερο σύνολο έτσι ώστε :

1. κάθε όνομα  $A \in \mathbf{Nc}$  έννοιας είναι μια SHIQ έννοια,
2. εάν το  $C$  και το  $D$  είναι SHIQ έννοιες και το  $r$  είναι ένας SHIQ ρόλος, τότε τα  $C \sqcap D$ ,  $C \sqcup D$ ,  $\neg C$ ,  $\forall r.C$  και  $\exists r.C$  είναι SHIQ έννοιες,

3. εάν το  $C$  είναι μια SHIQ έννοια, το  $r$  είναι ένας απλός SHIQ ρόλος, και  $n \in \mathbf{N}$ , κατόπιν  $(\leq n r.C)$  κ  $(\geq n r.C)$  είναι SHIQ έννοιες.

Η ερμηνευτική λειτουργία  $\cdot^I$  μιας ερμηνείας  $I = (\Delta^I, \cdot^I)$ , χαρτογραφεί πρόσθετα κάθε έννοια σε ένα υποσύνολο του  $\Delta^I$  έτσι ώστε

$$\begin{aligned} (C \sqcap D)^I &= C^I \cap D^I, & (C \sqcup D)^I &= C^I \cup D^I, & \neg C^I &= \Delta^I \setminus C^I, \\ (\exists r.C)^I &= \{x \in \Delta^I \mid \text{There is some } y \in \Delta^I \text{ with } (x, y) \in r^I \text{ and } y \in C^I\}, \\ (\forall r.C)^I &= \{x \in \Delta^I \mid \text{For all } y \in \Delta^I, \text{ if } (x, y) \in r^I, \text{ then } y \in C^I\}, \\ (\leq n r.C)^I &= \{x \in \Delta^I \mid \#_{r^I}(x, C) \leq n\}, \\ (\geq n r.C)^I &= \{x \in \Delta^I \mid \#_{r^I}(x, C) \geq n\}, \end{aligned}$$

Η ορολογία (TBox) εισάγει τις συντμήσεις (ονόματα) για τις σύνθετες έννοιες. Σε SHIQ, το TBox μας επιτρέπει να δηλώσουμε επίσης τους πιο σύνθετους περιορισμούς.

**Καθορισμός 3:** Ένας γενικός συνυπολογισμός έννοιας (GCI) είναι το έντυπο  $C \sqsubseteq D$ , όπου  $C, D$  είναι SHIQ έννοιες. Ένα σύνολο από GCIs καλείται TBox. Μια ερμηνεία  $I$  είναι ένα πρότυπο ενός TBox  $T$  εάν αυτό ικανοποιεί όλα τα GCIs στο  $T$ , δηλ.,  $C^I \sqsubseteq D^I$  κρατά για κάθε  $C \sqsubseteq D \in T$ .

Μία καθορισμένη έννοια είναι του εντύπου  $A \equiv C$ , όπου το A είναι ένα όνομα έννοιας. Αυτό μπορεί επιπλέον να θεωρηθεί και ως σύντμηση για τα δύο GCIs  $A \sqsubseteq C$  και το  $C \sqsubseteq A$ .

**Καθορισμός 4:** Η έννοια C καλείται ικανοποιήσιμη όσον αφορά την ιεραρχία R ρόλου και το TBox T εάν υπάρχει ένα πρότυπο I του R και του T με το  $C^I \neq \emptyset$ . Μια τέτοια ερμηνεία καλείται πρότυπο του C, π.χ. όπως τα R και T. Η έννοια D εντάσσει την έννοια C π.χ. ως  $\langle R, T \rangle$  (γραμμένο ως  $C \sqsubseteq_{\langle R, T \rangle} D$ ) εάν το  $C^I \subseteq D^I$  ισχύει για όλα τα πρότυπα I του R και του T. Δύο έννοιες C και D είναι ισοδύναμες π.χ. το R (γραμμένο ως) όταν εντάσσονται η μία στην άλλη.

Καθορίζοντας την, η ισοδυναμία, μπορεί να μειωθεί στην επαγωγή. Επιπλέον, η επαγωγή μπορεί να μειώσει την ικανότητα απόφασης με δεδομένη την ταυτότητα  $C \sqsubseteq_{\langle R, T \rangle} D$  εάν το  $C \sqcap \neg D$  είναι μη επιδεκτικό απόφασης όπως π.χ. τα R και T. Πριν προχωρήσουμε στην επίλυση του προβλήματος την ικανότητας απόφασης σε SHIQ, προσπαθούμε να δείξουμε πως μπορεί να χρησιμοποιηθεί η SHIQ στον καθορισμό των οντολογιών.

### 2.2.3 Περιγράφοντας τις Οντολογίες σε SHIQ

Μία οντολογία μπορεί να τυποποιηθεί σε ένα TBox ως εξής :

Αρχικά, περιορίζουμε το πιθανό περιβάλλον μας με την εισαγωγή των περιορισμών στις ερμηνείες. Παραδείγματος χάριν, για να εκφράσει ότι, στο περιβάλλον μας, θέλουμε να εξετάσουμε τους ανθρώπους, που είτε είναι μάγοι είτε δεν έχουν μαγικές ικανότητες, εμείς μπορούμε να χρησιμοποιήσουμε τον GCI

$$\text{Human} \sqsubseteq \text{Muggle} \sqcup \text{Sorcerer} \text{ and } \text{Muggle} \sqsubseteq \neg \text{Sorcerer}.$$

Έπειτα, για να εκφράσουμε ότι οι άνθρωποι έχουν ακριβώς δύο γονείς και ότι όλοι οι γονείς και τα παιδιά των ανθρώπων είναι άνθρωποι, μπορούμε να χρησιμοποιήσουμε το ακόλουθο GCI

$$\text{Human} \sqsubseteq \forall \text{hasParent}.\text{Human} \sqcap (\leq 2 \text{ hasParent}.\top) \sqcap (\geq 2 \text{ hasParent}.\top) \sqcap \forall \text{hasParent}^{\neg}.\text{Human},$$

Όπου  $T$  είναι μια σύντμηση για την κορυφαία έννοια  $A \sqcup \neg A$ .

Επιπλέον, εξετάζουμε το μεταβατικό ρόλο `hasAncestor`, και το συνυπολογισμό ρόλου:

$$\text{hasParent} \sqsubseteq \text{hasAncestor}.$$

Επομένως το GCI εκφράζει ότι οι άνθρωποι που έχουν έναν πρόγονο που είναι μάγος είναι και οι ίδιοι μάγοι:

$$\text{Human} \sqcap \exists \text{hasAncestor}.\text{Sorcerer} \sqsubseteq \text{Sorcerer}.$$

Αφετέρου, μπορούμε να καθορίσουμε τις σχετικές έννοιες της περιοχής εφαρμογής μας χρησιμοποιώντας τις καθορισμένες έννοιες. Επιπλέον η καθορισμένη έννοια  $\bar{A} \equiv C$  αντιπροσωπεύει τα δύο GCIs,  $A \sqsubseteq C$  και  $C \sqsubseteq A$ .

Χρειαζόμαστε τις καθορισμένες έννοιες για να ασκήσουμε καθοριστική επίδραση, δηλ., η ερμηνεία της πρωτόγονης έννοιας και τα ονόματα ρόλου πρέπει μεμονωμένα να καθορίσουν την ερμηνεία στα ονόματα των εννοιών. Για αυτό, το σύνολο των καθορισμένων εννοιών μαζί με το πρόσθετο GCIs πρέπει να ικανοποιήσει τρεις όρους:

1. Δεν υπάρχει κανένα προσδιορισμένο πολλαπλάσιο, δηλ., κάθε καθορισμένο όνομα μιας έννοιας μπορεί να εμφανιστεί το πολύ από μια φορά στην αριστερή πλευρά της προσδιορισμένης έννοιας.

2. Δεν υπάρχει κανένας κυκλικός προσδιορισμός, δηλ., καμία κυκλική εξάρτηση μεταξύ των ονομάτων στο σύνολο των προσδιορισμένων εννοιών.

3. Τα ονόματα δεν εμφανίζονται σε οποιαδήποτε από τα πρόσθετα GCIs.

Σε αντίθεση με τα τις προσδιορισμένες έννοιες, το GCIs σε SHIQ μπορεί να έχει και κυκλικές εξαρτήσεις μεταξύ των ονομάτων έννοιας. Ένα παράδειγμα είναι τα παραπάνω GCIs τα οποία περιγράφουν τους ανθρώπους.

Σαν απλό παράδειγμα ενός συνόλου προσδιορισμένων εννοιών που ικανοποιεί τους παραπάνω περιορισμούς, προσδιορίζουμε τις έννοιες γονείς, παππούς και γιαγιά.

$$\text{Parent} \equiv \text{Human} \sqcap \exists \text{hasParent}^- . T$$

$$\text{Grandparent} \equiv \exists \text{hasParent}^- . \text{Parent},$$

Το TBox που αποτελείται από τις παραπάνω προσδιορισμένες έννοιες και το GCIs, μαζί με το γεγονός ότι το `hasAncestor` είναι ένας μεταβατικός υπερρόλος του `hasParent`, υπονοεί την ακόλουθη σχέση υπαγωγής:

## **Grandparent $\sqcap$ Sorcerer $\sqsubseteq$ $\exists$ hasParent $^-$ . $\exists$ hasParent $^-$ . Sorcerer,**

Δηλ., οι παππούδες και γιαγιάδες που είναι μάγοι έχουν ένα εγγόνι που είναι μάγος.

Αν και αυτό το συμπέρασμα φαίνεται λογικό εάν λάβουμε υπόψη μας τις υποθέσεις αυτές, απαιτεί μεγάλη αιτιολόγηση για να μπορέσει κάποιος να το συλλάβει. Ειδικότερα, πρέπει να χρησιμοποιηθεί το γεγονός ότι το `hasAncestor` και το `hasAncestor-` είναι μεταβατικά, ότι το `hasParent-` είναι το αντίστροφο του `hasParent`, και ότι έχουμε το GCI που λέει ότι τα παιδιά των ανθρώπων είναι και πάλι άνθρωποι.

Για να συνοψίσουμε, ένα SHIQ-TBox μπορεί, αφ' ενός, να αξιολογήσει τις βασικές έννοιες μιας περιοχής εφαρμογής (τις πρωτόγονες έννοιες) από τα GCIs, τις δηλώσεις μεταβατικότητας, και τους συνυπολογισμούς ρόλου, υπό την έννοια ότι αυτές οι δηλώσεις περιορίζουν τις πιθανές ερμηνείες των βασικών εννοιών. Αφ' ετέρου, οι πιο σύνθετες έννοιες μπορούν να εισαχθούν από τις προσδιορισμένες έννοιες.

Η ταξινόμια ενός τέτοιου TBox δίνεται έπειτα από την ιεραρχία υπαγωγής των προσδιορισμένων εννοιών. Μπορεί να υπολογιστεί χρησιμοποιώντας έναν αλγόριθμο υπαγωγής για SHIQ. Ο μηχανικός γνώσης μπορεί να εξετάσει εάν το TBox συλλαμβάνει τη διαίσθησή του με τον έλεγχο της ικανότητας απόφασης των προσδιορισμένων εννοιών (δεδομένου ότι δεν έχει νόημα να δοθεί ένας σύνθετος προσδιορισμός για την κενή έννοια), και με τον έλεγχο εάν η θέση τους στην ταξινόμια αντιστοιχεί στη διαισθητική θέση τους. Η εκφραστική δύναμη της SHIQ μαζί με το γεγονός ότι κάποιος μπορεί να «εξακριβώσει» το TBox υπό την έννοια που αναφέρθηκε προηγουμένως είναι ο κύριος λόγος για τον οποίο το SHIQ ταιριάζει με γλώσσα οντολογίας [Sat00].

### **2.2.4 SHIQ και DAML+OIL**

Η DAML+OIL είναι μια γλώσσα οντολογίας του Semantic Web, η σημασιολογία της οποίας μπορεί να προσδιοριστεί μέσω μιας μετάφρασης σε ένα εκφραστικό DL. Αυτό δεν ήταν μια σύμπτωση, αλλά μια σχεδιασμένη κίνηση. Η χαρτογράφηση επιτρέπει στην DAML+OIL να εκμεταλλευτεί τα επίσημα αποτελέσματα από την έρευνα DL (π.χ., σχετικά με την ικανότητα του να λαμβάνεις εύκολα αποφάσεις και την πολυπλοκότητα των βασικών προβλημάτων συμπεράσματος) και την χρήση εφαρμοσμένων

αιτιολογήσεων DL (π.χ., FaCT και RACER) προκειμένου να παραχθούν οι αιτιολογημένες υπηρεσίες για τις εφαρμογές της DAML+OIL.

Η DAML+OIL χρησιμοποιεί μια σύνταξη που είναι βασισμένη στην RDF (το πλαίσιο περιγραφής των πόρων), και έτσι κατάλληλη για το Semantic Web. Το ελλοχεύον πρότυπο για RDF είναι μια επονομαζόμενη κατευθυνόμενη γραφική παράσταση όπου οι κόμβοι είναι είτε πόροι είτε στοιχεία (ουσιαστικά τα στοιχεία αυτά είναι σειρές συμβόλων, αλλά είναι σχεδιασμένες να επεκτείνουν τη γλώσσα και να υποστηρίζουν τις τιμές στοιχείων τύπων, π.χ., "integer 5"). Η γραφική παράσταση καθορίζεται από ένα σύνολο τριπλάσιων, δηλώσεις του εντύπου (θέμα, Ιδιοκτησία, αντικείμενο), όπου το θέμα είναι ένας πόρος, η ιδιοκτησία είναι η ετικέτα ακρών και το αντικείμενο είναι είτε ένας πόρος είτε κυριολεκτικό στοιχείο(literal).

Οτιδήποτε μπορεί να περιγράψει από την RDF είναι ένας πόρος. Ένας πόρος μπορεί να πάρει την ονομασία του από ένα URI, αλλά μερικοί πόροι (τους οποίους τους ονομάζουμε ανώνυμους πόρους) δεν μπορούν να ονομαστούν έτσι. Ένας πόρος μπορεί να είναι μια ολόκληρη σελίδα Web (αναγνωρίσιμη από το URL της), ένα μέρος ιστοσελίδας, αλλά και ένα αντικείμενο μη προσιτό μέσω του Web. Μια ιδιοκτησία είναι μια ιδιότητα ή μια σχέση που χρησιμοποιείται για να περιγράψει έναν πόρο, και παίρνει επίσης το όνομά της από ένα URI. Στην πράξη, τα τριπλάσια γράφονται χρησιμοποιώντας τυποποιημένη σειριοποίηση XML από τριπλά RDF (δείτε [http://www.w3.org/RDF/για\\_περισσότερες\\_λεπτομέρειες](http://www.w3.org/RDF/για_περισσότερες_λεπτομέρειες)).

Μια οντολογία DAML+OIL μπορεί να φαίνεται ότι αντιστοιχεί σε ένα DL TBox με ιεραρχικό ρόλο, περιγράφοντας τον τομέα από την άποψη των κατηγοριών (που αντιστοιχούν στις έννοιες) και των ιδιοτήτων (που αντιστοιχούν στους ρόλους). Μια οντολογία αποτελείται από ένα σύνολο αξιωμάτων που βεβαιώνουν, π.χ., τις σχέσεις μεταξύ των κατηγοριών ή των ιδιοτήτων.

Όπως σε ένα τυποποιημένο DLs, οι κατηγορίες της DAML+OIL μπορεί να είναι ονόματα ή εκφράσεις που ενισχύονται από απλούστερες κατηγορίες και ιδιότητες χρησιμοποιώντας ποικίλους κατασκευαστές. Το σύνολο των κατασκευαστών που υποστηρίζονται από την DAML+OIL, μαζί με την ισοδύναμη θεωρητική σύνταξη της DL, συνοψίζεται στο σχήμα 2.3. Η πλήρης XML σειριοποίηση της σύνταξης RDF δεν θα αναφερθεί γιατί είναι αρκετά πολύπλοκη.

Για παράδειγμα το Human u Male θα γραφόταν :

```
<daml:Class>  
<daml:intersectionOf rdf:parseType="daml:collection">  
<daml:Class rdf:about="#Human"/>  
<daml:Class rdf:about="#Male"/>  
</daml:intersectionOf>  
</daml:Class>
```

Ενώ το (> 2 hasChild:Lawyer) θα γραφόταν ως εξής :

```
<daml:Restriction daml:minCardinalityQ="2">  
<daml:onProperty rdf:resource="#hasChild"/>  
<daml:hasClassQ rdf:resource="#Lawyer"/>  
</daml:Restriction>
```

Ξαναφτιάχνει όπως το daml: καθορίζει στην XML χώρους ονομάτων(namespaces) για τους πόρους, ενώ η rdf:parseType="daml:collection" είναι μια επέκταση της DAML+OIL σε RDF που παρέχει μία στενογραφημένη σημείωση για τους καταλόγους της Lisp οι οποίοι καθορίζονται χρησιμοποιώντας τριπλά με τις ιδιότητες first και rest. Δηλαδή το παραπάνω παράδειγμα αποτελείται ανωτέρω από τα τριπλά < r1; daml : intersection Of; r2>,< r2; daml : ifrest; Human>,< r2; ruffs : type; Class >,<r2; daml : rest; r3>, κ.π., όπου r1b είναι ένας ανώνυμος πόρος, το Human αντιπροσωπεύει ένα URL το οποίο ονομάζει τον πόρο "Human", και το daml: intersection Of, daml:ifrest, daml:rest και ruffs:type αντιπροσωπεύει ένα Uri's που ονομάζει τις ιδιότητες.

Ένα σημαντικό χαρακτηριστικό γνώρισμα της DAML+OIL είναι ότι, εκτός από τις "αφηρημένες" κατηγορίες που προσδιορίζονται από την οντολογία, κάποιος μπορεί επίσης να χρησιμοποιήσει τύπους δεδομένων του σχήματος XML, στα hasClass, hasValue, και cardinality. Π.χ., η κατηγορία Adult θα μπορούσε να είναι ισοδύναμη με



Constructor	DL Syntax	Example
<b>intersectionOf</b>	$C_1 \sqcap \dots \sqcap C_n$	Human $\sqcap$ Male
<b>unionOf</b>	$C_1 \sqcup \dots \sqcup C_n$	Doctor $\sqcup$ Lawyer
<b>complementOf</b>	$\neg C$	$\neg$ Male
<b>oneOf</b>	$\{x_1 \dots x_n\}$	{john, mary}
<b>toClass</b>	$\forall P.C$	$\forall$ hasChild.Doctor
<b>hasClass</b>	$\exists r.C$	$\exists$ hasChild.Lawyer
<b>hasValue</b>	$\exists r.\{x\}$	$\exists$ citizenOf.{USA}
<b>minCardinalityQ</b>	$(\geq n r.C)$	$(\geq 2$ hasChild.Lawyer)
<b>maxCardinalityQ</b>	$(\leq n r.C)$	$(\leq 1$ hasChild.Male)
<b>inverseOf</b>	$r^-$	hasChild <sup>-</sup>

Σχήμα 2.3 : Μηχανισμοί της DAML+OIL

$\text{Person} \sqcap \exists \text{age. over17}$ , όταν το over17 είναι τύπος δεδομένων του σχήματος XML βασισμένο σε δεκαδικούς ,με τους επιπρόσθετους περιορισμούς ότι οι τιμές πρέπει να είναι τουλάχιστον 18 .Χρησιμοποιώντας έναν συνδυασμό του σχήματος XML και της RDF θα μπορούσε να γραφτεί ως εξής :

```

<xsd:simpleType name="over17">
  <xsd:restriction base="xsd:positiveInteger">
    <xsd:minInclusive value="18"/>
  </xsd:restriction>
</xsd:simpleType>
<daml:Class rdf:ID="Adult">
  <daml:intersectionOf rdf:parseType="daml:collection">
    <daml:Class rdf:about="#Person"/>
    <daml:Restriction>
      <daml:onProperty rdf:resource="#age"/>
      <daml:hasClass rdf:resource="#over17"/>
    </daml:Restriction>
  </daml:intersectionOf>
</daml:Class>

```

Όπως αναφέρεται ήδη, μια οντολογία DAML+OIL αποτελείται από ένα σύνολο αξιωμάτων. Το σχήμα 2.4 συνοψίζει τα αξιώματα που υποστηρίζονται από την DAML+OIL. Αυτά τα αξιώματα το καθιστούν πιθανό να επιβεβαιώσουν την επαγωγή ή την ισοδυναμία όσον αφορά τις κατηγορίες ή τις ιδιότητες, την έλλειψη συνοχής των κατηγοριών, την ισοδυναμία ή την μη-ισοδυναμία των ατόμων (πόροι), και τις διάφορες ιδιότητες των ιδιοτήτων.. Ειδικότερα, είναι δυνατό να βεβαιώσει η DAML+OIL ότι μια ιδιοκτησία είναι μοναδική (δηλ., λειτουργική), σαφής (δηλ., το αντίστροφό του λειτουργικού) ή μεταβατική.

Axiom	DL Syntax	Example
<b>subClassOf</b>	$C_1 \sqsubseteq C_2$	<b>Human</b> $\sqsubseteq$ <b>Animal</b> $\sqcap$ <b>Biped</b>
<b>sameClassAs</b>	$C_1 \equiv C_2$	<b>Man</b> $\equiv$ <b>Human</b> $\sqcap$ <b>Male</b>
<b>subPropertyOf</b>	$P_1 \sqsubseteq P_2$	<b>hasDaughter</b> $\sqsubseteq$ <b>hasChild</b>
<b>samePropertyAs</b>	$P_1 \equiv P_2$	<b>cost</b> $\equiv$ <b>price</b>
<b>disjointWith</b>	$C_1 \sqsubseteq \neg C_2$	<b>Male</b> $\sqsubseteq \neg$ <b>Female</b>
<b>sameIndividualAs</b>	$\{x_1\} \equiv \{x_2\}$	<b>{President_Bush}</b> $\equiv$ <b>{G.W_Bush}</b>
<b>differentIndividualFrom</b>	$\{x_1\} \sqsubseteq \neg \{x_2\}$	<b>{john}</b> $\sqsubseteq \neg$ <b>{peter}</b>
<b>transitiveProperty</b>	$P \in \mathbf{R}_+$	<b>hasAncestor</b> <sup>+</sup> $\in \mathbf{R}_+$
<b>uniqueProperty</b>	$\top \sqsubseteq (\leq 1 P.T)$	$\top \sqsubseteq (\leq 1 \text{ hasMother}.T)$
<b>unambiguousProperty</b>	$\top \sqsubseteq (\leq 1 P^-.T)$	$\top \sqsubseteq (\leq 1 \text{ isMotherOf}^-.T)$

Σχήμα 2.4 : Αξιώματα της DAML+OIL

Αυτό δείχνει ότι, εκτός από τα άτομα και τους τύπους δεδομένων, οι κατασκευαστές και τα αξιώματα της DAML+OIL μπορούν να μεταφραστούν σε SHIQ.

### 2.2.5 Αιτιολόγηση στην SHIQ

Αιτιολόγηση στην SHIQ σημαίνει μεγάλη ικανότητα ικανοποίησης και υπαγωγή των εννοιών της SHIQ, για παράδειγμα τα TBoxes (δηλ., τα σύνολα των γενικών συνυπολογισμών έννοιας) και των κατηγοριών ιεραρχίας. Όπως φαίνεται στην παράγραφο 2, η υπαγωγή μπορεί να μειωθεί (στο γραμμικό χρόνο) σε ικανοποιητικό επίπεδο. Επιπλέον, δεδομένου ότι η SHIQ επιτρέπει και υποπαράγοντες και μεταβατικούς παράγοντες, τα TBoxes μπορούν να ενσωματωθούν σε αυτή. Σε γενικές

γραμμές, αυτό επιτυγχάνεται με την εισαγωγή μιας (νέας) μεταβατικής  $U$  υπερκατηγορίας όλων των ρόλων που εμφανίζονται στο TBox  $T$  και την έννοια  $C_0$  για να εξεταστεί για την ικανότητα ικανοποίησης.

Κατόπιν αναλύουμε το  $C_0$  ως εξής :

$$\hat{C}_0 := C_0 \sqcap \bigsqcap_{C \sqsubseteq D \in T} (\neg C \sqcup D) \sqcap \forall u. (\neg C \sqcup D).$$

Έπειτα μπορούμε να δείξουμε ότι  $\hat{C}_0$  έχει μεγάλη ικανότητα ικανοποίησης π.χ. οι εκτεταμένες κατηγορίες ιεραρχίας εάν η αρχική έννοια  $C_0$  είναι ικανή να αποφασίσει για το TBox  $T$  των αρχικών κατηγοριών ιεραρχίας [BH91, HST99].

Συνεπώς, είναι αρκετό για να σχεδιάσει έναν αλγόριθμο που μπορεί να αποφασίσει για την ικανότητα των κατηγοριών ιεραρχίας των SHIQ-εννοιών, π.χ. και των μεταβατικών κατηγοριών. Αυτό το πρόβλημα είναι γνωστό ως EXPTIME-πλήρες (complete) [Tob01]. Στην πραγματικότητα, η EXPTIME –σκληρότητα (hardness) μπορεί να αναλυθεί με μια εύκολη προσαρμογή της απόδειξης EXPTIME –σκληρότητας (hardness) για την ικανότητα της προτασιακής δυναμικής λογικής. Χρησιμοποιώντας τις αυτόματοβασισμένες τεχνικές, Τα Tobies [Tob01] δείχνουν ότι η ικανότητα ικανοποίησης των κατηγοριών ιεραρχίας των SHIQ -εννοιών π.χ. παίζει πράγματι αποφασιστικό ρόλο μέσα στον εκθετικό χρόνο.

Στο υπόλοιπο αυτού του τμήματος, σκιαγραφούμε μια εικόνα-βασισμένη στην διαδικασία απόφασης για αυτό το πρόβλημα. Αυτή η διαδικασία, που περιγράφεται λεπτομερέστερα [HST99], τρέχει στο χειρότερο μη καθορισμένο διπλό εκθετικό χρόνοπερίπτωσης. Εντούτοις, σύμφωνα με την τρέχουσα κατάσταση προόδου, αυτές οι διαδικασίες είναι πρακτικότερες από την αυτόματο-βασισμένη [Tob01] στην ExpTime διαδικασία. Στην πραγματικότητα, είναι η βάση για την ιδιαίτερα βελτιστοποιημένη εφαρμογή του DL στο σύστημα FaCT [Hor98].

Όταν αρχίζει με μια SHIQ -έννοια  $C_0$ , μια κατηγορία ιεραρχίας  $R$ , και τις πληροφορίες στις οποίες οι κατηγορίες είναι μεταβατικές, αυτός ο αλγόριθμος προσπαθεί να κατασκευάσει ένα πρότυπο  $C_0$ , π.χ. το  $R$ . Δεδομένου ότι η SHIQ έχει μια αποκαλούμενη πρότυπη ιδιοκτησία δέντρων, μπορούμε να υποθέσουμε ότι αυτό το πρότυπο έχει τη μορφή ενός άπειρου δέντρου. Εάν θέλουμε να λάβουμε μια διαδικασία απόφασης, μπορούμε μόνο να κατασκευάσουμε ένα πεπερασμένο δέντρο που αντιπροσωπεύει το

άπειρο. Αυτό μπορεί να γίνει έτσι ώστε το πεπερασμένο δέντρο να μπορεί να διευκρινιστεί σε ένα πρότυπο I άπειρων δέντρων  $C0$ , π.χ το  $R$ . Στο πεπερασμένο δέντρο που αντιπροσωπεύει αυτό το πρότυπο, ένας κόμβος  $x$  αντιστοιχεί μεμονωμένα στο  $\pi(x) \in \Delta^I$ , και ονομάζουμε κάθε κόμβο με το σύνολο των εννοιών  $\mathcal{L}(x)$  του οποίου το  $\pi(x)$  υποτίθεται ότι είναι μια υποπερίπτωση. Ομοίως, τα άκρα που αντιπροσωπεύουν τις σχέσεις κατηγορίας-διαδοχής, και μια άκρη μεταξύ του  $X$  και του  $Y$  ονομάζεται σύμφωνα με τις σχέσεις που θα συνδεθούν το  $X$  και το  $Y$ .

Ο αλγόριθμος σταματάει είτε σε μία πεπερασμένη αντιπροσώπευση ενός προτύπου δέντρων, είτε με ένα ασυμβίβαστο (clash), δηλ., σε μια προφανή ασυνέπεια, όπως  $\{C, \neg C\} \subseteq \mathcal{L}(x)$ . Λέμε ότι το “ $C0$  έχει μεγάλη ικανότητα ικανοποίησης” π.χ. στο  $\rho$  στην πρώτη περίπτωση, και “ $C0$  δεν έχει μεγάλη ικανότητα ικανοποίησης” π.χ. στο  $\rho$  στην δεύτερη.

Ο αλγόριθμος μονογραφείται με το δέντρο που αποτελείται από έναν ενιαίο κόμβο  $X$  που χαρακτηρίζεται με το  $\mathcal{L}(x) = \{C_0\}$ . Κατόπιν εφαρμόζει τους αποκαλούμενους κανόνες ολοκλήρωσης, οι οποίοι χωρίζουν τις έννοιες στις ετικέτες κόμβων συντακτικά, συμπεραίνοντας κατά συνέπεια τους νέους περιορισμούς για το δεδομένο κόμβο, και επεκτείνουν έπειτα το δέντρο σύμφωνα με αυτούς τους περιορισμούς. Παραδείγματος χάριν, εάν  $C_1 \sqcap C_2 \in \mathcal{L}(x)$ , έπειτα ο  $\sqcap$ -κανόνας προσθέτει τα  $C_1$  και  $C_2$  στο  $\mathcal{L}(x)$ . Ο  $\geq$ -κανόνας παράγει τους  $n$  νέους κόμβους  $r$ -διαδόχους  $y_1 \dots y_n$  του  $X$  με το  $\mathcal{L}(x) = \{C_0\}$  εάν το  $(\geq n r.C) \in \mathcal{L}(x)$  και το  $x$  δεν έχει ακόμα τους  $n$  ευδιάκριτους  $r$ -διαδόχους με το  $C$  στην ετικέτα τους. Επιπλέον, βεβαιώνει ότι αυτοί οι νέοι διάδοχοι πρέπει να παραμείνουν ευδιάκριτοι (δηλ., δεν μπορεί να είναι στα τελευταία βήματα του αλγορίθμου).

### 2.2.6 Περιορισμοί Και Μεταβλητές Στην SHIQ

Η γλώσσα οντολογίας DAML+OIL είναι μια συντακτική μεταβλητή της SHIQ που επεκτείνεται με τα nominals (δηλ., έννοιες  $\{x_i\}$  που αντιπροσωπεύουν ένα μοναδικό σύνολο που αποτελείται από ένα άτομο) και τους συγκεκριμένους τύπους δεδομένων

(όπως πχ μια έννοια που αντιπροσωπεύει όλους τους ακέραιους αριθμούς μεταξύ 4 και 17).

Οι συγκεκριμένοι τύποι δεδομένων, όπως αυτοί που είναι διαθέσιμοι στην DAML+OIL, έχουν μια πολύ περιορισμένη μορφή η οποία ονομάζεται και συγκεκριμένη περιοχή [BH91]. Παραδείγματος χάριν, χρησιμοποιώντας τη συγκεκριμένη περιοχή όλων των μη αρνητικών ακέραιων αριθμών που εξοπλίζονται με το < κατηγορήμα, μια (λειτουργική) κατηγορία ηλικίας που αφορά (διάφορα) άτομα για την (συγκεκριμένη) ηλικία τους, και μια (λειτουργική) υποκατηγορία πατέρα του hasParent, το ακόλουθο αξίωμα δηλώνει ότι τα παιδιά είναι νεότερα από τους πατέρες τους:

**Animal  $\sqsubseteq$  (age < father o age).**

Η επέκταση εκφραστικού DLs με τις συγκεκριμένες περιοχές μπορεί εύκολα να οδηγήσει σε αναποφασιστικότητα. Εντούτοις, η DAML+OIL παρέχει μόνο μια πολύ περιορισμένη μορφή συγκεκριμένων περιοχών. Ειδικότερα, η συγκεκριμένη περιοχή δεν πρέπει να επιτρέψει arity κατηγορήματα μεγαλύτερα από 1, και οι περιορισμοί κατηγορήματος δεν πρέπει να περιέχουν τις κατηγορίες-αλυσίδες (ηλικία όπως των πατέρων στο παράδειγμά μας). Η ικανότητα απόφασης της SHIG επεκτείνεται ελαφρώς με έναν γενικότερο τρόπο σε ποιο υλικούς τομείς απ' ότι φαίνεται.

Σχετικά με τα nominals, τα πράγματα γίνονται λίγο πιο περίπλοκα. Αρχικά, μπορεί να αποδειχθεί ότι η SHIQ που επεκτείνεται με τα nominals είναι ένα τεμάχιο C2, ή ένας περιορισμός δύο μεταβλητών της λογικής πρώτης σειράς με τα quantifiers[Tob01](είναι ένας χειριστής ο οποίος μας δείχνει για ποιες τιμές μία φόρμα είναι σωστή. Υπολογισμού). Κατά συνέπεια, η ικανότητα ικανοποίησης και υπαγωγής παίζουν αποφασιστικό ρόλο στην NExpTime. Αυτό είναι το βέλτιστο δεδομένου ότι το πρόβλημα είναι επίσης στο NExpTime.-hard [Tob01]. Χοντρικά, ο συνδυασμός του GCIs (οι μεταβατικές κατηγορίες και οι κατηγορίες ιεραρχίας), των αντίστροφων κατηγοριών, και των περιορισμών αριθμού με τα nominals, είναι υπεύθυνος για την πολυπλοκότητα από το ExpTime για την SHIQ στο NExpTime

## **2.3 Από SHIQ και RDF στην OWL: Η παραγωγή μιας γλώσσας οντολογίας Web**

Η γλώσσα οντολογίας του Web, η OWL είναι μια νέα επίσημη γλώσσα για την αντιπροσώπευση των οντολογιών στο Semantic Web. Η OWL έχει τα χαρακτηριστικά γνωρίσματα από διάφορες οικογένειες γλωσσών, που περιλαμβάνουν πρώτιστα τις λογικές περιγραφής και τα πλαίσια. Η OWL έχει επίσης πολλά κοινά χαρακτηριστικά με την RDF, την W3C που είναι βάση του Semantic Web.

### **2.3.1 Εισαγωγή**

Η OWL [DCvH+03] είναι μια νέα γλώσσα οντολογίας για το Semantic Web, που αναπτύσσεται από την ομάδα εργασίας οντολογίας του Web στην κοινοπραξία World Wide Web (W3C). Η OWL είχε ως σκοπό πρώτιστα να αντιπροσωπεύσει τις πληροφορίες για τις κατηγορίες αντικειμένων και πώς τα αντικείμενα είναι αλληλένδετα - το είδος των πληροφοριών που καλείται συχνά οντολογία. Η OWL μπορεί επίσης να αντιπροσωπεύσει τις πληροφορίες για τα αντικείμενα και το είδος των πληροφοριών που θεωρούνται συχνά ως δεδομένα.

Η OWL δεν σχεδιάστηκε σε ένα κενό. Υπήρξαν πολλές επιρροές στο σχέδιο της OWL. Δεδομένου ότι η OWL είναι μια προσπάθεια στη δραστηριότητα του Semantic Web W3C'S, έπρεπε να προσαρμοστεί στο όραμα του Semantic Web σε αρκετές γλώσσες συμπεριλαμβανομένου της XML και της RDF. Για να είναι η OWL μια γλώσσα οντολογίας, έπρεπε να είναι σε θέση να αντιπροσωπεύσει μια χρήσιμη ομάδα χαρακτηριστικών γνωρισμάτων οντολογίας. Δεδομένου ότι υπήρξαν ήδη διάφορες γλώσσες οντολογίας που σχεδιάστηκαν για τη χρήση στον Ιστό, η OWL έπρεπε να διατηρήσει όσο το δυνατόν περισσότερη συμβατότητα με αυτές τις υπάρχουσες γλώσσες, συμπεριλαμβανομένου των SHOE, OIL, και DAML+OIL [CvHarHor+01] .

Οι πολλαπλάσιες επιρροές στην OWL οδήγησαν σε κάποια δυσκολία. Επίσης, και κάπως εκπληκτικά, η ιδιαίτερη τεχνική εργασία έπρεπε να εκτελεσθεί για να επινοήσει την OWL κατά τέτοιο τρόπο ώστε να μπορεί να αποδειχθεί για να έχει τα διάφορα επιθυμητά χαρακτηριστικά γνωρίσματα, διατηρώντας την συμβατότητα με τις ρίζες της. Εδώ περιγράφονται μερικές από τις αποφάσεις σχεδίου που έπρεπε να ληφθούν από την

ομάδα εργασίας οντολογίας του Web κατά τη διάρκεια του σχεδιασμού της OWL. Αν και πολλές από αυτές τις αποφάσεις βασίστηκαν στις απαιτήσεις που καταρτίστηκαν για την OWL [[DCvH+03], και στη συγκεντρωμένη επιστημονική γνώση, μερικές από αυτές βασίστηκαν στις μαλακότερες κρίσεις, και μερικές ήταν απλά ένα θέμα προτίμησης.

Εδώ παρουσιάζεται ένας απολογισμός των συνδιαλλαγών και των αποφάσεων σχεδιασμού στην OWL. Οι απόψεις που παρουσιάζονται είναι εκείνες των συντακτών, και δεν υποστηρίζονται απαραίτητως από όλα τα μέλη της ομάδας εργασίας οντολογίας του Web.

### **2.3.2 Επισκόπηση της OWL.**

Εδώ παρουσιάζεται είναι ένας απολογισμός των επιλογών σχεδιασμού και των συνδιαλλαγών που χρησιμοποιήθηκαν στην παραγωγή της OWL, και δεν σημαίνουν την εξαντλητική περιγραφή της γλώσσας OWL (για τα οποία ο αναγνώστης μπορεί να δει στα έγγραφα OWL, συμπεριλαμβανομένης της γλωσσικής αναφοράς OWL [DCvH+03] και του οδηγού.) Εντούτοις, για να καταστήσει το έγγραφο ανεξάρτητο, περιέχει μια σύντομη περιγραφή της γλώσσας και των σημαντικότερων χρήσεών της.

Στα πλαίσια του Semantic Web, οι οντολογίες αναμένεται να διαδραματίσουν έναν σημαντικό ρόλο στη βοήθεια των αυτοματοποιημένων διαδικασιών (αποκαλούμενοι "ευφυείς διαμεσολαβητές") στις πληροφορίες πρόσβασης. Ειδικότερα, οι οντολογίες που αναμένεται για να χρησιμοποιηθούν για να παρέχουν τα δομημένα λεξιλόγια που εξηγούν τις σχέσεις μεταξύ των διαφορετικών όρων, που επιτρέπουν στους ευφυείς διαμεσολαβητές (και τους ανθρώπους) για να ερμηνεύσουν την έννοιά τους ευέλικτα όμως σαφώς. Παραδείγματος χάριν, μια κατάλληλη οντολογία πίτσας μπορεί να περιλάβει τις πληροφορίες ότι η μοτσαρέλα και γκοργκοτζόλα είναι είδη τυριού, ότι αυτό το τυρί δεν είναι ένα είδος κρέατος ή ψαριών, και ότι μια πίτσα χορτοφάγων είναι μια τα της οποίας κάλυψη δεν περιλαμβάνει κρέας ή ψάρι. Αυτές οι πληροφορίες επιτρέπουν στον όρο "πίτσα που ολοκληρώνεται με (μόνο) τη μοτσαρέλα και τη γκοργκοτζόλα" για να ερμηνευθούν σαφώς (από, π.χ., μια πίτσα που διατάζει τον διαμεσολαβητή) ως ειδίκευση του όρου "πίτσα χορτοφάγων".

Οι όροι των οποίων η έννοια καθορίζεται στις οντολογίες μπορούν να χρησιμοποιηθούν στη σημασιολογική σήμανση που περιγράφει το περιεχόμενο και τη

λειτουργία των προσιτών πόρων Web. Οι οντολογίες και η οντολογία βασισμένη στη σημασιολογική σήμανση θα μπορούσαν να χρησιμοποιηθούν :

Ø Στο ηλεκτρονικό εμπόριο, μπορούν να διευκολύνουν την επικοινωνία μεταξύ της αγοράς και της πώλησης των διαμεσολαβητών με την παροχή ενός κοινού λεξιλογίου για να περιγράψουν τα αγαθά (όπως οι πίτσες) και τις υπηρεσίες (δείτε, π.χ., <http://www.verticalnet.com/>)

Ø Στις μηχανές αναζήτησης, μπορούν να βοηθήσουν στην εύρεση των σελίδων που περιέχουν σημασιολογικά τις παρόμοιες αλλά συντακτικά διαφορετικές λέξεις και φράσεις (δείτε, π.χ., <http://www.hotbot.com/>)

Ø Στις υπηρεσίες του Web και του πλέγματος, μπορούν να παρέχουν τις πλούσιες περιγραφές υπηρεσιών που μπορούν να βοηθήσουν στην εντόπιση των κατάλληλων υπηρεσιών.

Προκειμένου να υποστηριχθούν αυτά και άλλα σενάρια χρήσης, η OWL χρησιμοποιεί την δυνατότητα fact-stating της RDF [KC03] και της class- and και την δυνατότητα property-structuring των ικανοτήτων της RDF Schema [BG03] και τα επεκτείνει με τους σημαντικούς τρόπους. Η OWL μπορεί να δηλώσει τις κατηγορίες, αυτές, να τις οργανώσει και να τις συμψηφίσει σε μια ιεραρχία ("υποκατηγορία"), όπως μπορεί η RDF Schema. Οι κατηγορίες της OWL μπορούν να διευκρινιστούν ως λογικοί συνδυασμοί (διατομές, ενώσεις, ή συμπληρώματα) άλλων κατηγοριών, ή ως απαριθμήσεις των διευκρινισμένων αντικειμένων, που υπερβαίνουν τις ικανότητες RDFS. Η OWL μπορεί επίσης να δηλώσει τις ιδιότητες, να οργανώσουν αυτές τις ιδιότητες σε μια ιεραρχία "λιγότερο σωστή", και να παρέχει τις περιοχές και τις σειρές για αυτές τις ιδιότητες, πάλι όπως η RDFS. Οι περιοχές των ιδιοτήτων της OWL είναι κατηγορίες OWL , και οι σειρές μπορούν να είναι είτε κατηγορίες OWL είτε εξωτερικά καθορισμένοι τύποι δεδομένων όπως η συμβολοσειρά ή ο ακέραιος αριθμός. Η OWL μπορεί να δηλώσει ότι μια ιδιοκτησία είναι μεταβατική, συμμετρική, λειτουργική, ή είναι το αντίστροφο μιας άλλης ιδιοκτησίας, πάλι επεκτείνοντας εδώ την RDFS.

Η OWL μπορεί να εκφράσει ποια αντικείμενα (επίσης αποκαλούμενα ως "άτομα") και σε ποιες κατηγορίες ανήκουν, και ποιών οι τιμές ιδιοκτησίας ανήκουν σε συγκεκριμένα άτομα. Οι δηλώσεις ισοδυναμίας μπορούν να γίνουν στις κατηγορίες και στις ιδιότητες,



οι δηλώσεις των όχι καλά ενωμένων μπορούν να γίνουν στις κατηγορίες, και η ισότητα και η ανισότητα μπορούν να βεβαιωθούν μεταξύ των ατόμων.

Εντούτοις, η σημαντικότερη επέκταση πάνω στην RDFS είναι η δυνατότητα στην OWL να παραχθούν οι περιορισμοί με τρόπο με τον οποίο οι ιδιότητες συμπεριφέρονται σαν τοπικές σε μια κατηγορία. Η OWL μπορεί να καθορίσει τις κατηγορίες όπου μια ιδιαίτερη ιδιοκτησία είναι περιορισμένη έτσι ώστε όλες οι τιμές για την ιδιοκτησία στις περιπτώσεις της κατηγορίας πρέπει να ανήκουν σε μια ορισμένη κατηγορία (ή τύπο δεδομένου) τουλάχιστον μια αξία πρέπει να προέλθει από μια ορισμένη κατηγορία (ή τύπο δεδομένου) πρέπει να υπάρξουν τουλάχιστον ορισμένες συγκεκριμένες τιμές και πρέπει να υπάρξουν τουλάχιστον ή το πολύ-πολύ πιο ορισμένες ευδιάκριτες τιμές.

Παραδείγματα χάριν, χρησιμοποιώντας RDFS μπορούμε να:

- ∅ δηλώσουμε τις κατηγορίες όπως τη χώρα, το πρόσωπο, το σπουδαστή και έναν Καναδό.
- ∅ δηλώσουμε ότι ο σπουδαστής είναι μια υποκατηγορία του προσώπου.
- ∅ δηλώσουμε ότι ο Καναδάς και η Αγγλία είναι και οι δύο περιπτώσεις της κατηγορίας χώρας.
- ∅ κηρύξουμε την υπηκοότητα ως ιδιοκτησία που αφορά το πρόσωπο κατηγοριών (η περιοχή του) και τη χώρα (η σειρά του)
- ∅ δηλώσουμε ότι η ηλικία είναι μια ιδιοκτησία, με το πρόσωπο ως περιοχή της και τον ακέραιο αριθμό ως σειρά της και τέλος, να
- ∅ δηλώσουμε ότι ο Peter είναι μια περίπτωση της κατηγορίας Καναδός, και ότι η ηλικία του έχει την αξία 48.

Με την OWL μπορούμε πρόσθετα να:

- ∅ δηλώσουμε ότι η χώρα και το πρόσωπο χωρίζουν τις κατηγορίες
- ∅ δηλώσουμε ότι ο Καναδάς και η Αγγλία είναι ευδιάκριτα άτομα
- ∅ κηρύξουμε πρώην κατοίκους ως αντίστροφη ιδιοκτησία της υπηκοότητας
- ∅ δηλώσουμε ότι η κατηγορία άνευ υπηκοότητας καθορίζεται ακριβώς ως εκείνα τα μέλη του προσώπου κατηγορίας που δεν έχουν καμία τιμή για την υπηκοότητα ιδιοκτησίας

- ∅ δηλώσουμε ότι η κατηγορία, πολλές Εθνικότητες, καθορίζεται ακριβώς ως εκείνα τα μέλη του προσώπου κατηγορίας που έχουν τουλάχιστον 2 τιμές για την υπηκοότητα ιδιοκτησίας
- ∅ δηλώσουμε ότι η κατηγορία Καναδός καθορίζεται ακριβώς ως εκείνα τα μέλη του προσώπου κατηγορίας που έχουν τον Καναδά ως αξία της υπηκοότητας ιδιοκτησίας και να
- ∅ δηλώσουμε ότι η ηλικία είναι μια λειτουργική ιδιοκτησία.

Από τα προηγούμενα συμπεραίνουμε ότι η OWL είναι μια αρκετά περίπλοκη γλώσσα. Η OWL έχει και τη σύνταξη μιας ανταλλαγής RDF/XML και μ' ένα αφηρημένο πλαίσιο όπως η σύνταξη, και έχει τρεις ονομασμένες υπογλώσσες. Αυτή η πολλαπλότητα είναι το άμεσο αποτέλεσμα της προσπάθειας να ικανοποιηθεί ένας μεγάλος αριθμός μερικές φορές συγκρουόμενων επιρροών και απαιτήσεων.

### 2.3.3 Οι επιρροές στην OWL

Όπως έχει ήδη αναφερθεί παραπάνω, ο σχεδιασμός της OWL υπόκειται σε ποικίλες επιρροές. Αυτές οι συμπεριλαμβανόμενες επιρροές από τους καθιερωμένους φορμαλισμούς και τα παραδείγματα αντιπροσώπευσης γνώσης, επηρεάζονται από τις υπάρχουσες γλώσσες οντολογίας, και από τις επιρροές από τις υπάρχουσες γλώσσες του Semantic Web.

Μερικές από τις σημαντικότερες επιρροές στο σχέδιο της OWL προήλθαν, μέσω του προκατόχου της, DAML+OIL, από τις λογικές περιγραφές, από το παράδειγμα πλαισίων και από την RDF. Ειδικότερα, η επίσημη προδιαγραφή της γλώσσας επηρεάστηκε από τις λογικές περιγραφές, η δομή επιφάνειας της γλώσσας (όπως βλέπει στην αφηρημένη σύνταξη) επηρεάστηκε από το παράδειγμα πλαισίων και η σύνταξη ανταλλαγής RDF/XML επηρεάστηκε από μια απαίτηση για την ανοδική συμβατότητα με την RDF.

#### 2.3.3.1 Περιγραφές λογικής

Οι περιγραφές λογικής είναι μια οικογένεια των βασισμένων στο (έννοια βασισμένη) φορμαλισμό αντιπροσώπευσης γνώσης, κατηγοριών . Χαρακτηρίζονται με την χρήση των διάφορων κατασκευαστών για να χτίσουν σύνθετες κατηγορίες από τις απλούστερες, από μια έμφαση στην ικανότητα να λαμβάνονται αποφάσεις στα βασικά προβλήματα

συλλογισμού και από τις πλήρεις και (εμπειρικά) εύκολα διαλογίσιμες υπηρεσίες της παροχής ήχου. Οι λογικές περιγραφές, και οι ιδέες από την έρευνα λογικής περιγραφής, είχαν μια ισχυρή επιρροή στο σχεδιασμό της OWL, ιδιαίτερα στη διαμόρφωση της σημασιολογίας, την επιλογή των γλωσσικών κατασκευαστών και την ολοκλήρωση των τύπων και των τιμών των δεδομένων. Στην πραγματικότητα η OWL DL και η OWL LITE (δύο από τα τρία είδη της OWL) μπορούν να αντιμετωπισθούν ως έκφραση λογικής περιγραφής, με μια οντολογία που είναι ισοδύναμη με μια βάση γνώσεων λογικής περιγραφής.

Το κύριο χαρακτηριστικό σημασιολογίας  $A$  της λογικής περιγραφής είναι ότι είναι λογικές, δηλ., επίσημες γλώσσες με καλά καθορισμένη σημασιολογία. Η τυποποιημένη τεχνική για την έννοια μιας λογικής περιγραφής είναι μέσω μιας πρότυπης θεωρητικής σημασιολογίας, της οποίας ο σκοπός είναι να εξηγηθεί η σχέση μεταξύ της γλωσσικής σύνταξης και του προοριζόμενου πρότυπου της περιοχής. Ένα πρότυπο αποτελείται από μια περιοχή (συχνά γραμμένο  $\Delta^I$ ) και μια λειτουργία ερμηνείας (συχνά γραπτή  $\cdot^I$ ), όπου η περιοχή είναι ένα σύνολο αντικειμένων και η λειτουργία ερμηνείας είναι μια χαρτογράφηση από τα ονόματα ατόμων, κατηγορίας και ιδιοκτησίας στα στοιχεία της περιοχής, τα υποσύνολα της περιοχής και τις δυαδικές σχέσεις στην περιοχή, αντίστοιχα. Έτσι, για έναν μεμονωμένο John,  $\text{John}^I \in \Delta^I$ , για μια κατηγορία προσώπου Person,  $\text{Person}^I \subseteq \Delta^I$ , και για έναν φίλο ιδιοκτησίας friend,  $\text{friend}^I \subseteq \Delta^I \times \Delta^I$ .

Η λειτουργία ερμηνείας μπορεί να επεκταθεί από τα ονόματα κατηγορίας στις σύνθετες περιγραφές κατηγορίας με τον προφανή τρόπο. Παραδείγματος χάριν, δίνονται δύο κατηγορίες το αρσενικό-Male και το πρόσωπο-Person που ερμηνεύονται λαμβάνοντας υπόψη τα σύνολα  $\text{Male}^I = \{w,x,y\}$  και  $\text{Person}^I = \{x,y,z\}$ , κατόπιν η τομή του αρσενικού και του προσώπου (δηλ., αρσενικά πρόσωπα) ερμηνεύεται ως τομή  $\{w,x,y\}$  και  $\{x,y,z\}$ , δηλ.,  $(\text{Male και Person})^I = \{x,y\}$ .

Τα αντικείμενα στην περιοχή μόνα τους δεν έχουν οποιαδήποτε έννοια, αλλά ούτε και η επιλογή οποιουδήποτε συνόλου αντικειμένων που αποτελούν την περιοχή. Σε ένα δεδομένο πρότυπο, παραδείγματος χάριν, ένα μεμονωμένο  $i$  είναι μια περίπτωση μιας κατηγορίας  $C$  ακριβώς στη περίπτωση που το  $i$  ερμηνεύεται ως στοιχείο της ερμηνείας του  $C$  (δηλ.,  $i^I \in C^I$ ), και μια κατηγορία  $C$  είναι μια υποκατηγορία μιας κατηγορίας  $D$

ακριβώς σε περίπτωση που η ερμηνεία του  $C$  είναι ένα υποσύνολο της ερμηνείας του  $D$  (δηλ.,  $D^I \subseteq C^I$ ).

Η βάση γνώσεων στη λογική περιγραφής αποτελείται από ένα σύνολο αξιωμάτων βεβαιώνοντας, π.χ., ότι μια κατηγορία είναι μια υποκατηγορία σε άλλη, ή ότι ένα άτομο είναι μια περίπτωση μιας ιδιαίτερης κατηγορίας. Η έννοια αυτών των αξιωμάτων δίνεται από τους αντίστοιχους περιορισμούς στα πρότυπα. Εάν, παραδείγματος χάριν, η βάση γνώσεων περιέχει ένα αξίωμα δηλώνοντας ότι το πρόσωπο-Pearson είναι μια υποκατηγορία του ζώου-Animal (που γράφεται το `Pearson Animal`), κατόπιν σε ένα πρότυπο της βάσης γνώσεων η ερμηνεία του προσώπου-Pearson πρέπει πάντα να είναι ένα υποσύνολο της ερμηνείας του ζώου-Animal. Η έννοια μιας βάσης γνώσεων προέρχεται από τα χαρακτηριστικά γνωρίσματα και τις σχέσεις που είναι κοινά για όλα τα πιθανά πρότυπα. Εάν, παραδείγματος χάριν, η ερμηνεία μιας κατηγορίας πρέπει πάντα να είναι το κενό σύνολο, κατόπιν εκείνη η κατηγορία λέγεται ότι είναι ασυμβίβαστη, ενώ εάν δεν υπάρχει καμία πιθανή ερμηνεία, η ίδια η βάση γνώσεων λέγεται ότι είναι ασυμβίβαστη. Εάν η σχέση που διευκρινίζεται από ένα δεδομένο αξίωμα πρέπει να κρατήσει σε όλες τις ερμηνείες μιας βάσης γνώσεων, κατόπιν εκείνο το αξίωμα λέγεται ότι είναι συνεπαγόμενο από τη βάση γνώσεων, και εάν μια βάση γνώσεων συνεπάγεται κάθε αξίωμα σε μια άλλη βάση γνώσεων, κατόπιν η πρώτη βάση γνώσεων λέγεται ότι συνεπάγεται τη δεύτερη βάση γνώσεων. Μια βάση γνώσεων που περιέχει το αξίωμα `Pearson Animal`, παραδείγματος χάριν, συνεπάγεται ότι η διατομή του αρσενικού-Male και του προσώπου-Pearson είναι επίσης μια υποκατηγορία του ζώου-Animal. Αυτό που συνεπάγεται είναι αρκετά τετριμμένο, αλλά με μια γλώσσα τόσο σύνθετη όσο η OWL, που ελέγχει τα συνεπαγόμενα μπορεί, γενικά, να είναι ένας πολύ σκληρός στόχος.

Όπως η OIL και η DAML+OIL, η OWL χρησιμοποιεί μια πρότυπη θεωρία ύφους λογικής περιγραφής για να τυποποιήσει την έννοια της γλώσσας. Αυτό αναγνωρίστηκε ως ουσιαστικό χαρακτηριστικό γνώρισμα και σε στις τρεις γλώσσες, δεδομένου ότι επιτρέπει στις οντολογίες, και στις πληροφορίες να χρησιμοποιούν το λεξιλόγιο που καθορίζεται από τις οντολογίες, που το μοιράζονται και που ανταλλάσσεται χωρίς διαφωνίες στην ακριβή έννοια. Η ανάγκη για αυτό το είδος τυπικότητας ενισχύθηκε από την εμπειρία με τις πρώτες εκδόσεις της προδιαγραφής RDF και RDFS, όπου μια έλλειψη τυπικότητας οδήγησε σύντομα στα επιχειρήματα ως προς την έννοια των

γλωσσικών κατασκευασμάτων όπως οι περιορισμοί περιοχών και σειράς. Προκειμένου να αποφευχθούν τέτοια προβλήματα, η έννοια RDF τώρα επίσης καθορίζεται από την άποψη μιας πρότυπης θεωρίας.

Ένα άλλο πλεονέκτημα στην έννοια της γλώσσας κατ' αυτό τον τρόπο είναι ότι οι αυτοματοποιημένες τεχνικές συλλογισμού μπορούν να χρησιμοποιηθούν για να ελέγξουν τη συνέπεια των κατηγοριών και των οντολογιών, και για να ελέγξει τις συνεπαγόμενες σχέσεις. Αυτό είναι κρίσιμο εάν η πλήρης δύναμη των οντολογιών πρόκειται να χρησιμοποιηθεί από τους ευφυείς διαμεσολαβητές, και η δυνατότητα να παρασχεθεί τέτοια υποστήριξη συλλογισμού ήταν ένας βασικός στόχος σχεδίου για την OWL.

### **2.3.3.2 Κατασκευαστές Γλωσσών**

Η εκφραστική δύναμη μιας γλώσσας όπως στην OWL καθορίζεται από τους κατασκευαστές κατηγορίας (και ιδιοκτησίας) που υποστηρίζονται, και από τα είδη αξιωμάτων που μπορούν να εμφανιστούν σε μια οντολογία. Η φυσικά αυξανόμενη εκφραστική δύναμη σημαίνει αναπόφευκτα την αυξανόμενη υπολογιστική πολυπλοκότητα για τα βασικά προβλήματα συλλογισμού όπως τα συνεπαγόμενα.

Το σχέδιο της OWL επηρεάστηκε περισσότερο από 10 έτη από την έρευνα λογικής περιγραφής, η οποία έχει χαράξει με ιδιαίτερες λεπτομέρειες το τοπίο για ένα ευρύ φάσμα των κατασκευαστών και των αξιωμάτων, και διάφορους συνδυασμούς τους. Αυτή η γνώση επέτρεψε στο σύνολο κατασκευαστών και αξιωμάτων που υποστηρίχθηκαν από την OWL για να επιλεγεί προσεκτικά ώστε να ισορροπηθούν οι εκφραστικές απαιτήσεις των χαρακτηριστικών εφαρμογών με μια απαίτηση για την αξιόπιστη και χωρίς σπατάλη υποστήριξη συλλογισμού.

Ένας ιδιαίτερος στόχος αυτής της διαδικασίας σχεδίου ήταν να εξασφαλιστεί ότι αυτό που συνεπάγεται η OWL θα ήταν τουλάχιστον αποφασιστικό, δηλ., ότι θα ήταν δυνατό να σχεδιαστεί ένας αλγόριθμος που θα μπορούσε να εγγραφεί να καθορίσει εάν μια οντολογία OWL συνεπάγεται ή όχι άλλης (ένας τέτοιος αλγόριθμος καλείται συχνά διαδικασία απόφασης). Η διαθεσιμότητα των πρακτικών διαδικασιών απόφασης (για το τι συνεπάγονται), και ακόμη και τα εφαρμοσμένα συστήματα, ήταν επίσης μια σημαντική εκτίμηση.

Μια κατάλληλη ισορροπία μεταξύ αυτών των υπολογιστικών απαιτήσεων και των εκφραστικών απαιτήσεων που προσδιορίστηκαν [Hef03] επιτεύχθηκε με τη στήριξη του σχεδίου της OWL στη SH οικογένεια των λογικών περιγραφής. Οι κατασκευαστές και τα αξιώματα που υποστηρίζονται από τη SH είναι παρόμοιοι με εκείνους που περιγράφονται στην παράγραφο 2, και περιλαμβάνουν τις μπουουλιανές συνδέσεις (διατομή, ένωση και συμπλήρωμα), τις ιδιότητες των περιορισμών, τις μεταβατικές ιδιότητες και μια ιεραρχία ιδιοκτησίας, δηλ. ισοδύναμος με τη λογική περιγραφής ALC που επεκτείνεται με τις μεταβατικές ιδιότητες και μια ιεραρχία ιδιοκτησίας. Η ιεραρχία ιδιοκτησίας είναι σημαντική για την OWL δεδομένου ότι είναι ένα χαρακτηριστικό γνώρισμα RDFS, ενώ οι μεταβατικές ιδιότητες έχουν προσδιοριστεί ως σημαντική απαίτηση σε πολλές εφαρμογές [Hef03]. Τα μέλη της SH οικογένειας περιλαμβάνουν την επιδρούσα λογική περιγραφής SHIQ, που προσθέτει τις αντίστροφες ιδιότητες και τους γενικευμένους περιορισμούς αριθμητικού συνόλου στοιχείων, και SHOQ (D), το οποίο προσθέτει τη δυνατότητα να καθοριστεί μια κατηγορία με την απαρίθμηση των περιπτώσεών του (π.χ., η κατηγορία {Δευτέρα, Τρίτη, Τετάρτη, Πέμπτη, Παρασκευή}) και της υποστήριξης για τους τύπους δεδομένων και τις τιμές (π.χ., ακέραιος αριθμός και σειρά τύπου δεδομένων, και τιμές όπως "35").

Οι τύποι δεδομένων καθώς εξετάζονται επίσης και οι "αφηρημένες" κατηγορίες όπως το πρόσωπο-Pearson και το ζώο-Animal, πολλές πρακτικές εφαρμογές πρέπει να αντιπροσωπευθούν και ο λόγος για τους τύπους δεδομένων και τις τιμές όπως οι ακέραιοι αριθμοί και οι σειρές. Η ενσωμάτωση των τύπων δεδομένων στη γλώσσα OWL πάλι βαριά επηρεάζεται από την έρευνα λογικής περιγραφής, η οποία έχει καταδείξει ότι η προσοχή απαιτείται προκειμένου να αποφευχθεί να μεγαλώσει η πολυπλοκότητα ή ακόμα και η αναποφασιστικότητα που προκαλείται από τους τύπους δεδομένων [Lut02]. Στη λογική περιγραφής της SHOQ(D) αποδείχθηκε ότι αυτό θα μπορούσε να επιτευχθεί αν χωριστούν αυστηρά η ερμηνεία των τύπων δεδομένων και των τιμών από αυτήν των κατηγοριών και των ατόμων: στη SHOQ(D) οι ερμηνείες περιλαμβάνουν μια πρόσθετη περιοχή ερμηνείας για τη ταυτότητα τιμών των στοιχείων  $\Delta^I_D$  που χωρίζει την περιοχή από τα άτομα  $\Delta^I$ . Οι τύποι δεδομένων, όπως ο ακέραιος αριθμός, ερμηνεύεται ως υποσύνολο του  $\Delta^I_D$ , και οι τιμές όπως ο ακέραιος αριθμός "35" ερμηνεύονται ως στοιχεία του  $\Delta^I_D$ . Ο χωρισμός ενισχύεται περαιτέρω με τη διαίρεση των ιδιοτήτων σε δύο σύνολα

περίληψης και ιδιοτήτων τους τύπους δεδομένων. Οι αφηρημένες ιδιότητες όπως ο αδελφός ερμηνεύονται ως δυαδικές σχέσεις στο  $\Delta^I$  (δηλ., υποσύνολα του  $\Delta^I \times \Delta^I$ ), οι ιδιότητες των τύπων δεδομένων όπως η ηλικία ερμηνεύονται ως δυαδικές σχέσεις μεταξύ του  $\Delta^I$  και του  $\Delta^I_D$  (δηλ., υποσύνολα του  $\Delta^I \times \Delta^I_D$ ).

Αυτό το σχέδιο έχει το πλεονέκτημα ότι ο συλλογισμός με τους τύπους δεδομένων και τις τιμές μπορεί σχεδόν εξ ολοκλήρου να χωριστεί από το συλλογισμό με τις κατηγορίες και τα άτομα, μια κατηγορία που εδρεύει εξηγώντας απλά χρειάζεται την πρόσβαση σε ένα τύπο δεδομένων "χρησμό" που μπορεί να απαντήσει στις απλές ερωτήσεις όσον αφορά τους τύπους δεδομένων και τις τιμές (π.χ., "το 5 είναι ένας ακέραιος αριθμός όχι αρνητικός;"). Επιπλέον, η γλώσσα παραμένει αποφασιστική εάν ο τύπος δεδομένων και η αξία που διαλογίζεται είναι αποφασιστική, δηλ., εάν ο χρησμός (oracle) μπορεί να εγγυηθεί ότι θα απαντήσει σε όλα τα θέματα του σχετικού είδους για υποστηριζόμενους τύπους δεδομένων. Αυτό μπορεί εύκολα να επιτευχθεί για μια σειρά κοινών τύπων δεδομένων όπως οι ακέραιοι αριθμοί, τα δεκαδικά και οι αλφαριθμητικές [Lut02].

Όπως και αυτές οι πρακτικές εκτιμήσεις, μπορεί επίσης να υποστηριχτεί ότι ο χωρισμός των κατηγοριών και των τύπων δεδομένων έχει νόημα από μια φιλοσοφική σκοπιά όπως ότι οι τύποι δεδομένων κτίζονται ήδη από χτισμένους στα κατηγορήματα όπως > και <. Από αυτήν την άποψη, δεν έχει νόημα η χρήση των αξιωμάτων οντολογίας για να προσθέσει περαιτέρω δομή στους τύπους δεδομένων ή στις υβριδικές "κατηγορίες μορφής" όπως η κατηγορία κόκκινων ακέραιων αριθμών.

### 2.3.3.3 Το παράδειγμα των πλαισίων

Στο πλαίσιο του Semantic Web, όπου οι χρήστες με ένα ευρύ φάσμα πείρας αναμένεται να δημιουργήσουν ή να τροποποιήσουν τις οντολογίες, η αναγνωσιμότητα και η γενική ευκολία της χρήσης είναι σημαντικές εκτιμήσεις για μια γλώσσα οντολογίας. Στο σχεδιασμό της OIL, μια από τις γλώσσες στις οποίες η OWL είναι βασισμένη, αυτές οι απαιτήσεις εξετάστηκαν με την παροχή μιας σύνταξης επιφάνειας βασισμένης στο παράδειγμα πλαισίων. Τα πλαίσια συγκεντρώνουν τις πληροφορίες για κάθε κατηγορία, που καθιστά τις οντολογίες ευκολότερες να διαβάσουν και να κατανοηθούν, ιδιαίτερα από τους χρήστες που δεν είναι εξοικειωμένοι με τις λογικές (περιγραφής). Το παράδειγμα πλαισίων έχει χρησιμοποιηθεί σε διάφορα καλά γνωστά

συστήματα αντιπροσώπευσης γνώσης συμπεριλαμβανομένου του εργαλείου σχεδιασμού οντολογίας Protégé και του προτύπου γνώσης OKBC [CFarFik98]. Ο σχεδιασμός της OIL επηρεάστηκε από την XOL, μια πρόταση της XML για σύνταξη OKBC Lite (μια περικομμένη έκδοση του προτύπου γνώσης OKBC).

Στις γλώσσες που είναι βασισμένες στο πλαίσιο, κάθε κατηγορία περιγράφεται από ένα πλαίσιο. Το πλαίσιο περιλαμβάνει το όνομα της κατηγορίας, προσδιορίζει τη γενικότερη κατηγορία (ή τις κατηγορίες) που ειδικεύεται, και απαριθμεί ένα σύνολο "αυλακώσεων". Μια αυλάκωση μπορεί να αποτελείται από ένα ζευγάρι ιδιοκτησίας-αξίας, ή έναν περιορισμό στις τιμές που μπορούν να ενεργήσουν ως αυλάκωση "υλικά γεμίματος" (σε αυτό το πλαίσιο, να εκτιμήσουν τα μέσα είτε ένα άτομο είτε μια αξία στοιχείων). Αυτή η δομή χρησιμοποιήθηκε στη γλώσσα OIL, με κάποιο εμπλουτισμό της σύνταξης για τη διευκρίνιση των κατηγοριών και των περιορισμών αυλακώσεων ώστε να επιτραπεί η πλήρης δύναμη μιας γλώσσας ύφους λογικής περιγραφής να συλληφθεί. Επιπλέον, τα πλαίσια ιδιοκτησίας χρησιμοποιήθηκαν για να περιγράψουν τις ιδιότητες, π.χ., διευκρινίζοντας τις γενικότερες ιδιότητες, τους περιορισμούς σειράς και περιοχών, τη μεταβατικότητα και τις αντίστροφες σχέσεις ιδιοκτησίας.

Ένα πλαίσιο κατηγορίας είναι σημασιολογικά ισοδύναμο με ένα αξίωμα λογικής περιγραφής βεβαιώνοντας ότι η κατηγορία που περιγράφεται από το πλαίσιο είναι μια υποκατηγορία κάθε μια από τις κατηγορίες που ειδικεύεται και κάθε ένας από τους περιορισμούς ιδιοκτησίας που αντιστοιχούν στις αυλακώσεις. Όπως και μια πλουσιότερη σύνταξη αυλακώσεων, η δυνατότητα επίσης της OIL ότι η κατηγορία που περιγράφεται από το πλαίσιο ήταν ακριβώς ισοδύναμη με τη σχετική κατηγορία διατομής, (δηλ., την οποία ενέτασσαν αμοιβαία). Ένα πλαίσιο ιδιοκτησίας είναι ισοδύναμο με ένα σύνολο αξιωμάτων βεβαιώνοντας τις σχετικές σχέσεις υποϊδιοκτησίας, η σειρά περιορισμών και περιοχών κ.λπ. της OIL σχεδιάστηκε έτσι ώστε τα πλαίσια της OIL να μπορούν εύκολα να χαρτογραφηθούν στα ισοδύναμα αξιώματα στη λογική περιγραφής SHOQ(D).

Η επίσημη προδιαγραφή και σημασιολογία της OWL δίνονται από μια αφηρημένη σύνταξη που έχει επηρεαστεί βαριά από τα γενικά πλαίσια και ειδικότερα από το σχεδιασμό της OIL. Στην αφηρημένη σύνταξη, τα αξιώματα είναι σύνθετες κατασκευές που μοιάζουν πολύ σε ένα πλαίσιο ύφους OIL. Για τις κατηγορίες, που αποτελούνται από το όνομα του περιγραφής της κατηγορίας, μια μορφή "μερικού" ή "ολοκληρωμένου"



(δείχνοντας ότι το αξίωμα βεβαιώνει μια σχέση υποκατηγοριών ή ισοδυναμίας αντίστοιχα), και μια ακολουθία περιορισμών ιδιοκτησίας και ονομάτων των γενικότερων κατηγοριών. Ομοίως, ένα αξίωμα ιδιοκτησίας διευκρινίζει το όνομα της ιδιοκτησίας και των διάφορων χαρακτηριστικών γνωρισμάτων του.

Το ύφος πλαισίων της αφηρημένης σύνταξης το καθιστά πολύ εύκολο να διαβαστεί (έναντι στη σύνταξη RDF/XML), και επίσης ευκολότερο να κατανοηθεί και να χρησιμοποιηθεί. Επιπλέον, τα αφηρημένα αξιώματα σύνταξης έχουν άμεση αλληλογραφία με τα αξιώματα λογικής περιγραφής, και μπορούν επίσης να χαρτογραφηθούν σε ένα σύνολο τριπλάσιων RDF.

#### **2.3.3.4 Η RDF σύνταξη**

Η τρίτη σημαντική επιρροή στο σχεδιασμό της OWL ήταν η απαίτηση να διατηρηθεί η μέγιστη ανοδική συμβατότητα με τις υπάρχουσες γλώσσες του Web, και ειδικότερα με την RDF. Σε αυτή την απαίτηση που δίνει την καλή αίσθηση όπως η RDF (και ειδικότερα η RDF Schema) ήδη συμπεριλαμβανόμενη αρκετών από τα βασικά χαρακτηριστικά γνωρίσματα μιας κατηγορίας και εδρευόμενης της ιδιοκτησία μιας γλώσσας οντολογίας, π.χ., αυτό επιτρέπει στις σχέσεις υποκατηγοριών και υποιδιοκτησίας να βεβαιωθούν. Επιπλέον, η ανάπτυξη RDF προηγήθηκε αυτής της OWL, και αυτό φάνηκε λογικό να προσπαθήσει να απευθυνθεί σε οποιαδήποτε κοινότητα χρηστών που ιδρύθηκε ήδη από την RDF.

Μπορεί να φανεί εύκολο να καλυφθεί αυτή η απαίτηση απλά με το να δοθεί στην OWL μια σύνταξη βασισμένη στην RDF. Προκειμένου να παρασχεθεί η μέγιστη ανοδική συμβατότητα, θεωρήθηκε επίσης απαραίτητο να εξασφαλιστεί ότι η σημασιολογία των οντολογιών της OWL ήταν επίσης σύμφωνη με τη σημασιολογία RDF. Σε αυτό αποδείχθηκε δύσκολο να δοθεί η αυξανόμενη εκφραστική εξουσία που προβλέφθηκε από την OWL.

#### **2.3.4 Τα προηγούμενα συστήματα της OWL**

Η OWL δεν ήταν η πρώτη γλώσσα οντολογίας του Web, και ο σχεδιασμός της επηρεάστηκε από διάφορες προϋπάρχουσες γλώσσες συμπεριλαμβανομένου της RDFS, της SHOE, της OIL, της DAML-ONT και της DAML+OIL. Η DAML+OIL ειδικότερα

ήταν μια σημαντική επιρροή στην OWL, και ο χάρτης της ομάδας εργασίας οντολογίας Web ρητά δηλώνει ότι το σχέδιο της OWL πρέπει να βασιστεί στη DAML+OIL. Η DAML+OIL στη συνέχεια επηρεάστηκε αρκετά από τη γλώσσα OIL, με την πρόσθετη επιρροή από την εργασία πάνω στις DAML- ONT και RDFS.

#### **2.3.4.1 SHOE**

Μια από τις πρώτες προσπάθειες στον καθορισμό μιας γλώσσας οντολογίας για την επέκταση στο Web ήταν η SHOE. Η SHOE είναι μια γλώσσα βασισμένη σε πλαίσιο με τη σύνταξη της XML που θα μπορούσε να ενσωματωθεί ακίνδυνα στα υπάρχοντα έγγραφα HTML. Η SHOE χρησιμοποίησε τις αναφορές URI για τα ονόματα, μια σημαντική καινοτομία που υιοθετήθηκε στη συνέχεια και από τη DAML- ONT και από τη DAML+OIL. Η SHOE έδωσε επίσης έμφαση στο γεγονός ότι οι οντολογίες θα συνδέονταν στενά και υπαγόμενος στην αλλαγή. Συνεπώς, η SHOE περιέλαβε διάφορες οδηγίες που επέτρεψαν την εισαγωγή άλλων οντολογιών, τοπικά να μετονομάσουν τις εισαγόμενες σταθερές, και τη δήλωση και συμβατότητας των πληροφοριών μεταξύ των οντολογιών. Αυτή η γραμμή σκέψης έχει επηρεάσει το πρόσθετο λογικό λεξιλόγιο της OWL που έχει ως σκοπό να εξετάσει μερικώς τέτοια ζητήματα. Η SHOE ήταν μικρότερης επιρροής στο συντακτικό και σημασιολογικό σχεδιασμό της OWL δεδομένου ότι δεν βασίστηκε στην RDF, και δεν ήρθε με μια επίσημη σημασιολογία.

#### **2.3.4.2 DAML- ONT**

Το 1999 η γλώσσα σήμανσης διαμεσολαβητών του προγράμματος DARPA (DAML) άρχισε με το στόχο τη βάση ενός επόμενου "Semantic Web παραγωγής". Σαν πρώτο βήμα, αποφασίστηκε ότι η υιοθέτηση μιας κοινής γλώσσας οντολογίας που θα διευκόλυνε τη σημασιολογική διαλειτουργικότητας στα διάφορα έργα που αποτελούν το πρόγραμμα. Η RDFS (που είχε προταθεί ήδη ως W3C πρότυπο) φαινόταν ως καλή αφετηρία, αλλά δεν ήταν σαφέστατα εκφραστική για να καλύψει τις απαιτήσεις της DAML. Μια νέα γλώσσα αποκαλούμενη DAML- ONT επομένως αναπτύχθηκε και επέκτεινε την RDF με τους γλωσσικούς κατασκευαστές από τις αντικειμενοστραφείς και τις βασισμένες σε πλαίσιο στις γλώσσες αντιπροσώπευσης γνώσης.

Η DAML- ONT ενσωματώθηκε στενά με την RDFS, και ενώ αυτό ήταν χρήσιμο από μια άποψη συμβατότητας, οδήγησε σε μερικά σοβαρά προβλήματα στο σχέδιο για τη γλώσσα. Όπως η RDFS, η DAML- ONT υπέφερε από μια ανεπαρκή σημασιολογική προδιαγραφή, και σύντομα συνειδητοποιήθηκε ότι αυτό θα μπορούσε να οδηγήσει σε διαφωνία, μεταξύ των ανθρώπων και των μηχανών, ως προς την ακριβή έννοια των όρων σε μια οντολογία DAML-ONT. Επιπλέον, οι περιορισμοί ιδιοκτησίας DAML- ONT είχαν, όπως εκείνοι της RDFS, το σφαιρικό παρά τοπικό πεδίο, και ενώ αυτό ήταν λογικό για τους περιορισμούς περιοχών και σειράς που παρέχονταν από την RDFS, ο σφαιρικός αριθμός στοιχείων συνόλου περιορίζει, παραδείγματος χάριν, είναι δύσκολο να καταλάβει και την αμφισβητήσιμη χρησιμότητα (στην πραγματικότητα φαίνεται πιθανό ότι αυτό θα είχε αναγνωριστεί ως ρωγμή σχεδίου εάν η σημασιολογία της γλώσσας ήταν τυποποιημένη επαρκώς).

#### **2.3.4.3 OIL**

Περίπου τον ίδιο χρόνο που η DAML- ONT αναπτυσσόταν, μια ομάδα (κατά ένα μεγάλο μέρος ευρωπαίων) ερευνητών με στόχους παρόμοιους με εκείνους των ερευνητών DAML είχε σχεδιάσει μια άλλη οντολογία προσανατολισμένη προς το Web τη γλώσσα OIL (το στρώμα συμπεράσματος οντολογίας). Η OIL ήταν η πρώτη γλώσσα οντολογίας για να συνδυάσει τα στοιχεία από τις λογικές περιγραφές, τις γλώσσες πλαισίων και τα πρότυπα του Web όπως η XML και η RDF. Η OIL έδωσε μια ισχυρή έμφαση στην επίσημη ακαμψία, και η γλώσσα σχεδιάστηκε ρητά έτσι ώστε η σημασιολογία της να μπορεί να διευκρινιστεί μέσω μιας χαρτογράφησης στη λογική περιγραφής SHIQ . Η δομή της γλώσσας ήταν, εντούτοις, βασισμένη στο πλαίσιο, η χρησιμοποίηση της σύνθετης κατηγορίας οι "ορισμοί" στο ύφος που περιγράφηκαν πιο πάνω. Η OIL είχε και σύνταξη XML και RDF, η σύνταξη της RDF είχε ως σκοπό να διατηρήσει τη συμβατότητα με RDFS, δεν ασχολούταν με τις ακριβείς λεπτομέρειες της σημασιολογίας της RDF, οι οποίες εκείνη την περίοδο δεν ήταν τυπικά καθορισμένες.

#### **2.3.4.4 DAML+OIL**

Έγινε προφανές και στη DAML-ONT και στις ομάδες της OIL ότι οι στόχοι τους θα μπορούσαν καλύτερα να εξυπηρετηθούν με το συνδυασμό των προσπαθειών τους, το αποτέλεσμα που είναι η συγχώνευση της DAML-ONT και της OIL για να παραγάγουν DAML+OIL. Η ανάπτυξη DAML+OIL αναλήφθηκε από μια επιτροπή που αποτελείται κατά ένα μεγάλο μέρος από τα μέλη των δύο ομάδων γλωσσικού σχεδίου, και μάλλον κυρίως με τον τίτλο η κοινή γλώσσα αντιπροσώπευσης διαμεσολαβητών.

Η συγχωνευμένη γλώσσα δίνει μια επίσημη σημασιολογία από την πρότυπη θεωρία ύφους της DL αντί μέσω μιας μετάφρασης σε ένα κατάλληλο της DL. Οι παραγόμενοι από τη DL γλωσσικοί κατασκευαστές της OIL διατηρήθηκαν σε DAML+OIL, αλλά η δομή πλαισίων απορρίφθηκε κατά ένα μεγάλο μέρος υπέρ των αξιωμάτων ύφους της DL, τα οποία ενσωματώθηκαν ευκολότερα στη σύνταξη της RDF.

Επηρεασμένη από τη DAML-ONT, η DAML+OIL είναι πιο στενά ενσωματωμένη με την RDF. Η DAML+OIL, εντούτοις, παρείχε μόνο μια έννοια για εκείνα τα μέρη της RDF που ήταν σύμφωνα με την πρότυπη θεωρία της σύνταξης και του ύφους της DL. Αυτό δεν φάνηκε να είναι ενός προβλήματος δεδομένου ότι η RDF εκείνη την περίοδο δεν είχε μια τυπικά διευκρινισμένη δική της έννοια, αλλά ήταν η αιτία των σοβαρών δυσκολιών όταν χρησιμοποιήθηκε DAML+OIL ως βάση για την OWL.

#### **2.3.5 Τα προβλήματα κατά μήκος της πορείας**

Η πολλαπλή επιρροή στην OWL έχει το προβάδισμα σε διάφορα προβλήματα. Μερικά από αυτά τα προβλήματα είναι ένα αποτέλεσμα των συγκρουόμενων απαιτήσεων, όπως η σύγκρουση μεταξύ της χρησιμοποίησης RDF/XML ως επίσημη σύνταξη OWL και έχοντας μια εύκολη στο διάβασμα σύνταξη. Μερικά από αυτά τα προβλήματα προκύπτουν από μια ανάγκη να επεκταθούν οι προηγούμενες λύσεις, όπως στα προβλήματα που προκύπτουν στην επεξεργασία μιας επέκτασης σε RDF που ενσωματώνει τις πληροφορίες που δεν αρμόζουν καλά στην παγκόσμια RDF άποψη.

##### **2.3.5.1 Τα συντακτικά προβλήματα**

Για διάφορους λόγους, συμπεριλαμβανομένης και της διατήρησης των συνδέσεων στα πλαίσια και της λογικής περιγραφής, η OWL έπρεπε να έχει μια εύκολη στο διάβασμα

σύνταξη που να μπορεί να γίνει κατανοητή εύκολα και να δημιουργείται εύκολα. Εντούτοις, ήταν μια απαίτηση της OWL ότι θα χρησιμοποιεί την XML ως κανονική της σύνταξη, και, επιπλέον, θα χρησιμοποιεί την XML με τον ίδιο τρόπο που χρησιμοποιείται στην RDF [Bec02]. Αυτή η απαίτηση ήταν εξετασμένη ήδη από την OIL και, πιο πρόσφατα, από την DAML+OIL: Η OIL έχει και RDF/XML και XML σύνταξη [20], ενώ η DAML+OIL έχει μόνο RDF/XML σύνταξη [CvHarHor01].

Παίρνοντας απλώς ως σύνταξη για την OWL, η RDF υπό μορφή RDF/XML έχει έναν αριθμό προβλημάτων. Αυτά τα προβλήματα μπορούν να υπερνικηθούν, αλλά καθιστούν την OWL πιο σύνθετη από ότι είναι.

Ένα πρόβλημα είναι ότι οι RDF/XML είναι εξαιρετικά φλύαρες. Έστω για παράδειγμα να συγκρίνουμε κάποιες πληροφορίες για μια κατηγορία δεδομένου ότι θα δινόταν σε έναν περιγραφή λογικής σύνταξης

```
Student = Pearson Π> 1 enrolledIn
```

(ένας σπουδαστής είναι πρόσωπο που έχει ρόλο τουλάχιστον σε 1 πράγμα), με το πώς φυσικότερα θα γραφόταν χρησιμοποιώντας την OWL RDF/XML σύνταξη

```
< owl:Class rdf:ID="Student" >  
< owl:intersectionOf rdf:parsetype="Collection" >  
< owl:Class rdfs:about="Person"/>  
< owl:Restriction >  
< owl:onProperty rdf:resource="enrolledIn"/>  
< owl:minCardinality rdfs:datatype="&xsd;Integer" >  
1  
< owl:intersectionOf >
```

Ένα άλλο πρόβλημα είναι ότι η RDF σπάει όλα στα τριπλάσια RDF [KC03]. Αυτό σημαίνει ότι πολλά κατασκευάσματα της OWL, όπως οι περιορισμοί ιδιοκτησίας, πρέπει να κωδικοποιηθούν ως διάφορα τριπλάσια. Η OWL χρησιμοποιεί γενικά μια κωδικοποίηση παρόμοια με αυτήν που χρησιμοποιείται από την DAML+OIL. Παραδείγματος χάριν, ένας περιορισμός αξίας OWL που θα γραφόταν στη σύνταξη λογικής περιγραφής ως  $\exists$ child.person (η κατηγορία της οποίας οι περιπτώσεις έχουν κάποιο παιδί που είναι πρόσωπο) κωδικοποιείται ως δύο τριπλάσια RDF κάτι σαν

```
_:x owl:onProperty ex:child.
```

`_:x owl:someValuesFrom ex:Person.`

όπου `_:x` είναι συντακτικό κράτημα για τον περιορισμό συνολικά.

Ένα τρίτο πρόβλημα είναι ότι όλα τα τριπλάσια της RDF είναι ανεξάρτητα. Αυτό σημαίνει, παραδείγματος χάριν, ότι όσον αφορά την RDF δεν υπάρχει καμία απαίτηση ότι τα δύο παραπάνω τριπλάσια πρέπει πάντα να εμφανίζονται από κοινού. Ομοίως, δεν υπάρχει καμία απαίτηση ότι εκεί που δεν υπάρχουν τριπλάσια πρέπει να προστίθενται.

`_:x owl:onProperty ex:friend.`

`_:x owl:allValuesFrom ex:Doctor.`

τα παραπάνω δύο τριπλάσια δεν μπορούν να αποκλειστούν από την RDF.

Ένα τέταρτο πρόβλημα είναι ότι τα τριπλάσια της RDF είναι όλα προσιτά. Αυτό σημαίνει ότι η εγκύκλιος και άλλες ασυνήθιστες δομές δεν μπορούν να αποκλειστούν. Παραδείγματος χάριν, δεν υπάρχει κανένα πρόβλημα στην RDF με τις συλλογές των τριπλάσιων όπως

`_:x owl:onProperty ex:child. :`

`_:x owl:allValuesFrom _:x.`

Αυτά τα ζητήματα δεν αντιμετωπίζονται στην OIL, η οποία δεν παρέχει καμία οδηγία ως προς αυτό που πρέπει να συμβεί για τις συλλογές των τριπλάσιων που δεν ταιριάζουν με την παραγωγή σύνταξης της γλώσσας. Η DAML+OIL υιοθετεί μια διαφορετική μέθοδο, που επιτρέπει τις ασυνήθιστες κατασκευές αλλά που δυσκολεύεται να τους δώσει νόημα. Η OWL έχει ακολουθήσει κατά προσέγγιση τη λύση DAML+OIL, αλλά με διάφορες τροποποιήσεις.

### **2.3.5.2 Τα σημασιολογικά προβλήματα**

Τα ζητήματα της σύνταξης αντιμετωπίστηκαν, ζητήματα σχετικά με τη σημασία παραμένουν. Η RDF παρέχει έννοια για κάθε τριπλάσιο, έτσι εάν η OWL πρόκειται να θεωρηθεί επέκταση της RDF, η έννοια που η OWL παρέχει για τα τριπλάσια πρέπει να είναι μια επέκταση αυτής της έννοιας της RDF.

Αυτό δεν ήταν όπως ένα μεγάλο μέρος ενός ζητήματος όταν σχεδιάστηκαν η OIL και η DAML+OIL, δεδομένου ότι η έννοια της RDF κάλλιστα δεν διευκρινίστηκε. Η OIL ειδικότερα δεν ενοχλείται να συσχετίσει την έννοια της RDF με τη σύνταξη της RDF/XML στην έννοια της σύνταξης της OIL -- η σύνταξη RDF/XML για μερικά

κατασκευάσματα της OIL κάνει τη διάταξη περισσότερο-ή-λιγότερη με την έννοια RDF σε αυτά τα κατασκευάσματα αλλά δεν είναι με κανένα τρόπο η περίπτωση για όλα τα κατασκευάσματα. Παραδείγματος χάριν, η OIL έχει μια ειδική ιδιοκτησία (`oil:hasSlotConstraint`) που χρησιμοποιείται για να συσχετίσει μια κατηγορία με τις αυλακώσεις της, αλλά η έννοια της RDF αυτής της ιδιοκτησίας, δηλαδή η τυποποιημένη έννοια που ορίζεται σε οποιοδήποτε τριπλάσιο της RDF αγνοείται από τη σημασιολογία της OIL.

Η DAML+OIL κάνει καλύτερη εργασία στην τήρηση της έννοιας της σύνταξης της RDF. Η πρότυπη θεωρία της DAML+OIL περιλαμβάνει έναν σημασιολογικό όρο για τα τριπλάσια που είναι κοντά στην έννοια της RDF (όπως καθορίστηκε εκείνη την περίοδο) για τα τριπλάσια. Περαιτέρω, η DAML+OIL χρησιμοποιεί το ενσωματωμένο λεξιλόγιο των RDF και RDFS σε μια μεγαλύτερη έκταση από ότι η OIL, και το χρησιμοποιεί με έναν τρόπο γενικά συμβατό με την έννοια RDF ή RDFS (όπως καθορίζεται εκείνη την περίοδο) για αυτό το λεξιλόγιο. Παραδείγματος χάριν η DAML+OIL χρησιμοποιεί `rdfs:subClassOf` για να συσχετίσει τις κατηγορίες με τις υπερκατηγορίες, συμπεριλαμβανομένων των περιορισμών της DAML+OIL, ενώ η OIL χρησιμοποιεί `oil:hasSlotConstraint` σε μερικές από αυτές τις καταστάσεις.

Ακόμα και όταν αναπτυσσόταν η DAML+OIL υπήρξαν μερικές πτυχές της έννοιας της RDFS που δεν θα μπορούσε να συμφιλιωθεί με την κατάλληλη έννοια στη DAML+OIL. Ειδικότερα, η RDFS έπειτα είχε μια ασυνήθιστη έννοια για τις περιοχές και τις σειρές των ιδιοτήτων. Μόνο η ενιαία μιας σειρά επιτράπηκε για τις ιδιότητες και οι πολλαπλές περιοχές αντιμετωπίστηκαν διαχωριστικά. Παραδείγματος χάριν,

```
ex:foo rdfs:domain ex:Person.
```

```
ex:foo rdfs:domain ex:Rock.
```

θα επέτρεπε και σε Pearson και σε Rock να συμμετέχουν στην ιδιοκτησία .

Αυτή η ανάγνωση των περιοχών προκάλεσε τα προβλήματα για τη σημασιολογία της DAML+OIL έτσι μια επιλογή έγινε για να αλλάξει αυτό σε μια συνδετική ανάγνωση και να υποβάλει αίτηση στην πρόσφατα σχηματισμένη ομάδα εργασίας πυρήνων της RDF για να αλλάξει την RDFS για να επιτρέψει τις πολλαπλές περιοχές και τις σειρές, και στις δύο με μια συνδετική ανάγνωση. Ως τμήμα του καθορισμού της σημασιολογίας των RDF

και RDFS, η ομάδα εργασίας πυρήνων RDF έχει αποφασίσει να κάνει αυτήν την αλλαγή, που αποβάλλει ένα πρόβλημα για την OWL.

Λύνοντας τα προβλήματα με τις RDF και RDFS, η ομάδα εργασίας πυρήνων RDF αποφάσισε επίσης να βάλει RDF σε ένα σταθερότερο σημασιολογικό έδαφος. Αυτό το έκανε με την παροχή μιας πρότυπης θεωρίας για τις RDF και RDFS, μαζί με μια τυποποιημένη επεξεργασία του συμπεράσματος για αυτές. Αυτό σημαίνει ότι υπάρχει τώρα μεγαλύτερη έννοια που παρέχεται για τις RDF και RDFS και η OWL πρέπει να είναι συμβατή με αυτές. Ειδικότερα, όλα τα τριπλάσια που χρησιμοποιούνται για να κωδικοποιήσουν τη σύνταξη της OWL έχουν τώρα την έννοια της RDF, και αυτή η έννοια RDF πρέπει να ληφθεί υπόψη από την OWL εάν η σημασιολογία της OWL πρόκειται να είσαι πλήρως συμβατή με εκείνη των RDF και RDFS.

Ούτε η OIL ούτε η DAML+OIL δεν παρείχαν μια τυποποιημένη θεωρία του συμπεράσματος. Αυτό ήταν κοινό στους φορμαλισμούς που επηρέασαν την OIL και την DAML+OIL. Τα πλαίσια παρείχαν γενικά μια διεπαφή στις εσωτερικές δομές δεδομένων αντί οποιωνδήποτε άλλων συμπερασμάτων ή ακόμα και ερωτήσεων. Οι λογικές περιγραφές παρέχουν μια επίσημη θεωρία της συζήτησης, αλλά αυτό είναι κάπως διαφορετικό από μια τυποποιημένη θεωρία του συμπεράσματος.

Το διαφορετικό είναι ότι η συζήτηση λογικής περιγραφής θα μπορούσε να έχει καθοριστεί για την DAML+OIL με έναν τρόπο που θα είχε βοηθήσει να κρύψει την έννοια των τριπλάσιων της RDF. Παραδείγματος χάριν, η ερώτηση εάν ένα άτομο άνηκε σε μια κατηγορία θα μπορούσε να προσθέσει τη σύνταξη που χρησιμοποιήθηκε για να διευκρινίσει την κατηγορία στις εγκαταστάσεις της ερώτησης. Εντούτοις, μια τυποποιημένη θεωρία δεν μπορεί να το κάνει αυτό.

Τα αποτελέσματα αυτής της αλλαγής μπορούν να φανούν σε ένα απλό παράδειγμα. Λαμβάνοντας υπόψη τις ακόλουθες πληροφορίες

ex:John rdf:type ex:Student.

ex:John rdf:type ex:Employee.

Θα ήταν αρκετά εύκολο να τακτοποιηθεί έτσι ώστε η ερώτηση εάν ο John άνηκε στη διατομή του σπουδαστή και του υπαλλήλου αρχικά εξασφάλισε ότι αυτή η διατομή υπήρξε και έπειτα ρώτησε εάν John άνηκε σε αυτό. Εντούτοις, η μετατροπή αυτού σε κάτι απαραίτητο που επιθυμεί τις ανωτέρω πληροφορίες για να συνεπάγεται



```
:c owl:intersectionOf :l1.  
:l1 rdf:first ex:Student.  
:l1 rdf:rest :l2.  
:l2 rdf:first ex:Employee.  
:l2 rdf:rest rdf:nil  
. ex:John rdf:type :c.
```

που, λόγω της έννοιας της RDF που αποδίδεται σε όλα τα τριπλάσια, απαιτείται η ύπαρξη των τριπλάσιων που κωδικοποιούν την σύνταξη της OWL.

Έτσι έπρεπε να αναπτυχθεί μια μέθοδο που αυξάνει τη νέα σημασιολογία της RDF αρκετά ακριβώς για να υποστηρίξει τα παραπάνω συμπεράσματα χωρίς να είναι πάρα πολύ ισχυρή, και όντας καταλήγοντας έτσι με τα σημασιολογικά παράδοξα

### 2.3.5.3 Εκφραστική δύναμη

Επειδή πολλά πράγματα αναμένονταν από την OWL[Hef03], υπήρξαν πολλές απαιτήσεις για την εκφραστική δύναμη που υπερβαίνει αυτήν από τις λογικές περιγραφές. Παραδείγματος χάριν, πολλοί χρήστες θέλησαν να είναι σε θέση να συνδέσουν τις πληροφορίες με τις κατηγορίες και τις ιδιότητες και να κάνουν τις κατηγορίες να ανήκουν σε άλλες κατηγορίες, όπως είναι δυνατό στην RDF. Ομοίως, υπήρξαν πολλές απαιτήσεις για την εκφραστική δύναμη που υπερβαίνει τις: RDF και RDFS. Παραδείγματος χάριν, πολλοί χρήστες θέλησαν να είναι σε θέση να παρέχουν την τοπική δακτυλογράφηση για τις τιμές ιδιοκτησίας, όπως είναι δυνατό στις λογικές περιγραφές. Η OWL έπρεπε να έχει ως σκοπό να επιτρέψει κάπως αυτά τα είδη της εκφραστικότητας ακόμα διατηρώντας τις συνδέσεις στις ρίζες της.

Όταν η DAML+OIL αναπτύχθηκε, ο μόνος τύπος δεδομένων που υποστηρίχθηκε από RDF ήταν τυπογραφικά λάθη: κατά προσέγγιση οι τιμές που δόθηκαν ως σειρές. Έτσι η DAML+OIL έπρεπε να παρέχει τη λύση του για τους τύπους δεδομένων, και έκανε έτσι επιτρέποντας τη χρήση του XML Schema τύπου δεδομένων. Εντούτοις, οποιαδήποτε λογική λύση που χρησιμοποιεί μόνο ανάγκες σύνταξης RDF βοηθάτε από την RDF (δηλ., μια επέκταση στη σύνταξη της RDF), και έτσι η λύση DAML+OIL παρέμεινε ελλιπής.

Πρόσφατα η RDF έχει προσθέσει στην έκδοση της τύπωσης δεδομένων, παρόμοια αλλά διαφορετική από τη λύση DAML+OIL. Η OWL πρέπει έτσι να κινηθεί από την DAML+OIL προς την RDF.

#### **2.3.5.4 Τα υπολογιστικά προβλήματα**

Μια πτυχή της OWL που τη διακρίνει από τις RDF και RDFS είναι ότι υποστηρίζει ένα πλούσιο σύνολο συμπερασμάτων. Μερικά από αυτά τα συμπεράσματα είναι αρκετά προφανή, όπως το παράδειγμα που δόθηκε παραπάνω για τους σπουδαστές και τους υπαλλήλους, και εμφανίζονται έτσι εύκολα για υπολογισμό. Άλλα συμπεράσματα που υποστηρίζονται από την OWL, εντούτοις, είναι αρκετά σύνθετα, απαιτώντας, π.χ., το συλλογισμό από τις περιπτώσεις και μετά από τις αλυσίδες των ιδιοτήτων.

Η λήψη όλων των αντιπροσωπευτικών επιθυμιών για την OWL μαζί θα είχε δημιουργήσει έναν φορμαλισμό όπου τα βασικά προβλήματα συμπεράσματος ήταν αναποφάσιστα. Παραδείγματος χάριν, η άδεια των σχέσεων για να βεβαιωθεί μεταξύ των αλυσίδων ιδιοκτησίας (όπως το ρητό ότι ένας θείος είναι ακριβώς ο αδελφός ενός γονέα) θα καθιστούσε το συμπέρασμα της OWL αναποφάσιστο. Επιπλέον, μερικές πτυχές της RDF, όπως η χρήση των κατηγοριών ως περιπτώσεις, αλληλεπιδρούν με το υπόλοιπο της OWL για να δημιουργήσουν υπολογιστικές δυσκολίες, που παίρνουν την OWL πέρα από τη σειρά των πρακτικών αλγορίθμων και των εφαρμοσμένων συστημάτων συλλογισμού.

Η OWL έπρεπε έτσι να παρέχει μια λύση για αυτά τα υπολογιστικά ζητήματα ακόμα διατηρώντας την ανοδική συμβατότητα με τις RDF και RDFS.

#### **2.3.6 Οι λύσεις**

Η ομάδα εργασίας οντολογίας του Web πέρασε μεγάλο μέρος από ένα έτος για να υπερνικήσει τις βασικές εντάσεις που κρύβονται πίσω από τα παραπάνω προβλήματα. Οι δυσκολίες που έχει όχι κάθε πρόβλημα μεμονωμένο, αλλά ο συνδυασμός όλων των παραπάνω προβλημάτων και των περιορισμών που τοποθετούνται στο σχεδιασμό της OWL. Θα ήταν πολύ πιο εύκολο, παραδείγματος χάριν, να καλυφθούν όλες οι παραπάνω απαιτήσεις εάν μόνο η OWL θα μπορούσε να έχει χρησιμοποιήσει μια επέκταση της σύνταξης της RDF. Εάν αυτό είχε επιτραπεί, η OWL θα μπορούσε να έχει προσθέσει τη

νέα, φυσική σύνταξη για τα κατασκευάσματά της η σημασιολογία των οποίων δεν θα είχε απαιτηθεί για να φέρει κατά μήκος μια τριπλή έννοια της RDF.

Εντούτοις μια βιώσιμη λύση έχει βρεθεί που ικανοποιεί όλες τις παραπάνω απαιτήσεις. Ή, πραγματικά, είναι ακριβέστερο να πούμε ότι τρεις λύσεις έχουν βρεθεί, κάθε μια από τις οποίες ικανοποιεί σχεδόν όλες τις ανωτέρω απαιτήσεις.

OWL DL: Εάν η φιλική σύνταξη ή η αποφασιστικότητα ενός συμπεράσματος έχει πρωταρχική σπουδαιότητα, η OWL DL, μια έκδοση της OWL με αποφασιστικότητα στο συμπέρασμα που μπορεί να γραφτεί σε ένα πλαίσιο ή μια περιγραφή ο τρόπος λογικής, είναι κατάλληλος.

OWL Lite: Εάν μια ομαλά απλούστερη σύνταξη και κάτι πιο εύκολο στο να βγει συμπέρασμα έχουν πρωταρχική σπουδαιότητα, η OWL, ένα συντακτικό θέμα της OWL DL, θα ήταν το κατάλληλο.

OWL Full: Εάν η συμβατότητα με RDF και RDFS έχουν πρωταρχική σπουδαιότητα, κατόπιν η OWL Full, μια συντακτική και σημασιολογική επέκταση RDFS, είναι κατάλληλη.

### **2.3.6.1 Η αναγνωσιμότητα**

Όπως παρουσιάζεται στα παραπάνω παραδείγματα, η OWL δεν είναι πολύ αναγνώσιμη όταν γράφεται ως RDF/XML ή ακόμα και όταν ως RDF τριπλασιάζεται. Μέρος αυτού του προβλήματος είναι ότι η RDF/XML είναι εξαιρετικά φλύαρη, αλλά το σημαντικότερο μέρος του προβλήματος αναγνωσιμότητας είναι η κωδικοποίηση των κατασκευασμάτων OWL στα τριπλάσια των RDF/XML ή RDF.

Για να εξετάσει εν μέρει αυτό το πρόβλημα, μια αφηρημένη σύνταξη δημιουργήθηκε για την OWL, μαζί με μια χαρτογράφηση από την αφηρημένη σύνταξη στις γραφικές παραστάσεις της RDF. Αυτή η αφηρημένη σύνταξη είναι πιο στενή στη σύνταξη της OIL απ'ότι στη DAML+OIL, αλλά χωρίς ακραία έμφαση της OIL στην αναγνωσιμότητα. Σε αυτήν την αφηρημένη σύνταξη το παράδειγμα σπουδαστών ανωτέρω θα γραφόταν

```
Class(Student complete
    Pearson
    restriction(enrolledIn minCardinality (1))).
```

Η OWL DL ορίστηκε έπειτα ως το συντακτικό υποσύνολο της OWL που προκλήθηκε από τη μετάφραση από την αφηρημένη σύνταξη στις γραφικές παραστάσεις της RDF. Δηλαδή μια γραφική παράσταση της RDF είναι μια οντολογία της OWL DL όταν είναι η μετάφραση κάποιας οντολογίας στην αφηρημένη σύνταξη. Οι χρήστες και τα εργαλεία που ενδιαφέρονται για την αναγνωσιμότητα απ' ό,τι για RDF/XML μπορούν να χρησιμοποιήσουν αυτήν την αφηρημένη σύνταξη εσωτερικά, ή ακόμα και εξωτερικά για την παρουσίαση στους χρήστες, που διατηρούν τη σύνταξη RDF/XML για λόγους της ανταλλαγής μεταξύ των εφαρμογών.

### **2.3.6.2 Διαχειρίζοντας τις μη καλά σχεδιασμένες γραφικές παραστάσεις**

Επειδή η OWL Full επιτρέπει τις αυθαίρετες γραφικές παραστάσεις RDF, πρέπει να είναι σε θέση να χειριστεί και τις μη καλά φτιαγμένη σύνταξη της OWL. (Η OWL DL δεν υποφέρει από αυτό το πρόβλημα όπως καθορίζεται από την άποψη των απαραίτητων καλοσηματισμένων γραφικών παραστάσεων RDF που μπορούν να παραχθούν από την αφηρημένη σύνταξη.) Η OWL χρησιμοποιεί μια επέκταση της λύσης DAML+OIL: μόνο στα τριπλάσια που διαμορφώνουν μαζί τα καλοσηματισμένα κατασκευάσματα OWL δίνοντας ένα πρόσθετο νόημα, έτσι

```
:x owl:onProperty ex:child.
```

από μόνο του δεν έχει οποιαδήποτε ειδική έννοια στην OWL.

Για να χειριστεί τις περιπτώσεις πάρα πολλών τριπλάσιων, η OWL χρησιμοποιεί πάλι τη λύση DAML+OIL, να διαλέξει δηλαδή μεταξύ των καλοσηματισμένων υποσυνόλων και να τους δώσει το νόημα της OWL. Αυτό όμως έχει τις ασυνήθιστες συνέπειες -- παραδείγματος χάριν

```
_:x owl:onProperty ex:child.
```

```
_:x owl:someValuesFrom ex:Person.
```

```
_:x owl:onProperty ex:friend.
```

```
_:x owl:allValuesFrom ex:Doctor.
```

καταλήγοντας στην επέκταση τεσσάρων διαφορετικών περιορισμών της OWL (όλοι οι πιθανοί συνδυασμοί των δύο ιδιοτήτων με τις δύο κατηγορίες), που δεν είναι σχεδόν βέβαιο τι προορίστηκε από το χρήστη. Αυτή η λύση, εντούτοις, διατηρεί μονοτονία, και

(ενδεχομένως) η μη-διαισθητική έννοια είναι ένα δεύτερο πρόβλημα δεδομένου ότι τέτοιες όχι καλά σχεδιασμένες κατασκευές μπορούν εύκολα να αποφευχθούν.

Η έλλειψη δομής στις γραφικές παραστάσεις RDF πρέπει να αντιμετωπιστούν με τα σημασιολογικά μέσα.

### **2.3.6.3 Η παροχή μιας σημασιολογικής θεωρίας για την OWL**

Καθώς η RDF έχει τώρα μια πρότυπη θεωρία, με μια ολοκληρωμένη έννοια του συμπεράσματος, η OWL πρέπει να παρέχει μια προς τα πάνω συμβατή πρότυπη θεωρία που χειρίζεται κατάλληλα τα συμπεράσματα πέρα από τα κατασκευάσματα της OWL. Αυτό θα ήταν εύκολο εάν η OWL ήταν σε θέση να επεκτείνει τη σύνταξη της RDF, δεδομένου ότι έπειτα αυτά τα νέα κομμάτια της σύνταξης μπόρεσαν να έχουν μια μόνο την έννοια της OWL. Εντούτοις, ήταν μια απαίτηση η OWL να έχει τη σύνταξη της RDF, και αυτή η σύνταξη να έχει όλη την έννοια της RDF. Αυτή η απαίτηση για συμβατότητα διπλής κατεύθυνσης είναι πολύ ισχυρότερη από αυτή που επιβάλλεται συνήθως μεταξύ των πιο αδύνατων φορμαλισμών (όπως την προτασιακή λογική) και των ισχυρότερων φορμαλισμών (όπως την αρχική λογική) όπου ο ισχυρότερος φορμαλισμός επιτρέπεται για να επεκτείνει τη σύνταξη του πιο αδύνατου φορμαλισμού.

Η αυστηρότερη πτυχή αυτού του προβλήματος είναι ότι τα συντακτικά κατασκευάσματα της OWL που κωδικοποιούνται ως πολλαπλά τριπλάσια πρέπει να διατηρήσουν την έννοια της RDF για αυτά τα τριπλάσια. Δεδομένου ότι όλα τα τριπλάσια της RDF φέρνουν τους Semantic όρους, δεν μπορούν ακριβώς να προκύψουν από τίποτα. Άντ' αυτού, η σημασιολογία της OWL έπρεπε να προσθέσει τους ειδικούς περιορισμούς που ουσιαστικά δηλώνουν ότι κάθε ερμηνεία της OWL πρέπει να περιλάβει ορισμένα κατασκευάσματα. (Τέτοιοι περιορισμοί καλούνται συνήθως αρχές κατανόησης.) Παραδείγματος χάριν, μια αρχή κατανόησης για την OWL δηλώνει ότι κάθε πρότυπο πρέπει να περιλάβει όλους τους πεπερασμένους καταλόγους κατηγοριών, άλλος δηλώνει ότι κάθε τέτοιος κατάλογος πρέπει να έχει μια αντίστοιχη κατηγορία διατομής, με την απαίτηση ότι υπάρχει κάποια κατηγορία που συνδέει τον κατάλογο με μια ιδιοκτησία `owl:intersectionOf`.

Αυτές οι αρχές κατανόησης υποστηρίζουν τα συμπεράσματα από  
`ex:John rdf:type ex:Student.`

ex:John rdf:type ex:Employee.

to

\_:c owl:intersectionOf :I1.

\_:I1 rdf:first ex:Student.

\_:I1 rdf:rest :I2.

\_:I2 rdf:first ex:Employee.

\_:I2 rdf:nil rdf:rest.

ex:John rdf:type \_:c.

επειδή το `_:I1` μπορεί να είναι ο απαραίτητος κατάλογος του σπουδαστή και του υπαλλήλου, και το `I2` μπορεί να είναι η απαραίτητη ουρά αυτού του καταλόγου, και `:c` μπορεί να είναι η απαραίτητη διατομή του καταλόγου. Οι πρόσθετοι (και περισσότερο συνηθισμένοι) σημασιολογικοί όροι απαιτούν ότι `ex:John` ανήκουν στο `_:c`, η λήξη όλων αυτών απαιτείται για να κρατηθούν τα συμπεράσματα.

#### 2.3.6.4 Αποφεύγοντας τα παράδοξα

Οι αρχές κατανόησης είναι πολύ ισχυρές, δεδομένου ότι δημιουργούν κάτι από τίποτα (ή, τουλάχιστον, κάτι από πολύ λίγα). Αυτή η δύναμη μπορεί εύκολα να οδηγήσει στα σοβαρά προβλήματα.

Παραδείγματος χάριν, το παράδοξο του Russell είναι ένα παράδοξο ακριβώς λόγω των αρχών κατανόησης που χτίζονται σε μια πρόωρη έκδοση της καθορισμένης θεωρίας. Αυτή η πρόωρη έκδοση της καθορισμένης θεωρίας είχε μια αρχή κατανόησης ότι ένα σύνολο θα μπορούσε να κατασκευάσει τα πράγματα που ικανοποίησαν έναν τύπο με μια ελεύθερη μεταβλητή -- π.χ., ο τύπος της ύπαρξης του να είσαι άνθρωπος,  $x \in \text{human-άνθρωπος}$ , θα μπορούσαν να χρησιμοποιηθούν για να κατασκευάσουν το σύνολο ανθρώπων,  $\{x \mid x \in \text{human-άνθρωπος}\}$ . Δυστυχώς, από τον τύπο προκύπτει να μην ανήκει σε κάποιο,  $x \notin x$ , το σύνολο  $\{x \mid x \notin x\}$  προκύπτει. Αυτό το σύνολο προκαλεί τα προβλήματα οπουδήποτε υπάρχει επειδή είναι αδύνατο να καθοριστεί εάν ανήκει κάπου. Οι εξουσιοδοτήσεις αρχής κατανοούν ότι υπάρχει παντού, προκαλώντας κατά συνέπεια αυτήν την πρόωρη έκδοση της καθορισμένης θεωρίας στην κατάρρευση.

Μια παρόμοια κατάσταση μπορεί να προκύψει με την OWL. Φαίνεται φυσικό η OWL να θελήσει να έχει κυκλικό όπως τα κατασκευάσματα, παραδείγματος χάριν κατηγορίες

των οποίων περιπτώσεις συσχετίζονται μόνο με άλλες περιπτώσεις της κατηγορίας, όπως :

```
_:c owl:onProperty ex:child.
```

\_:c owl:allValuesFrom \_:c. Αυτό μπορεί να χρησιμοποιηθεί σε μια αφελή αντιπροσώπευση μερικών πτυχών της βιολογίας.

Εντούτοις, η κατοχή των αρχών κατανόησης για τέτοιες κυκλικές κατηγορίες μπορεί εύκολα να οδηγήσει σε μια απαίτηση για την ύπαρξη των κατηγοριών όπως

```
_:c owl:onProperty rdf:type.
```

```
_:c owl:allValuesFrom _:d.
```

```
_:d owl:complementOf _:l.
```

```
_:l rdf:first _:c.
```

```
_:l rdf:rest rdf:nil.
```

που είναι η κατηγορία πραγμάτων που δεν έχουν σχέση τύπων στην ίδια στην κατηγορία. Τα αντικείμενα που ανήκουν σε αυτήν την κατηγορία δεν μπορούν να ανήκουν σε αυτό, και αντίστροφα, έτσι εάν οι αρχές κατανόησης απαιτούσαν την ύπαρξη αυτής της κατηγορίας, κάθε οντολογία OWL θα ήταν παράδοξη.

Για να αποφύγουν αυτά τα παράδοξα, οι αρχές κατανόησης OWL δεν απαιτούν ποτέ την ύπαρξη των κυκλικών αλυσίδων της αναφοράς όπως παραπάνω. Εντούτοις, αυτό σημαίνει ότι αυτά που συνεπάγονται είναι αυτά που αναμένονται, όπως αυτά που περιλαμβάνει το κατασκευάσμα με `ex:child` ανωτέρω, παραδείγματος χάριν έχοντας ένα πρόσωπο χωρίς τα παιδιά ανήκετε σε ένα τέτοιο κατασκευάσμα, αυτά στην OWL δεν ισχύουν. Η επινόηση αυτών των αρχών κατανόησης πήρε ένα μεγάλο μέρος της προσπάθειας (ένα μεγάλο μέρος του οποίου περιέλαβε τον καθορισμό των βασικών κανόνων για τις αρχές).

### **2.3.6.5 Η διατήρηση της αποφασιστικότητας**

Το σύνολο της OWL είναι αναποφάσιστο (για διάφορους λόγους), και ακόμη και η OWL DL θα μπορούσε εύκολα να είναι αναποφάσιστη εάν περιλάμβανε ορισμένα κατασκευάσματα που ήταν γνωστά για να προκαλούν την αναποφασιστικότητα στις λογικές περιγραφές. Επομένως η OWL DL επεξεργάστηκε προσεκτικά για να παραμείνει αποφασιστική, και δεν περιλαμβάνει, παραδείγματος χάριν, τις σχέσεις μεταξύ των

αλυσίδων ρόλου, οι οποίες θα προκαλούσαν την αναποφασιστικότητα με την ενσωμάτωση της λέξης πρόβλημα στην OWL DL.

Αυτό δεν πρόκειται να πει ότι το συμπέρασμα στην OWL DL είναι εύκολο. Η OWL DL έχει ένα δυσκολία στο πρόβλημα συνεπαγόμενων, δεδομένου ότι το συμπέρασμα σε SHOIN (D) είναι στη χειρότερη περίπτωση όχι τυχαία στην εκθετική χρονική (NExpTime) πολυπλοκότητας, και η OWL DL πρέπει να έχει την ίδια πολυπλοκότητα. Χειρότερα, δεν υπάρχει ακόμα κανένας γνωστός "πρακτικός" πλήρης αλγόριθμος για το συμπέρασμα σε SHOIN (D), δηλ., κάτι που είναι πιθανό να αποδώσει καλά στα είδη προβλήματος που αντιμετωπίζονται στις χαρακτηριστικές εφαρμογές. Ελλείψει ενός τέτοιου αλγόριθμου, η συμπεριφορά των αιτιολογητών της OWL DL είναι πιθανό να είναι λιγότερο προβλέψιμη (και από την άποψη του χρόνου που λαμβάνονται για να αποκριθούν στις ερωτήσεις, και της χρήσης των πόρων συστημάτων), και μπορούν μερικές φορές να επιστρέψουν την απάντηση "άγνωστη" σε απάντηση της ερώτησης.

Η OWL Lite είναι καλύτερη εν προκειμένω. Το συμπέρασμα σε SHIF (D) είναι στη χειρότερη περίπτωση αιτιοκρατικής εκθετικής χρονικής (ExpTime) πολυπλοκότητας, και η OWL Lite έχει την ίδια πολυπλοκότητα. Επιπλέον, υπάρχουν πρακτικοί βελτιστοποιημένοι αλγόριθμοι για το συμπέρασμα στην OWL Lite, όπως ο αλγόριθμος που κρύβεται κάτω από τα συστήματα FaCT λογικής περιγραφής και RACER. Αυτά τα συστήματα έχουν αποδειχθεί για να εργαστούν καλά στις ρεαλιστικές εφαρμογές και για να μπορούν να αιτιολογούν τις μεγάλες οντολογίες.

### **2.3.7 OWL**

Εδώ περιγράφεται πώς οι λύσεις που αναλύθηκαν παραπάνω έχουν ενσωματωθεί στο τελικό σχέδιο της γλώσσας OWL. Δεν προορίζεται δεδομένου ότι σε μια πλήρη περιγραφή της γλώσσας οι αναγνώστες πρέπει να γυρίσουν στα W3C έγγραφα σχετικά με την OWL.

Για διάφορους λόγους, που περιγράφονται στα προηγούμενα τμήματα, υπάρχουν δύο μορφές της χρησιμοποίησης της OWL. Στο πρώτο ύφος, που ενσωματώνεται η OWL DL η OWL Lite, που μόνο ορισμένες κατασκευές επιτρέπονται, και αυτές οι κατασκευές μπορούν να συνδυαστούν μόνο με ορισμένους τρόπους. Τα οφέλη μέσα σε αυτούς τους περιορισμούς περιλαμβάνουν την αποφασιστικότητα των συμπερασμάτων και τη



δυνατότητα της σκέψης την OWL σε μια περισσότερο πρότυπη μόδα, ουσιαστικά ως εκφραστική λογική περιγραφής. Στο δεύτερο ύφος, που ενσωματώνεται στο σύνολο OWL, όλες οι γραφικές παραστάσεις RDF επιτρέπονται. Τα οφέλη αυτού του επεκτατικού ύφους περιλαμβάνουν τη συνολική ανοδική συμβατότητα με RDF και μια μεγαλύτερη εκφραστική δύναμη.

Ακόμη και οι περισσότερες-περιορισμένες εκδόσεις της OWL έχουν μερικές διαφορές από τις τυποποιημένες λογικές περιγραφές. Αυτές οι διαφορές κινούν αυτές τις εκδόσεις της OWL από τον επίσημο κόσμο της λογικής περιγραφής προς το κόσμο του Semantic Web.

- Ø Η OWL χρησιμοποιεί URI αναφορές ως ονόματα. Αυτές οι αναφορές URI κατασκευάζονται με τον ίδιο τρόπο που χρησιμοποιείται από την RDF. Είναι έτσι κοινό η OWL να χρησιμοποιεί τα κατάλληλα ονόματα ως συντομογραφίες για τις αναφορές URI, παραδείγματος χάριν, το κατάλληλο όνομα owl:Thing για τη URI αναφορά <http://www.w3.org/2002/07/owl#Thing>.
- Ø Η OWL συγκεντρώνει πληροφορίες για τις οντολογίες, οι οποίες αποθηκεύονται γενικά ως έγγραφα Web που γράφονται σε RDF/XML. Οι οντολογίες μπορούν να εισαγάγουν άλλες οντολογίες, προσθέτοντας τις πληροφορίες από την εισαγόμενη οντολογία στην τρέχουσα οντολογία.
- Ø Ακόμη και το DL/Lite ύφος που χρησιμοποιεί η OWL επιτρέπει τις ιδιότητες σχολιασμών της RDF να χρησιμοποιηθεί για να συνδέσει τις πληροφορίες με τις κατηγορίες, τις ιδιότητες, και τις οντολογίες, όπως owl:DeprecatedClass. Αυτοί οι σχολιασμοί είναι τριπλάσια της RDF, και επομένως απαιτούνται για να φέρουν ένα πλήρες σημασιολογικό βάρος. Δεν μπορούν να αντιμετωπιστούν ως άτυπα σχόλια χωρίς μια επίσημη έννοια. Αυτό χωρίζει εν μέρει τη σταθερή διάκριση λογικής περιγραφής μεταξύ των ατόμων, αφ' ενός, και των κατηγοριών και των ιδιοτήτων, αφ' ετέρου .
- Ø Η OWL χρησιμοποιεί τις ικανότητες του τύπου δεδομένων της RDF και τον XML Schema τύπο δεδομένων για να παρέχει τύπους δεδομένων και τις τιμές των στοιχείων.

Ø Οι DL και Lite εκδόσεις της OWL έχουν ένα πλαίσιο-όπως μια αφηρημένη σύνταξη, ενώ η RDF/XML είναι η επίσημη σύνταξη ανταλλαγής για την όλη την OWL.

### 2.3.7.1 Η OWL ως περιγραφική λογική

Η OWL DL, δηλαδή η λογική περιγραφής που χρησιμοποιεί η OWL, είναι πολύ κοντά στη SHOIN (D) λογική περιγραφής που είναι μια επέκταση της επιδρούσας SHOQ (D) λογικής περιγραφής (εκτεταμένος με τους αντίστροφους ρόλους και περιορισμένος στους αναρμόδιους αριθμητικούς περιορισμούς). Η OWL DL μπορεί να διαμορφώσει τις περιγραφές των κατηγοριών, των τύπων δεδομένων, των ατόμων και των τιμών στοιχείων χρησιμοποιώντας τα κατασκευάσματα που παρουσιάζονται στο σχήμα 2.5. Σε αυτόν τον πίνακα η πρώτη στήλη δίνει στην OWL την αφηρημένη σύνταξη για την κατασκευή, ενώ η δεύτερη στήλη δίνει την τυποποιημένη σύνταξη λογικής περιγραφής.

Η OWL DL χρησιμοποιεί αυτούς τους μηχανισμούς διαμόρφωσης και περιγραφής στα αξιώματα που παρέχουν τις πληροφορίες για τις κατηγορίες, τις ιδιότητες, και τα άτομα, όπως φαίνεται στο σχήμα 2.5. Πάλι, το πλαίσιο που μοιάζει με μια αφηρημένη σύνταξη δίνεται στην πρώτη στήλη, και η τυποποιημένη σύνταξη λογικής περιγραφής δίνεται στη δεύτερη στήλη.

### 2.3.7.2 Σημασιολογία για την OWL DL

Μια επίσημη σημασιολογία, πολύ παρόμοια με τη σημασιολογία με αυτή που έχουν οι λογικές περιγραφής, παρέχεται για αυτό το ύφος με τη χρησιμοποίηση της OWL. Τα πλήρη στοιχεία σε αυτήν την πρότυπη θεωρία μπορούν να βρεθούν στη σημασιολογία της OWL και στην αφηρημένη σύνταξη.

Επειδή η OWL περιλαμβάνει τύπους δεδομένων, η σημασιολογία για την OWL είναι πολύ παρόμοια με αυτήν των λογικών περιγραφής που ενσωματώνουν επίσης τύπους δεδομένων, ειδικότερα στην SHOQ (D). Εντούτοις, οι ιδιαίτεροι τύποι δεδομένων που χρησιμοποιούνται στην OWL λαμβάνονται από τύπους δεδομένων των RDF και της XML Schema. Οι τιμές στοιχείων όπως "44"^^xsd:integer σημαίνουν ότι τι θα σήμαιναν ως τιμές στοιχείων της XML Schema.

Η συγκεκριμένη έννοια που δίνεται στις περιγραφές OWL DL παρουσιάζεται στην τρίτη στήλη του σχήματος 2.5, όπου το  $\Delta^I$  είναι η περιοχή των ατόμων σε ένα πρότυπο και  $\Delta_D^I$

Abstract Syntax	DL Syntax	Semantics
<b>Descriptions (<math>C</math>)</b>		
$A$ (URI reference)	$A$	$A^I \subseteq \Delta^I$
<code>owl:Thing</code>	$\top$	$\text{owl:Thing}^I = \Delta^I$
<code>owl:Nothing</code>	$\perp$	$\text{owl:Nothing}^I = \{\}$
<code>intersectionOf(<math>C_1 C_2 \dots</math>)</code>	$C_1 \sqcap C_2$	$(C_1 \sqcap C_2)^I = C_1^I \cap C_2^I$
<code>unionOf(<math>C_1 C_2 \dots</math>)</code>	$C_1 \sqcup C_2$	$(C_1 \sqcup C_2)^I = C_1^I \cup C_2^I$
<code>complementOf(<math>C</math>)</code>	$\neg C$	$(\neg C)^I = \Delta^I \setminus C^I$
<code>oneOf(<math>o_1 \dots</math>)</code>	$\{o_1, \dots\}$	$\{o_1, \dots\}^I = \{o_1^I, \dots\}$
<code>restriction(<math>R</math> someValuesFrom(<math>C</math>))</code>	$\exists R.C$	$(\exists R.C)^I = \{x \mid \exists y. \langle x, y \rangle \in R^I \text{ and } y \in C^I\}$
<code>restriction(<math>R</math> allValuesFrom(<math>C</math>))</code>	$\forall R.C$	$(\forall R.C)^I = \{x \mid \forall y. \langle x, y \rangle \in R^I \rightarrow y \in C^I\}$
<code>restriction(<math>R</math> hasValue(<math>o</math>))</code>	$R : o$	$(R : o)^I = \{x \mid \langle x, o^I \rangle \in R^I\}$
<code>restriction(<math>R</math> minCardinality(<math>n</math>))</code>	$\geq n R$	$(\geq n R)^I = \{x \mid \#\{y. \langle x, y \rangle \in R^I\} \geq n\}$
<code>restriction(<math>R</math> maxCardinality(<math>n</math>))</code>	$\leq n R$	$(\leq n R)^I = \{x \mid \#\{y. \langle x, y \rangle \in R^I\} \leq n\}$
<code>restriction(<math>U</math> someValuesFrom(<math>D</math>))</code>	$\exists U.D$	$(\exists U.D)^I = \{x \mid \exists y. \langle x, y \rangle \in U^I \text{ and } y \in D^I\}$
<code>restriction(<math>U</math> allValuesFrom(<math>D</math>))</code>	$\forall U.D$	$(\forall U.D)^I = \{x \mid \forall y. \langle x, y \rangle \in U^I \rightarrow y \in D^I\}$
<code>restriction(<math>U</math> hasValue(<math>v</math>))</code>	$U : v$	$(U : v)^I = \{x \mid \langle x, v^I \rangle \in U^I\}$
<code>restriction(<math>U</math> minCardinality(<math>n</math>))</code>	$\geq n U$	$(\geq n U)^I = \{x \mid \#\{y. \langle x, y \rangle \in U^I\} \geq n\}$
<code>restriction(<math>U</math> maxCardinality(<math>n</math>))</code>	$\leq n U$	$(\leq n U)^I = \{x \mid \#\{y. \langle x, y \rangle \in U^I\} \leq n\}$
<b>Data Ranges (<math>D</math>)</b>		
$D$ (URI reference)	$D$	$D^I \subseteq \Delta_D^I$
<code>oneOf(<math>v_1 \dots</math>)</code>	$\{v_1, \dots\}$	$\{v_1, \dots\}^I = \{v_1^I, \dots\}$
<b>Object Properties (<math>R</math>)</b>		
$R$ (URI reference)	$R$	$R^I \subseteq \Delta^I \times \Delta^I$
	$R^-$	$(R^-)^I = (R^I)^-$
<b>Datatype Properties (<math>U</math>)</b>		
$U$ (URI reference)	$U$	$U^I \subseteq \Delta^I \times \Delta_D^I$
<b>Individuals (<math>o</math>)</b>		
$o$ (URI reference)	$o$	$o^I \in \Delta^I$
<b>Data Values (<math>v</math>)</b>		
$v$ (RDF literal)	$v$	$v^I = v^D$

Σχήμα 2.5 : OWL DL Περιγραφές, Σειρές δεδομένων, Ιδιότητες, Τύποι και Τιμές Δεδομένων

Abstract Syntax	DL Syntax	Semantics
Class( <i>A</i> partial $C_1 \dots C_n$ )	$A \sqsubseteq C_1 \sqcap \dots \sqcap C_n$	$A^{\mathcal{I}} \subseteq C_1^{\mathcal{I}} \cap \dots \cap C_n^{\mathcal{I}}$
Class( <i>A</i> complete $C_1 \dots C_n$ )	$A = C_1 \sqcap \dots \sqcap C_n$	$A^{\mathcal{I}} = C_1^{\mathcal{I}} \cap \dots \cap C_n^{\mathcal{I}}$
EnumeratedClass( <i>A</i> $a_1 \dots a_n$ )	$A = \{a_1, \dots, a_n\}$	$A^{\mathcal{I}} = \{a_1^{\mathcal{I}}, \dots, a_n^{\mathcal{I}}\}$
SubClassOf( $C_1 C_2$ )	$C_1 \sqsubseteq C_2$	$C_1^{\mathcal{I}} \subseteq C_2^{\mathcal{I}}$
EquivalentClasses( $C_1 \dots C_n$ )	$C_1 = \dots = C_n$	$C_1^{\mathcal{I}} = \dots = C_n^{\mathcal{I}}$
DisjointClasses( $C_1 \dots C_n$ )	$C_i \sqcap C_j = \perp, i \neq j$	$C_i^{\mathcal{I}} \cap C_j^{\mathcal{I}} = \emptyset, i \neq j$
Datatype( <i>D</i> )		$D^{\mathcal{I}} \subseteq \Delta_D^{\mathcal{I}}$
DatatypeProperty( <i>U</i> super( $U_1 \dots U_n$ ))	$U \sqsubseteq U_i$	$U^{\mathcal{I}} \subseteq U_i^{\mathcal{I}}$
domain( $C_1 \dots C_m$ )	$\geq 1 U \sqsubseteq C_i$	$U^{\mathcal{I}} \subseteq C_i^{\mathcal{I}} \times \Delta_D^{\mathcal{I}}$
range( $D_1 \dots D_l$ )	$\top \sqsubseteq \forall U. D_i$	$U^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times D_i^{\mathcal{I}}$
{Functional}	$\top \sqsubseteq \leq 1 U$	$U^{\mathcal{I}}$ is functional
SubPropertyOf( $U_1 U_2$ )	$U_1 \sqsubseteq U_2$	$U_1^{\mathcal{I}} \subseteq U_2^{\mathcal{I}}$
EquivalentProperties( $U_1 \dots U_n$ )	$U_1 = \dots = U_n$	$U_1^{\mathcal{I}} = \dots = U_n^{\mathcal{I}}$
ObjectProperty( <i>R</i> super( $R_1 \dots R_m$ ))	$R \sqsubseteq R_i$	$R^{\mathcal{I}} \subseteq R_i^{\mathcal{I}}$
domain( $C_1 \dots C_m$ )	$\geq 1 R \sqsubseteq C_i$	$R^{\mathcal{I}} \subseteq C_i^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
range( $C_1 \dots C_l$ )	$\top \sqsubseteq \forall R. C_i$	$R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times C_i^{\mathcal{I}}$
{InverseOf( $R_0$ )}	$R = (^- R_0)$	$R^{\mathcal{I}} = (R_0^{\mathcal{I}})^{-}$
{Symmetric}	$R = (^- R)$	$R^{\mathcal{I}} = (R^{\mathcal{I}})^{-}$
{Functional}	$\top \sqsubseteq \leq 1 R$	$R^{\mathcal{I}}$ is functional
{InverseFunctional}	$\top \sqsubseteq \leq 1 R^{-}$	$(R^{\mathcal{I}})^{-}$ is functional
{Transitive}	$Tr(R)$	$R^{\mathcal{I}} = (R^{\mathcal{I}})^+$
SubPropertyOf( $R_1 R_2$ )	$R_1 \sqsubseteq R_2$	$R_1^{\mathcal{I}} \subseteq R_2^{\mathcal{I}}$
EquivalentProperties( $R_1 \dots R_m$ )	$R_1 = \dots = R_m$	$R_1^{\mathcal{I}} = \dots = R_m^{\mathcal{I}}$
AnnotationProperty( <i>S</i> )		
Individual( <i>a</i> type( $C_1 \dots C_n$ ))	$a \in C_i$	$a^{\mathcal{I}} \in C_i^{\mathcal{I}}$
value( $R_1 a_1 \dots R_m a_m$ )	$\langle a, a_i \rangle \in R_i$	$\langle a^{\mathcal{I}}, a_i^{\mathcal{I}} \rangle \in R_i^{\mathcal{I}}$
value( $U_1 v_1 \dots U_n v_n$ )	$\langle a, v_i \rangle \in U_i$	$\langle a^{\mathcal{I}}, v_i^{\mathcal{I}} \rangle \in U_i^{\mathcal{I}}$
SameIndividual( $a_1 \dots a_n$ )	$a_1 = \dots = a_n$	$a_1^{\mathcal{I}} = \dots = a_n^{\mathcal{I}}$
DifferentIndividuals( $a_1 \dots a_n$ )	$a_i \neq a_j, i \neq j$	$a_i^{\mathcal{I}} \neq a_j^{\mathcal{I}}, i \neq j$

Σχήμα 2.6 : OWL DL Αξιώματα και γεγονότα

είναι η περιοχή των τιμών των στοιχείων. Όπως συνήθως, η έννοια των αξιωμάτων δίνεται από τους περιορισμούς στα πρότυπα, όπως φαίνεται στην τρίτη στήλη του σχήματος 2.6.

Η σημασιολογία για την OWL DL περιλαμβάνει μερικές ασυνήθιστες (για τις λογικές περιγραφής) πτυχές. Στους σχολιασμούς δίνεται ένα απλό χωριστό νόημα, που δεν παρουσιάζεται εδώ, το οποίο μπορεί να χρησιμοποιηθεί στις συνδυαζόμενες πληροφορίες με τις κατηγορίες, τις ιδιότητες, και τα άτομα κατά τρόπο συμβατό με τη σημασιολογία της RDF. Μέσα στη σημασιολογία των οντολογιών υπάρχουν και μπορούν να δοθούν οι πληροφορίες σχολιασμών. Τέλος, σε owl:imports δίνεται ένα νόημα που περιλαμβάνει την εύρεση της παραπεμφθείσας οντολογίας (αν είναι δυνατόν) και την προσθήκη της έννοιάς του στην έννοια της τρέχουσας οντολογίας.

Αυτό που κάνει την OWL DL μια σημασιολογική γλώσσα Web, δεν είναι η σημασιολογία της, που είναι αρκετά τυποποιημένη για μια λογική περιγραφής, αλλά άντ' αυτού τη χρήση των αναφορών URI για τα ονόματα, τη χρήση των τύπων

δεδομένων της XML Schema για τις τιμές στοιχείων, και τη δυνατότητα να συνδεθούν με τα έγγραφα στο World Wide Web.

### **2.3.7.3 Μια ευκολότερη OWL: Η OWL Lite**

Η OWL DL συσχετίζεται με την SHOIN(D), μια πολύ εκφραστική λογική περιγραφής. Αυτή η λογική περιγραφής πρόκειται να δυσκολευτεί κάπως να παρουσιαστεί στους αφελείς χρήστες, όπως είναι δυνατό να χτιστεί χρησιμοποιώντας μπουουλιανές περιγραφές, παραδείγματος χάριν, την ένωση και το συμπλήρωμα. Η SHOIN(D) είναι δύσκολο επίσης να αιτιολογήσει, δεδομένου ότι τα βασικά προβλήματα συμπεράσματος έχουν την πολυπλοκότητα NExpTime, και κάποια δυσκολία για να χτίσουν ακόμη και τα εργαλεία μη-συλλογισμού, λόγω των σύνθετων περιγραφών.

Για αυτούς τους λόγους, ένα υποσύνολο της OWL DL έχει προσδιοριστεί που πρέπει να είναι ευκολότερο σε όλες τις ανωτέρω μετρικές αυτό το υποσύνολο καλείται OWL Lite. Η OWL Lite απαγορεύει τις ενώσεις και τα συμπληρώματα, περιορίζει τις διατομές στις υπονοούμενες διατομές στα πλαίσιο-όπως τα αξιώματα κατηγορίας, περιορίζει όλες τις ενσωματωμένες περιγραφές στα ονόματα έννοιας, δεν επιτρέπει στα άτομα να παρουσιάσουν στις περιγραφές ή τα αξιώματα κατηγορίας, και περιορίζει τα σημαντικά σε 0 ή 1.

Αυτοί οι περιορισμοί κάνουν την OWL Lite παρόμοια με τη λογική περιγραφής SHIF (D). Όπως η SHIF (D), τα βασικά συμπεράσματα στην OWL Lite μπορούν να υπολογιστούν στη χειρότερη περίπτωση σε εκθετικό χρόνο (ExpTime), και υπάρχουν ήδη διάφορα βελτιωμένοι λόγοι για τις λογικές ισοδύναμες με την OWL Lite. Αυτή η βελτίωση στην ευπείθεια έρχεται με σχετικά λίγη απώλεια στην εκφραστική δύναμη - - αν και η σύνταξη OWL Lite είναι πιο περιορισμένη από αυτή της OWL DL που είναι ακόμα δυνατό να εκφραστούν οι σύνθετες περιγραφές με την εισαγωγή των νέων ονομάτων κατηγορίας και την εκμετάλλευση των υπονοούμενων αρνήσεων που εισάγονται από τα αξιώματα για αυτά που δεν ενώνονται. Χρησιμοποιώντας αυτές τις τεχνικές, όλες οι περιγραφές OWL DL μπορούν να συλληφθούν στην OWL Lite εκτός από εκείνους που περιέχουν είτε τα μεμονωμένα ονόματα είτε τα σημαντικά μεγαλύτερα από το σύνολο 1.

#### **2.3.7.4 OWL Full όπως η επέκταση RDF**

Η OWL DL και η OWL Lite είναι επεκτάσεις μιας περιορισμένης χρήσης των RDF και RDFS, επειδή, αντίθετα από τις RDF και RDFS, δεν επιτρέπουν στις κατηγορίες να χρησιμοποιηθούν ως άτομα, και έτσι οι γλωσσικοί κατασκευαστές δεν μπορούν να απευθυνθούν οι ίδιοι στη γλώσσα. Για τους χρήστες που χρειάζονται αυτές τις ικανότητες, μια έκδοση της OWL που είναι συμβατή προς τα πάνω με τις RDF και την RDFS έχει παρασχεθεί η έκδοση που καλείται OWL Full. Στην OWL Full, όλοι οι συνδυασμοί των RDF και RDFS επιτρέπονται. Παραδείγματος χάριν, στη OWL Full, είναι δυνατό στο `empose` ένας περιορισμός του συνόλου των αριθμητικών στοιχείων σε `rdfs:subClassOf`, εάν έτσι επιδιώκεται.

Η OWL Full περιέχει την OWL DL, αλλά στέκεται καλά έξω από το τυποποιημένο πλαίσιο λογικής περιγραφής. Η ποινική ρήτρα που πληρώνεται εδώ είναι διπλή. Κατ' αρχάς, ο συλλογισμός στην OWL Full είναι αναποφάσιτος (επειδή οι περιορισμοί που απαιτούνται προκειμένου να διατηρηθεί η αποφασιστικότητα της OWL DL δεν ισχύει για το σύνολο της OWL ). Δεύτερον, η αφηρημένη σύνταξη για την OWL DL είναι ανεπαρκής για την OWL Full, και η ανταλλαγή της επίσημης σύνταξης της OWL, RDF/XML, πρέπει να χρησιμοποιηθεί.

#### **2.3.7.5 Σημασιολογία για την OWL Full**

Στην OWL Full έχει δοθεί μια πρότυπη θεωρητική σημασιολογία που είναι μια επέκταση του λεξιλογίου της πρότυπης θεωρίας της RDF. Μια αλληλογραφία μεταξύ αυτής της σημασιολογίας και της σημασιολογίας της OWL DL έχει καθιερωθεί επίσης: έχει αποδειχθεί ότι η πρότυπη θεωρία για την OWL DL έχει τις ίδιες συνέπειες με αυτήν την πρότυπη θεωρία ύφους της RDF για εκείνες τις οντολογίες της OWL που μπορούν να γραφτούν στην αφηρημένη σύνταξη της OWL DL.

Η αλληλογραφία σημαίνει ότι, λαμβάνοντας υπόψη δύο οντολογίες της OWL DL την  $O_1$  και την  $O_2$ , που στην αφηρημένη σύνταξη γράφεται, η  $O_1$  συνεπάγεται τη  $O_2$  σύμφωνα με την πρότυπη θεωρία της OWL DL εάν και μόνο εάν η χαρτογράφηση  $O_1$  στα triple RDF συνεπάγεται τη χαρτογράφηση του  $O_2$  στα triple της RDF σύμφωνα

με την πρότυπη θεωρία της OWL Full. Η απόδειξη αυτής της αλληλογραφίας είναι μάλλον σύνθετη, και θα μπορούσε να χωριστεί, π.χ., ως αποτέλεσμα (προφανώς) των ελάχιστων αλλαγών στην προδιαγραφή της OWL ή της RDF. Λαμβάνοντας υπόψη το σχετικό εύθραυστο αυτής της αλληλογραφίας, και προκειμένου να αποφευχθεί οποιαδήποτε πιθανή σύγχυση ως προς την έννοια της OWL DL, στην πρότυπη θεωρία της OWL Full έχει δοθεί η "μη-κανονιστική" θέση (δηλ., είναι μόνο πληροφοριακή) για τις οντολογίες της OWL που μπορούν να γραφτούν στην αφηρημένη σύνταξη. Αυτό σημαίνει ότι η πρότυπη θεωρία OWL DL θα ληφθεί οριστική εάν η αλληλογραφία χωριστεί ή αποδειχθεί ότι είναι ελλιπής.

### **2.3.8 Οι μελλοντικές επεκτάσεις**

Σαφώς η OWL δεν είναι η τελική λέξη στις γλώσσες οντολογίας για το σημασιολογικό Ιστό. Διάφορα χαρακτηριστικά γνωρίσματα προσδιορίστηκαν ήδη στο έγγραφο απαιτήσεων της OWL [Hef03], και πολλές άλλες είναι υπό συζήτηση. Σε αυτό το τμήμα, εν συντομία (και μη-εξαντλητικά) απαριθμούμε μερικές από αυτές τις πιθανές επεκτάσεις και βελτιώσεις στην OWL:

*ενότητες και εισαγωγές.* Οι οντολογίες εισαγωγής που καθορίζονται από άλλες θα είναι ο κανόνας στο σημασιολογικό Ιστό. Εντούτοις, η δυνατότητα εισαγωγής της OWL είναι πολύ τετριμμένη: επιτρέπει μόνο να εισαχθεί μια ολόκληρη οντολογία, που διευκρινίζεται από τη θέση. Ακόμα κι αν κάποιος θα ήθελε μόνο να χρησιμοποιήσει μια μικρή μερίδα μιας άλλης οντολογίας, αναγκάζεται να εισάγει εκείνη την ολόκληρη οντολογία. Οι ενότητες-κατασκευές στον προγραμματισμό των γλωσσών είναι βασισμένες σε μια έννοια των πληροφοριών που κρύβονται: η ενότητα υπόσχεται να παρέχει κάποια λειτουργία στον εξωτερικό κόσμο (η εξαγωγή-πρόταση της ενότητας), αλλά η ενότητα εισαγωγής δεν χρειάζεται να ανησυχεί για το πώς αυτή η λειτουργία επιτυγχάνεται. Είναι μια ανοικτή ερευνητική ερώτηση για το πώς η έννοια του κρυψίματος πληροφοριών για τις οντολογίες μπορεί να είναι, και πώς θα μπορούσε να χρησιμοποιηθεί καθώς η βάση παρέχει σημαντικές πληροφορίες.

*προκαθορίζει.* Πολλά πρακτικά συστήματα αντιπροσώπευσης γνώσης επιτρέπουν να αγνοηθούν οι κληρονομημένες τιμές από τις πιο συγκεκριμένες κατηγορίες στην ιεραρχία, αντιμετωπίζοντας τις κληρονομημένες τιμές ως προεπιλογή. Αν και χρησιμοποιείται ευρέως στο πρακτικό KR, καμία συναίνεση δεν έχει επιτευχθεί στη σωστή διαμόρφωση για τη μη μονοτονική συμπεριφορά των προκαθορισμένων αξιών.

*Κλειστή παγκόσμια υπόθεση.* Ο Semantic της OWL υιοθετεί αυτήν την περίοδο το τυποποιημένο λογικό πρότυπο μιας ανοικτής παγκόσμιας υπόθεσης: μια δήλωση δεν μπορεί να υποτεθεί αληθινή με το να αποδείξει μια αποτυχία. Σαφώς, στο τεράστιο και μόνο μερικώς γνωστό World Wide Web αυτό είναι η σωστή υπόθεση. Εντούτοις, η αντίθετη προσέγγιση (μια κλειστή παγκόσμια υπόθεση: μια δήλωση είναι αληθινή όταν δεν μπορεί να αποδειχθεί η άρνησή της) είναι επίσης χρήσιμη σε ορισμένες εφαρμογές. Η κλειστή παγκόσμια υπόθεση είναι πολύ δεμένη στην έννοια των προεπιλογών, και οδηγεί στην ίδια μη μονοτονική συμπεριφορά, που είναι λόγος που δεν περιλαμβάνεται στην OWL ως έχει.

*Μοναδική υπόθεση ονομάτων.* Οι χαρακτηριστικές εφαρμογές των βάσεων δεδομένων υποθέτουν ότι τα άτομα με τα διαφορετικά ονόματα είναι πράγματι διαφορετικά άτομα. Η OWL ακολουθεί το συνηθισμένο λογικό παράδειγμα που δεν είναι αυτή η περίπτωση. Εάν δύο άτομα (ή κατηγορίες, ή ιδιότητες) έχουν διαφορετικά ονόματα, μπορούμε ακόμα να συμπεράνουμε ότι πρέπει να είναι τα ίδια. Όπως με τη μη-κλειστή παγκόσμια υπόθεση, η μη-μοναδική υπόθεση ονομάτων είναι η πιο εύλογη που κάνει στο World Wide Web, αλλά όπως πριν, οι καταστάσεις υπάρχουν όπου η μοναδική υπόθεση ονομάτων είναι χρήσιμη. *Διαδικαστική σύνδεση.* Μια κοινή έννοια στην αντιπροσώπευση γνώσης είναι να καθοριστεί η έννοια ενός όρου όχι μέσω των ρητών ορισμών στη γλώσσα (όπως γίνεται στην OWL), αλλά με την ένωση ενός κομματιού του κώδικα που εκτελείται για τον υπολογισμό της έννοιας του όρου. Αν και χρησιμοποιείται ευρέως, αυτή η έννοια δεν παραχωρεί πολύ καλά την ολοκλήρωση σε ένα σύστημα με μια επίσημη σημασιολογία, και δεν έχει περιληφθεί στην OWL.

*Ιδιοκτησία που αλυσοδένει, κανόνες.* Όπως εξηγείται ανωτέρω, για λόγους αποφασιστικότητας, η OWL αυτήν την περίοδο δεν επιτρέπει τη σύνθεση των



ιδιοτήτων, αλλά φυσικά σε πολλές εφαρμογές αυτό είναι μια χρήσιμη λειτουργία. Ακόμα γενικότερα, κάποιος θα ήθελε να καθορίσει τις ιδιότητες ως γενικούς κανόνες πέρα από άλλες ιδιότητες. Τέτοια ολοκλήρωση του βασισμένου στους κανόνες KR και του ύφους-DL, το KR είναι αυτήν την περίοδο ένας ενεργός τομέας της έρευνας.

## Κεφάλαιο 3

### 3.1 Σημασιολογικές Υπηρεσίες Web

Το Web είναι μια πολύ επιτυχημένη ιστορία. Ξεκίνησε σαν μια εσωτερική λύση για την ανταλλαγή επιστημονικών πληροφοριών και έχει γίνει, σε λιγότερο από μια δεκαετία, παγκοσμίως χρησιμοποιημένο μέσο για τη διάδοση πληροφοριών και την πρόσβαση σε αυτές. Η αυξομείωση, η άνεση και η ταχύτητα στη διάδοση των πληροφοριών δεν έχει κανένα προηγούμενο. Εντούτοις, είναι απλώς ένας Ιστός(Web) για τους ανθρώπους. Οι υπολογιστές δεν μπορούν "να καταλάβουν" τις παρεχόμενες πληροφορίες και σε αντάλλαγμα δεν μπορούν να παρέχουν οποιαδήποτε υποστήριξη στην επεξεργασία αυτών των πληροφοριών. Δύο συμπληρωματικές προσπάθειες, που γίνονται, είναι να μετασχηματίσουν τον Ιστό(Web), από το να είναι για ανθρώπους ,σε έναν Ιστό(Web) που περιπλέκει τους υπολογιστές για να παρέχει υποστήριξη για τις ανθρώπινες αλληλεπιδράσεις σε πιο υψηλό επίπεδο από ότι είναι διαθέσιμος με την τρέχουσα τεχνολογία του Web.

Ø Το SemanticWeb έχει σκοπό να προσθέσει τα σημασιολογικά στοιχεία που μπορούν να επεξεργαστούν από μια μηχανή σε κανονικά στοιχεία. Έτσι υπολογιστής μπορεί "να καταλάβει" τις πληροφορίες και επομένως να τις επεξεργαστεί εξ ονόματος του ανθρώπινου χρήστη. (βλ [Fen03]).

Ø Οι υπηρεσίες του Web(Web) προσπαθούν να υιοθετήσουν τον Ιστό(Web) ως σφαιρική υποδομή για το διανεμημένο υπολογισμό, για την ενσωμάτωση των διάφορων εφαρμογών, και για την αυτοματοποίηση των διαδικασιών της επιχείρησης (βλ. [ACKM03]).Ο Web(Web) δεν θα είναι μόνο ο τόπος όπου οι ανθρώπινες αναγνώσιμες πληροφορίες θα δημοσιεύονται αλλά θα είναι ο τόπος όπου ο σφαιρικός υπολογισμός θα πραγματοποιείται.

Εντούτοις, η τρέχουσα τεχνολογία υπηρεσιών Web(Web), βασίζεται στις SOAP [W3C03], WSDL [CCMW01] και UDDI [BCE+02], εξετάζει μόνο τις συντακτικές

πτυχές μιας υπηρεσίας Web(Web) και, επομένως, εξασφαλίζει μόνο ένα σύνολο άκαμπτων υπηρεσιών που δεν μπορούν να προσαρμοστούν σε μια αλλαγή του περιβάλλοντος χωρίς την ανθρώπινη επέμβαση. Ο ανθρώπινος προγραμματιστής πρέπει να διατηρηθεί σαν συνδετικός κρίκος καθώς η οικονομία των υπηρεσιών Web(Web) είναι περιορισμένη [FB02]. Το όραμα των υπηρεσιών του Semantic Web(Semantic Web) είναι να περιγράψει τις διάφορες πτυχές μιας υπηρεσίας Web χρησιμοποιώντας σαφήνεια, σημασιολογία κατανοητή από μηχανή (machine-understandable), να επιτρέπει την αυτόματη εγκατάσταση και να γίνεται συνδυασμός και χρήση των υπηρεσιών Web(Web). Η εργασία στον τομέα του Semantic Web (Semantic Web) εφαρμόζεται στις υπηρεσίες Web(Web) προκειμένου να περιοριστεί η επέμβαση του ανθρώπινου χρήστη στο ελάχιστο. Η σημασιολογική σήμανση μπορεί να χρησιμοποιηθεί για να αυτοματοποιήσει τους στόχους των υπηρεσιών αναζήτησης, εκτελώντας τους, συνθέτοντας τους και διευκολύνοντας την λειτουργικότητα μεταξύ τους [Coa04], επιτρέποντας κατά συνέπεια ευφείς υπηρεσίες Web.

Η περιγραφή των υπηρεσιών Web με έναν τρόπο κατανοητό από την μηχανή αναμένεται να έχει μεγάλη επιρροή στους τομείς του ηλεκτρονικού εμπορίου και των ολοκληρωμένων εφαρμογών των επιχειρήσεων, αφού μπορεί να επιτρέψει τη δυναμική και εξελικτική συνεργασία μεταξύ των διαφορετικών συστημάτων και των οργανώσεων.

Ένα σημαντικό βήμα προς τη δυναμική και εξελικτική ολοκλήρωση, μέσα και πέρα από τα επιχειρηματικά όρια, είναι η μηχανοποίηση της υπηρεσίας ανακάλυψης. Οι διαθέσιμες υπηρεσίες του αυτόματου εντοπισμού και του περιορισμού για να εκτελέσουν μια δεδομένη επιχειρησιακή δραστηριότητα μπορούν να μειώσουν το κόστος ολοκλήρωσης αρκετά και μπορούν να επιτρέψουν μια πιο εύκαμπτη ολοκλήρωση, όπου οι προμηθευτές επιλέγονται δυναμικά, βασισμένοι σε αυτό που παρέχουν και ενδεχομένως σε άλλες μη λειτουργικές ιδιότητες όπως η εμπιστοσύνη, η ασφάλεια, κ.λ.π.

Εάν η υπηρεσία ανακάλυψης δεν είναι ικανή να βρει μια υπηρεσία που να ταιριάζει με τις απαιτήσεις των χρηστών, μπορεί να είναι ακόμα δυνατό να συντεθούν (να ενσωματωθούν) διάφορες υπηρεσίες για να παράσχει την απαραίτητη λειτουργία. Αυτή η διαδικασία καλείται σύνθεση υπηρεσιών. Διακρίνουμε την αυτοματοποιημένη και την χειροκίνητη σύνθεση υπηρεσιών.

Στην περίπτωση της χειροκίνητης σύνθεσης υπηρεσιών, ο χρήστης διευκρινίζει ρητά πώς ορισμένες βασικές υπηρεσίες πρέπει να αλληλεπιδράσουν προκειμένου να παρασχεθεί η επιθυμητή λειτουργία. Οι προδιαγραφές από την υπηρεσία σύνθεσης μπορούν να απεικονιστούν και σαν τα αποτελέσματα της δουλειάς. Στην περίπτωση της αυτοματοποιημένης υπηρεσίας σύνθεσης, μια μηχανή σύνθεσης υπηρεσιών παράγει αυτόματα μια απεικόνιση της υπηρεσίας σύνθεσης που βασίζεται στις απαιτήσεις των χρηστών. Υπάρχουν επίσης υβριδικές προσεγγίσεις που αναμειγνύουν την χειρωνακτική και την αυτοματοποιημένη σύνθεση υπηρεσιών. Σε μια τέτοια ρύθμιση, η μηχανή σύνθεσης χρησιμεύει σαν ένα προγραμματιστικό εργαλείο αλληλεπίδρασης που μπορεί να βοηθήσει και να καθοδηγήσει τον προγραμματιστή για να σχεδιάσει σύνθετες υπηρεσίες. Ο προγραμματιστής μπορεί να διευκρινίσει μερικούς περιορισμούς και η μηχανή σύνθεσης να συμπληρώσει τις υπηρεσίες προκειμένου να ολοκληρωθούν τα αποτελέσματα της δουλειάς με έναν συνεπή τρόπο.

Σε ένα ανοικτό περιβάλλον, η υπηρεσία σύνθεσης περιλαμβάνει πάντα την ανακάλυψη υπηρεσιών. Ο προγραμματιστής μιας υπηρεσίας σύνθεσης πρέπει να ξέρει ποιες βασικές υπηρεσίες είναι διαθέσιμες, και λόγω του ελεύθερου περιβάλλοντος, το σύνολο των διαθέσιμων υπηρεσιών αλλάζει συνεχώς. Για την αυτοματοποιημένη υπηρεσία σύνθεσης, η μηχανή σύνθεσης υπηρεσιών πρέπει να έχει μια ενημερωμένη άποψη των διαθέσιμων υπηρεσιών. Ενώ ο αριθμός των δημοσιευμένων υπηρεσιών μπορεί να είναι εξαιρετικά μεγάλος (υποθέτοντας ότι είναι διαδεδομένες η αποδοχή και η υιοθέτηση των υπηρεσιών του Semantic Web(Semantic Web)), η μηχανή σύνθεσης υπηρεσιών μπορεί να μην είναι σε θέση να διατηρήσει ένα αντίγραφο όλων των δημοσιευμένων υπηρεσιών που περιγράφονται. Ως εκ τούτου, η μηχανή υπηρεσιών σύνθεσης πρέπει δυναμικά να αλληλεπιδράσει με τους καταλόγους υπηρεσιών για να ανακαλύψει τις σχετικές υπηρεσίες όταν αυτό απαιτείται.

Η αυτοματοποιημένη υπηρεσία σύνθεσης μπορεί να ληφθεί υπ'όψιν και σε διαφορετικά επίπεδα αφαίρεσης. Από μια υψηλού επιπέδου άποψη, μπορούμε να εστιάσουμε στην εισαγωγή των απαιτούμενων πληροφοριών και να δώσουμε τις εξαγόμενες πληροφορίες των υπηρεσιών και των υπηρεσιών σύνθεσης προκειμένου να παραχθούν τα αποτελέσματα που απαιτούνται από το χρήστη.

Καλούμε αυτό το είδος υψηλού επιπέδου σύνθεσης *λειτουργικές υπηρεσίες σύνθεσης*(*Functional Service Composition*), οι οποίες εξετάζονται στο κεφάλαιο 5. Εκεί δείχνουμε πώς οι τυποποιημένες τεχνικές προγραμματισμού μπορούν να προσαρμοστούν για τις λειτουργικές υπηρεσίες σύνθεσης(*Functional Service Composition*). Επιπλέον, συζητάμε την ολοκλήρωση λειτουργικών υπηρεσιών σύνθεσης(*Functional Service Composition*) με την ανακάλυψη δυναμικών υπηρεσιών.

Μόλις υπολογιστεί μια λειτουργική υπηρεσία σύνθεσης(*Functional Service Composition*), ενδιαφερόμαστε για συγκεκριμένες αλληλεπιδράσεις μεταξύ των διαφορετικών υπηρεσιών. Χαρακτηριστικά, μια υπηρεσία αποτελείται όχι μόνο από μια ενιαία λειτουργία (μέθοδος), αλλά σύμφωνα με το μερικό πρωτόκολλο υπηρεσιών, πρέπει να χρησιμοποιηθεί και μια δεδομένη υπηρεσία API π.χ. αυτό μπορεί να είναι απαραίτητο για να επικαλεσθούν διάφορες λειτουργίες υπηρεσιών (μέθοδοι) σε μια συγκεκριμένη εντολή. Ως εκ τούτου, μια άριστης ποιότητας ειδικού επιπέδου επεξεργασίας, σύνθεση απαιτείται με σκοπό να παράγει μια εκτελέσιμη υπηρεσία σύνθεσης. Στο κεφάλαιο 5 παρουσιάζουμε μια προσέγγιση για ειδικού επιπέδου επεξεργασίας σύνθεση βασισμένη στο πρότυπο που ελέγχει τις τεχνικές.

Τα πιθανά οφέλη των υπηρεσιών του *Semantic Web*(*Semantic Web*) έχουν οδηγήσει στην καθιέρωση σημαντικού ερευνητικού τομέα, στον ακαδημαϊκό κόσμο και στη βιομηχανία. Διάφορες πρωτοβουλίες έχουν εμφανιστεί για να σχολιάσουν τις υπηρεσίες του *Semantic Web*(*Semantic Web*), που παρέχουν τις διαφορετικές περιγραφές των υπηρεσιών *Web*(*Web*) και τις σχετικές πτυχές τους που, στη συνέχεια, φέρνουν ένα διαφορετικό είδος υποστήριξης για την ανακάλυψη και την σύνθεση. Στα επόμενα κεφάλαια, παρουσιάζουμε τις σημαντικότερες πρωτοβουλίες σε αυτόν τον τομέα.

### **3.2 OWL-S**

Η OWL-S [Coa04] είναι μια συνεργάσιμη προσπάθεια από τις BBN τεχνολογίες, τα Carnegie Mellon University, Stanford University, SRI international και Yale University για να καθορίσουν μία οντολογία για τη σήμανση των υπηρεσιών του *Semantic Web*(*Semantic Web*). Η OWL-S, αυτήν την περίοδο, προορίζεται να επιτρέψει την αυτοματοποίηση των υπηρεσιών *Web*(*Web*) στους τομείς της ανακάλυψης, της επίκλησης, της σύνθεσης, της λειτουργικότητας και του ελέγχου εκτέλεσης με την

παροχή των κατάλληλων περιγραφών των σημασιολογικών υπηρεσιών. Ο σκοπός της OWL-S είναι να καθορίσει ένα σύνολο βασικών κατηγοριών και ιδιοτήτων για να δηλώνονται και να περιγράφονται οι υπηρεσίες, δηλ. μια οντολογία στην OWL-S για την περιγραφή των υπηρεσιών Web(Web) που επιτρέπει στους χρήστες και στους διαμεσολαβητές(agents) λογισμικού να ανακαλύψουν αυτόματα, να επικαλεσθούν, να συνθέσουν και να ελέγξουν τους πόρους του Web(Web) που προσφέρουν τις υπηρεσίες, με κάποιους περιορισμούς.

### 3.2.1 Υποστήριξη για την ανακάλυψη

Η ανώτερη οντολογία OWL-S αποτελείται από τρία βασικά στοιχεία: το *προφίλ υπηρεσιών(service profile)*, το *μοντέλο υπηρεσίας(service model)*, και την *εκπαίδευση θεμελιώδους υπηρεσίας(service grounding)*. Από αυτούς, το *προφίλ υπηρεσιών(service profile)* παρέχει τις πληροφορίες που χρειάζεται ένας διαμεσολαβητή(agent) για να ανακαλύψει μια υπηρεσία [Coa04].

Το προφίλ της OWL-S ενοποιεί το όραμα και του αιτούντος και του προμηθευτή. Μπορεί να χρησιμοποιείται για να περιγράψει αυτό που ο αιτών αναμένει από την εκτέλεση υπηρεσιών αλλά και την πραγματική λειτουργικότητα των υπηρεσιών από την πλευρά που το βλέπει ο προμηθευτής. Επομένως, ο αιτών αναμένεται να καθορίσει ένα προφίλ υπηρεσιών που περιγράφει την υπηρεσία που ψάχνει, ενώ ο προμηθευτής θα περιγράψει μια υπηρεσία Web(Web) με την παροχή ενός ή περισσότερων προφίλ που δίνουν τις πληροφορίες για το τι κάνουν πραγματικά οι υπηρεσίες.

Το προφίλ προορίζεται να εντοπίζει τρεις διαφορετικούς τύπους πληροφοριών: ποια οργάνωση παρέχει την υπηρεσία, ποια λειτουργία υπολογίζει η υπηρεσία, και ένα πλήθος χαρακτηριστικών γνωρισμάτων που διευκρινίζουν τις προδιαγραφές της υπηρεσίας [Coa04]. Παρατηρήστε ότι και το πρώτο και το τρίτο είδος των πληροφοριών μπορούν να θεωρηθούν ως μη λειτουργικές ιδιότητες της υπηρεσίας Web(Web), ενώ το δεύτερο είδος πληροφοριών περιγράφει τη λειτουργία που παρέχεται από την υπηρεσία. Θα ακολουθήσουμε αυτή την κατηγοριοποίηση για να δώσουμε τις περαιτέρω λεπτομέρειες του προφίλ των υπηρεσιών. Όπως εξηγήσαμε πριν, το προφίλ χρησιμοποιείται επίσης για να περιγράψει ένα αίτημα των χρηστών, που περιέχει τις πληροφορίες για το τι είναι η οργάνωση που πρέπει να παρέχει την υπηρεσία που αναζητείται, για το ποια λειτουργία

πρέπει εκπληρώσει η υπηρεσία, και ποια μη λειτουργικά κριτήρια πρόκειται να ικανοποιηθούν.

**Μη λειτουργικές ιδιότητες.** Οι μη λειτουργικές ιδιότητες των υπηρεσιών του Web(Web) που περιέχονται στο προφίλ των υπηρεσιών είναι : το όνομα της υπηρεσίας, μια περιγραφή της υπηρεσίας, στοιχεία επικοινωνίας με τους υπεύθυνους της υπηρεσίας, μια εξωτερική κατηγοριοποίηση της υπηρεσίας και ένας εκτεταμένος κατάλογος μη-προκαθορισμένων ιδιοτήτων.

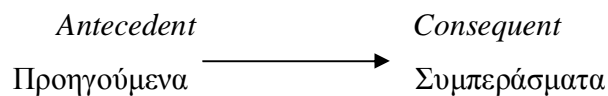
**Περιγραφή λειτουργίας.** Οι λειτουργικοί χαρακτηρισμοί των υπηρεσιών του Web(Web) εκφράζονται σε σχέση με τον μετασχηματισμό των πληροφοριών και της δήλωσης αλλαγής που γίνεται από την εκτέλεση της υπηρεσίας. Η δήλωση αλλαγής, που διαμορφώνεται από τις προϋποθέσεις και τα αποτελέσματα, αναφέρεται στην αλλαγή επάνω στη κατάσταση του κόσμου σαν συνεπεία της εκτέλεσης της υπηρεσίας, και ο μετασχηματισμός των πληροφοριών, διαμορφώνεται από τις εισαγόμενες πληροφορίες και τα αποτελέσματα, και αναφέρονται στις πληροφορίες που απαιτούνται και στις πληροφορίες που παράγονται (γενικά ανάλογα με τις πληροφορίες που παρέχονται ως εισαγωγή) από την υπηρεσία. Οι εισαγόμενες πληροφορίες, τα αποτελέσματα, οι προϋποθέσεις, και τα υπάρχοντα αποτελέσματα αναφέρονται κανονικά ως IOPEs [Coa04].

Το σχήμα που περιγράφει τις περιπτώσεις IOPEs καθορίζεται στο μοντέλο υπηρεσιών, όχι στο προφίλ της υπηρεσίας. Επομένως, αυτές οι περιπτώσεις περιγράφονται στο μοντέλο υπηρεσιών και δηλώνονται από το προφίλ υπηρεσιών. Η OWL-S προβλέπει ότι τα IOPEs του προφίλ είναι ένα υποσύνολο εκείνων που δημοσιεύονται από το μοντέλο [Coa04].

Η OWL-S είναι η πρώτη έκδοση των OWL-S που χρησιμοποιεί μια συγκεκριμένη γλώσσα κανόνα για να περιγράψει τα IOPEs, που αντιλαμβάνεται τη σχέση μεταξύ των εισαγωγών, των αποτελεσμάτων, των προϋποθέσεων και των υπάρχοντων αποτελεσμάτων. Η γλώσσα που χρησιμοποιείται για αυτόν το λόγο είναι η γλώσσα κανόνα Semantic Web [Semantic Web Rule Language, (SWRL)], [HPSB+04]. Παρακάτω παρέχουμε μια συνοπτική επισκόπηση της γλώσσας .

### 3.2.2 SWRL

Η SWRL είναι μια πρόταση για τον συνδυασμό της OWL DL [DS04] και της OWL Lite [DS04] με τις μοναδιαίες /δυναδικές υπογλώσσες Datalog RuleML5, που επιτρέπουν το συνδυασμό των Horn-Like κανόνων με την βάση γνώσης OWL. Οι κανόνες SWRL έχουν τη μορφή:



όπου και τα *antecedent* δηλ. το σώμα κανόνα και τα *consequent* δηλ. το κεφάλι κανόνα είναι μια συνύπαρξη ατόμων. Τα άτομα μπορούν να είναι της μορφής  $C(x)$ ,  $P(x, y)$ ,  $Q(x, z)$  ίδια με τα  $(x, y)$  ή διαφορετικά με τα  $(x, y)$ , όντας η  $C$  μια έννοια OWL DL (που δεν ονομάζεται απαραίτητως),  $P$  η OWL DL ιδιότητα αντικειμένου,  $Q$  μια OWL DL ιδιότητα στοιχείων, οι  $x$  και  $y$  είναι είτε μεταβλητές είτε περιπτώσεις OWL και το  $z$  είναι είτε μια μεταβλητή είτε μια OWL αξία στοιχείων.

Οι μεταβλητές αντιμετωπίζονται όπως παγκοσμίως ποσοτικοποιούνται μέσα στο πλαίσιο των κανόνων που εμφανίζονται, και μόνο ασφαλείς κανόνες επιτρέπονται. Η ύπαρξη της ποσοτικοποίησης μπορεί να γίνει αντιληπτή με τη χρησιμοποίηση του περιορισμού της OWL *someValuesFrom*.

*Εισαγόμενες εξαγόμενες πληροφορίες.* Οι OWL εισαγόμενες και οι εξαγόμενες πληροφορίες διαμορφώνονται ως υποκατηγορίες της *παραμέτρου*, οι όποιες είναι στη συνέχεια μια υποκατηγορία της μεταβλητής SWRL με μια ιδιότητα που δείχνει την κατηγορία ή προσδιορίζει τον τύπο της αξίας της παραμέτρου που ανήκουν [Coa04]. Οι τοπικές μεταβλητές μπορούν να είναι επίσης χρησιμοποιημένες, και διαμορφώνονται στην οντολογία ως υποκατηγορίες της παραμέτρου. Εισαγόμενες πληροφορίες, εξαγόμενες πληροφορίες, και τοπικές μεταβλητές έχουν ως πεδίο δράσης τη διαδικασία όπου εμφανίζονται.

Οι εισαγόμενες και εξαγόμενες πληροφορίες που καθορίζονται στο μοντέλο υπηρεσιών είναι παραπεμφθείς από το προφίλ μέσω των ιδιοτήτων *hasInput* και *hasOutput*,

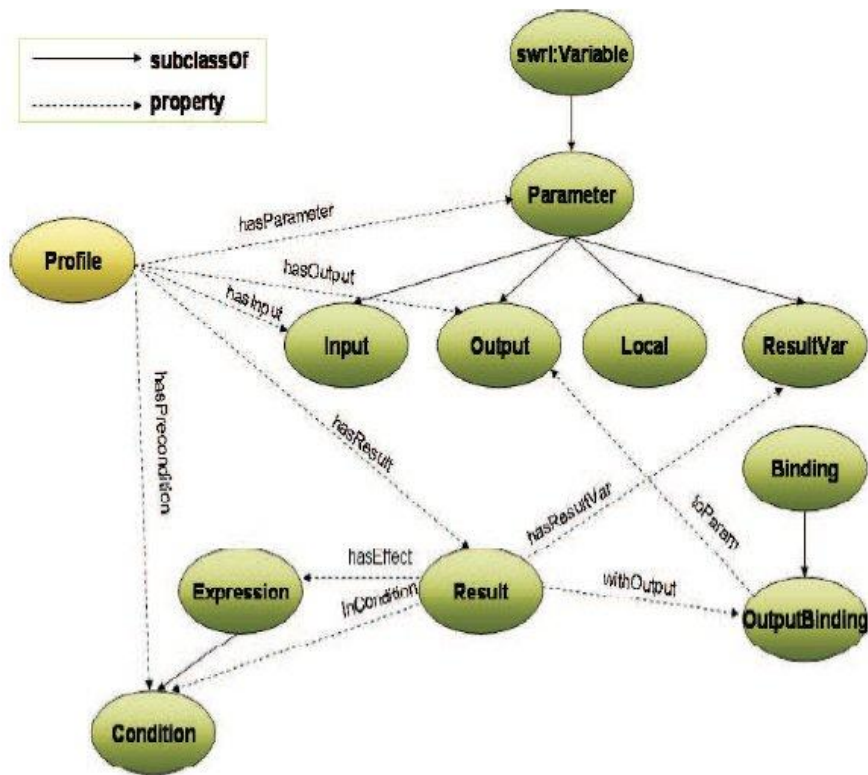


αντίστοιχα. Οι τοπικές μεταβλητές μπορούν να παραπεμφθούν μέσω της ιδιότητας *hasParameter*.

*Προϋποθέσεις και αποτελέσματα.* Οι προϋποθέσεις είναι όροι στην κατάσταση του κόσμου που πρέπει να ισχύουν για να εκτελεστεί επιτυχώς μια υπηρεσία. Διαμορφώνονται ως όροι, και είναι μια υποκατηγορία της έκφρασης. Οι εκφράσεις στην OWL-S διευκρινίζουν τη γλώσσα στην οποία η έκφραση περιγράφεται και η ίδια η έκφραση κωδικοποιείται σαν (σειρά ή XML) κυριολεκτική. Τα αποτελέσματα περιγράφουν τους όρους στην κατάσταση του κόσμου που είναι πραγματικά μετά από την εκτέλεση υπηρεσιών. Αυτά διαμορφώνονται σαν *αποτελέσματα*. Ένα αποτέλεσμα έχει ένα *inCondition*, ένα *ResultVar*, ένα *OutputBinding*, και ένα *effect* (*αποτελέσματα*). Το *inCondition* διευκρινίζει τον όρο που απαιτείται για την παράδοση του αποτελέσματος. Το *OutputBinding* δεσμεύει τη δηλωμένη εξαγόμενη πληροφορία στον κατάλληλο τύπο ή την αξία ανάλογα με *inCondition*. Τα *αποτελέσματα* περιγράφουν την κατάσταση του κόσμου ως αποτέλεσμα της εκτέλεσης της υπηρεσίας. Τα *ResultVars* διαδραματίζουν το ρόλο των τοπικών μεταβλητών για την περιγραφή των αποτελεσμάτων.

Οι όροι δηλ. οι προϋποθέσεις που καθορίζονται στο πρότυπο υπηρεσιών είναι παραπεμφθείσες από το προφίλ μέσω της ιδιότητας *hasPrecondition*, και τα αποτελέσματα μέσω της ιδιότητας *hasResult*.

Το σχήμα 3.1 επεξηγεί την περιγραφή των λειτουργικών πτυχών του προφίλ. Για λόγους απλότητας, ο πλήρης καθορισμός κάποιας *σύνδεσης* κατηγοριών π.χ. δέσιμο παραλείπεται.



Σχήμα 3.1 : Καθορισμός του προφίλ της OWL-S

**Υποστήριξη για τη σύνθεση :** Η αυτόματη σύνθεση και η λειτουργικότητα υπηρεσιών Web(Web) περιλαμβάνουν την αυτόματη επιλογή, σύνθεση, και λειτουργικότητα των υπηρεσιών Web(Web) για να εκτελέσουν κάποιο σύνθετο στόχο, λαμβάνοντας υπόψη την υψηλού επιπέδου περιγραφή ενός στόχου. Παραδείγματος χάριν, ο χρήστης μπορεί να θελήσει να κάνει όλες τις ρυθμίσεις ταξιδιού για ένα ταξίδι σε μια διάσκεψη. Τότε, ο χρήστης πρέπει να επιλέξει τις υπηρεσίες Web(Web), να καθορίσει τη σύνθεση με το χέρι, και να σιγουρέψει ότι οποιοδήποτε λογισμικό που απαιτείται για τη λειτουργικότητα των υπηρεσιών που πρέπει να μοιραστεί τις πληροφορίες είναι φτιαγμένο κατά παραγγελία. Με τη σήμανση OWL-S των υπηρεσιών Web(Web), οι πληροφορίες απαραίτητα επιλέγουν και συνθέτουν υπηρεσίες που κωδικοποιούνται στις ιστοσελίδες των υπηρεσιών. Το λογισμικό μπορεί να δημιουργηθεί για να χειριστεί αυτές τις αντιπροσωπεύσεις, μαζί με μια προδιαγραφή των αντικειμενικών στόχων του σκοπού, για να επιτύχει το στόχο αυτόματα. Για να το υποστηρίξει αυτό, η OWL-S παρέχει τις

επεξηγηματικές προδιαγραφές των προϋποθέσεων και των συνεπειών των εφαρμογών των ανεξάρτητων και των μεμονωμένων υπηρεσιών (σχετικών για το λειτουργικό επίπεδο σύνθεσης), και μια γλώσσα για την περιγραφή των συνθέσεων υπηρεσιών και των αλληλεπιδράσεων της ροής των στοιχείων (σχετική για process-level composition).

### 3.3 WSMO

Το WSMO είναι μια προδιαγραφή από την ομάδα εργασίας WSMO του SDK. Το WSMO στηρίζεται σε τέσσερα σημαντικά συστατικά που εμπνέονται από την εννοιολογική εργασία που γίνεται στον καθορισμό του WSMF (σχήμα 3.2), δηλαδή:

**Οντολογίες:** Παρέχουν την ορολογία και την επίσημη σημασιολογία για την περιγραφή άλλων στοιχείων σε WSMO.

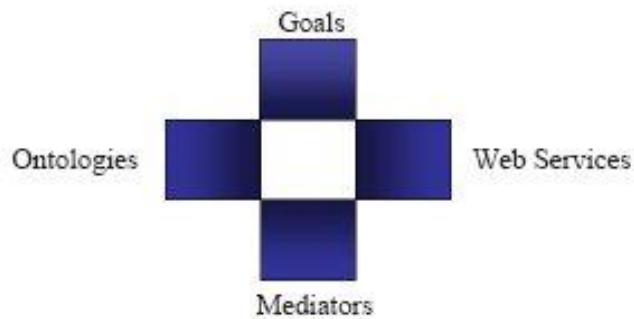
**Στόχοι:** Αυτά τα στοιχεία παρέχουν τα μέσα για να διευκρινιστούν οι πλευρές των αιτούντων όταν συμβουλευούνται μια υπηρεσία Web(Web), που περιγράφει σε ένα υψηλό επίπεδο έναν συγκεκριμένο στόχο που επιτυγχάνεται.

**Υπηρεσίες Web:** Παρέχουν την περιγραφή των υπηρεσιών του Semantic Web(Semantic Web), συμπεριλαμβανομένων των λειτουργικών και μη λειτουργικών ιδιοτήτων, καθώς επίσης και άλλων πτυχών σχετικών με την επικοινωνία τους.

**Μεσολαβητές:** Αυτά τα στοιχεία διαμόρφωσης είναι συνδετήρες που επιλύουν τα προβλήματα ετερογένειας προκειμένου να επιτραπεί η λειτουργικότητα μεταξύ των ετερογενών συμβαλλόμενων μερών.

#### Υποστήριξη για την ανακάλυψη

Από τα στοιχεία που παρουσιάζονται παραπάνω, τα πιο σχετικά με τη διαδικασία ανακαλύψεων είναι:



Σχήμα 3.2 : Πυρήνας στοιχείων WSMO

**Στόχοι:** Οι στόχοι στο WSMO καθορίζονται ως στόχοι που μπορεί να έχει ένας πελάτης κατά διαβούλευση σε μια υπηρεσία Web(Web). Αποτελούνται από τις μη λειτουργικές ιδιότητες (από ένα σύνολο προκαθορισμένων ουσιωδών ιδιοτήτων), τις εισαγόμενες οντολογίες, χρησιμοποιημένους μεσολαβητές, τις μετέπειτα καταστάσεις και τα αποτελέσματα.

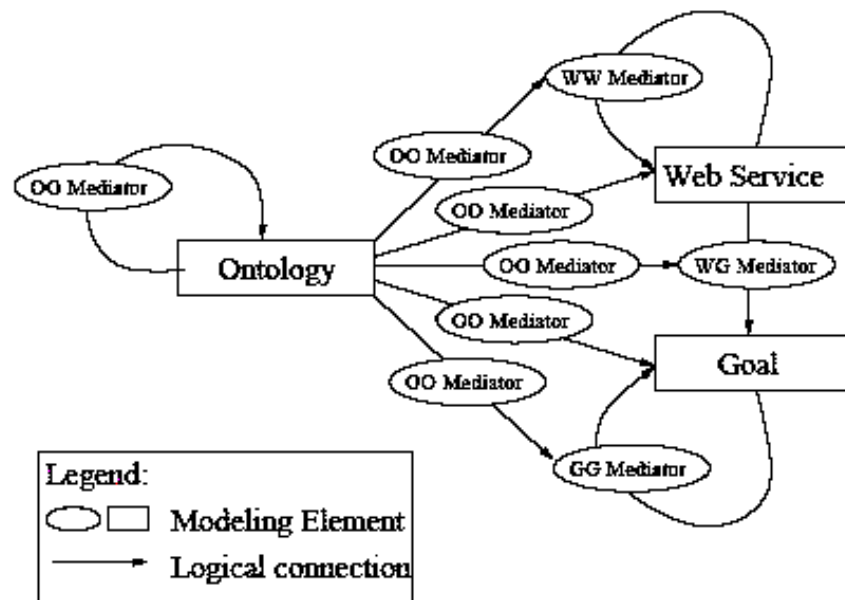
Η μετέπειτα κατάσταση και τα αποτελέσματα περιγράφουν την κατάσταση του διαστήματος πληροφοριών και του κόσμου που είναι επιθυμητά από τον αιτούντα, αντίστοιχα. Οι οντολογίες μπορούν να εισαχθούν άμεσα ως ορολογία για να καθορίσουν το στόχο όταν πρέπει να επιλυθεί χωρίς καμία σύγκρουση. Εντούτοις, εάν οποιαδήποτε ευθυγράμμιση, συγχωνευθεί, ή απαιτηθεί σύγκρουση διαχωρισμού, τότε είναι απαραίτητα απ' την αρχή τα *ooMediators*.

**Υπηρεσίες Web:** Διάφορες πτυχές των υπηρεσιών Web(Web) περιγράφονται σε WSMO. Η απαιτούμενη ορολογία για να τις περιγράψει, όπως και για τους στόχους, μπορεί να εισαχθεί άμεσα ή μέσω των *ooMediators* όταν οι συγκρούσεις πρέπει να επιλυθούν. Επιπλέον, η ικανότητα και οι διασυνδέσεις των υπηρεσιών περιγράφονται, από αυτό του οποίου η ικανότητα είναι το πιο σχετικό στοιχείο για την ανακάλυψη.

Οι ικανότητες WSMO καθορίζουν τις λειτουργικές πτυχές της προσφερθείσας υπηρεσίας, διαμορφωμένες μέσα σε όρους των προϋποθέσεων, των υποθέσεων, των μετέπειτα καταστάσεων και των αποτελεσμάτων. Καθορίζεται χωριστά από τους στόχους του αιτούντος, παρέχει τη διάκριση μεταξύ των απόψεων του αιτούντος και του προμηθευτή.

Οι *προϋποθέσεις* της ικανότητας περιγράφουν τις έγκυρες καταστάσεις του διαστήματος πληροφοριών πριν από την εκτέλεση υπηρεσιών. Οι *μετέπειτα καταστάσεις* περιγράφουν την κατάσταση του διαστήματος πληροφοριών που εγγυάται πως θα πάρει έκταση μετά από την εκτέλεση υπηρεσιών. Οι *υποθέσεις* είναι παρόμοιες με προϋποθέσεις, αλλά καθορίζουν τις έγκυρες καταστάσεις του κόσμου για μια σωστή εκτέλεση υπηρεσιών. Τα *αποτελέσματα* περιγράφουν την κατάσταση του κόσμου που εγγυάται πως θα πάρει έκταση μετά από την εκτέλεση της υπηρεσίας.

**Μεσολαβητές:** Σαν έναν από τους στυλοβάτες του, το WSMO εισάγει την έννοια των μεσολαβητών με σκοπό να επιλύσει τα ετερογενή προβλήματα. Οι μεσολαβητές σε WSMO είναι ειδικά στοιχεία που χρησιμοποιούνται σε ετερογενή τμήματα συνδέσεων που περιλαμβάνονται στη διαμόρφωση μιας υπηρεσίας Web(Web). Καθορίζουν την απαραίτητη σχεδίαση, τους μετασχηματισμούς ή την απλοποίηση μεταξύ των συνδεδεμένων στοιχείων. Όπως απεικονίζεται στο σχήμα 3.3, τέσσερις διαφορετικοί τύποι μεσολαβητών καθορίζονται. Από εκείνους, η καταλληλότητα για την ανακάλυψη είναι:



Σχήμα 3.3: Μεσολαβητές WSMO

Ø *ggMediators*: Συνδέουν δύο στόχους, εκφράζοντας την απλοποίηση ενός στόχου πηγής στο στόχο σκοπού. Μπορούν να χρησιμοποιήσουν *ooMediators* για να παρακάμψουν τις διαφορές στην ορολογία με σκοπό να καθορίσουν αυτούς τους στόχους. Επιπλέον, το WSMO επιτρέπει να συνδέονται όχι μόνο στόχοι, αλλά και στόχοι σε *ggMediators*, επιτρέποντας κατά συνέπεια την επαναχρησιμοποίηση των πολλαπλάσιων στόχων για να καθορίσει έναν νέο.

Ø *OoMediators*: Εισάγουν τις οντολογίες και επιλύουν τους πιθανούς κακούς συνδυασμούς αντιπροσώπευσης μεταξύ τους, όπως οι διαφορές στις γλώσσες αντιπροσώπευσης ή στο πώς αντιλαμβάνονται την ίδια περιοχή.

Ø *WgMediators*: Συνδέουν μια υπηρεσία Web(Web) με έναν στόχο. Αυτή η σύνδεση αντιπροσωπεύει (συνολικά ή μερικά) την εκπλήρωση του στόχου από την υπηρεσία Web(Web). Τα *wgMediators* μπορεί να χρησιμοποιήσουν *ooMediators* για να επιλύσουν τα προβλήματα ετερογένειας μεταξύ της υπηρεσίας Web και του στόχου.

### **Υποστήριξη για τη σύνθεση**

Η λειτουργικού επιπέδου σύνθεση μπορεί να εκμεταλλευτεί την επίσημη περιγραφή των ικανοτήτων και των στόχων που παρέχονται από το WSMO. Θεωρώντας την ειδικού επιπέδου επεξεργασία σύνθεσης, το WSMO παρέχει την περιγραφή διασύνδεσης υπηρεσιών που παρέχουν τις πληροφορίες για το πώς η υπηρεσία αλληλεπιδρά, και με ποιες άλλες υπηρεσίες συνεργάζεται για να επιτύχει τη λειτουργία του. Αυτές οι περιγραφές καλούνται χορογραφία και ενορχήστρωση (orchestration), αντίστοιχα. Η ειδικού επιπέδου επεξεργασία σύνθεσης μπορεί να εκμεταλλευτεί επίσημες περιγραφές χορογραφίας που παρέχονται από το WSMO για να ξέρει την εξωτερική επεξεργασία μιας δεδομένης υπηρεσίας, και της κατασκευής ενορχήστρωσης (orchestrations) για τη σύνθετη υπηρεσία που παράγεται.

**WSML:** Η WSML ( γλώσσα διαμόρφωσης υπηρεσιών Web(Web)) ομάδα εργασίας καθορίζει μια οικογένεια γλωσσών με τις βάσεις τους στις περιγραφές λογικής (description logics), την first-Order logic (λογική της πρώτης εντολής) και τον προγραμματισμό λογικής (logic programming) [ de04 ].

Η διαφορετική εκδοχή του πυρήνα της WSML(WSML Core) αντιστοιχεί σημασιολογικά με τη διασταύρωση της Λογικής περιγραφής (Description logic) και της λογικής Horn, που επεκτείνονται με την εκτενή υποστήριξη των τύπων δεδομένων (datatype) προκειμένου να είναι χρήσιμος στις πρακτικές εφαρμογές. Ο πυρήνας της WSML(WSML Core) συμμορφώνεται πλήρως με ένα υποσύνολο της OWL αν και η υποστήριξη των τύπων δεδομένων στον πυρήνα της WSML είναι ήδη πέρα από την OWL επειδή η υποστήριξη των τύπων δεδομένων στην OWL είναι πολύ περιορισμένη. Ο πυρήνας της WSML (WSML Core) επεκτείνεται, και στην κατεύθυνση της Λογικής περιγραφής (description logics) και στην κατεύθυνση του προγραμματισμού λογικής(logic programming).

Η WSML-DL επεκτείνει την WSML-Core σε μια εκφραστική λογική περιγραφής, που ονομάζεται *SHOIN*.

Η WSML-FLIGHT επεκτείνει την WSML Core στην κατεύθυνση του προγραμματισμού λογικής με περισσότερους διαισθητικούς περιορισμούς αξίας και περιορισμούς στον αριθμό των στοιχείων του συνόλου. Η WSML-FLIGHT είναι η προτιμημένη οντολογία γλώσσας διαμόρφωσης για WSMO, λόγω του πλούσιου συνόλου των διαμορφωμένων αρχικών στοιχείων για τη διαμόρφωση των διαφορετικών πτυχών των ιδιοτήτων, όπως οι περιορισμοί αξίας και οι περιορισμοί ακεραιότητας, και η πλούσια λογική γλώσσα του που επιτρέπει να γράφει κάτω τους αυθαίρετους κανόνες. Επιπλέον, η WSML-FLIGHT ενσωματώνει μια πλήρως αναπτυγμένη γλώσσα κανόνα, επιτρέποντας ακόμα έναν πιο αποδοτικό συλλογισμό.

Η WSML-RULE επεκτείνει την WSML-FLIGHT σε μια πλήρως αναπτυγμένη γλώσσα προγραμματισμού λογικής, συμπεριλαμβανομένων των συμβόλων λειτουργίας και των χαρακτηριστικών γνωρισμάτων υψηλού βαθμού (first-Order ) της HiLog [CKW93] και ενδεχομένως της λογικής συναλλαγής [BK98].

Η WSML-FULL ενοποιεί όλες τις παραλλαγές WSML κάτω από μια κοινή ομπρέλα first-Order με μη μονοτονικές επεκτάσεις. Όλες οι παραλλαγές WSML περιγράφονται από ένα κατανοητό και κανονικό κείμενο. Εκτός από το κατανοητό κείμενο, παρέχεται ένα XML και ένα RDF κείμενο για την ανταλλαγή μεταξύ μηχανών. Επιπλέον, δίνεται μια σχεδίαση από και προς την OWL για τη βασική λειτουργία μεταξύ των οντολογιών OWL μέσω ενός κοινού Semantic υποσυνόλου OWL και WSML.

### **3.4 METEOR-S**

Ο METEOR-S, αρχίζει το 2002 στο εργαστήριο LSDIS στο πανεπιστήμιο της Γεωργίας, με σκοπό να ενοποιήσει τα πρότυπα των υπηρεσιών Web(Web) όπως οι γλώσσες εκτέλεσης επιχειρηματικών διαδικασιών για τις υπηρεσίες Web [Business Process Execution Language (BPEL4WS)] [acd+ 03], γλώσσα περιγραφής υπηρεσιών Web (WSDL) [CCMW01] και καθολική περιγραφή, ανακάλυψη και ολοκλήρωση (UDDI) [BCE+ 02] με τεχνολογίες του Σημασιολογικές Web(Semantic Web).

Οι METEOR-S έχουν σκοπό να αυτοματοποιήσουν τους στόχους τους στην δημοσίευση, ανακάλυψη, περιγραφή, και στον έλεγχο ροής των υπηρεσιών Web(Web). Παρακάτω, θα εστιάσουμε στον τρόπο με τον οποίο οι METEOR-S, υποστηρίζουν την ανακάλυψη. Αν και οι METEOR-S παρέχουν επίσης μια υποδομή ανακαλύψεων για τις υπηρεσίες Web(Web) (MWSDI) [VSSP04], το ενδιαφέρον μας περιορίζεται στο πώς οι υπηρεσίες των METEOR-S περιγράφονται για να υποστηρίξουν τέτοια ανακάλυψη, όχι στην υποδομή που χρησιμοποιούν οι περιγραφές των METEOR-S.

#### **Υποστήριξη για την ανακάλυψη**

Οι METEOR-S προσθέτουν τη σημασιολογία (χρησιμοποιώντας οντολογίες) σε δύο επίπεδα [VSSP04]: στο επίπεδο των μεμονωμένων υπηρεσιών Web(Web) και στο επίπεδο των εγγεγραμμένων που αποθηκεύουν τις υπηρεσίες. Για τον σχολιασμό των μεμονωμένων υπηρεσιών Web(Web), δίνεται το παρακάτω παράδειγμα δηλ. το WSDL δηλώνει πως οι εισαγόμενες και εξαγόμενες πληροφορίες, σχεδιάζονται με τις κατάλληλες έννοιες στην περιοχή συγκεκριμένων οντολογιών. Σε αντίθεση από τις εισαγόμενες πληροφορίες του WSDL και των αποτελεσμάτων του WSDL, οι διαδικασίες του WSDL σχεδιάζονται επίσης στις οντολογικές έννοιες από μια διαδικασία οντολογιών [SVSM03]. Στο σχολιασμό των διαδικασιών του WSDL, οι προϋποθέσεις και τα αποτελέσματα είναι επίσης προστιθέμενα, ερμηνευμένα ως λογικοί όροι που πρέπει να ισχύσουν για την εκτέλεση της λειτουργίας, και σαν αλλαγές στον κόσμο που προκαλείται από την εκτέλεση της λειτουργίας, αντίστοιχα. Οι στόχοι του χρήστη εκφράζονται χρησιμοποιώντας τα πρότυπα υπηρεσιών βασισμένα στις έννοιες από τις οντολογίες περιοχών. Σε τέτοια πρότυπα, οι πληροφορίες για τη λειτουργία αναζητούνται



και οι εισαγόμενες και εξαγόμενες πληροφορίες τους δίνονται, και, προαιρετικά, οι προϋποθέσεις και τα αποτελέσματα μπορούν επίσης να διευκρινιστούν.

Ο στόχος αυτών που σχολιάζονται είναι να επιτραπεί η ταξινόμηση των υπηρεσιών Web που εδρεύουν στην περιοχή τους. Οι καταχωρήσεις είναι εξειδικευμένες σε μια δεδομένη περιοχή, και τα καταστήματα των υπηρεσιών Web(Web) σχετίζονται με εκείνη την περιοχή. Μια εξειδικευμένη οντολογία, η οντολογία καταχωρήσεων, χρησιμοποιείται για να σχολιάσει τις καταχωρήσεις, για να σχεδιάσει τις καταχωρήσεις σε μια δεδομένη περιοχή και να δώσει πρόσθετες πληροφορίες για τις καταχωρήσεις, να τις σχετίσει με άλλες καταχωρήσεις και με άλλες περιοχές.

### **Υποστήριξη για τη σύνθεση**

Η λειτουργικού επιπέδου σύνθεση μπορεί να εκμεταλλευτεί τη σημασιολογική περιγραφή των εισαγόμενων και εξαγόμενων πληροφοριών, των μετέπειτα καταστάσεων και των αποτελεσμάτων για να καθορίσει ποια μέρη ενός αιτήματος λύνονται από μια υπηρεσία και ποια άλλα παραμένουν άλυτα. Για την υψηλού επιπέδου επεξεργασία σύνθεσης, οι περιγραφές BPEL4WS που χρησιμοποιούνται στους METEOR-S μπορεί να χρησιμοποιηθούν. Εντούτοις, υπό αυτήν τη μορφή όσες περιγραφές έρχονται με μη επίσημη σημασιολογία, η λειτουργικού επιπέδου σύνθεση πρέπει να στηριχθεί στην εξωτερικά καθορισμένη σημασιολογία.

### **3.4.1 IRS-III**

Ο αρχικός στόχος της εκτέλεσης του προγράμματος IRS- III, από το KNOWLEDGE MEDIA INSTITUTE στο ανοικτό πανεπιστήμιο, είναι να υποστηρίξει την ανακάλυψη και την ανάκτηση των τμημάτων γνώσης (δηλ. υπηρεσίες) από τις βιβλιοθήκες που κατανέμονται στο Διαδίκτυο και την ημιαυτόματη διαμόρφωση τους προκειμένου να πραγματοποιηθούν οι συγκεκριμένοι στόχοι σύμφωνα με τις απαιτήσεις των χρηστών [MDCG03].

Το πλαίσιο εργασίας IRS- III μπορεί να θεωρηθεί ως προσαρμογή του πλαισίου εργασίας UPML [FMB+03] στην περιοχή των υπηρεσιών Web(Web).

### **Υποστήριξη για την ανακάλυψη και τη σύνθεση**

Το IRS- III έχει υιοθετήσει το εννοιολογικό πρότυπο WSMO. Επομένως, μπορούμε να υποθέσουμε ότι η υποστήριξη που παρέχεται σε WSMO για την ανακάλυψη και τη σύνθεση είναι ίδια όπως στο WSMO.

### **3.5 Μια περίπτωση χρήσης κίνητρου**

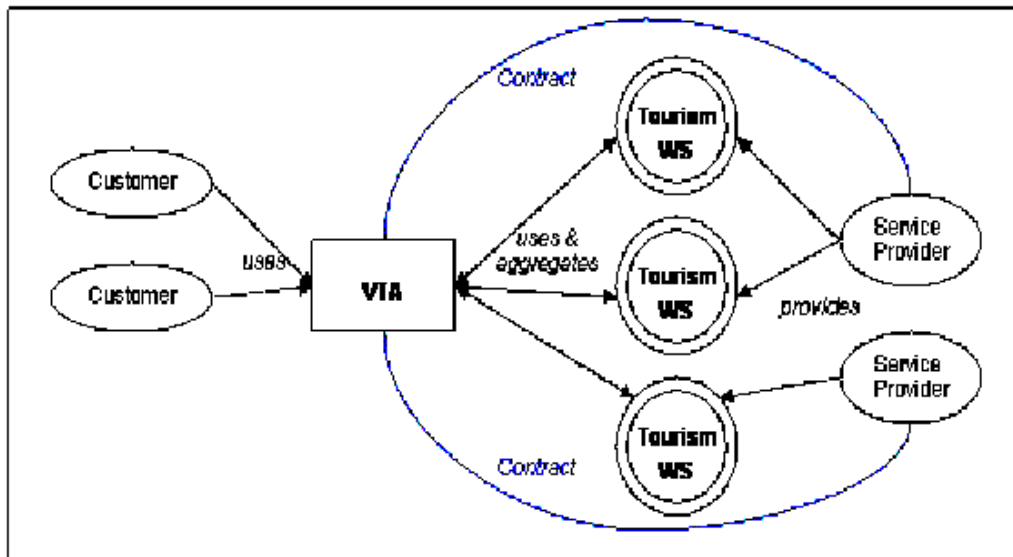
Οι διαφορετικές προσεγγίσεις για την περιγραφή των υπηρεσιών που παρουσιάζονται παραπάνω στοχεύουν στην υποστήριξη, την αυτόματη εγκατάσταση, την σύνθεση και την λειτουργικότητα με τις υπηρεσίες. Προκειμένου να διευκρινιστούν τα προβλήματα που στοχεύουμε να λύσουμε στο παρόν έγγραφο δηλ. την αυτόματη εγκατάσταση και τη σύνθεση από τις υπηρεσίες που βασίζονται σε τέτοιες περιγραφές, σε αυτό το τμήμα εισάγουμε μια περίπτωση χρήσης B2C που υποκινεί τέτοια προβλήματα και επεξηγεί πώς οι πραγματικές εφαρμογές μπορούν να ωφεληθούν από ένα υψηλότερο επίπεδο αυτοματοποίησης. Αυτή η περίπτωση χρήσης θα χρησιμοποιηθεί στην ακολουθία των κεφαλαίων για να επεξηγήσει τις προτάσεις μας για το πώς να πετύχουμε δυναμική ανακάλυψη και σύνθεση.

#### **3.5.1 Περιγραφή**

Φανταστείτε ένα εικονικό ταξιδιωτικό γραφείο [Virtual Traveling Agency (VTA εν συντομία) [Seditors] που είναι μια τελική πλατφόρμα χρήστη που παρέχει υπηρεσίες eTourism στους πελάτες. Αυτές οι υπηρεσίες μπορούν να καλύψουν όλα τα είδη υπηρεσιών πληροφοριών που αφορούν πληροφορίες τουρισμού, από τις πληροφορίες για τα αξιοθέατα και τους αρχαιολογικούς χώρους σε μια περιοχή έως και τις υπηρεσίες που υποστηρίζουν την κράτηση των πτήσεων, των ξενοδοχείων, των αυτοκινήτων ενοικίου, κ.λ.π. on-line. Τέτοια VTAs είναι ήδη υπάρχοντα, αλλά είναι συνήθως απλές πύλες πληροφοριών μαζί με μερικές βασισμένες στο Web εξυπηρετήσεις πελατών. Με την εφαρμογή των υπηρεσιών του Semantic Web(Semantic Web), ένα VTA θα επικαλεσθεί τις υπηρεσίες Web(Web) που παρέχονται από διάφορους προμηθευτές του eTourism και θα τους συναθροίσει μέσα σε μια νέα υπηρεσία εξυπηρέτησης πελατών σε μία (ημι)αυτόματη μόδα. Τέτοια VTAs παροχή που αυτοματοποιούν, eTourism υπηρεσίες αντιπροσωπεύουν μια εξέλιξη αυτήν την περίοδο να υπάρχουν VTAs, αφού μπορούν

δυναμικά να επιλέξουν (από ΟΛΕΣ τις διαθέσιμες υπηρεσίες τουρισμού) και να συνθέσουν τις κατάλληλες υπηρεσίες για να εκπληρώσουν ένα δεδομένο αίτημα τελικών χρηστών, που δεν περιορίζεται στους προσχεδιασμένους προμηθευτές με έναν προσχεδιασμένο τρόπο.

Η περίπτωση χρήσης VTA, που συναθροίζει υπηρεσίες Web(Web) από διαφορετικούς φορείς παροχής υπηρεσιών τουρισμού θα παρέχει την ακόλουθη γενική λειτουργία: ένας πελάτης χρησιμοποιεί την υπηρεσία VTA ως σημείο εισόδων για τις απαιτήσεις του. Τέτοιες υπηρεσίες τελικών χρηστών παρέχονται από το VTA με την ανακάλυψη, την υλοποίηση και τον συνδυασμό των υπηρεσιών Web(Web) που προσφέρονται από διάφορους φορείς παροχής υπηρεσιών τουρισμού.



Σχήμα 3.4 : Χρήση περιπτώσεων γενικής επισκόπησης .Εικονικός ταξιδιωτικός πράκτορας βασισμένος στις υπηρεσίες του Semantic Web(Semantic Web)

### 3.5.2 Πεδίο δράσης

Το σενάριο περιγράφει μια γενική δομή για VTAs που μπορεί να επεκταθεί σε πιο σύνθετα σενάρια όπου ο πελάτης μπορεί να είναι ο ίδιος μια υπηρεσία Web(Web), δημιουργώντας κατά συνέπεια ένα δίκτυο από συγκροτημένες υπηρεσίες που

προσφέρουν σύνθετες υπηρεσίες τουρισμού. Παραδείγματος χάριν, ένα VTA μπορεί να παρέχει υπηρεσίες κράτησης αεροπορικών εισιτηρίων για ένα σύνδεσμο αεροπορικών εταιρειών, ένα άλλο VTA συναθροίζει την υπηρεσία κρατήσεων για μια αλυσίδα ενός παγκόσμιου ξενοδοχείου, και ένα τρίτο VTA παρέχει υπηρεσίες κράτησης για ενοικιαζόμενα αυτοκίνητα σε συνδυασμό με υπηρεσίες διάφορων παγκόσμιων γραφείων ενοικίασης αυτοκινήτων. Κατόπιν, ένας άλλο VTA χρησιμοποιεί αυτές τις υπηρεσίες για την παροχή μιας υπηρεσίας τελικών χρηστών για την κράτηση των πλήρων διακοπών σε ταξίδια παγκοσμίως.

### 3.5.3 Δράστες, ρόλοι και στόχοι

Στην περίπτωση γενικής χρήσης υπάρχουν τρεις δράστες. Ο ακόλουθος καθορίζει γιατί συμμετέχουν σε αυτήν την περίπτωση χρήσης (στόχος) και τις ιδιαίτερες αλληλεπιδράσεις που περιλαμβάνονται μέσα σ' αυτές(ρόλοι).

Ø Πελάτης: ο τελικός χρήστης που ζητά μια υπηρεσία που παρέχεται από το VTA.

- Στόχος: αυτοματοποιημένη ανάλυση του αιτήματος υπηρεσιών τουρισμού.

- Ρόλος: ο τελικός χρήστης, αλληλεπιδρά με το VTA για τη χρήση, πληρωμής, και χωρίς τον υπολογισμό των προτερημάτων (π.χ. λαμβάνοντας το πραγματικό εισιτήριο κατά την κράτηση ενός ταξιδιού).

Ø Φορείς παροχής υπηρεσιών τουρισμού: εμπορικές επιχειρήσεις που παρέχουν συγκεκριμένες υπηρεσίες τουρισμού.

- Στόχος : πωλήσεις των υπηρεσιών στους πελάτες.

- Ρόλος: παρέχει τις υπηρεσίες τουρισμού σαν υπηρεσίες Web (επίσης παρέχει τις απαραίτητες περιγραφές των υπηρεσιών του Semantic Web(Semantic Web)).

Ø VTA: ο μεσάζων μεταξύ του πελάτη και των φορέων παροχής υπηρεσιών τουρισμού. Παρέχει υψηλής ποιότητας υπηρεσίες τουρισμού στους πελάτες με τη

συνάθροιση των ξεχωριστών υπηρεσιών που παρέχονται από τους ενιαίους φορείς παροχής υπηρεσιών.

- Στόχος: παρέχει τις υψηλής ποιότητας υπηρεσίες τουρισμού τελικών χρηστών, γίνεται χρήση των υπηρεσιών τουρισμού που ήδη υπάρχουν και τα συναθροίζει σε νέες υπηρεσίες. Αντιπροσωπεύει την ένωση των διαθέσιμων φορέων παροχής υπηρεσιών τουρισμού.
- Ρόλος: αλληλεπιδρά με τον πελάτη, εντοπίζει, συνθέτει (εάν είναι απαραίτητο) και χρησιμοποιεί τις υπηρεσίες που προσφέρονται από τους προμηθευτές τουρισμού και, τελικά, επιλύει το αίτημα των πελατών.

#### **3.5.4 Σενάρια χρήσης παραδείγματος**

Εδώ περιλαμβάνουμε μερικά πιθανά σενάρια που, μεταξύ των άλλων, μπορούν να εμφανιστούν στην περίπτωση χρήσης μας:

Ø Ο πελάτης ζητά το VTA για την έρευνα των προσφορών υπηρεσιών τουρισμού, το VTA ανιχνεύει και ερωτά τους κατάλληλους φορείς παροχής υπηρεσιών τουρισμού και δίνει τα αποτελέσματα στον πελάτη.

Ø Ο πελάτης επιλέγει μια συγκεκριμένη προσφορά και ζητά την κράτηση για αυτήν την προσφορά (που αλληλεπιδρά με το VTA), το VTA ανακαλύπτει και συναθροίζει τους φορείς παροχής υπηρεσιών τουρισμού για κράτηση, πληρωμή, κ.λπ. και επιστρέφει το αποτέλεσμα στον πελάτη.

Ø Το VTA αλληλεπιδρά με τον πελάτη και (με έναν ή περισσότερους) τους φορείς παροχής υπηρεσιών τουρισμού για τον έλεγχο της θέσης παράδοσης ενός δεδομένου εισιτηρίου πτήσης.

Μπορεί να φανεί ότι στην παραπάνω περίπτωση χρήσης η προστιθέμενη αξία του VTA ανήκει στη δυνατότητα να ανιχνευθούν αυτόματα οι αρμόδιοι φορείς παροχής υπηρεσιών τουρισμού, να αθροιστούν, και να προσφερθούν οι συναθροισμένες υπηρεσίες τουρισμού στον πελάτη σε διαφανή τρόπο.

Αυτή η περίπτωση χρήσης θα χρησιμοποιηθεί στα επόμενα κεφάλαια του παρόντος εγγράφου για να επεξηγήσει πώς εμείς προτείνουμε να επιτευχθεί η αυτόματη ανακάλυψη των φορέων παροχής υπηρεσιών και της σύνθεσής τους.

## Κεφάλαιο 4

### 4.1 Ένα εννοιολογικό πρότυπο για την ανακάλυψη

Μια εφαρμόσιμη προσέγγιση στην ανακάλυψη υπηρεσιών πρέπει να καθορίσει ακριβώς το πεδίο της και το εννοιολογικό πρότυπο και τις υποθέσεις πίσω από την προτεινόμενη λύση. Ενώ, όπως θα εκτεθεί λεπτομερώς μέσα στη παράγραφο 4.2, διάφορες προτάσεις είναι διαθέσιμες στον τομέα του ενδιαφέροντος μας, καμία τους δεν έχει συζητήσει ακριβώς αυτές τις πτυχές. Επομένως, αρχίζουμε με την εξέταση αυτού του στόχου και την παροχή του πεδίου, το πρότυπο και τις υποθέσεις που καθοδηγούν την πρότασή μας για την ανακάλυψη υπηρεσιών.

#### 4.1.1 Καθορισμός της υπηρεσίας

Θα καθορίσουμε τι ακριβώς εννοούμε με την υπηρεσία και, επομένως, τι είδους οντότητες στοχεύετε να ανακαλυφθούν για να φθάσει σε μια κοινή κατανόηση του προβλήματος και να περιγραφθούν ρητά οι υποθέσεις. Για την παροχή αυτού του ορισμού, εξετάζουμε τους ορισμούς που δίνονται στην εννοιολογική αρχιτεκτονική για τις σημασιολογικές υπηρεσίες Web που παρουσιάζονται στο [Pre04]:

∅ *Υπηρεσία ως παροχή αξίας σε κάποια περιοχή.* Αυτός ο καθορισμός θεωρεί μια υπηρεσία σαν παροχή αξίας (όχι απαραίτητως νομισματική αξία) σε κάποια δεδομένη περιοχή, ανεξάρτητα για το πώς ο προμηθευτής και ο καταναλωτής αλληλεπιδρούν. Παραδείγματα μιας υπηρεσίας από αυτή την άποψη είναι η παροχή πληροφοριών για τα εισιτήρια πτήσης από μια υπηρεσία προμήθειας τουρισμού, ή η παροχή κράτησης ταξιδιού με ορισμένα χαρακτηριστικά από ένα VTA.

∅ *Υπηρεσία ως οντότητα λογισμικού ικανή να παρέχει κάτι από την αξία.* Αυτό είναι η συνήθεις κατανόηση της υπηρεσίας στην κοινότητα IT σχετικά με την υπηρεσία ως λογισμικό οντότητας που παρέχει κάτι από την αξία (μια υπηρεσία υπό την ανωτέρω έννοια). Ένα παράδειγμα θα ήταν το λογισμικό που ένα VTA χρησιμοποιείται για την παροχή της αθροισμένης τουριστικής υπηρεσίας.

∅ *Υπηρεσία ως μέσο να αλληλεπιδράσης on-line με έναν φορέα παροχής υπηρεσιών.* Αυτός ο καθορισμός αναφέρεται σε υπηρεσίες όπως οι υπηρεσίες διαπραγμάτευσης, οι οποίες δεν παρέχουν τίποτα από την αξία τους αλλά καθιστούν

πιθανή την παροχή υπηρεσίας. Ένα παράδειγμα θα ήταν η χορογραφία WSMO που παρέχεται από έναν φορέα παροχής υπηρεσιών τουρισμού για να αλληλεπιδράσει με την κράτηση μιας δεδομένης πτήσης.

Στο υπόλοιπο αυτού του κεφαλαίου, θα αναφερθούμε στην υπηρεσία υπό την πρώτη έννοια, με την κατανόηση των υπηρεσιών Web ως υπηρεσίες που είναι προγραμματιστικά προσιτές πέρα από τον Ιστό δηλ. WSDL-όπως οι υπηρεσίες. Οι υπηρεσίες που προσφέρονται μέσω της ανθρώπινης αλληλεπίδρασης που χρησιμοποιεί τις παραδοσιακές σελίδες του Web δεν θα θεωρηθούν ρητά ως αυτές που είναι λιγότερο υποκείμενες στην αυτοματοποίηση, αλλά προσέξτε ότι μπορούν να ανακαλυφθούν και να συμβληθούν με τον ίδιο τρόπο που περιγράφεται ανωτέρω εάν κριθούν απαραίτητες οι περιγραφές που δίνονται.

#### **4.1.2 Επίπεδα αφαίρεσης**

Αφού καθορίσαμε τι καταλαβαίνουμε από την υπηρεσία, το επόμενο βήμα είναι να καθορίσουμε ποια είδη υπηρεσίας είναι σχετικά με μας. Για να κάνουμε κάτι τέτοιο, θα εξετάσουμε πάλι την εργασία που παρουσιάζεται στο [Pre04], όπου οι ακόλουθοι τύποι υπηρεσιών προσδιορίζονται:

∅ *Μια συγκεκριμένη υπηρεσία* είναι μια πραγματική ή πιθανή απόδοση ενός συνόλου στόχων που αντιπροσωπεύουν μια συνεπής λειτουργία μέσα σε κάποια περιοχή ενδιαφερόντων στις σχετικές οντότητές των αιτούντων και των προμηθευτών δηλ. μια συγκεκριμένη υπηρεσία είναι μια πραγματική υπηρεσία που θα είναι ή έχει παρασχεθεί, παραδείγματος χάριν η πραγματική κράτηση μιας πτήσης από ένα VTA.

∅ *Μια αφηρημένη υπηρεσία* είναι κάποιο σύνολο συγκεκριμένων υπηρεσιών, και μια αφηρημένη περιγραφή υπηρεσιών είναι κάποια περιγραφή D επεξεργάσιμη από μηχανή που έχει, ως πρότυπό της, την αφηρημένη υπηρεσία C δηλ. μια αφηρημένη περιγραφή υπηρεσιών διευκρινίζει το σύνολο πολλαπλών υπηρεσιών που μπορούν να είναι παρεχόμενες π.χ. κράτηση των πτήσεων που αναχωρούν από την Αυστρία.

∅ *Μια συμφωνηθείσα υπηρεσία* είναι μια αφηρημένη υπηρεσία που συμφωνείται μεταξύ δύο συμβαλλόμενων μερών δηλ. αντιπροσωπεύει τη συμφωνία μεταξύ ενός αιτούντος και ενός προμηθευτή για να λάβει και να εκτελέσει,

αντίστοιχα, μια δεδομένη υπηρεσία, παραδείγματος χάριν, η συμφωνία για την παροχή πληροφοριών πτήσεων για μια δεδομένη περιήγηση μεταξύ ενός φορέα παροχής υπηρεσιών τουρισμού και ενός VTA.

Ø *Μια σύμβαση υπηρεσιών είναι μια συμφωνία μεταξύ ενός φορέα παροχής υπηρεσιών και ενός αιτούντος όπου ο προμηθευτής θα παράσχει μια συμφωνηθείσα υπηρεσία στον αιτούντα π.χ. σε μια υπηρεσία τουρισμού ο προμηθευτής θα παράσχει την υπηρεσία που συμφωνείται με το VTA.*

Από αυτούς, η αφηρημένη υπηρεσία προσδιορίζεται όπως η σχετική για την ανακάλυψη των υπηρεσιών, ενώ οι συγκεκριμένες υπηρεσίες, οι συμφωνηθείσες υπηρεσίες, και οι συμβάσεις υπηρεσιών είναι σχετικές για τη φάση συμφωνητικού σύμβασης [Pre04].

#### **4.1.3 Πεδίο**

Όπως παρουσιάζεται στο [KLP+04], θα διαφοροποιήσουμε την ανακάλυψη υπηρεσιών και την υπηρεσία συνένωσης. Οι αφηρημένες περιγραφές υπηρεσιών που παρουσιάζονται ανωτέρω θα περιληφθούν στη φάση ανακαλύψεων. Στη φάση της συνένωσης, οι συγκεκριμένες υπηρεσίες θα χρησιμοποιηθούν, και αυτή η φάση θα οδηγήσει σε μια σύμβαση υπηρεσιών. Επομένως, το γενικό πεδίο για την ανακάλυψη όχι μόνο περιλαμβάνει την ανακάλυψη υπό την έννοια που παρουσιάζεται ανωτέρω, αλλά και την συνένωση των συγκεκριμένων σχετικών υπηρεσιών.

#### **4.1.4 Υποθέσεις**

Προκειμένου να καθοριστεί ένα πρότυπο για την ανακάλυψη και για τις Συνενώσεις υπηρεσιών, πρέπει να γίνουν σαφείς οι υποθέσεις μας στην περιοχή. Τέτοιες υποθέσεις συσχετίζονται κατωτέρω:

Ø υποθέσεις στους στόχους:

- Οι προκαθορισμένοι, γενικοί και επαναχρησιμοποιήσιμοι στόχοι θα είναι διαθέσιμοι στον αιτούντα, ο αιτών μπορεί να έχει το καθορισμό των γενικών στόχων.
- Οι προκαθορισμένοι στόχοι περιγράφονται κατά τρόπο επίσημο.
- Οι προκαθορισμένοι στόχοι μπορούν να ξαναβρεθούν (ή να παραμετροποιηθούν) από τον αιτούντα για να απεικονιστούν οι συγκεκριμένες ανάγκες του.



- Οι αιτούντες δεν θα γράψουν από την αρχή τους τυποποιημένους στόχους αλλά θα επαναχρησιμοποιήσουν τους προκαθορισμένους στόχους δηλ. δεν αναμένουμε τους αιτούντες για να τυποποιήσουμε τους στόχους τους από την αρχή.

-Οι αιτούντες πρέπει να είναι σε θέση να εντοπίσουν τους προκαθορισμένους στόχους που είναι σχετικοί μαζί τους δηλ. καθώς οι αιτούντες θα αναμένουν για να καθορίσουν τους προκαθορισμένους στόχους, υποθέτουμε ότι αυτός θα είναι ένας τρόπος για τους αιτούντες να εντοπίσουν τέτοιους προκαθορισμένους στόχους.

Ø υποθέσεις στις αφηρημένες περιγραφές υπηρεσιών:

-Οι αφηρημένες περιγραφές υπηρεσιών θα είναι πλήρεις αλλά όχι πάντα σωστές [Pre04] δηλ. κάθε συγκεκριμένη υπηρεσία που μπορεί να παρασχεθεί θα είναι πρότυπο της περιγραφής, αλλά θα υπάρξουν ενδεχομένως συγκεκριμένες υπηρεσίες που είναι πρότυπα αλλά η περιγραφή τους δεν μπορεί να παρασχεθεί πραγματικά. Παραδείγματος χάριν, μια υπηρεσία τουρισμού που παρέχει τις πτήσεις μέσα στην Ευρώπη (αλλά όχι όλες τις πιθανές πτήσεις) θα περιγράψει τις αφηρημένες υπηρεσίες της όντας σε θέση να παρέχει οποιαδήποτε πτήση μέσα στην Ευρώπη. Εντούτοις, θα υπάρξουν πτήσεις που είναι πρότυπο αυτής της περιγραφής δηλ. αυτές είναι οι πτήσεις μέσα στην Ευρώπη, αλλά δεν μπορούν πραγματικά να παρασχεθούν από αυτόν τον προμηθευτή. Αυτά τα λάθη είναι μια συνέπεια της απαραίτητης αφαίρεσης για να γίνουν οι περιγραφές εύχρηστες.

-Οι αφηρημένες περιγραφές υπηρεσιών θα περιλάβουν μόνο τα αναμενόμενα αποτελέσματα της υπηρεσίας, αλλά όχι ποια εισαγωγή πληροφοριών απαιτείται να επιτευχθεί.

-Ως αφηρημένη υπηρεσία η περιγραφή μπορεί να μην είναι σωστή, εάν μια συγκεκριμένη υπηρεσία μπορεί να παρασχεθεί πραγματικά και επομένως συμφωνηθεί και καθοριστεί κατά τη διάρκεια της φάσης της συνένωσης δηλ. κατά τη διάρκεια της φάσης της συμβαλλόμενης υπηρεσίας οι προμηθευτές θα συμφωνήσουν και θα καθιερώσουν τις συμβάσεις για τις συγκεκριμένες υπηρεσίες που αυτοί μπορούν πραγματικά να παρέχουν.

-Ένας φορέας παροχής υπηρεσιών που θα περιγράψει τις υπηρεσίες που είναι σε θέση να παρέχει την περιγραφή αφηρημένων υπηρεσιών, καλείται αφηρημένη ικανότητα.

Ø Υποθέσεις στις συγκεκριμένες υπηρεσίες:

-Ένας φορέας παροχής υπηρεσιών θα περιγράψει τις συγκεκριμένες υπηρεσίες που μπορεί να συμφωνήσει και να καθιερώσει μια σύμβαση με την παροχή γενικής ικανότητας. Η γενική ικανότητα θα περιλάβει επίσης την περιγραφή για το ποιοι όροι πρέπει να υπάρχουν για μια επιτυχή παροχή υπηρεσιών, καθώς επίσης και την απαιτούμενη σχέση για την εισαγωγή στα αποτελέσματα της υπηρεσίας.

Ø Υποθέσεις στη συνέπεια των περιγραφών:

-Η αφηρημένη ικανότητα μπορεί να προέρχεται αυτόματα από την ικανότητα και οι δύο πρέπει να είναι συνεπείς.

Ø Υποθέσεις στη φάση της συνένωσης.

-Ο στόχος του αιτούντος ως αποτέλεσμα του καθορισμού ενός προκαθορισμένου στόχου θα περιλάβει πληροφορίες απαραίτητες για την συνένωση, όπως οι πληροφορίες εισαγωγής πληροφοριών που μπορεί ο αιτών να παρέχει στην υπηρεσία. Δεν επιβάλλουμε ότι αυτό το (ενδεχομένως μεγάλο) σύνολο πληροφοριών πρέπει να απαριθμηθεί για κάθε στόχο, αλλά μπορεί να παρασχεθεί στη διαδικασία ανακαλύψεων με άλλα μέσα π.χ. μια πρόσθετη υπηρεσία αυτής παρέχει πληροφορίες για ό,τι ο αιτών έχει διαθέσιμο και είναι πρόθυμος να το γνωστοποιήσει.

-Μια σύμβαση δεν θα συμφωνηθεί εάν ο αιτών δεν είναι ικανός να παρέχει όλες τις πληροφορίες που απαιτούνται από τον προμηθευτή για να παραδώσει πραγματικά μια συγκεκριμένη υπηρεσία.

-Η επικοινωνία μεταξύ του αιτούντος και του προμηθευτή για να καθιερώσει μια σύμβαση που θα είναι διαφανής σε μας δηλ. δεν θα περιγράψουμε και δεν θα εξετάσουμε την υπηρεσία χρονογραφίας αλλά μόνο τα κατηγορήματα λογικής που, στην πράξη, θα περιλάβουν επικοινωνία είτε με τον αιτούντα είτε με τον προμηθευτή.

#### **4.1.5 Εννοιολογικό πρότυπο**

Με βάση τον καθορισμό της υπηρεσίας που επιλέχτηκε, τα διαφορετικά επίπεδα αφαίρεσης προσδιορίστηκαν, και τις υποθέσεις στην περιοχή που δίνονται ανωτέρω, παρέχουμε ένα εννοιολογικό πρότυπο για την ανακάλυψη αυτών που περιλαμβάνει την επαναχρησιμοποίηση των προκαθορισμένων στόχων, την ανακάλυψη των σχετικών αφηρημένων υπηρεσιών και την συνένωση των συγκεκριμένων υπηρεσιών για να εκπληρώσουν έναν συγκεκριμένο στόχο του αιτούντος. Το σχήμα 4.1 απεικονίζει τέτοιο

εννοιολογικό μοντέλο. Παρακάτω, παρέχουμε μια εξήγηση των διαφορετικών σταδίων από το εννοιολογικό πρότυπο.

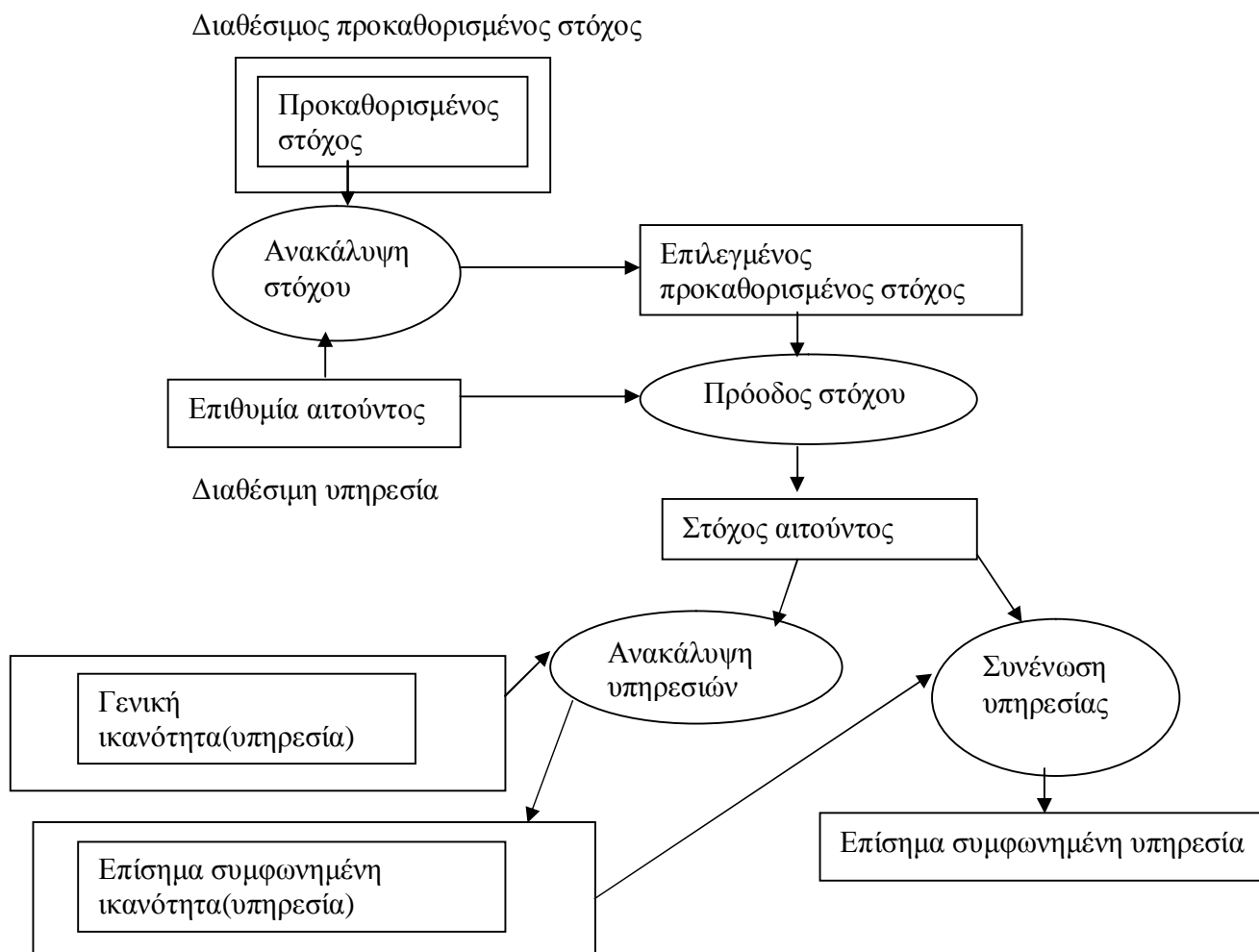
**Ανακάλυψη στόχου.** Ξεκινώντας από μια επιθυμία χρηστών (που εκφράζεται χρησιμοποιώντας τη φυσική γλώσσα ή οποιαδήποτε άλλα μέσα), η ανακάλυψη του στόχου θα εντοπίσει τον προκαθορισμένο στόχο που εγκαθιστά την επιθυμία του αιτούντος από το σύνολο προκαθορισμένων στόχων, που οδηγούν σε έναν επιλεγμένο προκαθορισμένο στόχο. Ένας προκαθορισμένος ο στόχος είναι μια αφαίρεση της επιθυμίας σε έναν γενικό και επαναχρησιμοποιήσιμο στόχο.

**Καθορισμός στόχου.** Ο επιλεγμένος προκαθορισμένος στόχος καθορίζεται, βασισμένος στη δοσμένη επιθυμία του αιτούντα, προκειμένου να απεικονιστεί πραγματικά μια τέτοια επιθυμία. Αυτό το βήμα θα οδηγήσει σε έναν τυποποιημένο στόχο του αιτούντα.

**Ανακάλυψη υπηρεσιών.** Οι διαθέσιμες υπηρεσίες που μπορούν, σύμφωνα με τις αφηρημένες ικανότητές τους, ενδεχομένως να εκπληρώσουν το στόχο του αιτούντα ανακαλύπτονται. Δεδομένου ότι η αφηρημένη ικανότητα δεν είναι εγγυημένη για να είμαστε σωστοί, δεν μπορούμε να βεβαιώσουμε σε αυτό το επίπεδο ότι η υπηρεσία θα εκπληρώσει πραγματικά το στόχο του αιτούντα.

**Συνένωση των υπηρεσιών.** Οι υπηρεσίες που ανακαλύπτονται που εδρεύουν στις αφηρημένες ικανότητές τους έχουν μια σχετική ικανότητα συνένωσης. Αυτή η ικανότητα συνένωσης θα χρησιμοποιηθεί στην υπηρεσία για να καθορίσει εάν η επιλεγμένη υπηρεσία μπορεί πραγματικά να εκπληρώσει το στόχο του αιτούντα, με τη σύναψη ενός συμφωνητικού σύμβασης. Εάν αυτό συμβαίνει, το αποτέλεσμα θα είναι η υπηρεσία.

Πάρτε για παράδειγμα έναν αιτούντα που θέλει να βρει τις πληροφορίες για τις πτήσεις από το Ίνσμπρουκ στη Μαδρίτη την 21η Δεκεμβρίου 2004. Αυτός ο αιτών μπορεί να εκφράσει την επιθυμία του ως κείμενο αναζήτησης πληροφοριών της μορφής «για τις πτήσεις από το Ίνσμπρουκ στη Μαδρίτη 21 Δεκεμβρίου 2004». Αυτό το κείμενο μπορεί να χρησιμοποιηθεί για να εκτελέσει το ταίριασμα βασισμένο στη λέξη που προκαθορίζει τους στόχους, όπως ένας προκαθορισμένος στόχος για τις πληροφορίες πτήσης.



Επίσημα συμφωνημένες ικανότητες  
 συσχετιζόμενων υπηρεσιών

Σχήμα 4.1: Εννοιολογικό πρότυπο ανακαλύψεων

Μόλις βρεθεί τέτοιος επίσημος προκαθορισμένος στόχος, θα καθοριστεί για να απεικονίσει τη συγκεκριμένη προέλευση και το προορισμό που δίνονται από τον αιτούντα, καθώς επίσης και την ημερομηνία. Αυτός ο καθορισμός μπορεί να γίνεται με το χέρι (υποστηριγμένος από τα κατάλληλα εργαλεία) ή αυτόματα από την επιθυμία κειμένου.

Εάν μια υπηρεσία VTA είναι διαθέσιμη να περιγράψει τις πτήσεις που μπορεί να παρέχει από οποιαδήποτε θέση στην Αυστρία σε οποιοδήποτε άλλο μέρος στην Ευρώπη (ως αφηρημένη ικανότητά του), αυτή η υπηρεσία θα είναι επιλεγμένη και εξετάζεται στη φάση της συνένωσης. Προσέξτε ότι σε αυτό το επίπεδο δεν αναμείνετε από το VTA να

περιγράψει ακριβώς όλες τις πραγματικές πτήσεις που μπορεί να παρέχει πληροφορίες, δεδομένου ότι γενικά δεν είναι ρεαλιστικό να αναμένεται οι προμηθευτές πληροφοριών πτήσης για να ξαναδιπλώσουν τις βάσεις δεδομένων για την πτήση στην υπηρεσία περιγραφής. Άντ' αυτού, θα παράσχουν μια αφαίρεση του είδους από την υπηρεσία που μπορούν να παρέχουν.

Κατά τη διάρκεια της φάσης συνένωσης, εάν η επιλεγμένη υπηρεσία μπορεί πραγματικά οι ζητούμενες πληροφορίες πτήσης θα ελεγχθούν δηλ. εάν το επιλεγμένο VTA μπορεί να παρέχει πληροφορίες για τις πτήσεις από το Ίνσμπρουκ στη Μαδρίτη κατά τη δεδομένη ημερομηνία θα εξεταστεί. Για αυτό, η ικανότητα συνένωσης του VTA θα χρησιμοποιηθεί, και τέτοια ικανότητα θα περιλαμβάνει κατηγορήματα λογικής που θα ρωτήσουν πραγματικά τη βάση δεδομένων του VTA (ή αθροισμένους φορείς παροχής υπηρεσιών τουρισμού) για να ελέγξουν εάν οι ζητούμενες πληροφορίες πτήσης είναι διαθέσιμες. Επιπλέον, και σε περίπτωση που το VTA επιθυμεί τις πρόσθετες πληροφορίες για να παρέχει την υπηρεσία του π.χ. οι λεπτομέρειες πελατών, θα ελεγχθούν εάν ο αιτών μπορεί να παρέχει τέτοιες πληροφορίες.

Εάν όλα τα ανωτέρω κριτήρια εκπληρώνονται, η υπηρεσία θα συμβληθεί και τελικά παρέχετε συγκεκριμένη υπηρεσία.

## **4.2 Κατάσταση προόδου στην ανάκτηση τμημάτων λογισμικού και ανακάλυψη υπηρεσιών Web**

Σε αυτό το τμήμα, παρουσιάζουμε και αναλύουμε το καταστάσεις στην τέχνη της ανακάλυψης υπηρεσιών. Λαμβάνοντας υπόψη την ισχυρή σχέση της με την ανακάλυψη υπηρεσιών Web, θα αρχίσουμε με την παρουσίαση στην παράγραφο 4.2.1 της εργασίας που γίνεται στον τομέα της ανάκτησης τμημάτων λογισμικού. Στην παράγραφο 4.2.2, αναλύουμε την διαθέσιμη εργασία στην αυτόματη ανακάλυψη υπηρεσιών Web.

### **4.2.1 Κατάσταση προόδου στην ανάκτηση τμημάτων λογισμικού**

Κατά τη διάρκεια της δεκαετίας του '90, η ερευνητική κοινότητα πληροφορικής έκανε μερικές προσπάθειες να βελτιωθεί η πρόοδος στην επαναχρησιμοποίηση των τμημάτων λογισμικού. Το κίνητρο για αυτήν την εργασία ήταν να υποστηρίξει την

επαναχρησιμοποίηση από τα ήδη να υπάρχοντα και εξέτασε τα τμήματα λογισμικού ως ένα βασικό παράγοντα για τα επιτυχή προγράμματα τεχνολογίας λογισμικού [SF97].

Η αποδοτική επαναχρησιμοποίηση των αξιόπιστων τμημάτων λογισμικού που παρέχουν μια δεδομένη λειτουργία προφανώς απαιτεί αποδοτικά μέσα να βρουν τέτοια συστατικά. Μια χειρωνακτική προσέγγιση, μέσα στην οποία ο μηχανικός λογισμικού πρέπει να κοιτάξει βιαστικά (ενδεχομένως ένα μεγάλο αριθμό) τις βιβλιοθήκες για να εντοπίσει το κατάλληλο συστατικό, σαφώς δεν μετριέται. Για αυτόν τον λόγο, οι ερευνητικές προσπάθειες προσανατολίστηκαν προς την επίσημη προδιαγραφή της συστατικής λειτουργίας και στην επίσημη περιγραφή του επιδιωκόμενου συστατικού προκειμένου να επιτραπεί η ημιαυτόματη ανάκτηση για τα κατάλληλα συστατικά.

Φαίνεται ότι το πρόβλημα για τα ημιαυτόματα τμήματα λογισμικού είναι ιδιαίτερα παρόμοια με την αυτόματη θέση των υπηρεσιών. Η συγκεκριμένη σχέση με το πρόβλημα να εντοπιστούν στο Web αυτόματα οι υπηρεσίες, θα συζητηθεί στο τέλος αυτού του τμήματος.

Παρακάτω, περιγράφουμε εν συντομία την εργασία που γίνεται στον τομέα του τμήματος λογισμικού ανάκτησης για να διευκρινίσει τη λειτουργία των τμημάτων λογισμικού και για να καθορίσει τυπικά τις διαφορετικές έννοιες της αντιστοιχίας.

### **Αντιστοιχία προδιαγραφών**

Η αντιστοιχία προδιαγραφών έχει προταθεί σε διάφορες εργασίες π.χ. [JC92, JC93, JC95, RW91, ZW95] για να αξιολογήσει πώς τα τμήματα λογισμικού αφορούν μία δεδομένη ερώτηση χρήστη δηλ. κάποια ανάγκη. Το ταίριασμα των προδιαγραφών στηρίζεται σε αξιώματα των τμημάτων λογισμικού και στις ερωτήσεις των χρηστών. Μια επίσημη (λογική) σχέση καθορίζεται και μια δεδομένη ερώτηση ελέγχει αν το συστατικό ικανοποιεί αυτήν την σχέση. Μια τέτοια σχέση πρέπει να συλλάβει την έννοια της ικανότητας επαναχρησιμοποίησης δηλ. εάν η σχέση ισχύει για τα τυπικά διευκρινισμένα συστατικά και τις ερωτήσεις, αυτό σημαίνει ότι το συστατικό μπορεί να επαναχρησιμοποιηθεί για να λύσει το πρόβλημα που συλλαμβάνεται από την ερώτηση.

Στο επίπεδο διαμόρφωσης, οι ακόλουθες ερωτήσεις πρέπει να απαντηθούν: α) πώς τα συστατικά και οι ερωτήσεις διευκρινίζονται, και β) ποια είναι η σχέση που ελέγχεται για να καθορίσει την ικανότητα επαναχρησιμοποίησης.

Όπως μέσα στο [CC00], τα ευρέως χρησιμοποιημένα αξιώματα των συστατικών και των ερωτήσεων βασίζεται στο [Hoa69]. Το [Hoa69] παρέχει μια λογική βάση για να αποδείξει μερικές ιδιότητες ενός προγράμματος, συμπεριλαμβανομένου του καθορισμού εάν ένα δεδομένο πρόγραμμα παρέχει μια ορισμένη λειτουργία. Η προοριζόμενη λειτουργία ενός προγράμματος (C) έχει διευκρινισμένη άποψη από τις αρχικές προϋποθέσεις ( $C_{pre}$ ) δηλ. οι ισχυρισμοί για ορισμένες ιδιότητες των τιμών που λαμβάνονται από τις σχετικές μεταβλητές πριν από την έναρξη του προγράμματος και των σχέσεων μεταξύ τους, και των έπειτα συνθηκών ( $C_{post}$ ) δηλ. το ίδιο είδος των ισχυρισμών όπως των προϋποθέσεων αλλά και των τιμών μετά από την εκτέλεση. Η σχέση μεταξύ των προϋποθέσεων και των μετέπειτα συνθηκών ενός δεδομένου προγράμματος διατυπώνεται ως εξής:

$$C_{pre}\{Q\}C_{post}$$

ερμηνευμένος όπως «εάν ο ισχυρισμός  $C_{pre}$  είναι αληθινός πριν από την έναρξη ενός προγράμματος Q, έπειτα ο ισχυρισμός  $C_{post}$  θα είναι αληθινός με την ολοκλήρωση του.» [Hoa69]

Με βάση αυτόν τον τύπο, το μεγαλύτερο μέρος της εργασίας που γίνεται στο ταίριασμα προδιαγραφών διευκρινίζει ένα συστατικό C ως διπλό των κατηγορημάτων ( $C_{pre}$ ,  $C_{post}$ ), όντας  $C_{pre}$  η προϋπόθεση από το συστατικό και το  $C_{post}$  θα είναι η κατάσταση έπειτα. Ομοίως, μια ερώτηση Q διευκρινίζεται ως ( $Q_{pre}$ ,  $Q_{post}$ ). Οι προϋποθέσεις του συστατικού είναι λογικοί τύποι που πρέπει να κρατήσουν προγενέστερα τη χρήση των συστατικών, και η μετέπειτα κατάσταση είναι λογικοί τύποι που είναι εγγυημένοι να είναι αληθινοί μετά από την εκτέλεση του συστατικού. Οι προϋποθέσεις και οι μετέπειτα καταστάσεις από την ερώτηση δίνουν έναν χαρακτηρισμό του επιθυμητού συστατικού από την άποψη των προϋποθέσεων του και της μετέπειτα κατάστασης. Οι προϋποθέσεις ερώτησης μπορούν να ερμηνευθούν ως αρχική περιγραφή για την οποία το επιδιωκόμενο συστατικό πρέπει να εγγυηθεί την εκπλήρωση της μετά της από την κατάσταση ερώτησης.

Match	Definition
1. $M_{\text{exact-pre/post}}$	$(Q_{\text{pre}} \leftrightarrow C_{\text{pre}}) \wedge (C_{\text{post}} \leftrightarrow Q_{\text{post}})$
2. $M_{\text{plug-in}}$	$(Q_{\text{pre}} \rightarrow C_{\text{pre}}) \wedge (C_{\text{post}} \rightarrow Q_{\text{post}})$
3. $M_{\text{plug-in-post}}$	$C_{\text{post}} \rightarrow Q_{\text{post}}$
4. $M_{\text{guarded-plug-in}}$	$(Q_{\text{pre}} \rightarrow C_{\text{pre}}) \wedge ((C_{\text{pre}} \wedge C_{\text{post}}) \rightarrow Q_{\text{post}})$
5. $M_{\text{relaxed-plug-in}}$	$(Q_{\text{pre}} \rightarrow C_{\text{pre}}) \wedge ((Q_{\text{pre}} \wedge C_{\text{post}}) \rightarrow Q_{\text{post}})$
6. $M_{\text{guarded-post}}$	$(C_{\text{pre}} \wedge C_{\text{post}}) \rightarrow Q_{\text{post}}$
7. $M_{\text{partial-comp}}$	$C_{\text{pre}} \wedge Q_{\text{pre}} \wedge C_{\text{post}} \rightarrow Q_{\text{post}}$
8. $M_{\text{exact-pred}}$	$(C_{\text{pre}} \rightarrow C_{\text{post}}) \leftrightarrow (Q_{\text{pre}} \rightarrow Q_{\text{post}})$
9. $M_{\text{gen-pred}}$	$(C_{\text{pre}} \rightarrow C_{\text{post}}) \rightarrow (Q_{\text{pre}} \rightarrow Q_{\text{post}})$
10. $M_{\text{spe-pred}}$	$(Q_{\text{pre}} \rightarrow Q_{\text{post}}) \rightarrow (C_{\text{pre}} \rightarrow C_{\text{post}})$
11. $M_{\text{exact-pred-2}}$	$(C_{\text{pre}} \wedge C_{\text{post}}) \leftrightarrow (Q_{\text{pre}} \wedge Q_{\text{post}})$
12. $M_{\text{gen-pred-2}}$	$(C_{\text{pre}} \wedge C_{\text{post}}) \rightarrow (Q_{\text{pre}} \wedge Q_{\text{post}})$
13. $M_{\text{guarded-gen-pred}}$	$(Q_{\text{pre}} \rightarrow C_{\text{pre}}) \wedge ((C_{\text{pre}} \rightarrow C_{\text{post}}) \rightarrow (Q_{\text{pre}} \rightarrow Q_{\text{post}}))$

Πίνακας 4.2: Περίληψη των αντιστοιχιών προδιαγραφών

Το [ZW97] ερευνά τις διαφορετικές έννοιες της αντιστοιχίας για την ανάκτηση του τυπικά διευκρινισμένου συστατικού του λογισμικού. Ένα τμήμα  $C$  λογισμικού περιγράφεται από την άποψη της υπογραφής τους,  $C_{\text{sig}}$ , και τη προδιαγραφή συμπεριφοράς τους,  $C_{\text{spec}}$ . Το πρώτο περιγράφει στατικά τις ελέγξιμες πληροφορίες (πληροφορίες συστατικών τύπων), ενώ το τελευταίο περιγράφει δυναμικά το συστατικό της λειτουργίας συμπεριφοράς. Αν και οι δύο πτυχές πρέπει να αντιστοιχηθούν για την ανάκτηση του τμήματος λογισμικού της δεδομένης ερώτησης, εδώ ενδιαφερόμαστε για την προδιαγραφή και το ταίριασμα από τη συστατική λειτουργία. Το  $C_{\text{spec}}$  είναι, όπως εξηγείται ανωτέρω, περιγραμμένο από τις προϋποθέσεις και τις μετέπειτα καταστάσεις του συστατικού. Ομοίως, μια ερώτηση  $Q$  περιγράφεται από της προϋποθέσεις και τις μετέπειτα καταστάσεις. Όλα οι προϋποθέσεις και οι μετέπειτα καταστάσεις είναι first-order τύποι.

Το [ZW97] προσδιορίζει την ακριβή αντιστοιχία βασισμένη στις προ και στις μετέπειτα καταστάσεις του  $Q$  και του  $C$  (1η σειρά στον πίνακα 4.2), ο οποίος αντιστοιχεί την περίπτωση στην οποία το  $C$  και το επιθυμητό συστατικό στο  $Q$  διευκρινίζονται ως ισοδύναμα. Δεδομένου ότι η ισοδυναμία μεταξύ των προδιαγραφών είναι μια ισχυρή απαίτηση και σε πολλές περιπτώσεις ένα γενικότερο ή πιο συγκεκριμένο



συστατικό μπορεί να είναι χρήσιμο, καθορίζονται οι διάφορες πιο χαλαρές έννοιες της αντιστοιχίας.

Στη **plug-in match** (2ρη σειρά στον πίνακα 4.2), το Q αντιστοιχείται από τα συστατικά με πιο αδύνατες προϋποθέσεις και με πιο ισχυρές μετέπειτα καταστάσεις δηλ. τα συστατικά των μετέπειτα καταστάσεων υπονοούν (και, επομένως, ικανοποιούν) της ερώτησης της μετέπειτα κατάστασης, και οι προϋποθέσεις της ερώτησης υπονοούν προϋποθέσεις του συστατικού. Διαισθητικά, το C μπορεί να χρησιμοποιηθεί για τη λήψη της συμπεριφοράς που διευκρινίζεται στο Q, αλλά όχι με άλλο τρόπο. Η *plug-in match* χρησιμοποιείται επίσης μέσα στα [SF97, PA97]. Στα [JC93, JC94] χρησιμοποιείται μια δοκιμή υποκατηγορίας που είναι ισοδύναμη με τη *plug-in match*, και τα [LW94, Ame91] χρησιμοποιούν τη *plug-in match* για τον καθορισμό μιας έννοιας.

Η **plug-in post match** (3τη σειρά στον πίνακα 4.2) δεν εξετάζει τις προϋποθέσεις για τον καθορισμό μιας αντιστοιχίας. Αυτή η έννοια αντιστοιχεί στις περιπτώσεις στις οποίες φροντίζουμε μόνο για τα αποτελέσματα της εκτέλεσης των συστατικών. Οι προϋποθέσεις του συστατικού μπορούν να βεβαιωθούν σε ένα μεταγενέστερο στάδιο από το χρήστη του συστατικού. Σε αυτήν την περίπτωση, η σχέση που πρέπει να κρατηθεί μεταξύ του C και του Q είναι ότι η μετέπειτα της κατάστασης της C πρέπει να υπονοηθεί δηλ. να εγγυηθεί τη μετέπειτα κατάσταση του Q. Αυτή η έννοια εξετάζεται επίσης μέσα στο [PA97].

Η **guarded plug-in match** (4τη σειρά στον πίνακα 4.2) περιορίζει τη *plug-in match* εισάγοντας μια φρουρά που δηλώνει ότι τα συστατικά της μετέπειτα κατάστασης απαιτούνται για να υπονοήσουν την ερώτηση της μετέπειτα κατάστασης μόνο σε περίπτωση όπου οι προϋποθέσεις των συστατικών ικανοποιούνται. Για ένα παράδειγμα μιας τέτοιας περίπτωσης όπου αυτή η υπόθεση είναι απαραίτητη, παραπέμπουμε τον αναγνώστη στο [ZW97]. Η ίδια έννοια χρησιμοποιείται μέσα στο [PA97], αλλά αποκαλείται αδύνατη σύνδεση.

Η **guarded post match** (6τη σειρά στον πίνακα 4.2) αφορά τη *guarded plug-in match* με τον ίδιο τρόπο που η *plug-in post match* σχετίζεται με την *plug-in match* δηλ. είναι το αποτέλεσμα της μείωσης της σχέσης της προϋπόθεσης που ελέγχεται από τη *guarded plug-in match*. Το [SF97] επίσης περιγράφει τη *guarded post match*. Αυτή η έννοια είναι ισοδύναμη με την *weak post* αντιστοιχία στο [PA97].

Στην **exact predicate match** (8η σειρά στον πίνακα 2.1), η σχέση μεταξύ των προϋποθέσεων και των μετέπειτα της κατάστασης του συστατικού πρέπει να είναι ισοδύναμες με τη σχέση μεταξύ της πριν και της μετέπειτα της κατάστασης ερώτησης, δηλαδή οι λειτουργίες (η σχέση μεταξύ της πριν και της μετέπειτα της κατάστασης) διευκρινίζουν την ερώτηση και το συστατικό πρέπει να είναι ισοδύναμο. Όπως δηλώνετε στο [ZW97],

$M_{\text{exact-pre/post}} \rightarrow M_{\text{exact-pred}}$ , και είναι ισοδύναμα σε περιπτώσεις όπου  $C_{\text{pre}} = Q_{\text{pre}} = (\text{true})$  αληθινό δηλ. η ακριβής αντιστοιχία (exact match) του κατηγορήματος είναι λιγότερο αυστηρή από την ακριβή αντιστοιχία (exact match).

Η **Generalized match** (9η σειρά στον πίνακα 4.2), αντιστοιχεί στη διαίσθηση που η περιγραφή από τα συστατικά θα είναι πλήρης, ενώ η περιγραφή των ερωτήσεων μπορεί απλά να κρατηθεί. Αυτή η έννοια της αντιστοιχίας ανακτά τα συστατικά που παρέχουν μια λειτουργία περισσότερο γενική από ένα που διευκρινίζεται στην ερώτηση. Η generalized match είναι λιγότερο αυστηρή από την plug-in match δηλ.  $M_{\text{plug-in}} \rightarrow M_{\text{gen-pred}}$ .

Οι εναλλακτικές εκδόσεις της ακριβής αντιστοιχίας κατηγορήματος και της generalized match μπορούν να ληφθούν με την αντικατάσταση της σχέσης επίπτωσης μεταξύ των προϋποθέσεων και των μετέπειτα της κατάστασης και στα δύο συστατικό και στην ερώτηση που έτυχαν μαζί (11τη και 12τη σειρά στον πίνακα 4.2, αντίστοιχα).

Η **guarded generalized predicate match** (13η σειρά στον πίνακα 4.2) προέρχεται από τη generalized match με τους περιορισμούς της generalized match στην περιοχή που καθορίζεται από τις προϋποθέσεις ( $Q_{\text{pre}}$ ) της ερώτησης.

Η **Specialized match** (10η σειρά στον πίνακα 4.2) ανακτά τα συστατικά των οποίων η λειτουργία είναι πιο εξειδικευμένη από τη ζητούμενη λειτουργία. Η λογική πίσω από αυτήν την έννοια από την αντιστοιχία είναι ότι μια λειτουργία της οποίας η προδιαγραφή είναι πιο αδύνατη από την ερώτηση που μπορεί ακόμα να έχει ενδιαφέρον ως βάση για να εφαρμόσει την επιθυμητή λειτουργία [ZW97].

Το [SF97] εισάγει τη **partial compatibility match** (7η σειρά στον πίνακα 4.2). Χρησιμοποιώντας αυτή την έννοια της αντιστοιχίας, ένα συστατικό θα αντιστοιχηθεί εάν υπολογίζει τα απαραίτητα αποτελέσματα για μια κοινή περιοχή. Αυτή η περιοχή καθορίζεται από τις προϋποθέσεις της ερώτησης και τις προϋποθέσεις του συστατικού

που τυχαίνουν μαζί. Εάν αυτό δεν είναι ικανοποιητικό δηλ. οι περιοχές από την ερώτηση και το συστατικό είναι χωρισμένα, ο τύπος θα είναι επίσης αληθινός. Εντούτοις, αναμένεται ότι οι χωρισμένες οι περιοχές θα απορριφθούν με το ταίριασμα υπογραφών.

Η **relaxed plug-in match** (5η σειρά στον πίνακα 4.2), καλείτε επίσης *satisfies match* στο [PA97], εισάγεται στο [SF97]. Εδώ, η φρουρά της σχέση μεταξύ του συστατικού και της ερώτησης της μετέπειτα κατάστασης είναι η προϋπόθεση της ερώτησης δηλ. ο έλεγχος της σχέσης είναι περιορισμένη περιοχή που δίνεται από τις προϋποθέσεις ερώτησης. Προσέξτε ότι αυτή η έννοια της αντιστοιχίας υπονοείτε στη guarded plug-in match λόγω της επίπτωσης στον τύπο για τη relaxed plug-in δηλ. τα συστατικά που ανακτώνται από τη guarded plug-in είναι ένα υποσύνολο αυτών που ανακτώνται από τη relaxed plug-in.

Ο πίνακας 4.2, που προσαρμόζεται και που επεκτείνεται από το [CC00], παρέχει μια περίληψη των διαφορετικών εννοιών από την αντιστοιχία που παρουσιάζεται μέσα στα [JC95, ZW95, PA97, SF97, Ame91, LW94, DL96].

Το σχήμα 4.3 παρουσιάζει τη σχέση μεταξύ των εννοιών της αντιστοιχίας που παρουσιάζονται ανωτέρω. Ένα βέλος από μια έννοια σε μια άλλη σημαίνει ότι ο πρώτος είναι ισχυρότερος δηλ. η πρώτη υπονοεί τη δεύτερος.

Τα Chen και Cheng [CC00] συζητούν ένα μέτρο της ικανότητας επαναχρησιμοποίησης και ταξινομούν μερικές από προαναφερθείσες έννοιες της αντιστοιχίας σύμφωνα με αυτό το μέτρο. Λαμβάνοντας υπόψη οποιοδήποτε συστατικό C και ερώτηση Q, εξετάζουν μια αντιστοιχία M (C, Q) για να εξασφαλίσει την επαναχρησιμοποίηση αν η ακόλουθος σχέση υπάρχει:

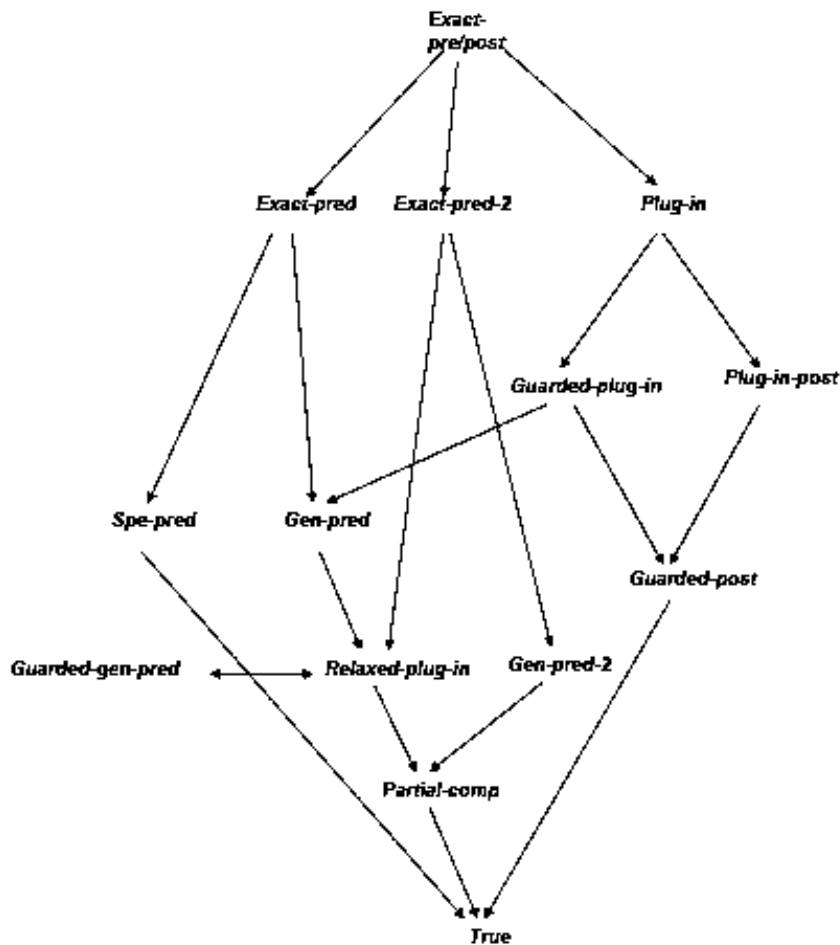
$$M(C, Q) \wedge \{C_{pre}\}C\{C_{post}\} \rightarrow \{Q_{pre}\}C\{Q_{post}\}$$

όπου  $\{p\}C\{q\}$  σημαίνει ότι εάν το p κρατά την αρχή της εκτέλεσης του C, η εκτέλεση του C θα ολοκληρώσει με το q να ικανοποιείται. Η διαισθητική έννοια του τύπου είναι ανώτερη από αυτή, εάν ένα συστατικό αντιστοιχείται, και έχει τις προϋποθέσεις  $C_{pre}$  και τις μετέπειτα καταστάσεις  $C_{post}$ , οι μετέπειτα καταστάσεις της ερώτησης θα ικανοποιηθούν από την εκτέλεση του C υπό τον όρο ότι οι προϋποθέσεις της ερώτησης ισχύουν πριν από την εκτέλεσή του.

Κάτω από αυτόν τον καθορισμό, αποδεικνύεται ότι η exact pre/post plug-in, η exact-pred-2, η relaxed plug-in, και η guarded-gen-pred αντιστοιχίες επαναχρησιμοποιούνται και εξασφαλίζουν αντιστοιχίες.

### Σχέση με τη σημασιολογική ανακάλυψη υπηρεσιών Web

Όπως έχει παρουσιαστεί πιο πάνω, οι υπηρεσίες Web περιγράφονται χρησιμοποιώντας μια παρόμοια προσέγγιση αξιωμάτων Hoare των τμημάτων λογισμικού. Στην πραγματικότητα, η μόνη διαφορά που προκύπτει στην περιγραφή των υπηρεσιών του Web είναι ότι, λόγω της σειράς εφαρμογής τους, μπορούν γενικά να έχουν αποτελέσματα στον πραγματικό κόσμο. Επομένως, είναι σημαντικό να χωρίσει (τις πληροφορίες)



Σχήμα 4.3 : Δικτυωτό πλέγμα των αντιστοιχιών προδιαγραφών λειτουργίας της μετέπειτα κατάστασης από τα πραγματικά αποτελέσματα, και τις (πληροφορίες) προϋποθέσεις από τις πραγματικές υποθέσεις, ως διαφορετικές έννοιες της αντιστοιχίας

που μπορούν να εφαρμοστούν σε αυτή. Ερωτήσεις στο λογισμικό της ανάκτησης συστατικών περιγράφεται με παρόμοιο τρόπο στα αιτήματα ή στους στόχους των υπηρεσιών του Web.

Μερικές από τις έννοιες της αντιστοιχίας που συζητούνται στα πλαίσια της ανάκτησης τμημάτων λογισμικού έχει χρησιμοποιηθεί στην ανακάλυψη των υπηρεσιών του Semantic Web, όπως θα δούμε παρακάτω οι έννοιες όπως η exact ή η plug-in αντιστοιχία εφαρμόζονται στην αυτόματη θέση των υπηρεσιών του Web. Εντούτοις, ένα εννοιολογικό πρότυπο για την ανακάλυψη των υπηρεσιών του Web εισάγει τα διαφορετικά στάδια όπου μερικές έννοιες της αντιστοιχίας που ερευνώνται μέσα στην ανάκτηση τμημάτων λογισμικού είναι ενδιαφέρουσες ενώ άλλες δεν έχουν το πρακτικό ενδιαφέρον.

**Ανακάλυψη στόχου.** Στην ανακάλυψη στόχου δεν αναμένουμε από τον αιτούντα να παρέχει άμεσα την επίσημη προδιαγραφή των επιθυμιών του. Στην πραγματικότητα, εάν δίνεται, θα πηδήσουμε την ανακάλυψη και τα refinement βήματα και συνεχίζουμε με τον τυποποιημένο στόχο στην υπηρεσία της ανακάλυψης των βημάτων. Επομένως, οι έννοιες της αντιστοιχίας που εισάγονται σε αυτό το τμήμα δεν εφαρμόζονται.

**Ανακάλυψη υπηρεσιών.** Κατά τη διάρκεια της ανακάλυψης υπηρεσιών θα συνεργαστούμε μόνο με την περιγραφή της αφηρημένης υπηρεσίας, οι οποίες περιέχουν μόνο μια περιγραφή των αποτελεσμάτων που μπορούν να παρασχεθούν από την υπηρεσία. Επομένως, θα εξετάσουμε μόνο τις μετέπειτα καταστάσεις και τα αποτελέσματα σε αυτό το βήμα, και μόνο οι έννοιες της αντιστοιχίας που εξετάζουν το σύνολο των αποτελεσμάτων που μπορούν να παρασχεθεί από την υπηρεσία wrt. Το σύνολο των ζητούμενων αποτελεσμάτων είναι ενδιαφέρον. Η plug-in-post έννοια της αντιστοιχίας που παρουσιάστηκε πριν είναι μοναδική αυτού του είδους και, όπως θα δούμε πιο κάτω, άλλες έννοιες που βασίζονται σε μερικές από τις έννοιες που ερευνώνται στην ανάκτηση τμημάτων λογισμικού όπου η προϋπόθεση του έλεγχου είναι απαραίτητη.

**Συνένωση υπηρεσιών.** Η συνένωση υπηρεσιών περιλαμβάνει Συνενώσεις μεταξύ των υπηρεσιών που επιλέγονται κατά τη διάρκεια της ανακάλυψης, μιας συγκεκριμένης υπηρεσίας που καλύπτει το στόχο του αιτούντος. Σε αυτήν την περίπτωση, η ικανότητα συνένωσης μιας υπηρεσίας θα περιγράψει τη πραγματική σχέση μεταξύ των εισαγωγών

πληροφοριών και των αποτελεσμάτων της υπηρεσίας, η οποία είναι ο τρόπος που τα τμήματα λογισμικού ήταν συνήθως. Επιπλέον, ο στόχος του αιτούντος περιλαμβάνει μια περιγραφή των πληροφοριών που μπορούν να παρασχεθούν, που μπορεί να αντιμετωπιστεί όπως οι προϋποθέσεις της ερώτησης στην ανάκτηση των τμημάτων λογισμικού. Οι έννοιες της αντιστοιχίας που παρουσιάζεται στα τμήματα είναι βασισμένες σε μερικές από τις έννοιες που ερευνώνται στη περιοχή ανάκτησης τμημάτων λογισμικού.

Αν και πολλές από τις έννοιες της αντιστοιχίας που παρουσιάζονται σε αυτό το τμήμα δεν μπορούν να είναι άμεσα χρησιμοποιημένες στο εννοιολογικό πρότυπό μας για την ανακάλυψη και την συνένωση υπηρεσιών, θα δούμε μέσα στο υπόλοιπο αυτού του κεφαλαίου πως οι περισσότερες από τις έννοιες της αντιστοιχίας που χρησιμοποιήθηκαν στο εννοιολογικό πρότυπο έχει τις ρίζες του στην εργασία που γίνεται στην ανάκτηση τμημάτων λογισμικού, με τις απαραίτητες προσαρμογές για την περιοχή των υπηρεσιών του Web. Ιδιαίτερα, ενώ οι έννοιες αντιστοιχία που συζητούνται εδώ εστιάζονται στην εντόπιση ενός τμήματος λογισμικού που μπορεί να χρησιμοποιηθεί στη θέση όπου το τμήμα λογισμικού που αντιπροσωπεύθηκε από την ερώτηση που ήταν δυνατόν, στην υπηρεσία ανακάλυψης εστιάζουμε ποια αποτελέσματα μπορούν να παραδοθούν από την υπηρεσία. Παραδείγματος χάριν, η plug-in έννοια της αντιστοιχίας που εισάγεται στην ανάκτηση τμημάτων λογισμικού απαιτεί το συστατικό της μετέπειτα κατάστασης να είναι πιο συγκεκριμένο από τις μετέπειτα καταστάσεις της ερώτησης, ενώ στην περίπτωση μας εμείς θα αντιστρέψουμε αυτήν την σχέση, δεδομένου ότι θέλουμε μια εγγύηση ότι η υπηρεσία θα παραδώσει ότι ήταν ζητούμενο στο στόχο δηλ. κάθε αποτέλεσμα που περιγράφεται από τη μετέπειτα κατάσταση στόχου πρέπει να είναι ένα αποτέλεσμα από την υπηρεσία και, επομένως, ο στόχος της μετέπειτα κατάστασης πρέπει να υπονοήσει τη μετέπειτα κατάσταση των υπηρεσιών.

#### **4.2.2 Κατάσταση προόδου στην ανακάλυψη υπηρεσιών Web**

Σε αυτό το τμήμα, θα συζητήσουμε την εργασία που γίνεται στην αυτόματη ανακάλυψη υπηρεσιών Web που βασίζεται στη σημασιολογική περιγραφή της λειτουργίας των υπηρεσιών του Web και των αιτημάτων των χρηστών. Αλλιώς η ανακάλυψη υπηρεσιών Web σημαίνει ότι προσφέρει ένα χαμηλό επίπεδο της

αυτοματοποίησης, όπως η χρήση [BCE+02], τα ληξιαρχεία UDDI για να εντοπίσουν τις υπηρεσίες Web, αυτά δεν θα συζητηθούν.

### **Ανακάλυψη στην OWL-S**

Το [PKPS02] προτείνει μια προσέγγιση βασισμένη στην DAML-S στο σημασιολογικό ταίριασμα μεταξύ της υπηρεσίας διαφήμισης και των αιτημάτων.

Στο [PKPS02], μόνο η πτυχή μετασχηματισμού πληροφοριών της υπηρεσίας εξετάζεται δηλ., μόνο οι εισαγωγές και τα αποτελέσματα λαμβάνονται υπόψη στη διαδικασία ανακαλύψεων. Ένα ταίριασμα από ένα σχεδιάγραμμα υπηρεσιών και του στόχου που έχει ζητηθεί (που διαμορφώνεται επίσης ως σχεδιάγραμμα) εμφανίζεται όταν όλα τα αποτελέσματα του στόχου αντιστοιχούνται από (ενδεχομένως ένα υποσύνολο) τα αποτελέσματα της ικανότητας, και όλες οι εισαγωγές της ικανότητας αντιστοιχούνται από (ενδεχομένως ένα υποσύνολο) τις εισαγωγές των στόχων. Λαμβάνοντας υπόψη ένα αίτημα, ο αλγόριθμος που τα ταιριάζει εκτελεί αυτόν τον έλεγχο για όλες τις διαθέσιμες υπηρεσίες. Για κάθε υπηρεσία, όλα τα αποτελέσματα του αιτήματος αντιστοιχούνται από τα αποτελέσματα της ικανότητας και, συμμετρικά, όλες οι εισαγωγές της ικανότητας αντιστοιχούνται ενάντια στις εισαγωγές του αιτήματος.

Η αντιστοιχία των εισαγωγών και των αποτελεσμάτων στηρίζεται στις υποκατηγορίες που αιτιολογούνται από την DAML+OIL. Οι διαφορετικοί βαθμοί αντιστοιχίας καθορίζονται για τον έλεγχο της εξάρτησης εισαγωγών και αποτελεσμάτων στη σχέση υποκατηγοριών που υπάρχει μεταξύ των ζευγαριών των αποτελεσμάτων (αντίστοιχα οι εισαγωγές), οι όποιοι οδηγούν στη συνέχεια σε μια ταξινόμηση των αντιστοιχισμένων υπηρεσιών. Οι προτιμημένες αντιστοιχίες είναι οι exact αντιστοιχίες, οι οποίες αντιστοιχούν στις περιπτώσεις όπου τα αποτελέσματα που (αντίστοιχα στις εισαγωγές) αντιστοιχούνται είναι ισοδύναμα ή η παραγωγή από το αίτημα είναι μια άμεση υποκατηγορία παραγωγής της υπηρεσίας. Οι δεύτερες καλύτερες περιπτώσεις είναι οι plug-in αντιστοιχίες, όπου η παραγωγή της υπηρεσίας εντάσσει την παραγωγή του αιτούντος. Η τρίτη περίπτωση, που θεωρείται χειρότερη από τις προηγούμενες, εντάσσει την αντιστοιχία, όπου η παραγωγή του αιτήματος εντάσσει την παραγωγή από την υπηρεσία. Η χειρότερη περίπτωση είναι όπου οι υποκατηγορίες δεν έχουν καμία σχέση με τα αποτελέσματα, που αντιστοιχούν στην αποτυχία. Το διαφορετικό αποτέλεσμα δίνεται στους διαφορετικούς τύπους αντιστοιχιών και, παίρνοντας τη χειρότερη

περίπτωση αντιστοιχίας, οι ταιριασμένες υπηρεσίες ταξινομούνται (βασισμένα στην παραγωγή τα αποτελέσματα αντιστοιχιών, τα αποτελέσματα αντιστοιχιών εισαγωγής εξετάζονται μόνο για να σημειώσουν τις υπηρεσίες).

Η προσέγγιση στο [PKPS02] παρουσιάζει μερικούς περιορισμούς και κάποια προβλήματα. Κατ' αρχάς, η παραγωγή κάθε αιτών πρέπει να ελέγχεται σε σχέση με όλα τα αποτελέσματα κάθε ενιαίας υπηρεσίας, έτσι ο αριθμός απαραίτητων δοκιμών υποκατηγοριών παραγωγής είναι  $n*m*s$ , όπου το  $n$  είναι ο αριθμός αποτελεσμάτων στο αίτημα, το  $m$  είναι ο αριθμός αποτελεσμάτων στην ικανότητα, και το  $s$  είναι ο αριθμός διαθέσιμων υπηρεσιών. Το ίδιο υποβάλλει αίτηση για τον έλεγχο εισαγωγής. Δεύτερον, μόνο εισαγωγές και αποτελέσματα εξετάζονται, κατά συνέπεια διαμορφώνεται μόνο οι πτυχές επικοινωνιακής ροής πληροφοριών δηλ. η υπηρεσία, αποκλείοντας τα αποτελέσματα στον πραγματικό κόσμο. Τρίτον, μερικοί περιορισμοί παράγονται από τη χρήση DAML+OIL. Η DAML+OIL δεν προσφέρει έναν φορμαλισμό για να εκφράσει τους κανόνες. Αν και, όπως περιγράφεται ανωτέρω η OWL-S εισάγει τη χρήση των DRS, KIF, ή SWRL για να εκφράσει τους κανόνες, η έκδοση του DAML-S χρησιμοποιούμενη μέσα στο [PKPS02] δεν τα υιοθετεί. Λόγω αυτής της έλλειψης κανόνων, τα αποτελέσματα της υπηρεσίας που αφορούν τις εισαγωγές του δεν καθορίζονται, και το ίδιο υποβάλλει αίτηση για τα αποτελέσματα και τις υποθέσεις. Επομένως η λειτουργία από την υπηρεσία δεν συλλαμβάνεται εντελώς. Από την εννοιολογική άποψη, αυτή η προσέγγιση δεν διαφοροποιεί την ανακάλυψη και την συνένωση, και δεν παρέχει τους πλήρως χρησιμοποιούμενους μηχανισμούς για οποιοδήποτε. Εάν θεωρηθεί ως ανακάλυψη υπηρεσιών, εισάγει την εκτίμηση των επίσημων συμφωνιών, η οποία δεν είναι η πρόθεσή μας. Εάν θεωρείται ως συνένωση υπηρεσιών, αυτό δεν διαμορφώνει τη σχέση μεταξύ των εισαγωγών και των αποτελεσμάτων και, επομένως, δεν μπορεί να οδηγήσει την συνένωση μιας συγκεκριμένης υπηρεσίας της οποίας η δυνατότητα χρησιμοποίησης είναι εγγυημένη.

Στο [LH03], επίσης οι υποκατηγορίες των DAML-S και DL χρησιμοποιούνται για την εκτέλεση της ανακάλυψης. Σε αυτήν την προσέγγιση, ολόκληρο το σχεδιάγραμμα (συμπεριλαμβανομένων των εισαγωγών και εξαγωγών των πληροφοριών, προϋποθέσεων και αποτελεσμάτων) ορίζεται ως ένα υποσενάριο του σχεδιαγράμματος των υπηρεσιών της DAML-S, και ταξινομείται έπειτα μέσα στην ιεραρχία υποκατηγοριών που



καθορίζεται από τα διαφημισμένα σχεδιαγράμματα υπηρεσιών. Επίσης σε αυτό το είδος οι εξεταζόμενες αντιστοιχίες είναι διαφορετικές από το [PKPS02]. Η exact match είναι περιορισμένη στο αντίτιμο των σεναρίων, η ίδια σχέση της υποκατηγορία εφαρμόζεται στις εισαγωγές και στα αποτελέσματα για τη plug-in και εντάσσει αντιστοιχίες, και προστίθεται η intersection match. Οι intersection matches αντιστοιχούν στις περιπτώσεις όπου η διατομή του αιτήματος και του σχεδιαγράμματος υπηρεσιών είναι ικανοποιητικές. Η αντιστοιχία εμφανίζεται όποτε ο αιτών και το σχεδιάγραμμα υπηρεσιών είναι συμβατά δηλ. υπάρχει μια intersection match, και όπως οι άλλοι τύποι αντιστοιχιών είναι πρόσθετες περιπτώσεις αυτή.

Κατ' αρχάς, η σχέση υποκατηγοριών μεταξύ του αιτούντος και του σχεδιαγράμματος υπηρεσιών υπολογίζονται χρησιμοποιώντας το RACER. Εάν καμία σχέση υποκατηγοριών δεν μπορεί να καθιερωθεί, έπειτα η ικανοποίηση από τη διατομή αυτών των εννοιών ελέγχεται. Για αυτό, το αρνούμενο αίτημα η αντιστοιχία είναι ταξινομημένη στην ιεραρχία των σχεδιαγραμμάτων των υπηρεσιών. Τα σχεδιαγράμματα που εντάσσει αλλά δεν είναι ίσα με το αρνούμενο αίτημα, θεωρούνται αντιστοιχίες διατομής. Τα σχεδιαγράμματα που εντάσσονται από το αρνούμενο αίτημα θεωρούνται αποτυχημένες αντιστοιχίες.

Το [LH03] προσφέρει μερικά πειραματικά αποτελέσματα με τα αυθαίρετα παραγόμενα σχεδιαγράμματα υπηρεσιών. Αποδεικνύεται ότι ο υπολογιστικά σκληρός στόχος είναι να ταξινομηθούν τα νέα σχεδιαγράμματα υπηρεσιών στην ιεραρχία σχεδιαγράμματος. Μόλις υπολογιστεί η ιεραρχία υποκατηγοριών του σχεδιαγράμματος, ταίριασμα του αιτήματος γίνεται σε λιγότερο από 20 χιλιοστά του δευτερολέπτου.

Εντούτοις, το [LH03] δεν λύνει μερικούς από τους περιορισμούς που αντιμετωπίζονται στο [PKPS02]: η έλλειψη κανόνων για την DAML+OIL οδηγεί σε μια ελλιπή περιγραφή της λειτουργίας υπηρεσιών και, επομένως, δεν μπορεί να χρησιμοποιηθεί για την συνένωση των υπηρεσιών. Αυτή η προσέγγιση μπορεί να είναι χρησιμοποιημένη για την ανακάλυψη υπηρεσιών εάν οι εισαγωγές και οι προϋποθέσεις αφαιρούνται, αλλά όχι για την συνένωση της υπηρεσίας.

Η εργασία στο [BHRT03] επίσης στηρίζεται στη χρήση της DAML-S ως οντολογία για την υπηρεσία περιγραφής. Ένα υποσύνολο της DAML+OIL για το οποίο η διαφορά χειρισμού που είναι σημασιολογικά μοναδική χρησιμοποιείται για να εκφράσει τις

εισαγωγές και τα αποτελέσματα του σχεδιαγράμματος υπηρεσιών (οι προϋποθέσεις και τα αποτελέσματα δεν εξετάζονται). Εντούτοις, η ανακάλυψη υπηρεσιών δεν διατυπώνεται ως πρόβλημα υποκατηγορίας συλλογισμού αλλά ως πρόβλημα που ξαναγράφεται δηλ. πώς να ξαναγράψει το αίτημα από την άποψη των διαθέσιμων υπηρεσιών. Λαμβάνοντας υπόψη ένα αίτημα υπηρεσιών και ένα σχεδιάγραμμα υπηρεσιών, ένας συνδυασμός υπηρεσιών Web που ικανοποιούν όσο το δυνατόν περισσότερο τα ζητούμενα αποτελέσματα και που απαιτούν μόνο τις πιθανές εισαγωγές που δεν παρέχονται στο αίτημα επιλέγονται. Τέτοιος συνδυασμός είναι αυτός που λέγεται καλύτερη κάλυψη σχεδιαγράμματος του αιτήματος που χρησιμοποιεί το σύνολο διαθέσιμων (διαφημισμένων) υπηρεσιών.

Χρησιμοποιώντας το χειριστή διαφοράς, οι υπηρεσίες των οποίων τα αποτελέσματα ικανοποιούν τουλάχιστον ένα αποτελέσματα στο αίτημα δηλ. η διαφορά μεταξύ των αποτελεσμάτων του αιτήματος και των αποτελεσμάτων από την υπηρεσία δεν είναι ολόκληρο το σύνολο αποτελεσμάτων στο αίτημα, προσδιορίζεται. Η ίδια λειτουργία (που χρησιμοποιεί το χειριστή διαφοράς στην αντίστροφη διαταγή) εκτελείται για τις εισαγωγές. Το σύνολο υπηρεσιών που έχουν το μικρότερο σύνολο αποτελεσμάτων στο αίτημα και στις εισαγωγές των μη ικανοποιημένων υπηρεσιών επιλέγεται. Κατά αυτόν τον τρόπο, ο καλύτερος συνδυασμός υπηρεσιών που παρέχει τον υψηλότερο αριθμό αποτελεσμάτων του αιτούντος και απαιτεί ότι οι λιγότερες εισαγωγές που δεν δίνονται στο αίτημα είναι επιλεγμένες. Το πρόβλημα τέτοιου συνόλου υπηρεσιών περιορίζεται στο πρόβλημα ελάχιστου υπολογισμού μετακίνησης με τις ελάχιστες δαπάνες υπερδιαγράμματος.

Αυτή η προσέγγιση έχει πλεονέκτημα τη διαδικασία ανακαλύψεων που χρησιμοποιεί έναν απλό τύπο σύνθεσης υπηρεσιών. Επιπλέον, στις ελλειπείς αντιστοιχίες μπορούν να βρεθούν και πληροφορίες για τα αποτελέσματα ή τις εισαγωγές λείπουν για μια πλήρη αντιστοιχία που δίνεται, εξυπηρετώντας την εξήγηση που ο αιτών μπορεί να χρησιμοποιήσει για να καθαρίσει το αίτημά του. Εντούτοις, το πρόβλημα έλλειψης κανόνων για την DAML+OIL και των σχετικών προβλημάτων για να συλλάβουν τη λειτουργία υπηρεσιών παραμένει. Επιπλέον, η εκφραστικότητα της γλώσσας μειώνεται στα υποσύνολα για τα οποία ο χειριστής διαφοράς είναι σημασιολογικά μοναδικός. Μερικά πειραματικά αποτελέσματα δίνονται για τα διαφορετικά μεγέθη του

προβλήματος και για τις διαφορετικές παραλλαγές του αλγορίθμου. Τα αποτελέσματα όχι ξεκάθαρα παρουσιάζουν την αποδοτικότητα του αλγορίθμου, και οι अपαράδεκτοι χρόνοι απόκρισης βρίσκονται για ορισμένους συνδυασμούς στοιχείων των πειραμάτων και των παραλλαγών του αλγορίθμου. Αυτή η προσέγγιση, όπως αυτή που παρουσιάστηκε πριν, μπορεί (εάν να προσαρμοστεί) να χρησιμοποιηθεί για την ανακάλυψη υπηρεσιών, αλλά όχι για την συνένωση υπηρεσίας.

Γενικά, όλες οι εργασίες που είναι βασισμένες στην OWL-S χάνουν τη σχέση μεταξύ της εισαγωγής και της παραγωγής και των αποτελεσμάτων, κατά συνέπεια δεν συλλαμβάνεται η λειτουργία των υπηρεσιών Web. Αν και η χρήση SWRL, DRS και KIF έχουν εισαχθεί στην πιο πρόσφατη απελευθέρωση της OWL, δεν υπάρχει καμία εργασία διαθέσιμη μέχρι τώρα στην εκμετάλλευση των όρων και της σχέσης μεταξύ των IOPEs για την ανακάλυψη. Σαν συνέπεια, αυτές οι προσεγγίσεις δεν είναι άμεσα κατάλληλες για την συνένωση υπηρεσιών, και πρέπει να προσαρμοστεί για την ανακάλυψη υπηρεσιών στο εννοιολογικό πρότυπό μας.

### **Ανακάλυψη στο METEOR-S**

Όπως παρουσιάστηκε πιο πάνω, το METEOR-S σχολιάζει σημασιολογικά τις διαδικασίες της WSDL, τις εισαγωγές και τις εξαγωγές, και προσθέτει τις προϋποθέσεις και τα αποτελέσματα στην περιγραφή διαδικασιών, χρησιμοποιώντας τα στοιχεία του WSDL. Η διαδικασία ταιριάσματος χρησιμοποιεί το συλλογισμό των υποκατηγοριών για να ταιριάζουν οι διαδικασίες, οι εισαγωγές, οι εξαγωγές, οι προϋποθέσεις και τα αποτελέσματα των σχολιασμένων υπηρεσιών ενάντια στο πρότυπο που περιγράφει το αίτημα.

Ο σχολιασμός των ληξιαρχείων και της ειδικεύσής του στις περιοχές βοηθά να εξετάσει τον ενδεχομένως τεράστιο αριθμό δημοσιευμένων υπηρεσιών. Εντούτοις, η προσέγγιση του METEOR-S στην υπηρεσία περιγραφής και ανακάλυψης παρουσιάζει μερικούς περιορισμούς. Ομοίως με τις προηγούμενες απελευθερώσεις από την OWL, ο σχολιασμός του METEOR-S δεν αφορά τις εισαγωγές, τις εξαγωγές, τις προϋποθέσεις και τα αποτελέσματα, επομένως παρέχεται μία μη ακριβή περιγραφή της λειτουργίας υπηρεσιών. Επιπλέον, η περιγραφή METEOR-S κεντροθετείται από την WSDL, ενώ

προσεγγίσεις όπως οι OWL και WSMO υιοθετούν μια πιο εύκαμπτη μέθοδο, μη επιβάλλοντας οποιαδήποτε γλώσσα για να στηρίξει τις σημασιολογικές υπηρεσίες Web.

Πρέπει να παρατηρηθεί ότι η ανακάλυψη στις χρήσεις METEOR-S ζητά πρότυπα παρόμοια με τους προκαθορισμένους στόχους, οι οποίοι πρόκειται παραμετροποιηθούν από τον αιτούντα. Εντούτοις, και όπως με τις βασισμένες στην OWL προσεγγίσεις, οι Συνενώσεις υπηρεσιών δεν εξετάζονται.

### **Ανακάλυψη σε LARKS**

Το [SWKL02] καθορίζει την λεγόμενη γλώσσα για τη διαφήμιση και το αίτημα για τη μοιρασμένη γνώση (LARKS), που χρησιμοποιείται για να περιγράψει τις ικανότητες των διαμεσολαβητών. Μια προδιαγραφή ικανότητας σε LARKS καθορίζει το πλαίσιο της προδιαγραφής, των μεταβλητών εισαγωγής και παραγωγής, τους περιορισμούς σε αυτές τις μεταβλητές, τις οντολογικές χρησιμοποιούμενες περιγραφές, και μια κειμενική περιγραφή. Όπως στις OWL, μια προδιαγραφή ικανότητας μπορεί να αντιμετωπιστεί ως αίτημα ή ως διαφήμιση. Λαμβάνοντας υπόψη ένα αίτημα και μια διαφήμιση, η διαδικασία αντιστοιχίας μπορεί να εφαρμόσει πέντε διαφορετικά φίλτρα, δηλαδή [SWKL02]: 1) ταίριασμα πλαισίου, 2) σύγκριση σχεδιαγράμματος, 3) ομοιότητα αντιστοιχιών, 4) ταίριασμα υπογραφών, και 5) ταίριασμα περιορισμού. Ο συνδυασμός αυτών των φίλτρων μπορεί πραγματικά να εφαρμοστεί και να επιλεγεί από τον αιτούντα.

Κατά αυτόν τον τρόπο, η ανταλλαγή μεταξύ της ακρίβειας και της αποδοτικότητας της ανακάλυψης μπορεί να επιλεγεί. Το πλαίσιο σε LARKS ορίζεται ως ένα σύνολο λέξεων κλειδιών που περιγράφουν την περιοχή του διαμεσολαβητή. Το ταίριασμα πλαισίου υπολογίζει την απόσταση [Ros94] μεταξύ των λέξεων κλειδιών του αιτήματος και της διαφήμιση, και η σχέση υποκατηγοριών μεταξύ των εννοιών αντιστοιχείται στα ζευγάρια των περισσότερων παρόμοιων λέξεων. Η υπολογισμένη ομοιότητα, που χρησιμοποιείται στο κατώτατο όριο, θα καθορίσει εάν υπάρξει μια αντιστοιχία.

Η σύγκριση σχεδιαγράμματος μεταχειρίζεται το αίτημα και τις διαφημίσεις ως έγγραφα και καθορίζει το βαθμός ομοιότητας μεταξύ των βασισμένων στη συχνότητα και της σχετικότητας των λέξεων στο έγγραφο. Εάν η ομοιότητα υπερβαίνει ένα δεδομένο κατώτατο όριο, υπάρχει μια αντιστοιχία.

Η σύγκριση σχεδιαγράμματος δεν εξετάζει τη δομή της προδιαγραφής, αλλά την ομοιότητα του ταιριάσματος. Τέτοια ομοιότητα υπολογίζεται συνδυάζοντας τις τιμές απόστασης για τα ζευγάρια των δηλώσεων εισαγωγής και παραγωγής, συμπεριλαμβανομένων των περιορισμών τους. Πάλι, εάν υπολογιστεί ότι η ομοιότητα υπερβαίνει ένα κατώτατο όριο, το αποτέλεσμα θα είναι μια αντιστοιχία.

Το ταιρίασμα υπογραφών ελέγχεται εάν οι δηλώσεις εισαγωγής και παραγωγής του αιτήματος  $R$  και η αντιστοιχία διαφημίσεων  $A$  ταιριάζουν. Αυτό γίνεται χρησιμοποιώντας τους κανόνες και τις υποκατηγορίες συμπεράσματος συλλογισμού.

Το ταιρίασμα περιορισμού χρησιμοποιεί την έννοια της *plug-in match*. Οι λογικές επιπτώσεις είναι ελεγμένες χρησιμοποιώντας τις υποκατηγορίες που διαλογίζεται για τις προτάσεις HORN. Το ταιρίασμα περιορισμού χρησιμοποιεί το φίλτρο υπογραφών και, επομένως, αυτά τα δύο φίλτρα λειτουργούν από κοινού.

Η εφαρμογή των διαφορετικών φίλτρων για την ανακάλυψη σε LARKS έχει το πλεονέκτημα η ανταλλαγή μεταξύ της ακρίβειας και της αποδοτικότητας με την απόφαση των φίλτρων θα είναι εφαρμοσμένα. Επιπλέον, οι περιορισμοί εισαγωγής και παραγωγής μπορούν να περιλάβουν τη σχέση μεταξύ εισαγωγής και παραγωγής, που συλλαμβάνει ακριβέστερα τη λειτουργία του διαμεσολαβητή. Εντούτοις, οι αλλαγές στο διάστημα πληροφοριών και στην ξεκάθαρη κατάσταση δεν διαφοροποιούνται, δεδομένου ότι τα αποτελέσματα της δράσης ενός διαμεσολαβητή κωδικοποιούν σιωπηρά τους περιορισμούς παραγωγής. Η προσθήκη ανακάλυψης σε LARKS δεν διαφοροποιείται μεταξύ της ανακάλυψης υπηρεσιών και των επίσημων συμφωνιών.

### **Άλλες προσεγγίσεις**

Στο [GCTB01], οι υποκατηγορίες της DAML+OIL που διαλογίζεται είναι ο κεντρικός μηχανισμός συλλογισμού για την ανακάλυψη. Περιγράφονται υπηρεσίες Web ως μπουουλιανός συνδυασμός ενός συνόλου περιορισμών πέρα από τους τύπους δεδομένων και τις αφηρημένες ιδιότητες υπηρεσιών. Οι διαφημισμένες υπηρεσίες είναι ταξινομημένες σε μια ιεραρχία υποκατηγορίας, και το διαφορετικό είδος αντιστοιχιών προσδιορίζονται για ένα δεδομένο αίτημα  $S$  και μια ιεραρχία περιγραφής υπηρεσιών, δηλαδή: ισοδύναμες έννοιες στο  $S$ , υπό-έννοιες  $S$ , υπέρ-έννοιες του  $S$  που εντάσσονται από την έννοια περιγραφής γενικών υπηρεσιών, και από υπό-έννοιες οποιασδήποτε

άμεσης υπέρ-έννοιας του  $S$  του οποίου η διατομή με το  $S$  είναι ικανοποιητική. Όπως για τις προηγούμενες προσεγγίσεις βασισμένες στη DAML+OIL, η περιγραφή των λειτουργιών των υπηρεσιών δεν είναι πλήρης, και επομένως η συνένωση δεν μπορεί να επιτευχθεί. Εντούτοις, ως προσέγγιση στο [LH03], είναι ενδιαφέρουσα για την ανακάλυψη των υπηρεσιών, όπως στηρίζεται στην ταξινόμηση του Web οι υπηρεσίες σε ένα T-κιβώτιο, και δεν απαιτούν τον έλεγχο κάθε υπηρεσίας χωριστά ενάντια στο δεδομένο στόχο.

Στο [ZK04], μια βασισμένη στην F-λογική προσέγγιση για την περιγραφή των υπηρεσιών Web παρουσιάζεται. Η έννοια των υπηρεσιών περιγράφει τις μη λειτουργικές ιδιότητες υπηρεσιών από την άποψη του προμηθευτής, της θέση, του τύπος υπηρεσιών, κ.λ.π., της συμπεριφορά του (εισαγωγές πληροφοριών, αποτελέσματα, και η σχέση μεταξύ τους), και τις διαδικασίες του (συμπεριλαμβανομένων της εισαγωγής πληροφοριών και των αποτελεσμάτων για κάθε λειτουργία). Ένας αιτών εκφράζει το στόχο του ως ερώτηση από την άποψη της υπηρεσίας μη λειτουργικών ιδιοτήτων, συμπεριφοράς, και διαδικασιών. Εντούτοις, η περιγραφή υπηρεσιών δεν μπορεί να διαμορφώσει τον κόσμο αλλάζοντας τις εναλλακτικές υπηρεσίες δηλ. με τα αποτελέσματα στον κόσμο. Επιπλέον, καθώς χρησιμοποιείται η απλή ερώτηση που απαντάται, η περιγραφή των υπηρεσιών περιορίζεται στα επίγεια γεγονότα, τα οποία μειώνουν αρκετά την εκφραστικότητα που επιτρέπεται για την περιγραφή της λειτουργίας υπηρεσιών και δεν επιτρέπει την περιγραφή των αφηρημένων ικανοτήτων.

### **4.3 Περιγραφή των υπηρεσιών και των στόχων του Web**

Όπως παρουσιάζεται πιο πάνω τα τμήματα λογισμικού και οι ερωτήσεις περιγράφονταν χρησιμοποιώντας αξιώματα Hoare [Hoa69]. Η προοριζόμενη λειτουργία ενός προγράμματος ( $C$ ) διευκρινίζεται από τις αρχικές προϋποθέσεις ( $C_{pre}$ ) και τις μετέπειτα καταστάσεις ( $C_{post}$ ). Μια ερώτηση  $Q$  είναι όμοια διευκρινισμένη ως ( $Q_{pre}, Q_{post}$ ).

Σημασιολογικές περιγραφές των υπηρεσιών του Web που προτείνονται από τις OWL-S, WSMO, METEOR-S και IRS-III ακολουθούν μια παρόμοια προσέγγιση, αλλά διαφοροποιούνται επίσης μεταξύ των πραγματικών αποτελεσμάτων και των μετέπειτα καταστάσεων των πληροφοριών, και μεταξύ των πραγματικών υποθέσεων και των πληροφοριών προϋποθέσεις.

Παρακάτω, θα υποθέσουμε ότι οι υπηρεσίες του Web περιέγραψαν τις προϋποθέσεις, τις υποθέσεις, τις μετέπειτα καταστάσεις και τα αποτελέσματα, που χρησιμοποιούν την ορολογία WSMO [RLeditors04]. Θα το δείξουμε αυτό με  $W_{pre}$ ,  $W_{ass}$ ,  $W_{post}$ , και  $W_{eff}$ , αντίστοιχα. Το  $W_{pre}$  μπορεί να θεωρηθεί δεδομένου της OWL-S και της METEOR-S ως εισαγωγή, το  $W_{ass}$  ως προϋποθέσεις, το  $W_{post}$  ως εξαγωγή, και το  $W_{eff}$  ως αποτελέσματα. Από αυτά, μόνο οι μετέπειτα καταστάσεις και τα αποτελέσματα θα είναι μέρος από τις αφηρημένες ικανότητες των υπηρεσιών. Παρατηρήστε ότι το WSMO δεν επιτρέπει τις πολλαπλάσιες ικανότητες και δεν διαφοροποιεί τις αφηρημένες και τις ικανότητες επίσημων συμφωνιών. Εντούτοις, και όπως έχει αναφερθεί πιο πάνω, η αφηρημένη ικανότητας εξάγεται αυτόματα από την ικανότητα της συνένωσης, που είναι ακριβώς μια αφαίρεση από τα τελευταία και πρέπει να είναι συνεπείς.

Όσον αφορά τους στόχους, θα ακολουθήσουμε επίσης την ορολογία WSMO και θα δείξουμε τις μετέπειτα καταστάσεις και τα αποτελέσματα από τα  $G_{post}$  και  $G_{eff}$ , αντίστοιχα, που ερμηνεύονται ως κατάσταση των πληροφοριών και της κατάσταση του κόσμου που επιδιώκεται μετά από την εκτέλεση υπηρεσιών. Αυτά μπορούν να θεωρούνται σαν εξαγωγές και αποτελέσματα του αιτήματος των OWL-S και METEOR-S. Εντούτοις, είναι σημαντική η εννοιολογική διαφοροποίηση της άποψης του αιτούντος (στόχοι) και της άποψης των υπηρεσιών (ικανότητες), η οποία λείπει από την OWL-S. Όσον αφορά τη WSMO, εισάγουμε ένα νέο στοιχείο  $G_{input}$ , το οποίο περιγράφει τις πληροφορίες εισαγωγής που μπορεί να παρέχεται από τον αιτούντα και είναι (άμεσα ή έμμεσα) μέρος του στόχου.

Φαίνεται, ακόμα κι αν θα υιοθετήσουμε μια ορολογία WSMO, οι προσεγγίσεις που συζητηθούν στα επόμενα τμήματα είναι ουδέτερες όσον αφορά την πρόταση για τις Υπηρεσίες του Web. Η βασική προϋπόθεση είναι ότι οι περιγραφές συλλαμβάνουν τις πτυχές της λειτουργίας των υπηρεσιών του Web και τους σχετικούς προηγούμενους στόχους. Εντούτοις, διαφορετικές προσεγγίσεις στην υπηρεσία ανακάλυψης του Web θα απαιτήσει τη διαφορετική εκφραστικότητα για την περιγραφή τέτοιων πτυχών, στις οποίες η WSMO είναι καταλληλότερη από άλλες προσεγγίσεις ειδικά όταν η σχέση μεταξύ των εισαγωγών και των μετέπειτα καταστάσεων και των αποτελεσμάτων πρέπει να περιγραφεί.

Όπως φαίνεται στο [LPL+], οι διαφορές μεταξύ του χαρακτηρισμού των υπηρεσιών στις OWL-S και WSMO δεν είναι πολλές, οι πιο σχετικές διαφορές στη γλώσσα που χρησιμοποιούνται για να εκφράσουν το χωρισμό των στόχων και των υπηρεσιών σε WSMO. Μια επίσημη η χαρτογράφηση μεταξύ των OWL-S και WSMO παρέχεται μέσα σε [Leditors04a], το οποίο μπορεί να χρησιμοποιείται στο μέλλον για να μεταφράσει τις περιγραφές της OWL-S στις περιγραφές WSMO. Εάν παρουσιαστεί ότι η χαρτογράφηση δεν υπονοεί μια απώλεια σχετικών πληροφοριών, επιβεβαιώνεται ότι η πρόταση των ανακαλύψεων ισχύει επίσης στις υπηρεσίες OWL-S.

Παρακάτω, θα πάμε στις λεπτομέρειες της διαφορετικής υπηρεσίας προσεγγίσεων ανακαλύψεων του Web, αρχίζοντας από την λέξη που βασίζετε στην ανακάλυψη και τελειώνοντας με την ανακάλυψη που βασίζετε στις ακριβείς περιγραφές της λειτουργίας των υπηρεσιών του Web.

#### **4.4 Ανακάλυψη των αυτόματων υπηρεσιών του Web**

Σε αυτό το τμήμα, συζητάμε εν συντομία τη χρήση της λέξης που βασίζεται στην ανακάλυψη του στόχου, και παρουσιάζουμε μια συνολική θεωρητική ανακάλυψη των υπηρεσιών προσέγγισης του Web. Η διαμόρφωση αυτής της προσέγγισης πρώτα (χωρίς περιορισμούς) χρησιμοποιεί τη first-order λογική και περιορίζει αργότερα τη γλωσσική εκφραστικότητα. Οι ενδιαφέρουσες γλώσσες DL θα συζητηθούν.

Όπως θα συζητηθεί πιο κάτω οι ενιαίες προσεγγίσεις μπορούν να απαιτήσουν τις διαφορετικές τεχνικές για τη μεσολάβηση. Η σημαντικότερη τεχνική για μας είναι βεβαίως η συγχώνευση της οντολογίας και η ευθυγράμμιση. Εντούτοις, η ίδια η μεσολάβηση είναι εξωτερική όψη του παραδοτέου αυτού πεδίο. Επομένως, θα υποθέσουμε πιο κάτω ότι η μεσολάβηση είναι διαθέσιμη και ότι έχει ήδη διενεργηθεί.

##### **4.4.1 Ανακάλυψη βασισμένη στη λέξη**

Η λέξη που βασίζεται στην ανακάλυψη είναι ένα βασικό συστατικό σε ένα πλήρες πλαίσιο για την ανακάλυψη υπηρεσιών στο σημασιολογικό Ιστό. Με την εκτέλεση μιας λέξης βασισμένη στην αναζήτηση, το τεράστιο ποσό διαθέσιμων οι υπηρεσιών μπορεί να φιλτραριστεί ή να ταξινομηθεί αποτελεσματικά. Η εστίαση της ανακάλυψης των υπηρεσιών δεν είναι ανοικτή λέξη που βασίζετε στην ανακάλυψη αλλά θεωρούμε αυτό



το είδος ανακάλυψης μια χρήσιμη τεχνική στο πλήρες σημασιολογικό πλαίσιο ανακαλύψεων των υπηρεσιών Web.

Σε ένα χαρακτηριστικό σενάριο βασισμένο στη λέξη, μια λέξη βασισμένη στην ερώτηση της μηχανή χρησιμοποιείται για την ανακάλυψη υπηρεσιών. Μια ερώτηση, που είναι βασική σε ένα σύνολο λέξεων κλειδιών, παρέχεται ως εισαγωγή στην ερώτηση της μηχανής. Η ερώτηση της μηχανής ταιριάζει με τις λέξεις κλειδιά από την ερώτηση ενάντια στις λέξεις κλειδιά που χρησιμοποιούνται για να περιγράψουν την υπηρεσία. Μια ερώτηση με την ίδια έννοια μπορεί να διατυπωθεί με τη χρησιμοποίηση ενός συνώνυμου λεξικού, όπως το WordNet [Fe198]. Η σημασιολογία της ερώτησης παραμένει η ίδια αλλά λόγω των διαφορετικών λέξεων κλειδιών που χρησιμοποιήθηκαν, ως συνώνυμα προηγουμένως, οι περισσότερες υπηρεσίες πιθανός να εκπληρώνουν το αίτημα των χρηστών. Επιπλέον, με τη χρησιμοποίηση των λεξικών όπως το WordNet, όπως και των τεχνικών επεξεργασίας φυσικής γλώσσας, μια αύξηση της σημασιολογικής σχετικότητας από τα αποτελέσματα αναζήτησης (wrt. το αίτημα στην αναζήτηση) μπορεί να επιτευχθεί [RS95] εν τούτοις, τέτοιες τεχνικές περιορίζονται εγγενώς από την ασάφεια της φυσικής γλώσσας και την έλλειψη της σημασιολογικής κατανόηση των περιγραφών της φυσικής γλώσσας από τα αλγοριθμικά συστήματα.

Οι περιγραφές υπηρεσιών είναι παρεχόμενες από τη μεριά των διαφημίσεων μαζί με τις λέξεις κλειδιά για την κατηγοριοποίηση. Αυτό σημαίνει ότι καθεμία λέξη κλειδί πρέπει να εξαχθεί από τις αφηρημένες ικανότητες και οι περιγραφές του στόχου που αναφέρονται στο προηγούμενο τμήμα, ή τέτοιες περιγραφές πρέπει να επεκταθούν για να περιλάβουν τις σχετικές λέξεις κλειδιά.

Ενώ η λέξη που βασίζεται η ανακάλυψη είναι μια ενδιαφέρουσα τεχνική που πρέπει να κρατηθεί στη διαταγή για να εξετάσει τις ήδη υπάρχουσες βασισμένες στη WSDL υπηρεσίες, μπορεί να χρησιμοποιηθεί καλύτερα για την ανακάλυψη του στόχο. Δεδομένου ότι δεν θέλουμε να θέσουμε τις μη ρεαλιστικές απαιτήσεις στη γνώση οι δεξιότητες αντιπροσώπευσης του αιτούντος, ο πλέον πιθανός τρόπος ένας αιτών να εκφράσει την επιθυμία του είναι με την παροχή μιας κειμενικής περιγραφής του. Η λέξη που βασίζεται στην αντιστοιχία μπορεί να χρησιμοποιηθεί για να εντοπίσει σχετικούς προκαθορισμένοι στόχους που παρέχουν επίσης μια διαμόρφωση της επιθυμίας δηλ. ο στόχος του αιτούντος. Μόλις βρεθεί τέτοιος προκαθορισμένος στόχος, τα κατάλληλα

αυτόματα εργαλεία καθαρίζουν ή παραμετροποιούν τον προκαθορισμένο στόχο για να απεικονιστεί πραγματικά η επιθυμία του, ή μόνος του με το χέρι θα καθορίσει τέτοιο στόχο.

Οι προκαθορισμένοι στόχοι μπορούν φυσικά να κοιταχτούν βιαστικά, αλλά εάν ο αριθμός εκείνων είναι μεγάλος, η λέξη που βασίζεται στην αντιστοιχία είναι μια χρήσιμη τεχνική με σκοπό την εντόπιση αυτών που είναι σχετικά για τον αιτούντα.

Παρακάτω, θα υποθέσουμε ότι ένας στόχος αιτούντος έχει παραχθεί ήδη από έναν προκαθορισμένο στόχο, που έχει μια διαμόρφωση του στόχου του αιτούντος που περιλαμβάνει πληροφορίες εισαγωγής που μπορεί να επιτρέψει την παροχή υπηρεσιών.

#### **4.4.2 Σημασιολογικός χαρακτηρισμός των αποτελεσμάτων**

Αν και η λέξη που βασίζεται η αναζήτηση είναι μια ευρέως χρησιμοποιημένη τεχνική για την ανάκτηση πληροφοριών, αυτό δεν χρησιμοποιεί τη ρητή, καθορισμένη με σαφήνεια σημασιολογία. Οι λέξεις κλειδιά που χρησιμοποιούνται για να ανακτήσουν πληροφορίες δεν έχουν μια ρητή διαμόρφωση και, επομένως, δεν επιτρέπουν να βελτιωθούν τα αποτελέσματα αναζήτησης.

Για αυτούς τους λόγους, ως δεύτερη προσέγγιση εξετάζουμε τη χρήση των ελεγχόμενων λεξιλογίων με τη ρητή, επίσημη σημασιολογία. Οι οντολογίες είναι καλοταίριασμένες και προεξέχουσες εννοιολογικά για αυτόν το λόγο. Παρέχουν μια ρητή και κοινή ορολογία, εξηγούν τις αλληλεξαρτήσεις μεταξύ των ενιαίων εννοιών και, επομένως, είναι καλοταίριασμένες για την περιγραφή των αφηρημένων ικανοτήτων και των στόχων του αιτούντος. Επιπλέον, οι οντολογίες μπορούν να τυποποιηθούν στις λογικές οι οποίες επιτρέπουν τη χρήση των υπηρεσιών συμπεράσματος για την εκμετάλλευση της γνώσης για το πρόβλημα περιοχής κατά τη διάρκεια της διαδικασίας ανακαλύψεων.

Σε αυτό το τμήμα παρουσιάζουμε μια προσέγγιση επίσημης διαμόρφωσης για τις υπηρεσίες και τους στόχους του Web ο οποίος είναι βασισμένος στην καθορισμένη θεωρία και εκμεταλλεύεται τις οντολογίες ως επίσημες, την αντιπροσώπευση που είναι επεξεργάσιμη από τη μηχανή των περιοχών γνώσης. Συζητάμε την ανακάλυψη υπηρεσιών βασισμένη σε αυτήν την προσέγγιση για τις απλές σημασιολογικές

περιγραφές και πώς να εφαρμοστεί τέτοιο σύνολο βασισμένο στο πρότυπο μέσα στο επίσημο πλαίσιο της λογικής.

Το πλαίσιο που παρουσιάζουμε σε αυτό το τμήμα της ανακάλυψης υπηρεσιών Web βασίζεται σε απλούς Semantic σχολιασμούς που είναι στην πραγματικότητα παρόμοιο με ένα πρότυπο που έχει προταθεί πρόσφατα [GMP04] για την ανακάλυψη σε μια ρύθμιση ηλεκτρονικού εμπορίου. Κατά κάποιον τρόπο η προσέγγισή μας είναι γενίκευση του προτύπου που συζητείται εκεί.

**Ένα παράδειγμα.** Θα επεξηγήσουμε την προσέγγισή μας με τα απλά παραδείγματα στο πλαίσιο VTA: διαμορφώνουμε μερικούς στόχους του αιτούντος και υπηρεσίες Web σχετικές με τις πληροφορίες για τις πτήσεις και τις συνδέσεις των τρενών. Χάριν της απλότητας, θα εξετάσουμε στα ακόλουθα μόνο τις μετέπειτα καταστάσεις (αποτελέσματα) μιας υπηρεσίας και δεν μεταχειριζόμαστε τα αποτελέσματα ρητά. Αν και είναι εννοιολογικά διαφορετικά και η δύναμη του αιτούντος θέλει να εφαρμόσει τις διαφορετικές έννοιες της αντιστοιχίας τους, η συζήτησή μας μπορεί να εφαρμοστεί εξίσου και στα δύο.

Η άτυπη περιγραφή των στόχων αιτούντος είναι η ακόλουθη:

G1: Θέλω να ξέρω όλες τις πτήσεις από οποιοδήποτε μέρος στην Ιρλανδία σε οποιοδήποτε μέρος στην Αυστρία.

G2: Θέλω να ξέρω μερικές πτήσεις από οποιοδήποτε μέρος στην Ιρλανδία σε οποιοδήποτε μέρος στην Ιρλανδία.

G3: Θέλω να ξέρω όλες τις πτήσεις από το Galway (Ιρλανδία) στο Δουβλίνο (Ιρλανδία).

G4: Θέλω να ξέρω μερικές πτήσεις από το Ίνσμπρουκ (Αυστρία) σε κάποιο μέρος στην Ιρλανδία (ο πελάτης δεν νοιάζεται απαραίτητως για κάποιο).

G5: Θέλω να ξέρω μερικές συνδέσεις τρενών από το Galway (Ιρλανδία) στο Δουβλίνο (Ιρλανδία).

Υποθέστε περισσότερες από τέσσερις διαθέσιμες υπηρεσίες Web W1,....., W4 εκθέτοντας τις ακόλουθες (άτυπες) αφηρημένες ικανότητες:

W1 :πληροφορίες προσφορών για όλες τις πτήσεις από οποιοδήποτε μέρος στην Ευρώπη σε οποιοδήποτε μέρος στην Ευρώπη.

W2: παρέχει πληροφορίες για τις πτήσεις από οποιοδήποτε μέρος της Ιρλανδίας σε οποιοδήποτε μέρος της Ιρλανδίας αλλά όχι απαραίτητα πληροφορίες για όλες αυτές τις πτήσεις.

W3: Παρέχει πληροφορίες για τις πτήσεις από οποιοδήποτε μέρος της Ιρλανδίας σε οποιοδήποτε μέρος του Ίνσμπρουκ (Αυστρία)

W4: παρέχει πληροφορίες μερικές συνδέσεις τραίνων από το Galway (Ιρλανδία) στο Δουβλίνο (Ιρλανδία).

### **Μια προσέγγιση βασισμένη στο σύνολο διαμόρφωσης**

Οι βασισμένες στο πλαίσιο ή οι αντικειμενοστραφείς τεχνικές διαμόρφωσης έχουν γίνει δημοφιλείς προσεγγίσεις στη διαμόρφωση συστημάτων και περιοχών και στη βιομηχανία και στον ακαδημαϊκό κόσμο. Ένα κύριο χαρακτηριστικό από αυτές τις προσεγγίσεις είναι ότι η περιοχή του προβλήματος (ή ο κόσμος) γίνεται κατανοητή ως σύνολο από τα αντικείμενα, και τα ενιαία αντικείμενα μπορεί να συγκεντρωθούν στα σύνολα (ή τις κατηγορίες). Κάθε κατηγορία συλλαμβάνει τα κοινά (συντακτικά και σημασιολογικά) χαρακτηριστικά γνωρίσματα των στοιχείων τους. Τα χαρακτηριστικά γνωρίσματα μπορούν να είναι κληρονομημένα μεταξύ των κατηγοριών με τον καθορισμό των ιεραρχιών κατηγορίας. Αυτός ο τρόπος, σε μια περιοχή προβλήματος μπορεί να κτίζεται ως κατηγορίες αντικειμένων και είναι βασικά κατανοητός ως συλλογή των κατηγοριών (ή του συνόλου πραγμάτων). Συγκεκριμένα, οι οντολογίες είναι μια δημοφιλής τεχνική αντιπροσώπευσης γνώσης που εκμεταλλεύονται συνήθως το ίδιο παράδειγμα διαμόρφωσης.

Σε μια τέτοια διαμόρφωση πλησιάζονται οι κύριες σημασιολογικές ιδιότητες όπου κάποια ενδιαφέρεται για ορισμένες σχέσεις μεταξύ τέτοιων συνόλων αντικειμένων του κόσμου. Καθιερώνετε και ο έλεγχος τέτοιων σχέσεων που είναι ο κύριος στόχος του συλλογισμού που επιτρέπει στους διαμεσολαβητές να εκμεταλλευτούν τη γνώση που τυποποιείται στο πρότυπο περιοχών (ή στην οντολογία). Από μια αντιπροσώπευση γνώσης η προοπτική, είναι μια πολύ φυσική και απλή προσέγγιση διαμόρφωσης για τα ανθρώπινα όντα.

**Διαμορφώνοντας υπηρεσίες και τους στόχους του Web.** Η παροχή υπηρεσίας οδηγεί μέσα (wrt. ένα σύνολο από τις τιμές εισαγωγής που παρέχονται από τον πελάτη) σε

ορισμένες πληροφορίες που δίνονται στον αιτούντα (παραγωγή) και στο επίτευγμα ορισμένων αλλαγών στην κατάσταση του κόσμου. Ως εκ τούτου, μια περίληψη υπηρεσίας μπορεί να περιγραφεί από την άποψη των αποτελεσμάτων που μπορούν ενδεχομένως να παραδοθούν από την υπηρεσία.

Οι στόχοι περιγράφουν τις πληροφορίες που ο πελάτης θέλει να λάβει ως συνέπεια μιας παροχής υπηρεσίας (αποτελέσματα), καθώς επίσης και των αποτελεσμάτων στην κατάσταση του κόσμου που ο πελάτης σκοπεύει για να επιτύχει. Αυτή η επιθυμία μπορεί να αντιπροσωπευθεί όπως το σύνολο των στοιχείων που είναι σχετικά με τον πελάτη ως αποτελέσματα και αποτελέσματα μιας εκτέλεσης υπηρεσιών.

Σύμφωνα με αυτήν την άποψη, οι υπηρεσίες και οι στόχοι του Web αντιπροσωπεύονται ως το σύνολο των αντικειμένων. Οι περιγραφές αυτών των συνόλων αναφέρονται στις οντολογίες που συλλαμβάνουν τη γενική γνώση για περιοχές του προβλήματος που εξετάζονται.

Μια σημαντική παρατήρηση στην προσέγγισή μας είναι ότι η περιγραφή ενός συνόλου αντικειμένων για την αντιπροσώπευση ενός στόχου ή μιας υπηρεσίας του Web που μπορεί πραγματικά να ερμηνευθεί με διαφορετικούς τρόπους και, κατά συνέπεια, η περιγραφή με τη βοήθεια ενός συνόλου δεν είναι σημασιολογικά μοναδική: Ένα μοντέλο θέλει για να εκφράσει ότι καθένα από όλα τα στοιχεία που περιλαμβάνονται στο ζητούμενο σύνολο ( η περίπτωση μιας περιγραφής στόχου) ή μπορεί να παραδοθούν (στη περίπτωση περιγραφής υπηρεσιών Web), ή μόνο μερικά από αυτά τα στοιχεία ζητούνται (ή μπορεί να παραδοθούν). Ένα μοντέλο έχει κάποια συγκεκριμένη διαίσθηση στο μυαλό κατά τη διευκρίνιση τέτοιου συνόλου σχετικών αντικειμένων για έναν στόχο ή της περιγραφής υπηρεσιών του Web και αυτή η πρόθεση ουσιαστικά καθορίζεται εάν εξετάζουμε δύο περιγραφές που ταιριάζουν ή όχι. Κατά συνέπεια, αυτή η διαίσθηση πρέπει να δηλώσει ρητά τις περιγραφές των ερωτώμενων στόχων και των αφηρημένων υπηρεσιών.

Εάν παίρνουμε για παράδειγμα το στόχο G4 και την υπηρεσία W1, τα σχετικά αντικείμενα που περιγράφονται είναι:

Στόχος/Υπηρεσία Web	Σύνολο R σχετικών αντικειμένων	Πρόθεση του R
G <sub>4</sub>	{f   f είναι μια πτήση που αρχίζει στο Ίνσμπρουκ στην Αυστρία και τελειώνει οποιαδήποτε πόλη C που βρίσκεται στην Ιρλανδία}	Ύπαρξη (∃)
W <sub>1</sub>	{f   f είναι μια πτήση που αρχίζει στην πόλη s και τελειώνει στην πόλη e, s είναι οποιαδήποτε πόλη στην Ευρώπη, e είναι οποιαδήποτε πόλη στην Ευρώπη}	Καθολικό

**Σημασιολογικό ταίριασμα.** Η σημασιολογία ενός στόχου ή μιας αφηρημένης περιγραφής D υπηρεσιών αντιπροσωπεύεται από ένα σύνολο αντικειμένων  $R_D \subseteq U$  (στη συνηθισμένη κατάσταση U) που αντιπροσωπεύει το σύνολο των σχετικών αντικειμένων για την περιγραφή καθώς επίσης και μια ρητή προδιαγραφή για το αντίστοιχο  $I_D \in \{\forall, \exists\}$  του συνόλου.

Προκειμένου να θεωρηθεί ένας στόχος C και μια υπηρεσία Web για να ταιριάζει σε ένα σημασιολογικό επίπεδο, τα σύνολα  $R_G$  και  $R_W$  που περιγράφουν αυτά τα στοιχεία πρέπει να συσχετιστούν ακριβώς προφορικά, και αναμένουμε ότι κάποια θεωρητική συνολική σχέση μεταξύ  $R_G$  και του  $R_W$  πρέπει να υπάρξει. Οι πιο βασικές θεωρητικές συνολικές σχέσεις που εξετάζονται είναι οι ακόλουθες:

- Συνολική ισότητα:  $R_G = R_W$
- Υποσύνολο περιγραφής του στόχου της περιγραφής των υπηρεσιών Web:

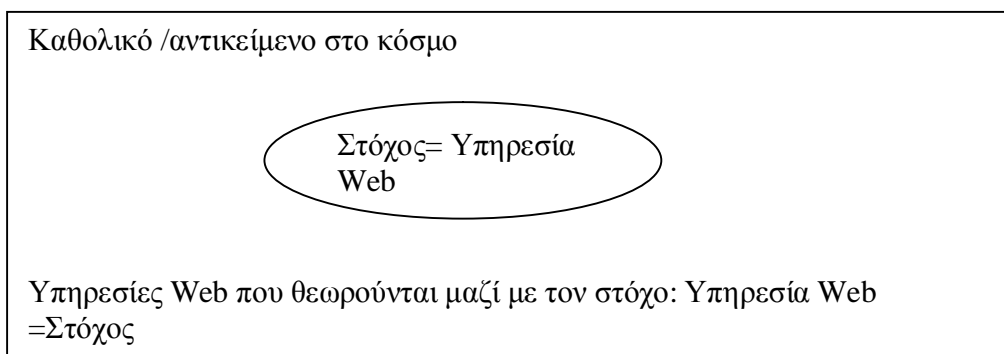
$$R_G \subseteq R_W$$

- Υποσύνολο περιγραφής υπηρεσιών Web της περιγραφής στόχου:  $R_W \subseteq R_G$
- Κοινό στοιχείο της περιγραφής υπηρεσιών στόχου και Web:  $R_G \cap R_W \neq \emptyset$
- Κανένα κοινό στοιχείο της περιγραφής υπηρεσιών στόχου και Web:  $R_G \cap R_W = \emptyset$

Αυτό το σύνολο θεωρητικών σχέσεων παρέχουν τα βασικά μέσα για τη διαισθητική κατανόηση μιας αντιστοιχίας μεταξύ των στόχων και των υπηρεσιών Web στον πραγματικό κόσμο. Για αυτό το λόγο, έχουν εξεταστεί ως ένα ορισμένο βαθμό ήδη στη λογοτεχνία, για παράδειγμα στο [LH03] ή στο [PKPS02] τα πλαίσια της αντιστοιχίας των υπηρεσιών που βασίζονται στην περιγραφή

Αφ' ενός, πρέπει να λάβουμε υπόψη ότι στο πρότυπό μας αυτά τα σύνολα περιλαμβάνουν μόνο ένα μέρος της σημασιολογίας των στόχων και της περιγραφής των υπηρεσιών, δηλαδή τα σχετικά αντικείμενα για τον αιτούντα υπηρεσιών ή το φορέα παροχής υπηρεσιών. Οι προθέσεις αυτών των συνόλων στη σημασιολογική περιγραφή της υπηρεσίας του στόχου ή στον Ιστό δεν εξετάζεται αλλά σαφώς έχει επιπτώσεις εάν μια ορισμένη υπάρχουσα συνολική θεωρητική σχέση μεταξύ του  $R_G$  και του  $R_W$  εξετάζει την πραγματική αντιστοιχία, σε μια αναμενόμενη αντιστοιχία. Πιο κάτω θα συζητήσουμε ενιαίες για σύνολο των θεωρητικών σχέσεων καθώς επίσης και την ερμηνεία τους λεπτομερώς:

- **καθορισμένη ισότητα:  $R_G = R_W$**



Εδώ τα αντικείμενα που διαφημίζονται από την αφηρημένη ικανότητα του  $W$  (και που έτσι μπορείτε ενδεχομένως να παραδοθεί από την υπηρεσία) και το σύνολο των σχετικών αντικειμένων για την αντιστοιχία του αιτούντος (που δίνεται από το  $C$ ) ταιριάζουν τέλεια, δηλ. συμπίπτουν.

Εντούτοις, κατά την εξέταση των πιθανών συνδυασμών προθέσεων των  $I_W$  και  $I_G$  για  $R_W$  και  $R_G$ , παίρνουμε τις ακόλουθες διαισθητικές ερμηνείες της συνολικής θεωρητικής σχέσης:

1. Ο αιτών υπηρεσίας θέλει να πάρει όλα τα αντικείμενα που έχει διευκρινίσει σχετικά ( $I_G = \forall$ ), και η υπηρεσία είναι σε θέση να παραδώσει όλα τα αντικείμενα που διευκρινίζονται μέσα στο  $R_W$  ( $I_W = \forall$ ). Σε αυτήν την περίπτωση, οι ανάγκες του αιτούντος και η διαφημισμένη υπηρεσία ικανότητας αντιστοιχίζονται τέλεια.

2. Ο αιτών υπηρεσιών θέλει να πάρει μερικά από τα αντικείμενα που έχει διευκρινίσει σαν σχετικά ( $I_G = \subseteq$ ), ενώ η υπηρεσία είναι σε θέση να παραδώσει όλα τα αντικείμενα διευκρινισμένα στο  $R_W$  ( $I_W = \forall$ ). Σε αυτήν την περίπτωση, οι ανάγκες του αιτούντος και η διαφημισμένη υπηρεσία επίσης ταιριάζουν τέλεια. Από μία άποψη, η υπηρεσία υπερικανοποιεί τις ανάγκες του αιτούντος.

3. Ο αιτών υπηρεσιών θέλει να πάρει όλα τα αντικείμενα που έχει διευκρινίσει σχετικά ( $I_G = \forall$ ), ενώ ο φορέας παροχής υπηρεσιών υποστηρίζει ότι η υπηρεσία Web είναι ικανή να παραδώσει μόνο μερικά από τα αντικείμενα που διευκρινίζονται στο  $R_W$  ( $I_W = \exists$ ). Σε αυτή τη περίπτωση, οι ανάγκες του αιτούντος δεν μπορούν να ικανοποιηθούν πλήρως από την υπηρεσία. Στην καλύτερη περίπτωση, η υπηρεσία μπορεί να συμβάλει για να επιλύσει την επιθυμία του πελάτη. Κατά συνέπεια, εξετάζουμε αυτή τη περίπτωση ως μερική αντιστοιχία (partial match).

4. Ο αιτών υπηρεσιών θέλει να πάρει μερικά από τα αντικείμενα που έχει διευκρινίσει σχετικά ( $I_G = \exists$ ), και ο φορέας παροχής υπηρεσιών υποστηρίζει ότι η υπηρεσία Web είναι σε θέση για να παραδώσει μόνο μερικά από τα αντικείμενα που διευκρινίζονται στο  $R_W$  ( $I_W = \exists$ ). Σε αυτήν την περίπτωση, οι ανάγκες του



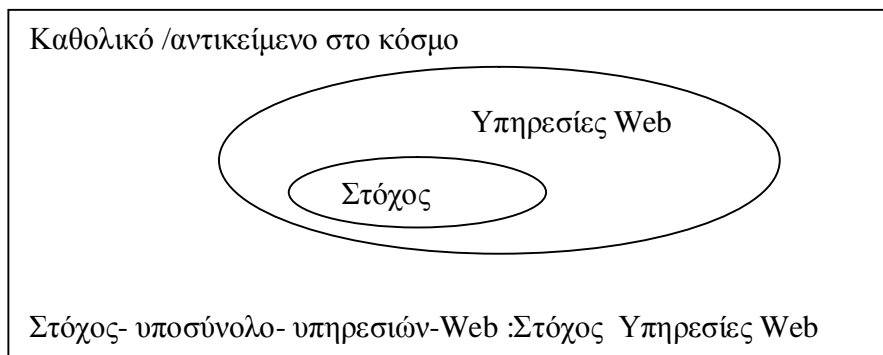
αιτούντος η διαφημισμένη υπηρεσία πάλι ταιριάζουν . Αυτή τη φορά, η υπηρεσία Web πέρα από αυτό δεν ικανοποιεί απαραίτητως τις ανάγκες του αιτούντος.

Στο [LH03] η κατάσταση όπου  $R_G = R_W$  μπορεί να καθιερωθεί καλείται ακριβή αντιστοιχία. Εντούτοις, στο πρότυπό μας δεν εξετάζουμε απαραίτητως το στόχο και την υπηρεσία του Web που ταιριάζει τέλεια, δεδομένου ότι θα εξαρτηθεί από τις αντίστοιχες προθέσεις του αιτούντος και του προμηθευτής.

- **Υποσύνολο περιγραφής του στόχου της περιγραφής των υπηρεσιών του Web:  $R_G \subseteq R_W$**

Εδώ τα σχετικά αντικείμενα που διαφημίζονται από το φορέα παροχής υπηρεσιών διαμορφώνουν υπερσύνολο από το σύνολο σχετικών αντικειμένων για τον αιτούντα όπως διευκρινίζεται στο στόχο G.

Με άλλα λόγια, η δύναμη υπηρεσιών είναι σε θέση να παραδώσει όλα τα σχετικά αντικείμενα (που εξαρτώνται στην αντίστοιχη πρόθεση του  $R_W$ ).



Κατά την εξέταση των πιθανών συνδυασμών προθέσεων των  $I_W$  και  $I_G$  για τα  $R_W$  και  $R_G$ , παίρνουμε τις ακόλουθες διαισθητικές ερμηνείες της συνολικής θεωρητικής σχέσης:

1. Για τις προθέσεις  $I_G = \forall$  και  $I_W = \forall$  , οι ανάγκες του αιτούντος καλύπτονται πλήρως κοντά στην υπηρεσία Web και έχουμε μια αντιστοιχία.
2. Για τις προθέσεις  $I_G = \exists$  και  $I_W = \forall$ , οι ανάγκες του αιτούντος καλύπτονται πλήρως κοντά στην υπηρεσία Web. Από μία άποψη, η υπηρεσία υπέρ-ικανοποιεί ακόμη και τις ανάγκες του αιτούντος, δεδομένου ότι είναι σε θέση πραγματικά να παραδώσει όλα τα σχετικά αντικείμενα για τον πελάτη.

3. Για τις προθέσεις  $I_G = \forall$  και  $I_W = \exists$ , δεν μπορούμε να καθορίσουμε εάν υπάρχει πραγματικά μια αντιστοιχία, δεδομένου ότι δεν ξέρουμε ποια στοιχεία του  $R_W$  έχουν πραγματικά παραδοθεί. Μπορεί να περιλαμβάνει το πραγματικά παραδομένο σύνολο όλων των στοιχεία του  $R_G$  (σε εκείνη την περίπτωση θα είχαμε πράγματι μια αντιστοιχία), αλλά αυτό επίσης μπορεί να συμβαίνει γιατί δεν περιέχει κανένα στοιχείο. Ως εκ τούτου, θεωρούμε ένα από αυτά πιθανή αντιστοιχία. Στην πιθανή αντιστοιχία του χρόνου, μπορούμε να βρούμε τρεις δυνατότητες: μια τέλεια αντιστοιχία, μία μερική αντιστοιχία, ή μια μη-αντιστοιχία.

4. Για τις προθέσεις  $I_G = \exists$  και  $I_W = \exists$ , δεν είναι εγγυημένα ότι ένα σχετικό στοιχείο για τον αιτούντα θα παραδοθεί από τον προμηθευτή. Εντούτοις, είναι επίσης δυνατό ότι η υπηρεσία παραδίδει κάποιο ενδιαφέρον αντικείμενο για τον αιτούντα και, επομένως, θεωρούμε αυτήν την κατάσταση μια πιθανή αντιστοιχία. Στην χρονική συνένωση, θα υπάρξει είτε μια τέλεια αντιστοιχία είτε μια μη-αντιστοιχία.

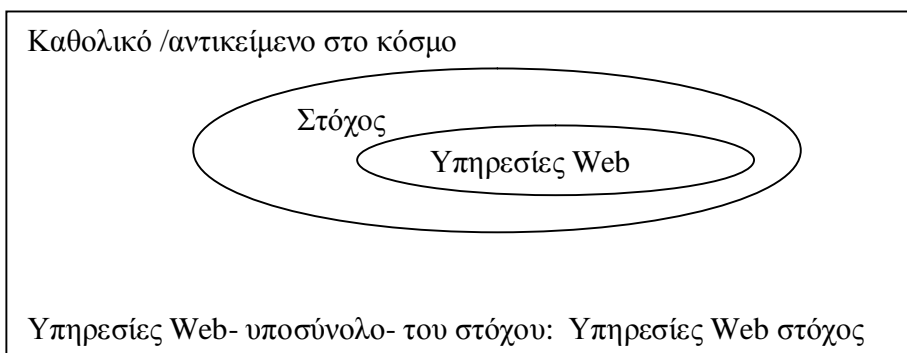
Στο [LH03] η κατάσταση όπου  $R_G \subseteq R_W$  μπορεί να καθιερωθεί λέγεται plug-in αντιστοιχία. Εντούτοις, όπως έχει αναφερθεί ανωτέρω, μπορεί να αντιστοιχεί σε μια αντιστοιχία ή σε μια πιθανή αντιστοιχία.

- **Υποσύνολο περιγραφής υπηρεσιών Web της περιγραφής του στόχου:**

$$R_W \subseteq R_G$$

Εδώ τα σχετικά αντικείμενα που διαφημίζονται από το φορέα παροχής υπηρεσιών διαμορφώνουν μόνο ένα υποσύνολο του σχετικού συνόλου αντικειμένων για τον αιτούντα όπως διευκρινίζεται στο στόχο G.

Με άλλα λόγια, η υπηρεσία δεν είναι γενικά ικανή να παραδώσει όλα τα αντικείμενα που είναι σχετικά αντικείμενα για τον αιτούντα.



Κατά την εξέταση των πιθανών συνδυασμών προθέσεων  $I_W$  και  $I_G$  για τα  $R_W$  και  $R_G$ , έχουμε τις ακόλουθες καταστάσεις:

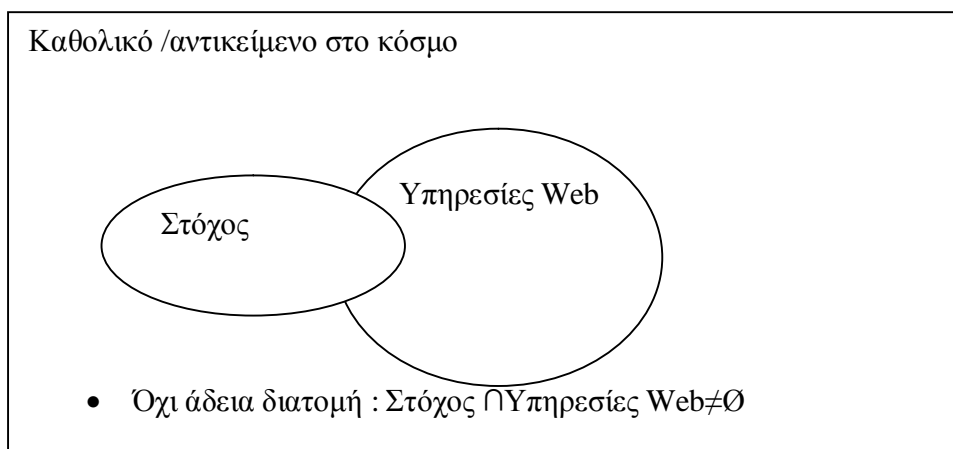
1. Για τις προθέσεις  $I_G = \forall$  και  $I_W = \forall$ , οι ανάγκες του αιτούντος δεν μπορούν να είναι πλήρως ικανοποιημένες από την υπηρεσία. Στην καλύτερη περίπτωση, η υπηρεσία μπορεί να συμβάλει για να επιλύσει την επιθυμία του πελάτη. Κατά συνέπεια, θεωρούμε αυτήν την περίπτωση ως μερική αντιστοιχία.
2. Για τις προθέσεις  $I_G = \exists$  και  $I_W = \forall$ , οι ανάγκες του αιτούντος καλύπτονται πλήρως κοντά στις υπηρεσίες του Web και, επομένως, έχουν μια αντιστοιχία.
3. Για τις προθέσεις  $I_G = \forall$  και  $I_W = \exists$ , οι ανάγκες του αιτούντος δεν μπορούν να είναι πλήρως ικανοποιημένες από την υπηρεσία αλλά η υπηρεσία μπορεί τουλάχιστον να συμβάλει για να επιλύσει την επιθυμία του πελάτη. Κατά συνέπεια, θεωρούμε αυτήν την περίπτωση μερική αντιστοιχία.
4. Για τις προθέσεις  $I_G = \exists$  και  $I_W = \exists$ , έχουμε την ίδια κατάσταση όπως στη δεύτερη περίπτωση και θεωρούμε αυτήν την κατάσταση μια αντιστοιχία.

Στο [LH03] η κατάσταση όπου το  $R_W \subseteq R_G$  μπορεί να καθιερωθεί καλείται εντάσσει αντιστοιχία. Εντούτοις, στο πρότυπό μας μπορούμε να έχουμε μια συνολική ή μερική αντιστοιχία.

- **κοινό στοιχείο της περιγραφής υπηρεσιών στόχου και Web:  $R_G \cap R_W \neq \emptyset$**

Εδώ το σύνολο σχετικών αντικειμένων που διαφημίζονται από το φορέα παροχής υπηρεσιών και το σύνολο των σχετικών αντικειμένων για τον αιτούντα έχει μια όχι άδεια διατομή, δηλ. εκεί είναι τουλάχιστον ένα αντικείμενο στον κοινό κόσμο που δηλώνεται σχετικό και από τα δύο συμβαλλόμενα μέρη.

Με άλλα λόγια, η υπηρεσία δεν είναι γενικά ικανή να παραδώσει όλα τα αντικείμενα που είναι σχετικά αντικείμενα για τον αιτούντα, αλλά τουλάχιστον ένα τέτοιο στοιχείο μπορεί να παραδοθεί.



Κατά την εξέταση των πιθανών συνδυασμών των προθέσεων των  $I_W$  και  $I_G$  για τα  $R_W$  και  $R_G$ , διακρίνουμε τις ακόλουθες περιπτώσεις:

1. Για τις προθέσεις  $I_G = \forall$  και  $I_W = \forall$ , οι ανάγκες του αιτούντος δεν μπορούν να είναι πλήρως ικανοποιημένες από την υπηρεσία. Στην καλύτερη περίπτωση, η υπηρεσία μπορεί (αδύναμα) να συμβάλει για να επιλύσει την επιθυμία του πελάτη. Κατά συνέπεια, θεωρούμε αυτήν την περίπτωση ως μερική αντιστοιχία.

2. Για τις των προθέσεις  $I_G = \exists$  και  $I_W = \forall$ , οι ανάγκες του αιτούντος καλύπτονται πλήρως κοντά στην υπηρεσία Web και έχουμε μια αντιστοιχία.

3. Για τις προθέσεις  $I_G = \forall$  και  $I_W = \exists$ , οι ανάγκες του αιτούντος δεν καλύπτονται πλήρως. Δεν είμαστε ικανοί να καθορίσουμε εάν η υπηρεσία μπορεί πραγματικά να παραδώσει το κοινό στοιχείο του  $R_G$  και του  $R_W$ . Εντούτοις μπορεί η υπηρεσία να μπόρεσε να παραδώσει μερικά ζητούμενα αποτελέσματα, έτσι θεωρούμε αυτήν την κατάσταση ένα πιθανό μέρος αντιστοιχίας. Στην χρονική συνένωση, μπορούμε στην καλύτερη περίπτωση να έχουμε μια μερική αντιστοιχία, και στη χειρότερη περίπτωση θα έχουμε μια μη αντιστοιχία.

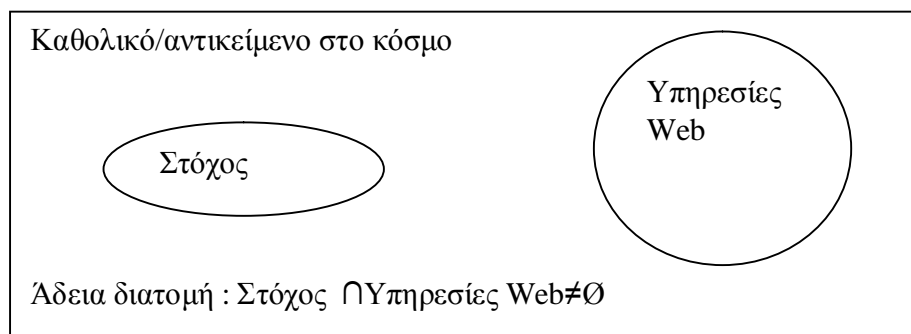
4. Για τις προθέσεις των  $I_G = \exists$  και  $I_W = \exists$ , θεωρούμε την κατάσταση μια πιθανή αντιστοιχία. Στην χρονική συνένωση, μπορούμε να έχουμε μια τέλεια αντιστοιχία ή μια μη-αντιστοιχία.

Στο [LH03] η κατάσταση όπου  $R_G \cap R_W \neq \emptyset$  μπορεί να καθιερωθεί να καλείται διατομή αντιστοιχίας. Εντούτοις, στο πρότυπό μας μπορούμε να έχουμε τις διαφορετικές γεύσεις της αντιστοιχίας.

- **κανένα κοινό στοιχείο της περιγραφής υπηρεσιών στόχου και Web:**

$$R_G \cap R_W \neq \emptyset$$

Εδώ, περιγράφονται τα αντικείμενα υπηρεσιών Web που αναφέρονται σε αυτό και τα αντικείμενα που αναφέρεται ο στόχος του αιτούντος είναι χωρισμένα. Αυτό σημαίνει ότι δεν υπάρχει καμία σημασιολογική σύνδεση μεταξύ των δύο περιγραφών και, επομένως, δεν υπάρχει κανένα στοιχείο κοινό και για τις δύο.



Ως εκ τούτου, ανεξάρτητα από τις αντίστοιχες διαισθήσεις για τα  $R_G$  και το  $R_W$ , εξετάζουμε αυτή τη κατάσταση για μη-αντιστοιχία.

Στο [LH03] η κατάσταση όπου  $R_G \cap R_W \neq \emptyset$  μπορεί να καθιερωθεί να καλείται ως μη ένωση.

Λαμβάνοντας υπόψη κάποιο στόχο  $G$  και κάποια υπηρεσία  $W$  του Web, το σχήμα συνοψίζει τη συζήτηση και παρουσιάζει κάτω από ποιες περιστάσεις η παρουσία της συνολικής θεωρητικής σχέσης μεταξύ των  $R_G$  και  $R_W$  θεωρείται ως αντιστοιχία (Match), μια μερική αντιστοιχία (PMatch), μια πιθανή αντιστοιχία (poMatch), μια πιθανή μερική αντιστοιχία (ppMatch), ή μια μη-αντιστοιχία (NoMatch).

Intention of $\mathcal{G} / \mathcal{W}$	$I_{\mathcal{W}} = \forall$		$I_{\mathcal{W}} = \exists$	
$I_{\mathcal{G}} = \forall$	$R_{\mathcal{G}} = R_{\mathcal{W}}$	Match	$R_{\mathcal{G}} = R_{\mathcal{W}}$	PMatch
	$R_{\mathcal{G}} \subseteq R_{\mathcal{W}}$	Match	$R_{\mathcal{G}} \subseteq R_{\mathcal{W}}$	poMatch
	$R_{\mathcal{G}} \supseteq R_{\mathcal{W}}$	PMatch	$R_{\mathcal{G}} \supseteq R_{\mathcal{W}}$	PMatch
	$R_{\mathcal{G}} \cap R_{\mathcal{W}} \neq \emptyset$	PMatch	$R_{\mathcal{G}} \cap R_{\mathcal{W}} \neq \emptyset$	ppMatch
	$R_{\mathcal{G}} \cap R_{\mathcal{W}} = \emptyset$	NoMatch	$R_{\mathcal{G}} \cap R_{\mathcal{W}} = \emptyset$	NoMatch
$I_{\mathcal{G}} = \exists$	$R_{\mathcal{G}} = R_{\mathcal{W}}$	Match	$R_{\mathcal{G}} = R_{\mathcal{W}}$	Match
	$R_{\mathcal{G}} \subseteq R_{\mathcal{W}}$	Match	$R_{\mathcal{G}} \subseteq R_{\mathcal{W}}$	poMatch
	$R_{\mathcal{G}} \supseteq R_{\mathcal{W}}$	Match	$R_{\mathcal{G}} \supseteq R_{\mathcal{W}}$	Match
	$R_{\mathcal{G}} \cap R_{\mathcal{W}} \neq \emptyset$	Match	$R_{\mathcal{G}} \cap R_{\mathcal{W}} \neq \emptyset$	poMatch
	$R_{\mathcal{G}} \cap R_{\mathcal{W}} = \emptyset$	NoMatch	$R_{\mathcal{G}} \cap R_{\mathcal{W}} = \emptyset$	NoMatch

Σχήμα 4.4 : Αλληλεπίδραση μεταξύ των σύνολο-θεωρητικών κριτηρίων, των προθέσεων και της διαισθητικής κατανόησής μας από το ταίριασμα.

**Ασυμβίβαστες περιγραφές για τους στόχους και τις υπηρεσίες.** Δεν έχουμε εξετάσει μέχρι τώρα τη δυνατότητα των ασυμβίβαστων περιγραφών για τις περιγραφές των στόχων και των υπηρεσιών του Web δηλ. όπου το  $R_{\mathcal{G}}$  ή το  $R_{\mathcal{W}}$  είναι κενό. Μπορεί να εμφανίζονται σε περιπτώσεις όπου υπάρχει ένα λάθος περιγραφής π.χ. όταν οι περιγραφές είναι αρκετά σύνθετες ή αναφέρονται σε διάφορα συκροτήματα οντολογιών που δεν σχεδιάζονται από το μοντέλο.

Επιπλέον, όταν αγνοηθούν μπορούν να ασκήσουν την ανεπιθύμητη επίδραση στο ταίριασμα και έτσι ανακαλύπτετε. Εξετάστε παραδείγματος χάριν μια ασυμβίβαστη περιγραφή στόχου, δηλ.  $R_{\mathcal{G}} = \emptyset$ . Εάν ελέγχουμε το  $\mathcal{G}$  για το ταίριασμα των υπηρεσιών του Web χρησιμοποιώντας το plug-in κριτήριο, δηλ.  $R_{\mathcal{G}} \subseteq R_{\mathcal{W}}$ , κάθε υπηρεσία Web θα ταιριάζει. Για έναν χρήστη (που δεν γνωρίζει ότι η περιγραφή του είναι ασυμβίβαστη, δεδομένου ότι ειδιάλλως συνήθως δεν θα έθετε την ερώτηση) το αποτέλεσμα θα φαινόταν μάλλον παράξενο και ακόμα και ανακριβές επειδή όλες οι ελεγχόμενες υπηρεσίες πραγματικά θα αντιστοιχηθούν.

Προβλέπουμε δύο τρόπους για να εξετάσουμε τις ασυμβίβαστες περιγραφές:

Ø Μην επιτρέψτε τη διαφήμιση των ασυμβίβαστων περιγραφών των υπηρεσιών στόχου και Web. Εάν κάποιος διαφημίζει μια περιγραφή στόχου ή υπηρεσίας που δείχνει ένα κενό σύνολο, απορρίψτε την περιγραφή προτού να μπορέσει να συμμετέχει σε οποιαδήποτε ανακάλυψη. Κατά τη διάρκεια της ανακάλυψη θα εργαστούμε μόνο με τις συνεπείς περιγραφές. Κάποιος πρέπει να είναι προσεκτικός κατά την εξέταση του κράτους συνέπειας κάποιας περιγραφής υπηρεσιών στόχου ή Web: η συνέπεια δεν εξαρτάται αποκλειστικά από την ίδια την περιγραφή αλλά επίσης από όλες τις οντολογίες στις οποίες η περιγραφή αναφέρεται. Ως εκ τούτου, αλλαγές σε τέτοιες οντολογίες ενδεχομένως μπορεί να οδηγήσουν στις ασυμβίβαστες περιγραφές υπηρεσιών στόχου και Web. Για αυτό το λόγο, μπορεί να πρέπει να επανεξετάσει τη συνέπεια όλου του εξαρτημένου στόχου και τις περιγραφές των υπηρεσιών Web που αναφέρονται σε κάποια οντολογία κατά την αλλαγή της οντολογίας.

Ø Ελέγξτε για την ασυνέπεια των περιγραφών των υπηρεσιών στόχου και Web κατά τη διάρκεια της ανακάλυψης και της επιστροφής αποδεκτών αποτελεσμάτων. Σε αυτήν την περίπτωση δεν υποθέτουμε τη συνέπεια οποιασδήποτε από τις περιγραφές που περιλαμβάνονται, αλλά ελέγχουμε τη συνέπεια κατά τη διάρκεια της φάση ανακαλύψεων και παράδοσης των λογικών αποτελεσμάτων ή ενημέρωσης του αιτούντα ή την υπηρεσία Web του προμηθευτή ότι κάτι κάνει λάθος με τη διαφημισμένη περιγραφή. Ελέγχεται η συνέπεια σημαίνει να καθοριστεί εάν υπάρχει κάποιο στοιχείο eeU έτσι ώστε να ικανοποιεί τις ιδιότητες της δεδομένης περιγραφής του στόχου (ή των υπηρεσιών Web).

Συστήνουμε να ακολουθήσουμε την πρώτη επιλογή για τις περιγραφές των υπηρεσιών και των προκαθορισμένων στόχων. Αν και θα απαιτηθεί να ελεγχθεί η συνέπεια των περιγραφών των υπηρεσιών και των προκαθορισμένων στόχων που διαφημίζονται ήδη κάθε φορά αλλάζοντας οντολογία, αυτό γίνεται off-line δηλ. χωριστά από τη διαδικασία ανακαλύψεων. Επομένως, υπάρχουν επιπτώσεις στην αποδοτικότητα της διαδικασίας ανακαλύψεων.

**Προθέσεις.** Οι προσεγγίσεις βασισμένες στο DL για την ανακάλυψη υπηρεσιών όπως στα [PKPS02], [LH03], η έννοια της «πρόθεσης» αυτή τη στιγμή δεν έχει απεικονιστεί

ρητά. Όπως έχουμε παρουσιάσει ανωτέρω, οι προθέσεις συλλαμβάνουν μια σημαντική πτυχή των περιγραφών των υπηρεσιών των στόχου και του Web και καθορίζουν εάν ορισμένα συνολικά θεωρητικά κριτήρια αντιπροσωπεύουν τα αναμενόμενα αποτελέσματα αντιστοιχιών.

Η προσέγγιση DL που είναι διαθέσιμη μπορεί μέχρι τώρα να έχει γίνει κατανοητή καλύπτοντας μόνο το χαμηλότερο αριστερό μέρος του πίνακα δηλ. η κατάσταση όπου ένας στόχος έχει μια υπαρξιακή πρόθεση και μία υπηρεσία Web έχει την καθολική πρόθεση, δηλ.  $I_G = \exists; I_W = \forall$ .

Το σχήμα επιδεικνύει λεπτομερώς πώς η γενική εικόνα επηρεάζεται όταν έρχονται οι προθέσεις στο παιχνίδι. Σε αντίθεση με τις υπάρχουσες προσεγγίσεις όπου οι προθέσεις της υπηρεσίας στόχου και οι περιγραφές του Web καθορίζονται (και δεν μπορεί έτσι να επηρεαστεί από τα μοντέλα), πιστεύουμε ότι είναι χρήσιμο να δώσει στα μοντέλα η πρόσθετη ελευθερία για να συλλάβει ακριβώς την έννοια της περιγραφής.

Επιπλέον, πιστεύουμε ότι ορισμένα ζευγάρια των προθέσεων θα εμφανιστούν συχνότερα στην πράξη από άλλα: Οι φορείς παροχής υπηρεσιών Web έχουν ένα ισχυρό ενδιαφέρον για τις υπηρεσίες Web που έχουν ανακαλυφθεί. Εάν συγκρίνουμε τον αριθμό των πιθανών αντιστοιχιών με έναν δεδομένο στόχο κάτω από τον υπαρξιακό και με τις καθολικές προθέσεις, φαίνεται πλέον πιθανό ότι οι προμηθευτές τείνουν να χρησιμοποιήσουν τις καθολικές προθέσεις, ακόμα κι αν η περιγραφή δεν διαμορφώνει απαραίτητως την ακριβή ικανότητα της υπηρεσία αλλά μόνο μιας αφαίρεσης. Εντούτοις, εάν ένας φορέας παροχής υπηρεσιών θέλει να είναι περισσότερο ακριβής με την περιγραφή υπηρεσιών του Web έπειτα σε πολλές καταστάσεις θα έπρεπε να χρησιμοποιήσει την υπαρξιακή πρόθεση. Ακόμη και στην υπαρξιακή πρόθεση, υπάρχουν πραγματικές αντιστοιχίες ανιχνεύσιμες, όταν ο αιτών χρησιμοποιεί τους στόχους με την υπαρξιακή πρόθεση.

Για τους αιτούντες υπηρεσιών (συγκεκριμένα σε ηλεκτρονικό εμπόριο που) αναμένουμε ότι η υπαρξιακή η πρόθεση θα αρκестεί σε πολλές καταστάσεις, εντούτοις ο αιτών έχει την κατάλληλη ελευθερία για σαφή ισχυρότερα αιτήματα από τους υπαρξιακούς στόχους (που χρησιμοποιούν την καθολική πρόθεση) εάν αυτός πρέπει να πάρει τα ακριβέστερα αποτελέσματα.



Οι διαφορετικές προθέσεις που συζητούνται ανωτέρω θα καθορίσουν ποια επίσημη σχέση μεταξύ του στόχου του αιτούντος και της αφηρημένης ικανότητα πρέπει να εξεταστεί. Το σχήμα περιγράφει τη σχέση μεταξύ του στόχου και των αφηρημένων περιγραφών υπηρεσιών που πρέπει να ελεγχθούν για κάθε έννοια της αντιστοιχίας.

Intention of $G / W$	$I_W = \forall$		$I_W = \exists$	
$I_G = \forall$	Match	$R_G \subseteq R_{GW}$	Match	Not possible
	PMatch	$R_G \cap R_{GW} \neq \emptyset$	PMatch	$R_G \supseteq R_{GW}$
	poMatch	Not possible	poMatch	$R_G \subseteq R_{GW}$
	ppMatch	Not possible	ppMatch	$R_G \cap R_{GW} \neq \emptyset$
	NoMatch	$R_G \cap R_{GW} = \emptyset$	NoMatch	$R_G \cap R_{GW} = \emptyset$
$I_G = \exists$	Match	$R_G \cap R_{GW} \neq \emptyset$	Match	$R_G \supseteq R_{GW}$
	PMatch	Not possible	PMatch	Not possible
	poMatch	Not possible	poMatch	$R_G \cap R_{GW} \neq \emptyset$
	ppMatch	Not possible	ppMatch	Not possible
	NoMatch	$R_G \cap R_{GW} = \emptyset$	NoMatch	$R_G \cap R_{GW} = \emptyset$

Σχήμα 4.5 : Ποια επίσημα κριτήρια πρέπει να χρησιμοποιηθούν για τον έλεγχο των διαφορετικών βαθμών ταιριάσματος.

Μπορούμε επίσης να καθορίσουμε ότι μερικές έννοιες της αντιστοιχίας θα προτιμηθούν. Είναι σαφές ότι η αντιστοιχία θα προτιμηθεί από άλλες έννοιες. Θεωρούμε επίσης μια πιθανή αντιστοιχία προτιμητέα από μια μερική αντιστοιχία, ως πιθανή αντιστοιχία μπορεί να αποδειχθεί μια τέλεια αντιστοιχία στην χρονική συνένωση, ενώ μια μερική αντιστοιχία δεν θα παρέχει ποτέ όλα αυτά που ζητούνται. Προφανώς, μια μερική αντιστοιχία θα προτιμηθεί πέρα από μια πιθανή μερική αντιστοιχία, δεδομένου ότι η πρώτη προσφέρει μια εγγύηση ότι στον κατάλογο μερικά σχετικά αποτελέσματα θα παρασχεθούν, ενώ τα τελευταία μπορούν να αποδειχθούν ότι δεν αντιστοιχούνται. Παρουσιάζουμε αυτήν την διαταγή κατωτέρω, με την έννοια  $\succ$  η αριστερή πλευρά προτιμάται πέρα από αυτήν στη δεξιά πλευρά:

Αντιστοιχία  $\xrightarrow{\lambda}$  πιθανή-αντιστοιχία  $\xrightarrow{\lambda}$  μερική-αντιστοιχιών  $\xrightarrow{\lambda}$  πιθανός-μερική-  
αντιστοιχία  $\xrightarrow{\lambda}$  μη-αντιστοιχία

Για τα παραδείγματα υπηρεσιών στόχου και Web που δίνονται πριν, οι ακόλουθες αντιστοιχίες θα είναι καθιερωμένος:

Αντιστοιχία:

$G_1-W_1, G_2-W_1, G_2-W_2, G_3-W_1, G_4-W_1, G_5-W_4$

Πιθανός-αντιστοιχία:

$G_3-W_2$

Μερικός-αντιστοιχία:

$G_1-W_3$

**Ταιριάζοντας με το σενάριο.** Κατά τη διάρκεια της διαδικασίας ανακαλύψεων το σενάριο για το ταίριασμα μεταξύ των περιγραφών υπηρεσιών στόχου και Web είναι γενικά το ακόλουθο: Ένας αιτών διευκρινίζει το στόχος του (ο καθαρισμός ενός προκαθορισμένου στόχου) με τη βοήθεια ενός συνόλου σχετικών αντικειμένων και η αντίστοιχη πρόθεση  $((R_G, I_G))$ , και ελέγχουμε για μια αντιστοιχία χρησιμοποιώντας τα αντίστοιχα κριτήρια για τις προθέσεις  $(I_G, I_W)$  από το σχήμα. Θα αρχίσουμε τους διαφορετικούς τύπους αντιστοιχιών μετά από τη διαταγή που δίνεται κατωτέρω, αρχίζοντας από την προτιμημένη έννοια της αντιστοιχίας.

Η συζήτηση δείχνει ότι κατά τη διάρκεια της ανακάλυψης και του ταυριάζματος με προθέσεις μπορεί να εξεταστεί σε επόμενα επίπεδα (σε σύγκριση με τις συνολικές θεωρητικές έννοιες), δηλ. δεν έχουν επιπτώσεις στα άμεσα ενιαία συνολικά θεωρητικά κριτήρια και στους απαραίτητους ελέγχους. Ως εκ τούτου, χρειαζόμαστε ακριβώς μια εφαρμογή από τα διαφορετικά συνολικά θεωρητικά κριτήρια προκειμένου να πραγματοποιηθεί ένα σύστημα για την αντιστοιχία.

### **Πραγματοποίηση της προσέγγισης στη λογική**

Στη διαμόρφωση της προσέγγισης που παρουσιάζεται ανωτέρω, δεν θα επιβάλουμε προς το παρόν οποιοδήποτε συγκεκριμένο περιορισμό στην έκφραση της γλώσσας δηλ. θα επιτρέψουμε ένα σύνολο first-order γλώσσας για την περιγραφή των στόχων του αιτούντος και των αφηρημένων ικανοτήτων.

Τυπικά, αντιπροσωπεύουμε τα σύνολα  $R_W$  και  $R_G$  που χρησιμοποιούνται για την περιγραφή της σημασιολογίας της Υπηρεσίας Web με τη βοήθεια των μοναδιαίων κατηγορημάτων σε first-order γλώσσα. Όλα τα στοιχεία των καταστάσεων που ικανοποιούν το κατηγορήμα (σύμφωνα με τον ορισμό του) θεωρούνται στοιχεία του συνόλου.

Η ακριβής σημασιολογία του συμβόλου  $W_{post}(X)$  κατηγορήματος καθορίζεται με τη βοήθεια του καθορισμού  $W$  των πρωταρχικών τύπων. Γενικά, υποθέτουμε ότι οι τύποι στο  $W$  αναφέρονται σε κάποιο σύνολο οντολογιών. Ως εκ τούτου, αυτές οι οντολογίες πρέπει επίσης να περιγραφούν τυπικά με διαταγή να καθοριστεί η σημασιολογία του συμβόλου  $W_{post}$  κατηγορήματος ( $X$ ) κατά τρόπο επίσημο. Η επίσημη αντιπροσώπευση αυτών των οντολογιών ως λογική θεωρία στο first-order πλαίσιο δείχνεται από το  $O$ .

Το καθορισμένο  $W$  του καθορισμού των τύπων πρέπει να επιλεγεί κατά τέτοιο τρόπο ώστε κάτω από κάθε ερμηνεία  $I$  που είναι ένα πρότυπο του  $W$  όπως οι οντολογίες  $O$  στις οποίες ο καθορισμός αναφέρεται, το κατηγορήμα ερμηνεύεται ως αντίστοιχο σύνολο σχετικών αντικειμένων, δηλ.

$$I \models \mathcal{W} \Leftrightarrow I(W_{post}(x)) = R_{\mathcal{W}}$$

για κάθε ερμηνεία  $I$ .

Υπό αυτήν την επίσημη έννοια, το  $W$  (και  $O$ ) καθορίζουν την ερμηνεία του συμβόλου  $W_{post}$  μέσα στο προοριζόμενο τρόπο. Προφανώς, το σύμβολο  $W_{post}$  πρέπει να εμφανιστεί στο  $W$ .

Στην απλούστερη περίπτωση, μπορούμε ακριβώς να χρησιμοποιήσουμε έναν ενιαίο τύπο της μορφής

$$\mathcal{W} : \forall x. (\psi(x) \leftrightarrow W_{post}(x))$$

όπου  $\psi(x)$  είναι ένας πρωταρχικός αυθαίρετος τύπος με ακριβώς μια ελεύθερη μεταβλητή  $x$ .

Για τους στόχους  $G$  κάνουμε το ίδιο πράγμα και καθορίζουμε ένα μοναδιαίο σύμβολο  $G_{post}$  κατηγορήματος ( $X$ ) με τη σημασία από ένα (κλειστό) σύνολο  $G$  first-order τύπων και τη διαμόρφωση  $O$  του συνόλου οντολογιών τους οποίους ο καθορισμός αναφέρει. Πάλι, ο καθορισμός  $G$  πρέπει να ικανοποιήσει την ιδιοκτησία

$$I \models \mathcal{G} \Leftrightarrow I(G_{post}) = R_{\mathcal{G}}$$

για κάθε ερμηνεία I.

Πάλι, στην απλούστερη περίπτωση  $G_{post}(X)$  μπορεί να καθοριστεί από έναν ενιαίο τύπο της μορφής

$$\mathcal{G} : \forall \mathbf{x}. \{ \phi(\mathbf{x}) \leftrightarrow G_{post}(\mathbf{x}) \}$$

όπου  $\phi(x)$  είναι ένας πρωταρχικός αυθαίρετος τύπος με ακριβώς μια ελεύθερη μεταβλητή X.

**Λογική διαμόρφωση του συνόλου των θεωρητικών σχέσεων.** Λαμβάνοντας υπόψη το λογικό προορισμό W μιας αφηρημένης ικανότητας και G ενός στόχου, θέλουμε τώρα να επιδείξουμε πώς να αντιπροσωπεύσουμε τα συνολικά θεωρητικά κριτήρια από την προηγούμενη παράγραφο στο λογικό πλαίσιο μας.

Παρόμοια με το [LH03], διακρίναμε πέντε τύπους κριτηρίων. Παρατηρήστε ότι αυτά τα κριτήρια αντιστοιχούν στη διαμόρφωση των καθορισμένων σχέσεων που χρησιμοποιούνται σε προηγούμενο σχήμα, όχι σε έννοιες που αντιστοιχούν οι ίδιοι δηλ. η αντιστοιχία λέξης κατωτέρω αναφέρεται στη σχέση μεταξύ συνόλων που περιγράφονται χωρίς εξέταση των προθέσεων:

- **Exact-match.** Εδώ τα σύνολα των σχετικών αντικειμένων της περιγραφής υπηρεσιών του Web και η περιγραφή του στόχου συμπίπτει:  $R_G = R_W$ . Κάθε στοιχείο  $e \in U$  του U κόσμου ο οποίος είναι στο  $R_G \subseteq U$  είναι επίσης στο  $R_W \subseteq U$  και αντίστροφα.

Τυπικά, αυτό σημαίνει ότι πρέπει να αποδείξουμε τα εξής:

$$\mathcal{W}, \mathcal{G}, \mathcal{O} \models \forall \mathbf{x}. (G_{post}(\mathbf{x}) \leftrightarrow W_{post}(\mathbf{x}))$$

όπου το W είναι ο καθορισμός της υπηρεσίας Web, το G είναι ο καθορισμός του στόχου, και το O είναι ένα σύνολο οντολογιών στο οποίο και οι δύο περιγραφές αναφέρονται.

Σε αυτήν την περίπτωση γράφουμε το  $\mathcal{W} \equiv_{\mathcal{O}} \mathcal{G}$  για να δείξουμε αυτό το ιδιαίτερο είδος αντιστοιχίας.

- **Subsumption-match.** Εδώ το σύνολο σχετικών αντικειμένων για τις υπηρεσίες Web είναι ένα υποσύνολο του συνόλου σχετικών αντικειμένων του

στόχου:  $R_w \subseteq R_G$ . Κάθε στοιχείο  $e \in U$  του κόσμου  $U$  που είναι  $R_w \subseteq U$  είναι επίσης  $R_G \subseteq U$ .

Τυπικά, αυτό σημαίνει ότι πρέπει να αποδείξουμε τα εξής:

$$\mathcal{W}, \mathcal{G}, \mathcal{O} \models \forall x. (W_{post}(x) \rightarrow G_{post}(x))$$

Σε αυτήν την περίπτωση γράφουμε το  $\mathcal{W} \subseteq_o \mathcal{G}$  για να δείξουμε αυτό το ιδιαίτερο είδος αντιστοιχίας.

- **Plug in-match.** Εδώ το σύνολο σχετικών αντικειμένων για την υπηρεσία Web είναι ένα υπερσύνολο από το σύνολο σχετικών αντικειμένων του στόχου:  $R_G \subseteq R_w$ . Κάθε στοιχείο  $e \in U$  του κόσμου  $U$  που είναι  $R_G \subseteq U$  είναι επίσης  $R_w \subseteq U$ .

Τυπικά, το οποίο σημαίνει ότι πρέπει να αποδείξουμε τα εξής:

$$\mathcal{W}, \mathcal{G}, \mathcal{O} \models \forall x. (G_{post}(x) \rightarrow W_{post}(x))$$

Σε αυτήν την περίπτωση γράφουμε το  $\mathcal{W} \supseteq_o \mathcal{G}$  για να δείξουμε αυτό το ιδιαίτερο είδος αντιστοιχίας.

- **Intersection-match.** Εδώ η διατομή μεταξύ του συνόλου σχετικών αντικειμένων για την υπηρεσία Web και το σύνολο σχετικών αντικειμένων του στόχου δεν είναι το κενό σύνολο:  $R_w \cap R_G \neq \emptyset$ . Υπάρχει ένα στοιχείο  $e \in U$  του κόσμου  $U$  που είναι και στα δύο  $R_G \subseteq U$ ,  $R_w \subseteq U$ .

Τυπικά, αυτό σημαίνει ότι πρέπει να αποδείξουμε τα εξής:

$$\mathcal{W}, \mathcal{G}, \mathcal{O} \models \exists x. (G_{post}(x) \wedge W_{post}(x))$$

Σε αυτήν την περίπτωση γράφουμε  $\mathcal{W} \sqcap_o \mathcal{G}$  για να δείξουμε αυτό το ιδιαίτερο είδος αντιστοιχίας.

- **Non-match** Εδώ η διατομή μεταξύ του συνόλου σχετικών αντικειμένων για την υπηρεσία Web και το σύνολο σχετικών αντικειμένων του στόχου είναι το κενό σύνολο:  $R_w \cap R_G = \emptyset$ . Δεν υπάρχει κανένα στοιχείο  $e \in U$  του κόσμου  $U$  που είναι και στα δύο  $R_G \subseteq U$  και στο  $R_w \subseteq U$ .

Τυπικά, το οποίο σημαίνει ότι πρέπει να αποδείξουμε τα εξής:

$$\mathcal{W}, \mathcal{G}, \mathcal{O} \models \neg \exists x. (G_{post}(x) \wedge W_{post}(x))$$

Σε αυτήν την περίπτωση γράφουμε  $\mathcal{W} \parallel \mathcal{O} \mathcal{G}$  για να δείξουμε την non-match.

Για τη διαμόρφωση των παραδειγμάτων που δίνονται στην αρχή της ακολουθίας τμημάτων οι διαμορφώσεις ανωτέρω υποθέτουμε ότι είναι ένα κατάλληλο σύνολο οντολογιών  $\mathcal{O}$  για γεωγραφικά στοιχεία και ταξίδια που είναι σε ισχύ. Συγκεκριμένα, έχουμε μια σχέση  $(X, Y)$  όπου το αντικείμενο  $X$  βρίσκεται γεωγραφικά στο αντικείμενο  $Y$ , μια πτήση σχέσης  $(f, s, e)$  όπου το  $f$  είναι μια πτήση από τη θέση  $s$  στη θέση  $e$  και ένα τραίνο σχέσης  $(t, s, e)$  που δηλώνει ότι το  $t$  είναι μια σύνδεση τρενών από τη θέση  $s$  στη θέση  $e$ .

Η διαμόρφωση του συνόλου αντικειμένων σχετικών με παραδείγματος χάριν  $G_1$  θα ήταν:

$G_1: \forall x (s \in \text{flight}(x, s, e) \wedge \text{in}(s, \text{ireland}) \wedge \text{in}(e, \text{austria})) \leftrightarrow G_{1post}(X)$  όπου η Ιρλανδία και η Αυστρία είναι σταθερές στο  $\mathcal{O}$ . Η διαμόρφωση των σχετικών αντικειμένων για, παραδείγματος χάριν,  $W_3$  θα ήταν:

$$W_3: x. (\exists s, \text{flight}(x, s, \text{innsburg}) \wedge \text{in}(s, \text{ireland})) \leftrightarrow W_{3post}(X)$$

όπου το Ίνσμπρουκ είναι μια σταθερά στο  $\mathcal{O}$  για το οποίο η σχέση μέσα στο  $(\text{innsburg}, \text{austria})$  κρατά.

Δεδομένου ότι έχουμε τις καθολικές προθέσεις και για το στόχο και για την αφηρημένη ικανότητα, πρέπει να ελέγξουμε για μια αντιστοιχία την ακόλουθη επίσημη σχέση:

$$\mathcal{W}_3, \mathcal{G}_1, \mathcal{O} \models \forall x. (G_{1post}(x) \rightarrow W_{3post}(x))$$

που αντιστοιχεί στη σχέση:

$$\mathcal{W}_3, \mathcal{G}_1, \mathcal{O} \models \forall x. (\exists s, e (\text{flight}(x, s, e) \wedge \text{in}(s, \text{ireland}) \wedge \text{in}(e, \text{austria})) \rightarrow \text{flight}(x, s, \text{innsbruck}) \wedge \text{in}(s, \text{ireland}))$$

που δεν κρατά.

Για τον έλεγχο της έπειτα προτιμημένης πιθανής έννοιας της αντιστοιχίας δηλ. μερική αντιστοιχία, πρέπει να ελέγξουμε ότι:

$\mathcal{W}_3, \mathcal{G}_1, \mathcal{O} \models \exists x. (\exists s, e (flight(x, s, e) \wedge in(s, ireland) \wedge in(e, austria)) \wedge$   
 $(\exists s flight(x, s, innsbruck) \wedge in(s, ireland)))$   
 που ισχύει για  $e = innsburg$  εάν η οντολογία περιοχών λέει ότι είναι μέσα στο  
 (innsburg, austria).

Επομένως, η  $\mathcal{W}_3$  είναι μια μερική αντιστοιχία για το  $\mathcal{G}_1$ .

#### 4.5 Χρησιμοποιώντας τη DL για το χαρακτηρισμό των αποτελεσμάτων

Σε αυτό το τμήμα, ερευνάμε τον περιορισμό της εκφραστικότητας που έχει την άδεια για να περιγράψει τις αφηρημένες ικανότητες και τους στόχους των αιτούντων προκειμένου να αυξηθεί η αποδοτικότητα της υπηρεσίας για τη διαδικασία ανακαλύψεων.

Θα εισαγάγουμε εν συντομία τις λογικές περιγραφής (DLs), να περιγράψουν πώς η βασισμένη στις γλώσσες οντολογία DL μπορούν να εφαρμοστούν στην ανακάλυψη υπηρεσιών Web, και να αναλύσουν τα υπολογιστικά οφέλη που παίρνουμε από τη χρήση της διαμόρφωσης DL και της αιτιολογίας DL. Θα συζητήσουμε επίσης για DLs που παρέχει προσαρμοσμένους τύπους δεδομένων και τύπους δεδομένων για τα κατηγορήματα, που μπορούν να είναι ενδιαφέροντα για την ανακάλυψη υπηρεσιών Web.

##### Λογικές περιγραφής

Οι λογικές περιγραφής (DLs) [BN03] είναι οικογένεια της κατηγορίας που βασίζεται στην αντιπροσώπευση γνώσης φορμαλισμού, που εξοπλίζονται από την καθορισμένη με σαφήνεια πρότυπη θεωρητική σημασιολογία [BMNPS02]. Αυτές αναπτύχθηκαν αρχικά για να παρέχουν την επίσημη, δηλωτική έννοια στα σημασιολογικά δίκτυα [Qui67] και στα πλαίσια [Min81], και για να επιδείξουν πώς τέτοιες δομημένες αντιπροσωπεύσεις μπορούν να εξοπλιστούν με τα αποδοτικά εργαλεία συλλογισμού. Οι βασικές έννοιες των λογικών περιγραφής είναι κατηγορίες, δηλ., μοναδιαία κατηγορήματα που ερμηνεύονται όπως τα σύνολα αντικειμένων, και ιδιότητες, δηλ., δυαδικά κατηγορήματα που ερμηνεύεται όπως τα σύνολα ζευγαριών.

Οι λογικές περιγραφές χαρακτηρίζονται από τους κατασκευαστές που παρέχουν για να χτίσουν σύνθετες περιγραφές κατηγορίας και ιδιοκτησίας από ατομικές. Παραδείγματος χάριν, «ελέφαντες με ηλικίες τους μεγαλύτερες από 20» μπορούν να περιγραφούν από την ακόλουθη περιγραφή κατηγορίας DL:

$$\text{Elephant} \sqcap \exists \text{age. } >_{20}$$

όπου ο ελέφαντας είναι μια ατομική κατηγορία, η ηλικία είναι μια ατομική ιδιοκτησία τύπου δεδομένων, το  $>_{20}$  είναι προσαρμοσμένος τύπος δεδομένων (μεταχειρισμένος ως μοναδιαίο κατηγορήμα τύπου δεδομένων) και  $\sqcap, \exists$  είναι κατασκευαστές κατηγορίας. Όπως παρουσιάζεται ανωτέρω, οι τύποι δεδομένων και τα κατηγορήματα (όπως, =, >, +) είναι καθορισμένα πέρα από όσα μπορούν να χρησιμοποιήσουν οι κατασκευές των περιγραφών κατηγορίας. Αντίθετα από τις κατηγορίες, τύπων δεδομένων και τύπων δεδομένων για τα κατηγορήματα υπάρχουν οι προφανείς (σταθερές) επεκτάσεις π.χ., η επέκταση  $>_{20}$  είναι όλοι οι ακέραιοι αριθμοί που είναι μεγαλύτεροι από 20. Λόγω των διαφορών μεταξύ των κατηγοριών και των τύπων δεδομένων, υπάρχουν δύο είδη ιδιοτήτων: (i) ιδιότητες αντικειμένου, οι οποίες σχετίζουν τα αντικείμενα στα αντικείμενα, και (ii) ιδιότητες τύπων δεδομένων, οι οποίες σχετίζουν τα αντικείμενα και τις τιμές στοιχείων, οι οποίες είναι περιπτώσεις τύπων δεδομένων.

Οι περιγραφές κατηγορίας και ιδιοκτησίας μπορούν να χρησιμοποιηθούν στα αξιώματα στις βάσεις γνώσεων DL. DL Τα αξιώματα είναι δηλώσεις που περιγράφουν (i) τις σχέσεις μεταξύ των περιγραφών κατηγορίας (ιδιοκτησία), (ii) τα χαρακτηριστικά των ιδιοτήτων, όπως η βεβαίωση μιας ιδιοκτησίας είναι λειτουργική, ή (iii) περίπτωση-σχέσεις μεταξύ (ζευγάρια) των ατόμων και των κατηγοριών (ιδιότητες). Μπορούμε να χρησιμοποιήσουμε τα αξιώματα DL για να αντιπροσωπεύσει τις έννοιες και τους περιορισμούς σε μια οντολογία. Παραδείγματος χάριν, μπορούμε να καθορίσουμε την κατηγορία AdultElephant με το ακόλουθο αξίωμα DL

$$\text{AdultElephant} \equiv \text{Elephant} \sqcap \exists \text{age. } >_{20};$$

μπορούμε να βεβαιώσουμε ότι η ηλικία ιδιοκτησίας είναι λειτουργική (π.χ., οι ελέφαντες μπορούν να έχουν το πολύ-πολύ 1 ηλικία):



Func (age);

μπορούμε επίσης να βεβαιώσουμε ότι το αντικείμενο Ganesh είναι μια περίπτωση της κατηγορίας περιγραφής των ελεφάντων για το ποιοι είναι μεγαλύτεροι από 25 ετών»:

**Ganesh : (Elephant  $\sqcap$   $\exists$ age.  $>25$ ).**

Ένα σύστημα DL όχι μόνο αποθηκεύει τα αξιώματα, αλλά και προσφέρει τις υπηρεσίες που τα αιτιολογούν. Χαρακτηριστικά, ο συλλογισμός με μια βάση γνώσεων DL είναι μια διαδικασία στην υπονοούμενη γνώση συνεπαγόμενη από τη βάση γνώσεων. Οι αιτιολογημένες υπηρεσίες μπορούν να ταξινομηθούν κατά προσέγγιση ως βασικές υπηρεσίες, οι οποίες αναμειγνύουν τον έλεγχο της αξίας αλήθειας για μια δήλωση, και σύνθετες υπηρεσίες, οι οποίες μπορούν να χτιστούν στις βασικές. Αφήστε το  $\Sigma$  να είναι μια βάση γνώσεων, το L μια περιγραφή Λογική, τα C,D κατηγορίες, το a ένα μεμονωμένο όνομα. Οι βασικές αιτιολογίσιμες υπηρεσίες αρχής περιλαμβάνουν:

Η **βάση γνώσεων Satisfiability** είναι το πρόβλημα εάν υπάρχει ένα πρότυπο I του  $\Sigma$ .

Η **έννοια Satisfiability** είναι το πρόβλημα εάν υπάρχει ένα πρότυπο I του  $\Sigma$  στο οποίο  $C^I \neq 0$ .

Το **Subsumption** είναι το πρόβλημα εάν σε κάθε πρότυπο I του  $\Sigma$  έχουμε το  $C^I \subseteq D^I$ .

Ο **έλεγχος περίπτωσης** είναι το πρόβλημα εάν σε κάθε πρότυπο I του  $\Sigma$  έχουμε  $a^I \in C^I$ .

Για παράδειγμα, λαμβάνοντας υπόψη τα προηγούμενα αξιώματα μία DL αιτιολόγηση (με την υποστήριξη τύπου δεδομένων) πρέπει να είστε σε θέση να συμπεράνει ότι Ganesh είναι ένα AdultElephant.

Οι πιο κοινές σύνθετες υπηρεσίες περιλαμβάνουν την ταξινόμηση και την ανάκτηση. Η ταξινόμηση είναι ένα πρόβλημα μια νέα κατηγορία στην κατάλληλη θέση σε μια ταξινομική ιεραρχία για τα ονόματα κατηγορίας αυτό μπορεί να γίνει από τις υποκατηγορίες ελέγχοντας μεταξύ κάθε ονομασμένης κατηγορίας μέσα στην ιεραρχία και τη νέα κατηγορία. Η θέση της νέας κατηγορίας, θα την καλούμε C, στην ιεραρχία θα είναι μεταξύ των πιο συγκεκριμένων ονομασμένων κατηγοριών που εντάσσουν το C και πιο πολύ γενικά ονομασμένες κατηγορίες που το C εντάσσει. Η ταξινόμηση TBox, η οποία υπολογίζει την ταξινομική ιεραρχία των ονομάτων που αναφέρονται σε ένα TBox,

είναι μια πρόσθετη περίπτωση της ταξινόμησης, όπου το T επιλέγεται ως «νέα» έννοια. Η ανάκτηση (ή ερώτηση που απαντά) είναι ένα πρόβλημα καθοριστικό του συνόλου ατόμων που στιγμιαία σε μια δεδομένη κατηγορία αυτό μπορεί να γίνει αφελώς στην περίπτωση που γίνεται έλεγχος μεταξύ κάθε ονομασμένου ατόμου και της δεδομένης κατηγορίας.

### **Οντολογίες και λογικές περιγραφής**

Η οντολογία είναι ένας όρος που δανείζεται από τη φιλοσοφία που αναφέρεται στην επιστήμη της περιγραφής για τα είδη οντοτήτων στον κόσμο και πώς συσχετίζονται. Στην πληροφορική, η οντολογία είναι, γενικά, μια αντιπροσώπευση «μιας κοινής σύλληψης» μιας συγκεκριμένης περιοχής [Gru93a,UG96]. Οι οντολογίες [Gru93a,UG96] έχουν προταθεί για να παρέχουν κοινούς και καθορισμένους όρους και περιορισμούς για να περιγράψουν την έννοια των πόρων Web (συμπεριλαμβανομένου του Web υπηρεσίας).

Μια οντολογία αποτελείται χαρακτηριστικά από μια ιεραρχική περιγραφή των σημαντικών εννοιών στη περιοχή, μαζί με τις περιγραφές των ιδιοτήτων κάθε έννοιας, και περιορισμών για αυτές τις έννοιες και τις ιδιότητες. Μια οντολογία αντιστοιχεί συνήθως σε ένα σύνολο κατηγορίας και ιδιοκτησίας αξιωμάτων στις λογικές περιγραφής. Το λεξιλόγιο σε μια οντολογία μπορεί να εκφραστεί από ονομασμένες κατηγορίες και ιδιότητες. Οι υποθέσεις /οι περιορισμοί υποβάθρου μπορούν να αντιπροσωπευθούν από γενικά αξιώματα κατηγορίας και ιδιοκτησίας. Μερικές φορές, μια οντολογία αντιστοιχεί σε βάση γνώσης DL. Παραδείγματος χάριν, στη γλώσσα οντολογίας Web OWL [BvHH+04], μια οντολογία επίσης περιέχει τις περιπτώσεις σημαντικών κατηγοριών και σχέσεων μεταξύ αυτών των περιπτώσεων, οι οποίες μπορούν να αντιπροσωπευθούν από τα μεμονωμένα αξιώματα DL.

Οι υψηλής ποιότητας οντολογίες εξαρτώνται αποφασιστικά από τη διαθεσιμότητα μιας καθορισμένης με σαφήνεια σημασιολογίας και από τα ισχυρά εργαλεία συλλογισμού [BHS02, Pan04a]. Οι λογικές περιγραφής εξετάζουν και τις δύο αυτές οι ανάγκες οντολογίας. Κατά τρόπο που δεν προκαλεί έκπληξη, οι γνωστές γλώσσες οντολογίας όπως οι OIL, DAML+OIL και OWL χρησιμοποιούν τη πρότυπη θεωρητική σημασιολογία DL.

Μεταξύ αυτών των γλωσσών οντολογίας Web, η OWL είναι ιδιαίτερα σημαντική. Η OWL έχει υιοθετήσει πρότυπα (W3C σύσταση) για την έκφραση των οντολογιών στο Σημασιολογικό Web. Υπάρχουν τρεις υπογλώσσες της OWL: η OWL Lite, η OWL DL και η OWL Full. Σε αυτό το τμήμα, όταν αναφέρουμε την «OWL» θα σημαίνει συνήθως την «OWL DL» επειδή η OWL DL, ισοδυναμεί με τη SHOIN (D+) DL, που είναι η πιο εκφραστική αποφασιστική υπογλώσσα της OWL, ενώ η OWL Lite είναι απλά μια υπογλώσσα της OWL DL και η OWL Full μπορεί να θεωρηθεί ως ανεπιτυχής προσπάθεια στην ενσωμάτωση RDF και OWL DL [Pan04b]. Ο αναγνώστης αναφέρεται στο [PSHH04] για τις λεπτομέρειες της αφηρημένης σύνταξης και της πρότυπης θεωρητικής σημασιολογίας από την OWL DL.

### Περιγραφές λογικής και ανακάλυψη υπηρεσιών

Οι λογικές περιγραφής έχουν διακρίνει τις λογικές ιδιότητες. Υπογραμμίζουν την αποφασιστικότητα από τα βασικά προβλήματα συλλογισμού. Σύγχρονοι αιτιολογητές DL, όπως το FaCT [Hor98], FaCT++, Racer [HM01], DLP [PS99], και Pellet έχουν καταδείξει ότι, ακόμη και με εκφραστικό DLs, ιδιαίτερα βελτιστοποιημένες εφαρμογές μπορεί να παρέχουν την αποδεκτή απόδοση σε ρεαλιστικές εφαρμογές. Με άλλα λόγια, στοχαστικές τεχνικές βελτιστοποίησης ([Hor97,HPS98,HS02,Hor03]) έχουν κινήσει τα όρια της «ευπείθειας» κάπου πολύ κοντά σε EXP-TIME σκληρά, ή χειρότερα ([Don03]).

Συγκεκριμένα, μπορεί να δει που υπολογίζεται τη σχέση υποσυνόλων μεταξύ των εννοιών που είναι ισοδύναμες για να καθορίσει την καθορισμένη σχέση συνυπολογισμού μεταξύ των συνόλων που περιγράφονται από τέτοια έννοιες. Μπορούμε να ξαναγράψουμε τις επίσημες σχέσεις που εισάγονται στο προηγούμενο τμήμα σε επίπεδο από τα υποσύνολα DL των εννοιών, την ισοδυναμία των εννοιών, και την ικανοποίηση έννοιας. Λαμβάνοντας υπόψη τα μετέπειτα κατάσταση μιας αφηρημένης ικανότητας ( $W_{\text{post}}$ ) και ενός στόχου του αιτούντος ( $G_{\text{post}}$ ) διαμορφωμένου σαν έννοιες DL, πρέπει να ελέγξουμε τα εξής:

1. **Exact- Match** εάν  $W_{\text{post}}$  και  $G_{\text{post}}$  είναι ισοδύναμες έννοιες, τότε έχουμε exact-match δηλ. Το  $C_A \equiv C_B$  πρέπει να ελεγχθεί.

2. **Plugin-Match** εάν  $G_{\text{post}}$  είναι μια υποκατηγορία  $W_{\text{post}}$  δηλ.  $W_{\text{post}}$  εντάσσει  $G_{\text{post}}$  τότε έχουμε plugin-match δηλ.  $C_R.C_A$  πρέπει να ελεγχθεί.

3. **Subsumption-Match** εάν  $G_{\text{post}}$  είναι μια έξοχη κατηγορία  $W_{\text{post}}$  δηλ.  $G_{\text{post}}$  εντάσσει  $W_{\text{post}}$ , έχουμε subsumption-match δηλ. το  $C_A.C_R$  πρέπει να ελεγχθεί.

4. **Intersection-Match** εάν η διατομή  $W_{\text{post}}$  και  $G_{\text{post}}$  είναι ικανοποιητική, έχουμε intersection-match δηλ.:  $\neg (C_A \sqcap C_R, \perp)$  πρέπει να ελεγχθεί.

5. **Non-Match** εάν δεν υπάρχει καμία από τις ανωτέρω σχέσεις, έχουμε non-match δηλ.:  $C_A \sqcap C_R, \perp$  πρέπει να ελεγχθεί.

Εάν περιορίσουμε την επιτρεπόμενη εκφραστικότητα για την περιγραφή των αφηρημένων ικανοτήτων και των στόχων του αιτούντος (για την ανακάλυψη) στις γλώσσες DL που οι τρέχοντες αιτίες DL μπορούν αποτελεσματικά να ασχοληθούν, μπορούμε αποτελεσματικά να εκμεταλλευτούμε το συλλογισμό υποκατηγοριών. Ο RACER παρέχει τον αποδοτικό συλλογισμό υποκατηγοριών για SHIQ με τον ελλιπή συλλογισμό για τις συνδέσεις. Η FACT++ υποστηρίζει την SHIF (D). Ο Pellet παρέχει τον υγιή και πλήρη συλλογισμό για το SHIN(D) και SHON (D), και ήχο αλλά ελλιπή για SHOIN (D).

Στο [LH03], ο RACER χρησιμοποιείται για να ταξινομήσει τις υπηρεσίες Web στο T-Box, το οποίο είναι χρονοβόρο αλλά μπορεί να γίνει off-line στην έκδοση του χρόνου. Μόλις ταξινομηθεί το TBox, τα πειραματικά αποτελέσματα δείχνουν ότι ελέγχοντας τη σχέση υποσυνόλων μεταξύ του αιτήματος των χρηστών και των υπηρεσιών του Web στο TBox μπορούν να γίνουν μέσα σε 20 χιλιοστά του δευτερολέπτου. Στη περίπτωση, και όπως παρουσιάστηκε στο σημασιολογικό πρόγραμμα Fred του Web (SWF) ([KSF04]), χρησιμοποιήθηκαν περισσότερο εκφραστικές λογικές (first-order λογική) για την βασισμένη στο σύνολο διαμόρφωση και ένα θεώρημα που αποδεικνύει τη συνολικά βασισμένη ανακάλυψη υπονοεί σε γενικές γραμμές κάθε διαθέσιμη υπηρεσία χωριστά, όπως οι αποδείξεις θεωρήματος δεν βελτιστοποιούνται για αυτόν τον στόχο συλλογισμού. Για ένα μειωμένο σύνολο δοκιμής τεσσάρων διαθέσιμων υπηρεσιών, η απόδειξη στο θεώρημα απαιτείται μεταξύ ενός και δύο δευτερολέπτων για να καθορίσει την ύπαρξη μιας ταιριασμένης υπηρεσίας. Στις περιπτώσεις που τελικά ένας μεγάλος αριθμός από τις υπηρεσίες θα είναι διαθέσιμος στη μηχανή ανακαλύψεων, ένα γρηγορότερο φιλτράρισμα των σχετικών υπηρεσιών πριν αξιολογήσει ένα-ένα τα

ουσιαστικά για να καταστήσει την ανακάλυψη εξελικτική. Αυτοί τα αποτελέσματα σαφώς προτείνουν ότι χρησιμοποιώντας τους αιτιολογητές DL στις υπηρεσίες δεικτών Web στο χρόνο δημοσιεύσεων μπορεί να είναι χρήσιμα, περιορίζοντας την πιο λεπτομερή και την πιο ακριβή συνένωση στο (ιδανικά μικρό) υποσύνολο όλων των διαθέσιμων υπηρεσιών Web. Προσέξτε ότι αν ένα θεώρημα αποδεικνύει ότι θα μπορούσε να είναι επίσης βελτιστοποιημένο για την ταξινόμηση, αλλά θα απαιτούσε μια μεγαλύτερη προσπάθεια, και δεν είναι σαφές εάν η πρόσθετη εκφραστικότητα απαιτείται για τις αφηρημένες ικανότητες.

**Τι να ταξινομήσει;** Ένα από τα σημαντικότερα οφέλη των αιτιολογητών DL είναι ότι μπορούμε να έχουμε τις διαθέσιμες υπηρεσίες Web που ταξινομούνται πριν να πραγματοποιηθεί η διαδικασία ανακαλύψεων, ως DL οι αιτιολογητές βελτιστοποιούνται για αυτόν τον στόχο. Μια άλλη δυνατότητα είναι να ταξινομηθούν οι προκαθορισμένοι στόχοι. Αυτές οι δύο επιλογές έχουν τα ακόλουθα πλεονεκτήματα και τα μειονεκτήματα:

Ø Η χρήση των προκαθορισμένων, τυποποιημένων στόχων αναμένεται στο εννοιολογικό πρότυπό μας. Εάν εμείς περιορίσουμε τέτοιους στόχους που περιγράφονται σε γλώσσα DL, μπορούμε να τους ταξινομήσουμε off-line σε DL αιτιολογητή TBox. Κατά την έκδοση μιας υπηρεσίας Web, η σχέση υποκατηγοριών με τους ταξινομημένους προκαθορισμένους στόχους μπορεί να ελεγχθεί, και ένα wgMediator [RLeditors04] μπορεί να παραχθεί για να συνδέσει την υπηρεσία Web με έναν προκαθορισμένο στόχο που (συνολικά ή μερικώς, ανάλογα με τις υποκατηγορίες της σχέσης που υπολογίζεται) εκπληρώνει. Με αυτό το τρόπο, και η ανακάλυψη στόχου πρέπει να οδηγήσει σε έναν (καθαρισμένο) προκαθορισμένο στόχο, και στον Ιστό η ανακάλυψη υπηρεσιών αναμένεται για να απαιτήσει ότι το ταίριασμα αυτών καθόρισε τους προκαθορισμένους στόχους ενάντια στις δημοσιευμένες υπηρεσίες Web, η ανακάλυψη υπηρεσιών Web μειώνεται στην εξερεύνηση. Οι υπηρεσίες Web συνδέθηκαν μέσω των wgMediators με το χρησιμοποιημένο προκαθορισμένο στόχο.

Ø Μια άλλη επιλογή είναι να ταξινομηθούν οι υπηρεσίες Web όταν δημοσιεύονται. Αναμένεται ότι ο αριθμός διαθέσιμων υπηρεσιών Web θα

είναι αρκετά υψηλότερος από τον αριθμό των προκαθορισμένων στόχων και, επομένως, θα έπρεπε να εξετάσει τα μεγαλύτερα TBoxes και τους χειρότερους χρόνους ταξινόμησης. Εντούτοις, αυτοί οι χρόνοι ταξινόμησης όχι απαραίτητως έχουν επιπτώσεις στο χρόνο για να απαντηθεί ένας εισερχόμενος στόχος. Μια άλλη εκτίμηση είναι ότι σε αυτήν την περίπτωση η σχέση υποκατηγοριών μεταξύ ενός καθαρισμένου προκαθορισμένου στόχου και οι ταξινομημένες υπηρεσίες Web θα έπρεπε να υπολογιστούν για κάθε αίτημα ανακαλύψεων, ενώ στη προηγούμενη υποκατηγορία λύσης ο έλεγχος απαιτείται μόνο μια φορά για κάθε μία Υπηρεσία Web, και μόνο στο χρόνο δημοσιεύσεων. Εντούτοις, παρατηρήστε ότι σε αυτήν την περίπτωση εμείς άμεσα λάβαμε τη σχέση υποκατηγορίας του συγκεκριμένου που καθαρίζεται μεταξύ του προκαθορισμένου στόχου και των υπηρεσιών Web, ενώ στην προηγούμενη περίπτωση η άμεση σχέση μεταξύ εκείνων δεν είναι γνωστή αλλά μόνο η σχέση τους μέσω του προκαθορισμένου στόχου.

Προτείνουμε μετά από τη δεύτερη πορεία, υπολογίζοντας την άμεση σχέση μεταξύ του συγκεκριμένου στόχου του αιτούντος και των αφηρημένων ικανοτήτων των δημοσιευμένων υπηρεσιών θα παράσχουν καλύτερα αποτελέσματα φιλτράροντας μόνο ελέγχοντας τη σχέση αυτή με τον προκαθορισμένο στόχο. Επιπλέον, όπως φαίνεται στο [LH03], που υπολογίζει τη σχέση του στόχου του αιτούντος όσον αφορά ένα ταξινομημένο TBox περιέχοντας τις διαθέσιμες υπηρεσίες που μπορεί να γίνουν σωστά. Παρατηρήστε ότι αυτή η επιλογή είναι στις περιπτώσεις που οι αφηρημένες ικανότητες των διαθέσιμων υπηρεσιών θα αποτελέσουν τη δομημένη περιοχή και όχι ένα σύνολο μετά βίας σχετικών εννοιών.

Προσέξτε ότι η πρόθεση του προμηθευτή δεν μπορεί να διαμορφωθεί άμεσα χρησιμοποιώντας DL. Επομένως, η πρόθεση θα είναι ένας σχολιασμός της έννοιας περιγράφοντας τα αφηρημένα αποτελέσματα της υπηρεσίας, και αυτό θα χρησιμοποιηθεί για να επιλέξει ποιες υπηρεσίες έχουν μια δεδομένη σχέση με το στόχο που θα επιλεγεί.

Στο [Leditors04b] η χρήση του RACER και του Pellet έχει ερευνηθεί για να ταξινομήσει την περίληψη υπηρεσιών και στόχων. Τα αποτελέσματα δείχνουν ότι η

επεξεργασία των μικρότερων δεν είναι πλήρως υποστηριγμένη για το SHOIN DL, το οποίο περιορίζει τη χρήση τους στις αφηρημένες ικανότητες και τους στόχους του αιτούντος για την ανακάλυψη υπηρεσιών. Εντούτοις, δύο πιθανές λύσεις προβλέπονται:

- Απαγορεύοντας τη χρήση των μικρότερων για την ανακάλυψη υπηρεσιών και εξετάζοντας τις τιμές περίπτωσης μόνο στην συνένωση, ή
- παρέχοντας έναν αλγόριθμο για περιπτώσεις στις περιγραφές υπηρεσιών και ζητούμενου στόχου από τις αντίστοιχες έννοιες.

### **OWL-E: Υποστήριξη προσαρμοσμένων τύπων δεδομένων και Κατηγορήματα τύπων δεδομένων στις οντολογίες**

Ένα από τα πιο χρήσιμα χαρακτηριστικά γνωρίσματα των ανωτέρω γλωσσών οντολογίας Web είναι η υποστήριξη τύπων δεδομένων (όπως οι σειρές και οι ακέραιοι αριθμοί) και οι τιμές (όπως οι ακέραιοι αριθμοί 1.2.3 κ.λ.π.). Συγκεκριμένα, κάποιος μπορεί να περιγράψει όχι μόνο τις σχέσεις μεταξύ των όρων, αλλά και τις σχέσεις μεταξύ των όρων και τύπων δεδομένων, π.χ., της ηλικίας ενός ελέφαντα, η τιμή ενός προϊόντος, ημερομηνία μιας διάσκεψης και ο ταχυδρομικός κώδικας ενός εστιατορίου. Είναι αυτά τα είδη σχέσεων «πραγματικού κόσμου», οι οποίες παρέχουν τις χρήσιμες συγκεκριμένες πληροφορίες, εφαρμογές εκείνης της οντολογίας που συχνά απαιτείται.

Οι προσαρμοσμένοι τύποι δεδομένων και τα κατηγορήματα τύπων δεδομένων (όπως>+) είναι πολύ χρήσιμα επειδή είναι συχνά απαραίτητο να επιτραπούν οι χρήστες για να καθορίσουν τους τύπους δεδομένων και τα κατηγορήματά τύπων δεδομένων για τις αιτήσεις τους. Για παράδειγμα, σε μια οντολογία πωλήσεων υπολογιστών, μια απαίτηση υπηρεσιών μπορεί να ζητήσει ένα PC με το μέγεθος μνήμης μεγαλύτερο από 512Mb, με τιμή μονάδων λιγότερο από 700 λίβρες και ημερομηνία παράδοσης νωρίτερα από τις 15/03/2004. Εδώ το «μεγαλύτερο από 512», το «λιγότερο από 700» και το «νωρίτερα από το 15/03/2004» προσαρμόζεται σε τύπους δεδομένων του ακέραιου αριθμού βάσεων τύπων δεδομένων, ακέραιος αριθμός και ημερομηνία, αντίστοιχα. Οι παρόμοιες περιπτώσεις χρήσης προκύπτουν στην περιοχή VTA, όπου ένας πελάτης μπορεί να ρωτήσει για μια πτήση φτηνότερη από 300 ευρώ στις 15/03/2004 και άφιξη στις 20:00.

Η OWL DL έχει έναν ισχυρό περιορισμό στην υποστήριξη τύπων δεδομένων [Pan04a]. Δεν υποστηρίζει κανένα κατηγορημα τύπων δεδομένων με τα αυθαίρετα arities, ούτε προσαρμοσμένους τύπους δεδομένων, για να μην αναφέρουμε προσαρμοσμένα κατηγορήματα τύπων δεδομένων. Για να υπερνικήσει αυτούς τους περιορισμούς, [PH04, Pan04b] προτείνει την OWL-E, ισοδύναμη με τη SHOIQ (G) DL, που είναι μια αποφασιστική επέκταση και της OWL DL και της DAML+OIL, που παρέχει προσαρμοσμένους τύπους δεδομένων και κατηγορήματα στην πραγματικότητα, όλες οι βασικές αιτιολογίσιμες υπηρεσίες της OWL-E είναι αποφασιστικές.

Η OWL-E παρέχει τύπους δεδομένων στις εκφράσεις βασισμένες στην προσέγγιση ομάδα τύπων δεδομένων [Pan04b], που μπορεί να χρησιμοποιηθεί για να αντιπροσωπεύσει προσαρμοσμένους τύπους δεδομένων και κατηγορήματα τύπων δεδομένων. Στο πίνακα παρουσιάζεται το είδος υποστηρίξεων έκφρασης OWL-E τύπων δεδομένων, όπου το  $u$  είναι ένα κατηγορημα τύπων δεδομένων. Το  $\text{URIref}, "s_i" \wedge d_i$  είναι τυπογραφικό λάθος,  $U_1, \dots, U_n$  είναι (ενδεχομένως αρνητικό) μοναδιαίο που υποστηρίζεται από το κατηγορημα  $\text{URIrefs}$ ,  $P, Q$  είναι εκφράσεις τύπων δεδομένων και  $\Phi_G$  είναι το σύνολο υποστηριγμένου κατηγορήματος  $\text{URIrefs}$  σε μια ομάδα  $G$ . Οι τύποι δεδομένων της OWL-E παρέχουν μερικές νέες περιγραφές κατηγοριών, οι οποίες παρατίθεται στον πίνακα, όπου  $T, T_1, \dots, T_N$  είναι ιδιότητες τύπων δεδομένων (όπου  $T_i \not\subseteq T_j, T_j \not\subseteq T_i$



Abstract Syntax	DL Syntax	Semantics
$rdfs:Literal$ $owl:DatatypeBottom$ $u$ a predicate URIref	$\top_D$ $\perp_D$ $u$	$\Delta_D$ $\emptyset$ $u^D$
$not(u)$	$\bar{u}$	if $u \in D_G, \Delta_D \setminus u^D$ if $u \in \Phi_G \setminus D_G, (dom(u))^D \setminus u^D$ if $u \notin \Phi_G, \bigcup_{n>1} (\Delta_D)^n \setminus u^D$
$oneOf("s_1" \dots "s_n")$ $domain(v_1, \dots, v_n)$ $and(P, Q)$ $or(P, Q)$	$\{ "s_1" \dots "s_n" \}$ $[v_1, \dots, v_n]$ $P \wedge Q$ $P \vee Q$	$\{ ("s_1")^D \} \cup \dots \cup \{ ("s_n")^D \}$ $v_1^D \times \dots \times v_n^D$ $P^D \cap Q^D$ $P^D \cup Q^D$

Πίνακας 4.6 : OWL-S εκφράσεις δεδομένων

Abstract Syntax	DL Syntax	Semantics
$restriction(\{T\}$ $someTuplesSatisfy(E))$	$\exists T_1, \dots, T_n.E$	$\{x \in \Delta^I \mid \exists t_1, \dots, t_n. \langle x, t_i \rangle \in T^I \text{ (for all } 1 \leq i \leq n) \wedge \langle t_1, \dots, t_n \rangle \in E^D\}$
$restriction(\{T\}$ $allTuplesSatisfy(E))$	$\forall T_1, \dots, T_n.E$	$\{x \in \Delta^I \mid \forall t_1, \dots, t_n. \langle x, t_i \rangle \in T^I \text{ (for all } 1 \leq i \leq n) \rightarrow \langle t_1, \dots, t_n \rangle \in E^D\}$
$restriction(\{T\}$ $minCardinality(m)$ $someTuplesSatisfy(E))$	$\geq m T_1, \dots, T_n.E$	$\{x \in \Delta^I \mid \#\{\langle t_1, \dots, t_n \rangle \mid \langle x, t_i \rangle \in T^I \text{ (for all } 1 \leq i \leq n) \wedge \langle t_1, \dots, t_n \rangle \in E^D\} \geq m\}$
$restriction(\{T\}$ $maxCardinality(m)$ $someTuplesSatisfy(E))$	$\leq m T_1, \dots, T_n.E$	$\{x \in \Delta^I \mid \#\{\langle t_1, \dots, t_n \rangle \mid \langle x, t_i \rangle \in T^I \text{ (for all } 1 \leq i \leq n) \wedge \langle t_1, \dots, t_n \rangle \in E^D\} \leq m\}$
$restriction(R$ $minCardinality(m)$ $someValuesFrom(C))$	$\geq m R.C$	$\{x \in \Delta^I \mid \#\{y \mid \langle x, y \rangle \in R^I \wedge y \in C^I\} \geq m\}$
$restriction(R$ $maxCardinality(m)$ $someValuesFrom(C))$	$\leq m R.C$	$\{x \in \Delta^I \mid \#\{y \mid \langle x, y \rangle \in R^I \wedge y \in C^I\} \leq m\}$

Πίνακας 4.7 : OWL-S εισαχθείσες περιγραφές κατηγορίας

για το  $1 \leq i \leq j \leq v$ ), το R είναι μια ιδιοκτησία αντικειμένου, το G είναι μια κατηγορία, το E είναι μια έκφραση τύπου δεδομένων ή μια έκφραση URIref τύπου δεδομένων, και το # δείχνει τον αριθμό στοιχείων συνόλου. Παρατηρήστε ότι τα πρώτα τέσσερα είναι τύποι δεδομένων από ομάδες βασισμένες στις περιγραφές κατηγορίας, και τα τελευταία δύο είναι κατάλληλοι περιορισμοί αριθμού.

### Case Study: Αντιστοιχία στην OWL-E

Ας χρησιμοποιήσουμε ένα παράδειγμα για να επεξηγήσουμε πώς χρησιμοποιείτε η OWL-E για την ανακάλυψη υπηρεσιών.

Μπορούμε να χρησιμοποιήσουμε την OWL-E για να περιγράψουμε και τις αφηρημένες υπηρεσίες και τον στόχο του αιτούντα. Ακριβέστερα, μπορούν να αντιπροσωπευθούν ως

περιορισμοί κατηγορίας OWL-E ή κατηγορίας, π.χ. η ικανότητα ότι το μέγεθος μνήμης πρέπει να είναι είτε 256Mb είτε 512Mb, μπορεί να είναι αντιπροσωπευόμενος τύπος δεδομένων μια σχετική έκφραση έννοιας  $\exists \text{memoryUnitSizeInMb. } (=_{256V} =_{512})$ .

Για να αποκτήσει μια περαιτέρω επίγνωση στις ανωτέρω πέντε σχέσεις που ελέγχονται για να καθιερώσει την αντιστοιχία, είναι συχνά χρήσιμο να έχετε μερικά παραδείγματα εργασίας. Υποθέστε, σε ένα σενάριο πώληση υπολογιστών, όπου ένας διαμεσολαβητής θα επιθυμούσε να αγοράσει ένα PC με τις ακόλουθες ικανότητες:

- ο επεξεργαστής πρέπει να είναι Pentium4
- το μέγεθος της μνήμης σε Mb πρέπει να είναι 128
- η τιμή σε Pound πρέπει να είναι λιγότερη από 500.

Αυτό μπορεί να αντιπροσωπευθεί από την ακόλουθη περιγραφή στην κατηγορία OWL-

:

$$C_{R1} \equiv PC \sqcap \exists \text{processor.Pentium4} \sqcap \exists \text{memoryUnitSizeInMb. } =_{[128]}^{\text{int}} \sqcap \exists \text{priceInPound. } <_{[500]}^{\text{int}}$$

<b>Exact match:</b> $C_{A1}$	$\equiv PC \sqcap \exists \text{processor.Pentium4} \sqcap \geq_{[128]}^{\text{int}} \text{memoryUnitSizeInMb.} \sqcap \exists \text{priceInPound. } <_{[500]}^{\text{int}}$
<b>PuIn match:</b> $C_{A2}$	$\equiv PC \sqcap \exists \text{processor.Pentium4} \sqcap \geq_{[128]}^{\text{int}} \text{memoryUnitSizeInMb.} (=_{[128]}^{\text{int}} \vee =_{[256]}^{\text{int}}) \sqcap \exists \text{priceInPound. } <_{[700]}^{\text{int}}$
<b>Subsume match:</b> $C_{A3}$	$\equiv PC \sqcap \exists \text{processor.Pentium4} \sqcap \geq_{[128]}^{\text{int}} \text{memoryUnitSizeInMb.} \sqcap \exists \text{priceInPound. } <_{[500]}^{\text{int}} \sqcap \forall \text{orderDate. } ((\geq_{[20040901]}^{\text{int}} \wedge \leq_{[20040931]}^{\text{int}}) \vee (\geq_{[20040901]}^{\text{int}} \wedge \leq_{[20040930]}^{\text{int}})) \sqcap \forall \text{orderDate, deliverDate. } <_{[ ]}^{\text{int}}$
<b>Intersect. match:</b> $C_{A4}$	$\equiv PC \sqcap \exists \text{processor.Pentium4} \sqcap \geq_{[128]}^{\text{int}} \text{memoryUnitSizeInMb.} \sqcap \exists \text{priceInPound. } >_{[400]}^{\text{int}} \sqcap \leq_{[ ]}^{\text{int}} \text{priceInPound.} \sqcap \exists \text{CPUFreqInGHz. } =_{[2.8]}^{\text{real}}$
<b>Disjoint match:</b> $C_{A5}$	$\equiv PC \sqcap \exists \text{processor.Pentium4} \sqcap \geq_{[256]}^{\text{int}} \text{memoryUnitSizeInMb.} \sqcap \exists \text{priceInPound. } <_{[500]}^{\text{int}} \sqcap \leq_{[256]}^{\text{int}} \text{memoryUnitSizeInMb.} \sqcap \forall \text{HardDiskBrand, USBKeyBrand. } =^{\text{str}}$

Σχήμα 4.8 Παράδειγμα ταιριάσματος Διαφημίσεων

Το σχήμα παρουσιάζει πέντε παραδείγματα σχέσεων για το  $C_{R1}$  και τις διαφημισμένες υπηρεσίες. Σε ρεαλιστικές καταστάσεις, δεν είναι εύκολο να υπάρχει exact-match από όταν οι διαφημίσεις μπορούν να παρέχουν πιο γενικές και πιο ειδικές πληροφορίες.

### **Συζήτηση: Συλλογισμός με την OWL-E**

Σε αυτό το τμήμα, συζητάμε εν συντομία πώς να παρέχουμε τις πρακτικές διαδικασίες απόφασης για την OWL-E, το SHOIQ (G) DL. Το [Pan04a] δείχνει ότι το SHOIQ (G) DL είναι αποφασιστικό, και ότι εάν έχουμε έναν αλγόριθμο εικόνων για το SHOIQ, μπορούμε εύκολα να το αναβαθμίσουμε σε SHOIQ (G). Το πρόβλημα είναι ότι, στη καλύτερη γνώσης μας, δεν υπάρχουν αλγόριθμοι εικόνων που να δημοσιεύονται για να χειριστούν το SHOIQ DL.

**Προσέγγιση «Divide and Conquer»(Προσέγγιση διαιρέστε και κατακτήστε)** Το κίνητρο αυτής της προσέγγισης είναι ότι στην εφαρμογή οντολογίας μπορεί συχνά να μη χρειαστούμε το πλήρες SHOIQ (G) DL. Επομένως, άντ' αυτού από την παροχή μιας διαδικασίας απόφασης για το πλήρες SHOIQ (G) DL, μπορούμε να παρέχουμε την απόφαση διαδικασίας για ένα σύνολο υπογλωσσών του SHOIQ(G)DL έτσι ώστε να μπορούν να καλύψουν όλα τα χαρακτηριστικά γνωρίσματα της SHOIQ(G)DL. Τώρα πρέπει να αποφασίσουμε πιο σύνολο υπογλωσσών πρέπει να εξετάσουμε. Από την SHOQ(D)DL έχει υποστηριχτεί για να είναι χρήσιμο στα πλαίσια του Semantic Web [HS01], και την SHIQ DL είναι η υποστήριξη από την γλώσσα οντολογίας Web OIL και εφαρμόζεται στα δημοφιλή συστήματα DL όπως το FaCT[Hor98] και το RACER[HM01], μια πιθανή επιλογή θα ήταν το ακόλουθο σύνολο υπογλωσσών {SHOQ(G), SHIQ(G), SHIO(G)} SHOIQ(G). Εικόνες αλγορίθμων για τις SHOQ(G), SHIQ(G) και SHIO(G) DLs παρουσιάζονται στο [Pan04a]. Μεταξύ τους, ο ένας για την SHIQ (G) έχει εφαρμοστεί εκτεταμένα για τη FaCT αιτιολογία.

**FOL προσέγγιση** Αυτή η προσέγγιση είναι βασισμένη στην παρατήρηση ότι η SHOIQ DL μπορεί να μεταφράζεται στο  $C^2$  τεμάχιο της first-order λογικής(FOL). Επομένως, οι διαδικασίες για την απόφαση  $C^2$  μπορούν να χρησιμοποιηθούν όπως αυτά για την SHOIQ. Για να υποστηρίξει τη SHOIQ (G) DL, πρέπει να επεκτείνουμε τη διαδικασία απόφασης για  $C^2$ , και να υποστηρίξουμε επίσης τους τύπους δεδομένων. Στο καλύτερο

μέρος της γνώσης μας, δεν υπάρχει καμία τέτοια δημοσιευμένη διαδικασία απόφασης για την επέκταση τύπων δεδομένων  $C^2$  δεν υπάρχει κανένα δημοσιευμένο θεώρημα που να αποδεικνύει FOL που να εφαρμόζετε στις διαδικασίες απόφασης  $C^2$ .

Παρατηρήστε ότι σε αυτό το τμήμα δεν έχουμε εισαγάγει το σχολιασμό των ληξιαρχείων όπως προτείνεται για τους METEOR-S(δείτε σε προηγούμενη παράγραφο). Εντούτοις, αυτός ο σχολιασμός μπορεί να είναι χρήσιμος σε επίπεδο ανακαλύψεων, όταν μπορούν οι αφηρημένες υπηρεσίες να αφαιρεθούν περαιτέρω σε μια κοινή περιοχή υπηρεσίας και επομένως ομαδοποιημένη στα κατάλληλα ληξιαρχεία. Αν και αυτό μπορεί να είναι μια ενδιαφέρουσα τεχνική για να βελτιωθεί η αποδοτικότητα, που παρέχει ένα πρόσθετο φίλτρο στο όριο της διαδικασίας ανακαλύψεων υπηρεσιών που προτείνεται σε ένα μικρότερο υποσύνολο των υπηρεσιών, αυτό δεν είναι στον πυρήνα του προβλήματος ανακαλύψεων έτσι δεν θα ασχοληθούμε περισσότερο με αυτό το ζήτημα.

#### **4.6 Συνενώσεις υπηρεσιών στον αυτοματοποιημένο Ιστό**

Στο προηγούμενο τμήμα έχουμε μελετήσει πώς να διαμορφώσουμε και πώς να ταιριάζουμε με τις ικανότητες της αφηρημένη(abstract) υπηρεσίας και τους στόχους του αιτούντος. Σε εκείνο το επίπεδο, ενδιαφερθήκαμε για έναν αφηρημένο χαρακτηρισμό από τις υπηρεσίες που μπορούν να παρασχεθούν, χωρίς εξέταση της ακριβούς περιγραφής από τις συγκεκριμένες υπηρεσίες και τη σχέση μεταξύ των διαθέσιμων πληροφοριών από τον αιτούντα και της παρεχόμενης υπηρεσίας.

Εντούτοις, στη φάση της συνένωσης μια συγκεκριμένη υπηρεσία που συμφωνείται και που παραδίδεται τελικά πρέπει να βρεθεί και, επομένως, οι ακριβείς λεπτομέρειες τέτοιας υπηρεσίας πρέπει να έχουν (άμεσα ή έμμεσα) συλληφθεί, συμπεριλαμβάνων των πληροφοριών που πρέπει να παρασχεθούν από τη διαταγή του αιτούντα για να αποκτηθεί η επιθυμητή υπηρεσία.

Παρακάτω, θα συζητήσουμε την επέκταση του προηγούμενου προτύπου μας που συλλαμβάνει τη σχέση της ικανότητας της συνένωσης μεταξύ της εισαγωγής στην υπηρεσία και των αποτελεσμάτων υπό τον όρο ότι, ο στόχος του αιτούντος και μια τέτοια ικανότητα αντιστοιχούνται.

Χρησιμοποιώντας τους απλούς Semantic σχολιασμούς για μια υπηρεσία, όπως περιγράφηκε στο προηγούμενο το τμήμα, προσθέτει στη μηχανή επεξεργάσιμες

σημασιολογικές πληροφορίες στις περιγραφές υπηρεσιών που επιτρέπει σε έναν μηχανισμό ανακαλύψεων να εκμεταλλευτεί αυτήν την σημασιολογία κατά τη διάρκεια της διαδικασίας ανακαλύψεων και να παραδώσει τα αποτελέσματα με υψηλή ακρίβεια και ανάκληση.

Εν τούτοις, το είδος σημασιολογικών πληροφοριών που μπορούν να εκφραστούν σε εκείνη την προσέγγιση είναι περιορισμένος w.r.t. οι λεπτομέρειες των συγκεκριμένων υπηρεσιών που μπορούν να συμφωνηθούν και να συμβληθούν, όπως αυτό αντιπροσωπεύεται, στη γενική περίπτωση, είναι απλώς μια αφαίρεση τέτοιων συγκεκριμένων υπηρεσιών.

Πιο κάτω δείχνουμε πώς μπορούμε να επεκτείνουμε τη συνολική βασισμένη προσέγγιση διαμόρφωσης που συζητείται σε προηγούμενη παράγραφο στην κατεύθυνση της συνένωσης υπηρεσιών. Επιπλέον, θα συζητήσουμε μια προσέγγιση βασισμένη στη λογική συναλλαγής [BK98], που είναι μια επέκταση της first-order λογικής που επιτρέπει να διευκρινίσουμε τη δυναμική των λογικών θεωριών (ή των βάσεων γνώσης) με έναν δηλωτικό τρόπο. Εδώ, η κατάσταση του κόσμου αντιπροσωπεύεται από μια λογική θεωρία και από την παροχή από τις αλλαγές υπηρεσιών της κατάσταση του κόσμου και του διαστήματος πληροφοριών, που αυτό οδηγεί μια αναπροσαρμογή της λογικής θεωρίας. Η περιγεγραμμένη προσέγγιση μπορεί να εφαρμόζεται χρησιμοποιώντας το σύστημα FLORA-2 [KLP<sup>+</sup>04].

### **Μια συνολικά βασισμένη προσέγγιση διαμόρφωσης για τις πλούσιες περιγραφές υπηρεσιών**

**Το άτυπο πρότυπο νέων υπηρεσιών.** Η παροχή υπηρεσίας παράγει (w.r.t. το σύνολο τιμών εισαγωγής) ορισμένες πληροφορίες ως παραγωγή και επιτυγχάνει ορισμένα αποτελέσματα στη κατάσταση του κόσμου. Και η παραγωγή και η επίδραση μπορούν να θεωρηθούν ως αντικείμενα που μπορούν να ενσωματώνονται σε κάποια περιοχή οντολογιών.

Μέχρι τώρα αγνοήσαμε τις εισαγωγές και τη σχέση τους με τις εξαγωγές και τα αποτελέσματα της παροχής υπηρεσιών Web. Εντούτοις, η εξέταση των συγκεκριμένων υπηρεσιών, θα εξαρτηθεί από τις παρεχόμενες τιμές εισαγωγής. Ως εκ τούτου, μια ικανότητα συνένωσης μπορεί να περιγραφεί από τα σύνολα εξαγωγών και αποτελέσματα

για τις συγκεκριμένες τιμές εισαγωγής. Επιπλέον, οι πληροφορίες που ο αιτών είναι σε θέση να παρέχει, δειγμένος από  $G_{input}$  θα διαδραματίσει έναν ρόλο στον καθορισμό εάν μια συγκεκριμένη υπηρεσία μπορεί να συμφωνείται και να συμβάλλεται. Τέλος, η ικανότητα συνένωσης μιας δεδομένης υπηρεσίας μέσα στη γενική περίπτωση συλλαμβάνει μερικές λεπτομέρειες που δεν εξετάστηκαν προηγουμένως και που περιλαμβάνει την πραγματική επικοινωνία με τον προμηθευτή π.χ. που ελέγχει εάν ζητούμενες πληροφορίες πτήσης μπορούν πραγματικά να ορίζονται δηλ. εάν μια συγκεκριμένη υπηρεσία που εκπληρώνει το στόχος του αιτούντος μπορεί να συμφωνηθεί από τον προμηθευτή και, επομένως, να παρασχεθεί.

Επιπλέον, μπορούμε να εμπλουτίσουμε το σύνολο ταιριάζοντας τις έννοιες που δίνονται σε προηγούμενη παράγραφο με μια ορθογώνια διάσταση: μπορούμε να εκφράσουμε ότι μπορούμε να ικανοποιήσουμε μια ιδιαίτερη έννοια ταιριάζοντας με  $wrt.$  μια ενιαία συγκεκριμένης υπηρεσία καθώς επίσης και  $wrt.$  ένα αυθαίρετο αριθμό συγκεκριμένων υπηρεσιών. Αυτό οδηγεί σε πρόσθετο ταίριασμα με έννοιες που συλλαμβάνουν την πρόσθετη σημασιολογία του στόχου του αιτούντος.

Ας επεξηγήσουμε τη διαφορά με ένα απλό παράδειγμα. Φανταστείτε τον ακόλουθο (άτυπο) στόχο του αιτούντος:

G: «Θέλω να ξέρω όλες τις πτήσεις από το Ίνσμπρουκ και το Σάλτςμπουργκ στη Μαδρίτη στις 28 Δεκεμβρίου 2004»

και μια υπηρεσία Web με την ακόλουθη (άτυπη) ικανότητα

W: «Η υπηρεσία παρέχει τις πληροφορίες για όλες τις πτήσεις από οποιαδήποτε θέση από την Αυστρία στη Μαδρίτη, κατά οποιαδήποτε ημερομηνία. Εντούτοις, παρέχει τις πληροφορίες για ενιαία περιήγηση π.χ. Ίνσμπρουκ-Μαδρίτη σε ένα συγκεκριμένο χρόνο»

Επομένως, μια ενιαία συγκεκριμένη υπηρεσία δεν μπορεί να εκπληρώσει το στόχο του αιτούντος, αλλά και τις δύο υπηρεσίες (μία για την περιήγηση Ίνσμπρουκ-Μαδρίτη και μία για την περιήγηση Σάλτςμπουργκ-Μαδρίτη) μπορεί να συμφωνηθεί και να εκπληρώνουν μαζί το στόχο του αιτούντος. Αυτές οι δύο υπηρεσίες αντιστοιχούν στις διαφορετικές πληροφορίες εισαγωγής που παρέχονται από τον αιτούντα. Αυτό μπορεί να αντιμετωπιστεί ως απλή μορφή σύνθεσης, αλλά μπορεί ακόμα να συλληφθεί στο πλαίσιο μιας συνένωσης και στον καθορισμό των επίσημων υποχρεώσεων απόδειξης που πρέπει

να ελεγχθούν για να καθορίσουν εάν οι συγκεκριμένες υπηρεσίες που εκπληρώνουν το στόχο μπορούν να συμβληθούν.

Στις ακόλουθες παραγράφους θα δείξουμε πώς να τυποποιήσουμε την εκτεταμένη υπηρεσία και το αντίστοιχο ταίριασμα εννοιών.

**Τυποποίηση του εκτεταμένου προτύπου υπηρεσιών.** Αντί της χρησιμοποίησης ενός μοναδιαίου  $W_{\text{post}}(x)$  για την περιγραφή της ικανότητας συνένωσης μιας υπηρεσίας Web  $W$ , πρέπει να εκφράσουμε την εξάρτηση από τις υπηρεσίες που μπορούν να παραδοθούν στη συγκεκριμένη εισαγωγή που απαιτήθηκαν για την παροχή τέτοιων υπηρεσιών  $i_1, \dots, i_n$ .

Ας αφήσουμε το  $W$  να είναι υπηρεσία Web με τις παραμέτρους εισαγωγής  $i_1, \dots, i_n$ , έπειτα τυποποιούμε την ικανότητα της συνένωσης της υπηρεσίας ως εξής:

$$W: \forall x, i_1, \dots, i_n. (ws(x, i_1, \dots, i_n) \leftrightarrow W_{\text{pre}}(i_1, \dots, i_n) \wedge W_{\text{post}}(i_1, \dots, i_n, x))$$

Όπου  $W_{\text{pre}}(i_1, \dots, i_n)$  είναι ένας αυθαίρετος first-order τύπος που περιγράφει τις προϋποθέσεις της υπηρεσία Web και το  $W_{\text{post}}(i_1, \dots, i_n)$  είναι ένας αυθαίρετος πρωταρχικός τύπος που περιγράφει τις μετέπειτα καταστάσεις του. Στο  $W_{\text{post}}(i_1, \dots, i_n, x)$  η μεταβλητή  $x$  αναφέρεται στην αξία παραγωγής ως αποτέλεσμα παροχής υπηρεσιών. Το καθορισμένο κατηγορημα  $ws(x, i_1, \dots, i_n)$  έχει μια φυσική ερμηνεία: Η αξία  $x$  θα παραδοθεί από την υπηρεσία Web  $W$  για τις τιμές εισαγωγής  $i_1, \dots, i_n$ .

**Προσαρμογή των ταιριασμένων εννοιών.** Δεδομένου ότι προσαρμόσαμε τον τρόπο περιγραφής υπηρεσιών για τις Συνενώσεις, πρέπει να προσαρμόσουμε τα επίσημα κριτήρια που χρησιμοποιούνται για να καθορίσουν μια αντιστοιχία. Πιο κάτω θα δείξουμε πώς να προσαρμόσουμε τις ενιαίες σχέσεις αναλόγως και θα δώσουμε έναν καθορισμό για την περίπτωση στην οποία εξετάζουμε μόνο τις ενιαίες συγκεκριμένες υπηρεσίες όπως την περίπτωση για τα σύνολα συγκεκριμένων υπηρεσιών.

Οι εισαγωγές που εξετάζονται κατωτέρω στις διαμορφώσεις πρέπει να παρασχεθούν από τον αιτούντα δηλ.  $i_1, \dots, i_n \in G_{\text{input}}$

- **Exact-Match**  $(W \equiv_{\mathcal{O}}^1 G, W \equiv_{\mathcal{O}}^+ G)$ .

Τυπικά, η σχέση που πρέπει να αποδείξουμε εάν περιοριζόμαστε σε μια ενιαία υπηρεσία είναι η ακόλουθη:

$$W, G, \mathcal{O} \models \exists i_1, \dots, i_n. (\forall x. (g(x) \leftrightarrow ws(x, i_1 \dots i_n)))$$

όπου το  $W$  είναι ο καθορισμός της ικανότητας συνένωσης υπηρεσιών, το  $G$  είναι ο καθορισμός του στόχου του αιτούντος, και το  $\mathcal{O}$  είναι ένα σύνολο οντολογιών στο οποίο και οι δύο περιγραφές αναφέρονται. Σε αυτήν την περίπτωση γράφουμε το  $W \equiv_{\mathcal{O}}^1 G$  για να δείξει αυτό το ιδιαίτερο είδος σχέσης.

Για τις πολλαπλάσιες συγκεκριμένες υπηρεσίες άντ' αυτού θα πρέπει να αποδείξουμε τα εξής:

$$W, G, \mathcal{O} \models \forall x. (\exists i_1, \dots, i_n. (g(x) \leftrightarrow ws(x, i_1 \dots i_n)))$$

Σε αυτήν την περίπτωση γράφουμε το  $W \equiv_{\mathcal{O}}^+ G$  για να δείξουμε αυτό το ιδιαίτερο είδος σχέσης.

- **Subsumption-Match**  $(W \sqsubseteq_{\mathcal{O}}^1 G, W \sqsubseteq_{\mathcal{O}}^+ G)$ .

Τυπικά, η σχέση που πρέπει να αποδείξουμε εάν περιοριζόμαστε σε μια ενιαία πολλαπλή υπηρεσία είναι η ακόλουθη:

$$W, G, \mathcal{O} \models \exists i_1, \dots, i_n. (\forall x. (g(x) \leftarrow ws(x, i_1 \dots i_n)))$$

Τον δείχνουμε από το  $W \sqsubseteq_{\mathcal{O}}^1 G$ .

Και για τις πολλαπλάσιες συγκεκριμένες υπηρεσίες:

$$W, G, \mathcal{O} \models \forall x. (\exists i_1, \dots, i_n. (g(x) \leftarrow ws(x, i_1 \dots i_n)))$$

Που δείχνεται από το  $W \sqsubseteq_{\mathcal{O}}^+ G$ .

- **Plugin-Match**  $(W \supseteq_{\mathcal{O}}^1 G, W \supseteq_{\mathcal{O}}^+ G)$ .



Τυπικά, η σχέση που πρέπει να αποδείξουμε εάν περιοριζόμαστε σε μια ενιαία πολλαπλή υπηρεσία είναι η ακόλουθη:

$$\mathcal{W}, \mathcal{G}, \mathcal{O} \models \exists i_1, \dots, i_n. (\forall x. (g(x) \rightarrow ws(x, i_1 \dots i_n)))$$

Δειγμένη από το  $\mathcal{W} \sqsupseteq_{\mathcal{O}}^+ \mathcal{G}$ .

Για τις πολλαπλάσιες συγκεκριμένες υπηρεσίες πρέπει να αποδείξουμε τα εξής:

$$\mathcal{W}, \mathcal{G}, \mathcal{O} \models \forall x. (\exists i_1, \dots, i_n. (g(x) \rightarrow ws(x, i_1 \dots i_n)))$$

Που γράφεται  $\mathcal{W} \sqsupseteq_{\mathcal{O}}^+ \mathcal{G}$ .

Παρατηρήστε ότι οι ασυμβίβαστοι ορισμοί, και στη Plugin και στην subsumption-match μπορούν να εξεταστούν με τον ίδιο τρόπο που συζητήσαμε στην προηγούμενη παράγραφο.

- **Intersection-match**  $(\mathcal{W} \sqcap_{\mathcal{O}}^- \mathcal{G}, \mathcal{W} \sqcap_{\mathcal{O}}^+ \mathcal{G})$ .

Η σχέση που πρέπει να αποδείξουμε για μια ενιαία συγκεκριμένη υπηρεσία είναι:

$$\mathcal{W}, \mathcal{G}, \mathcal{O} \models \exists i_1, \dots, i_n. (\exists x. (g(x) \wedge ws(x, i_1 \dots i_n)))$$

Που γράφεται  $\mathcal{W} \sqcap_{\mathcal{O}}^1 \mathcal{G}$ .

Για τις πολλαπλάσιες συγκεκριμένες υπηρεσίες πρέπει να αποδείξουμε τα εξής:

$$\mathcal{W}, \mathcal{G}, \mathcal{O} \models \exists x. (\exists i_1, \dots, i_n. (g(x) \wedge ws(x, i_1 \dots i_n)))$$

Που δείχνεται από το  $\mathcal{W} \sqcap_{\mathcal{O}}^+ \mathcal{G}$ .

**Έννοια της αντιστοιχίας.** Όπως συζητείται για την ανακάλυψη υπηρεσιών, οι επίσημες ανωτέρω σχέσεις δεν αντιπροσωπεύουν την έννοια της αντιστοιχίας που αναμένουμε αλλά τη σχέση που πρέπει να ελεγχθεί για να καθιερώσει μια δεδομένη σχέση αντιστοιχιών. Στο σχήμα οι σχέσεις που πρέπει να ελεγχθούν για τις διαφορετικές έννοιες της αντιστοιχίας παρουσιάστηκαν. Παρατηρήστε ότι οι σχέσεις που πρέπει να ελέγχονται για την συνένωση είναι ουσιαστικά οι ίδιες αλλά εκτεταμένες με την

εκτίμηση από την εισαγωγή. Για παράδειγμα, για τον έλεγχο μιας τέλειαν αντιστοιχίας εάν και ο αιτών και ο προμηθευτής έχουν μια καθολική πρόθεση, θα πρέπει να ελέγξουμε τις σχέσεις Plug-in ή τις Exact-match.

**Εισαγωγή αιτούντος.** Παρατηρείστε ότι η εισαγωγή που περιέλαβε τις υποχρεώσεις απόδειξης ανωτέρω πρέπει να παρέχεται από τον αιτούντα. Αυτό δεν επιβάλλει ότι ο αιτών να πρέπει να απαριθμήσει για κάθε στόχο όλες τις πληροφορίες που έχει διαθέσιμες, αλλά μπορεί παραδείγματος χάριν να προσφέρει μια υπηρεσία που παρέχει τις διαθέσιμες πληροφορίες του για τη ζήτηση. Επιπλέον, κάποιες πληροφορίες εισαγωγής μπορούν αυτόματα να εξαχθούν από την περιγραφή του στόχου π.χ. εάν ο αιτών θέλει να πετάξει από το Ίνσμπρουκ στη Μαδρίτη ήδη ξέρουμε ότι μπορεί να παρέχει το Ίνσμπρουκ ως αναχώρηση θέσης και τη Μαδρίτη ως θέση άφιξης. Εντούτοις, πως αυτές οι πληροφορίες κάνουν διαθέσιμη τη διαδικασία στην συνένωση είναι πέρα από το παραδοτέο πεδίο, και υποθέτουμε ότι θα είναι διαθέσιμο με κάποιο τρόπο κατά τη διάρκεια της φάσης συνένωσης.

**Ένα παράδειγμα.** Για παράδειγμα, εξετάστε το στόχο  $G$  και την άτυπη ικανότητα συνένωσης της υπηρεσίας  $W$  που δίνεται ανωτέρω. Επιπλέον, υποθέστε ότι η υπηρεσία Web απαιτεί την υπηκοότητα και τέτοια υπηκοότητα πρέπει να είναι από οποιαδήποτε ευρωπαϊκή χώρα, και ο αιτών μπορεί να παρέχει τη θέση έναρξης, τη θέση τελών, την ημερομηνία και την υπηκοότητα:

$$G : \forall x. ((\exists s \text{ flight}(x, s, \text{madrid}, \text{Dec282004}) \wedge (s = \text{innsbruck} \vee s = \text{salzburg})) \leftrightarrow G_{\text{post}}(x))$$

$$\text{startlocation}(\text{innsbruck}), \text{startlocation}(\text{salzburg}), \text{endlocation}(\text{madrid}), \\ \text{date}(\text{Dec282004}), \text{nationality}(\text{austrian}) \in G_{\text{input}}$$

$$W : \forall x, i_1, i_2, i_3, i_4. (ws(x, i_1, i_2, i_3, i_4) \leftrightarrow \\ (\text{startlocation}(i_1) \wedge \text{in}(i_1, \text{austria}) \wedge \text{endlocation}(i_2) \wedge i_2 = \text{madrid} \\ \wedge \text{date}(i_3) \wedge \text{nationality}(i_4) \wedge \text{european}(i_4)) \\ \wedge \text{flight}(x, i_1, i_2, i_3) \wedge \text{availableinfo}(x))$$

όπου το  $\text{availableinfo}(X)$  είναι ένα κατηγορήμα που περιλαμβάνει τον έλεγχο εάν πληροφορίες για τη πτήση  $X$  είναι διαθέσιμες στη βάση γνώσεων των προμηθευτών.

Δεδομένου ότι και ο αιτών και ο προμηθευτής έχουν καθολικές προθέσεις, μπορούμε να έχουμε ανάμειξη αντιστοιχιών πολλαπλάσιων στις συγκεκριμένες υπηρεσίες εάν η επίσημη σχέση  $\mathcal{W} \sqsupseteq_{\mathcal{O}}^+ \mathcal{G}$  εκπληρώνεται, το οποίο συμβαίνει εάν το  $\text{availableinfo}$  κατηγορήματος  $(x)$  αξιολογηθεί αληθινό. Εντούτοις, εάν περιοριζόμαστε σε ενιαίες συγκεκριμένες υπηρεσίες, θα είμαστε σε θέση μόνο να αποδείξουμε τη σχέση  $\mathcal{W} \sqsupseteq_{\mathcal{O}}^1 \mathcal{G}$  και, επομένως, μόνο μια μερική αντιστοιχία θα είναι δυνατή. Συμπερασματικά, εάν το  $\text{availableinfo}(x)$  είναι αληθινό, θα συμφωνήσουμε με το να συμβληθούν δύο συγκεκριμένες υπηρεσίες (μια για την περιήγηση Ίνσμπρουκ- Μαδρίτη και μια για την περιήγηση Σάλτζμπουργκ-Μαδρίτη), και ειδάλλως θα συμφωνήσουμε μόνο πάνω σε μια ενιαία συγκεκριμένη υπηρεσία που εκπληρώνει μερικώς το στόχο (για μόνο μια από τις περιηγήσεις).

Παρατηρείστε ότι η αξιολόγηση του  $\text{availableinfo}$  κατηγορήματος  $(X)$  θα απαιτήσει την πρόσβαση στη βάση γνώσεων προμηθευτών. Εντούτοις, και όπως δηλώνεται πριν, εξετάζουμε αυτήν την διαδικασία να είναι διαφανής σε μας δηλ. ακριβώς απαιτούμε ότι παίρνουμε μια αξία αλήθειας από το κατηγορήμα αλλά δεν μας ενδιαφέρει για το εάν η επικοινωνία με τον προμηθευτή εκτελείται πραγματικά. Αυτό θα είναι θέμα της μελλοντικής εργασίας.

### **Ανακάλυψη με τη λογική συναλλαγής**

Η σχετική προσέγγιση ανωτέρω δεν περιορίζει την εκφραστικότητα που επιτρέπεται και απαιτεί τη χρήση ενός θεωρήματος απόδειξης, το οποίο δεν είναι σαφές για τις αυθαίρετες ικανότητες και τους στόχους.

Προκειμένου να υπερνικηθούν αυτά τα προβλήματα, προτείνουμε τη χρήση της λογικής συναλλαγής που διαμορφώνουν οι υποχρεώσεις απόδειξης για την συνένωση υπηρεσιών. Η λογική συναλλαγής είναι μια επέκταση του κατηγορήματος που ο υπολογισμός του αποτελεί τη λογική βάση για τις αλλαγές κατάστασης στα προγράμματα λογικής και στις βάσεις δεδομένων. Έρχεται με μια πρότυπη θεωρία και την υγιή και

πλήρη θεωρία απόδειξης. Εντυπωσιακά, η απόδειξη όχι μόνο ελέγχει τα προγράμματα αλλά και τα εκτελεί. Ενδιαφέροντα χαρακτηριστικά γνωρίσματα για μας είναι οι υποθετικές αναπροσαρμογές, οι περιορισμοί στην εκτέλεση συναλλαγής, και οι μαζικές αναπροσαρμογές [BK98].

**Υποχρεώσεις απόδειξης** Εάν η ικανότητα συνένωσης μιας υπηρεσίας διαμορφώνει τη σχέση μεταξύ της εισαγωγή του και των αποτελεσμάτων του δηλ. εάν οι μετέπειτα καταστάσεις και τα αποτελέσματα που παραδίδονται από την υπηρεσία είναι εξαρτώμενα από τις πληροφορίες εισαγωγής που τίθενται στην διάθεση του από τον αιτούντα, μπορούμε να ελέγξουμε εάν οι δεδομένες πληροφορίες εισαγωγής θα οδηγήσουν στις μετέπειτα καταστάσεις και τα αποτελέσματα που τις ικανοποιούν και τα αποτελέσματα που ζητούνται στο στόχο.

Για αυτόν το λόγο, πρέπει αρχικά να ελέγξουμε εάν υπάρχει η εισαγωγή πληροφοριών  $Input$  διαθέσιμες στην υπηρεσία δηλ.  $Input.G_{input}$  που ικανοποιούν τις προϋποθέσεις ( $W_{pre}$ ) της λαμβάνοντας υπόψη την υπηρεσία  $W$ .

Εάν τα ανωτέρω ισχύουν, μπορούμε υποθετικά να υποθέσουμε ότι οι μετέπειτα καταστάσεις και τα αποτελέσματα για τις Υπηρεσίες Web συντηρούν αυτήν την εισαγωγή ( $W_{post}(Input)$  and  $W_{eff}(Input)$ ) ισχύει. Σε αυτές τις περιπτώσεις, έχουμε τα αποτελέσματα και τις μετέπειτα καταστάσεις των διαθέσιμων υπονήφιων υπηρεσιών; εάν ο στόχος των μετέπειτα καταστάσεων και αποτελεσμάτων σε αυτή τη περίπτωση ισχύει, σημαίνει ότι η υπηρεσία Web παρέχει τα επιθυμητά αποτελέσματα για τη διαθέσιμη εισαγωγή και, επομένως η συγκεκριμένη αντιστοιχία υπηρεσιών σε αυτήν την εισαγωγή μπορεί να συμφωνηθεί και να συμβληθεί.

Αυτό μπορεί να τυποποιηθεί από την ακόλουθη υποχρέωση απόδειξης χρησιμοποιώντας τη λογική συναλλαγής, όπου  $\mathcal{V}$  είναι ο υποθετικός χειριστής,  $\mathbf{V}$  είναι ο χειριστής ακολουθίας, και το  $O$  είναι το σύνολο περιοχών οντολογίας του στόχου του αιτούντος ( $G$ ) και της ικανότητας συνένωσης από την υπηρεσία  $W$  που αναφέρετε σε:

$$O, W, G \models \exists Input \\ \diamond(W_{pre}(Input) \otimes insert\{W_{post}(Input), W_{eff}(Input)\}) \\ \otimes G_{post} \wedge G_{eff})$$

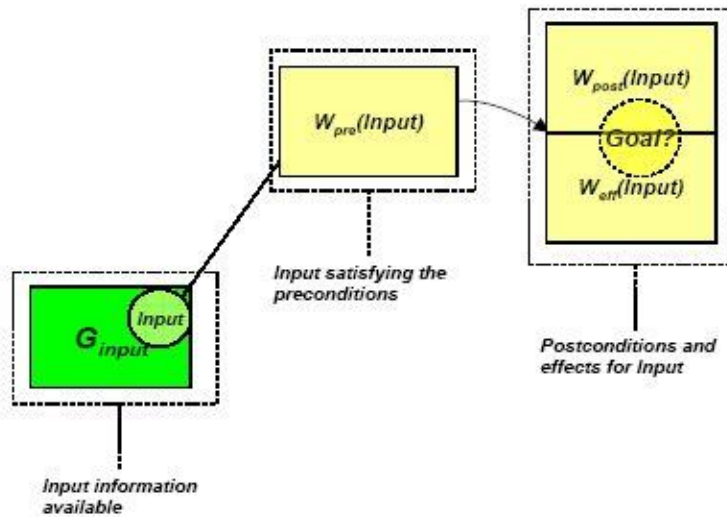
Δεδομένου ότι χρησιμοποιούμε τον υποθετικό χειριστή ( $\rightarrow$ ), ο ισχυρισμός των μετέπειτα καταστάσεων των υπηρεσιών και τα αποτελέσματα θα κυληθούν πίσω δηλ. αποσύρονται αφότου τελειώσουν τον έλεγχο.

Η χρήση του χειριστή ακολουθίας ( $\mathbf{V}$ ) σημαίνει ότι οι προϋποθέσεις πρέπει να εξεταστούν πριν από τις μετέπειτα καταστάσεις και τα αποτελέσματα της υπηρεσίας μπορεί να βεβαιωθούν, και ότι ο στόχος μόνο εξετάζεται μετά από αυτό. Κατά αυτόν τον τρόπο, διακρίνουμε μεταξύ του προ και του μετά-κράτους μια συγκεκριμένη υπηρεσία που αντιστοιχεί στην εισαγωγή πληροφοριών Input. Αυτό είναι ιδιαίτερα σχετικό για τον έλεγχο προϋποθέσεων, ως ισχυρισμός των αποτελεσμάτων των υπηρεσιών Web θα αλλάξουν τις προ-καταστάσεις (και τις πληροφοριών και τα πραγματικά διαστήματα) και, εξ ορισμού, οι προϋποθέσεις πρέπει να ελέγχονται προτού συμβούν τέτοιες αλλαγές. Ομοίως, ο στόχος πρέπει να εξεταστεί μόνο τις μετά-καταστάσεις δηλ. όταν βεβαιωθούν οι μετέπειτα καταστάσεις και τα αποτελέσματα της υπηρεσίας Web.

Η υποχρέωση απόδειξης διευκρινίζεται στο σχήμα. Στην αριστερή πλευρά της εικόνας, αντιπροσωπεύουμε τις διαθέσιμες πληροφορίες εισαγωγής. Από αυτό, η εισαγωγή που ικανοποιεί την συνένωση ικανότητας Web, οι προϋποθέσεις  $W_{pre}$  (εισαγωγή) εξετάζονται. Ανάλογα με αυτήν την εισαγωγή, υποθέτουμε ότι (αντιπροσωπευόμενο από ένα βέλος στον αριθμό) οι μετέπειτα καταστάσεις και τα αποτελέσματα της συγκεκριμένης υπηρεσίας αντιστοιχούν στην εισαγωγή. Αυτά, αντιπροσωπεύονται από το κιβώτιο επάνω στη δεξιά πλευρά του αριθμού, που είναι τα αποτελέσματα που παρέχονται από τη συγκεκριμένη υπηρεσία. Τέλος, ελέγχουμε εάν ο στόχος ικανοποιείται από αυτά τα αποτελέσματα δηλ. τα αποτελέσματα που ζητούνται στο στόχο παράγονται.

Για την περιγραφή του στόχου, επιτρέπουμε τη χρήση και των δύο παγκοσμίως ποσολογημένων στόχων δηλ. όλα τα αποτελέσματα που ικανοποιούν τους όρους που δηλώνονται στο στόχο πρέπει να παραδοθούν από την υπηρεσία Web, καθώς επίσης και των συνδεδεμένων ποσολογημένων στόχων δηλ. μερικά αποτελέσματα που ικανοποιούν τέτοιους όρους πρέπει να παραδοθούν. Εάν το αίτημα είναι υπαρξιακό ή καθολικό θα είναι μέρος από την περιγραφή του ίδιου του στόχου. Ομοίως, η ικανότητα συνένωσης θα διευκρινίσει επίσης εάν μπορεί να παρέχει ένα ή όλα τα περιγραφόμενα στοιχεία. Εάν ο υπαρξιακός προσδιορισμός της ποσότητας χρησιμοποιείται, οι υπηρεσίες Web που

παρέχουν τουλάχιστον ένα αποτέλεσμα που ικανοποιεί τους όρους που δηλώνονται ο στόχος θα αντιστοιχηθεί, ενώ μόνο οι υπηρεσίες Web που παρέχουν όλα αυτά τα αποτελέσματα θα αντιστοιχηθούν εάν ο καθολικός προσδιορισμός της ποσότητας χρησιμοποιείται



Σχήμα 4.9 : Προϋποθέσεις και αποτελέσματα των υπηρεσιών του Web

### Σχέση μεταξύ της ανακάλυψης και της μεσολάβησης

Σε ένα διανεμημένο περιβάλλον, οι διαφορετικοί χρήστες και οι υπηρεσίες Web μπορούν να χρησιμοποιήσουν διαφορετικές ορολογίες, όποια οδηγεί στην ανάγκη για τη μεσολάβηση προκειμένου τα ετερογενή συμβαλλόμενα μέρη να επικοινωνήσουν. Σε αυτό το τμήμα αναλύουμε τη σχέση μεταξύ της ανακάλυψης και της μεσολάβησης, προσδιορίζοντας τι είδους μεσολάβηση απαιτείται στα διαφορετικά σενάρια. Παρέχοντας τη μεσολάβηση υποστήριξης που απαιτείται να είναι εκτός του παραδοτέου πεδίου. Εντούτοις, συζητάμε υποθέσεις απαραίτητες στα διαφορετικά τεχνολογικά σενάρια και μέχρι ποίο σημείο είναι ρεαλιστικά. Παρατηρήστε ότι ένα έντονο παρόμοιο πρόβλημα μεσολάβησης προκύπτει στη σύνθεση, και οι υποθέσεις και οι απαιτήσεις που παρουσιάζονται εδώ ισχύουν στη σύνθεση ή οποιασδήποτε άλλης διαδικασίας που περιλαμβάνει τις υπηρεσίες Web.

## 4.7 Σχέση μεταξύ ανακάλυψης και μεσολάβησης

### 4.7.1 Υποθέσεις στη μεσολάβηση

Θα μπορούσαμε να υποθέσουμε ότι οι υπηρεσίες και οι στόχοι του Web περιγράφονται χρησιμοποιώντας την ίδια ορολογία. Σε εκείνη την περίπτωση κανένα πρόβλημα μεσολάβησης δεν υπάρχει κατά τη διάρκεια της διαδικασίας ανακαλύψεων. Εντούτοις, είναι απίθανο που ένας ενδεχομένως τεράστιος αριθμός διανεμημένων και αυτόνομων συμβαλλόμενων μερών θα συμφωνηθεί εκ των προτέρων σε μια κοινή ορολογία.

Εναλλακτικά, θα μπορούσαμε να υποθέσουμε ότι οι στόχοι και οι υπηρεσίες του Web περιγράφονται χρησιμοποιώντας εντελώς ανεξάρτητα λεξιλόγια. Αν αυτό συνέβαινε σε ένα πραγματικό περιβάλλον, η ανακάλυψη θα ήταν αδύνατο να επιτευχθεί. Κατά συνέπεια, εξετάζουμε ένα ενδιάμεσο σενάριο όπου δεν αγνοούμε την έμφυτη ετερογένεια του πεδίου, και που η μεσολάβηση και συνεπώς η ανακάλυψη είναι δυνατή. Τέτοιο σενάριο στηρίζεται σε τρεις κύριες υποθέσεις:

- Ø Οι στόχοι και οι υπηρεσίες Web χρησιμοποιούν πιθανότατα διαφορετικά λεξιλόγια, ή με άλλα λόγια, δεν περιορίζουμε την προσέγγισή μας στην περίπτωση όπου και οι δύο θα χρησιμοποιούν το ίδιο λεξιλόγιο.
- Ø Οι στόχοι και οι υπηρεσίες Web χρησιμοποιούν κάποιο ελεγχόμενο λεξιλόγιο ή κάποια ελεγχόμενη οντολογία που περιγράφει τις απαιτούμενες και τις παρεχόμενες υπηρεσίες.
- Ø Υπάρχει κάποια υπηρεσία μεσολάβησης σε ισχύ. Λαμβάνοντας υπόψη την προηγούμενη υπόθεση, μπορούμε να υποθέσουμε ότι μια σχεδίαση έχει καθιερωθεί ήδη μεταξύ της ήδη χρησιμοποιημένης ορολογίας, όχι αποκλειστικά για να διευκολύνει το συγκεκριμένο πρόβλημα ανακαλύψεων, αλλά μάλλον για να υποστηρίξει την ανταλλαγή γενικών πληροφοριών μεταξύ των συμβαλλόμενων μερών που χρησιμοποιούν αυτές τις ορολογίες.

Κάτω από αυτές τις υποθέσεις, θα πάρουμε μια υποστήριξη ελάχιστης μεσολάβησης που είναι αναγκαία προϋπόθεση για επιτυχή ανακάλυψη.

Παρατηρούμε ότι το IBROW (βλ. [BPM<sup>+</sup>98]), είναι ένα πρόγραμμα βασισμένο στον τομέα του Διαδίκτυου, αντιστοιχώντας τις περιγραφές των στόχων και τους ορισμούς ικανότητας των μεθόδων επίλυσης προβλήματος, έχει υιοθετήσει μια παρόμοια

προσέγγιση. Και οι στόχοι και οι μέθοδοι χρησιμοποιούν διαφορετικές οντολογίες που περιγράφουν τα αιτήματα και τις υπηρεσίες τους. Εντούτοις, και οι δύο οντολογίες περιγραφής στηρίζονται σε μια κοινή βασική οντολογία που επιτρέπει να ξαναδιατυπωθούν τα αιτήματα και οι υπηρεσίες σε σχέση με μία κοινή οντολογία.

### **4.7.2 Απαιτήσεις μεσολάβησης**

Παρακάτω, θα συζητήσουμε τα διαφορετικά τεχνολογικά σενάρια για τη διαδικασία ανακαλύψεων και πώς αυτά αναδιαμορφώνουν την υπόθεσή μας στην βασική υποστήριξη μεσολάβησης που χρειάζεται να είναι σε ισχύ.

#### **Επεξεργασία φυσικής γλώσσας και λέξεις κλειδιά**

Η επεξεργασία γραπτής φυσικής γλώσσας, που δημιουργείται από έναν ανθρώπινο χρήστη χρησιμοποιείται συνήθως για να παράγει πληροφορίες για την κατανάλωση υπολογιστών. Η προέλευση, η επικόλληση μέρων του λόγου, η διατύπωση φράσεων, η αναζήτηση συνωνύμου κ.λ.π., μπορούν να χρησιμοποιηθούν για να παράγουν την επεξεργάσιμη από τη μηχανή σημασιολογία από τις περιγραφές των στόχων και των υπηρεσιών. Κατά αυτόν τον τρόπο, ένα σύνολο λέξεων κλειδιών που εξάγονται από τον απαιτούμενο στόχο αντιστοιχούνται εναντίον ενός συνόλου λέξεων κλειδιών που εξάγονται από μια περιγραφή υπηρεσιών ή από προκαθορισμένους στόχους. Εναλλακτικά, και ο αιτών και ο προμηθευτής μπορούν άμεσα να χρησιμοποιήσουν τις λέξεις κλειδιά για να περιγράψουν τα αιτήματα και τις προσφορές τους.

Η προέλευση και η αναγνώριση συνωνύμου προσπαθεί ήδη να μετατρέψει τις διαφορετικές λέξεις σε μια κοινή τους εννοιολογική έννοια. Ακόμα, αυτή η υποστήριξη μεσολάβησης είναι πολύ γενική και στηρίζεται μόνο στους γενικούς κανόνες της επεξεργασίας γλωσσών. Η μεσολάβηση μπορεί επίσης να απαιτηθεί για τη μετάφραση από μια δεδομένη ανθρώπινη γλώσσα σε μια άλλη μια. Επίσης σε αυτήν την περίπτωση η μεσολάβηση που απαιτείται είναι γενική και μπορεί να υπολογιστεί ρεαλιστικά.



### **Ελεγχόμενα λεξιλόγια & οντολογίες**

Ένα διαφορετικό σενάριο είναι να υποθέσουμε ότι ο αιτών και ο προμηθευτής χρησιμοποιούν (όχι απαραίτητα το ίδιο) ελεγχόμενο λεξιλόγιο. Προκειμένου να διευκρινίσουμε αυτό το σενάριο θα αναφερθούμε σε ελεγχόμενα λεξιλόγια για προϊόντα και υπηρεσίες. Προσπάθειες όπως UNSPSC ή eCI@ss (μεταξύ των άλλων) παρέχουν ελεγχόμενα λεξιλόγια για να περιγράψουν προϊόντα και υπηρεσίες με περίπου 15.000 έννοιες η κάθε μια. Μια υπηρεσία περιγράφεται από (μίας ή περισσότερων) κατηγορίες αναφοράς μέσα από τα σχήματα ταξινόμησης και, ομοίως, ένας στόχος περιγράφεται από μια έννοια που λαμβάνεται από ένα (όχι απαραίτητα το ίδιο) ελεγχόμενο λεξιλόγιο.

Μια υπηρεσία μεσολάβησης απαιτείται σε περίπτωση που ο αιτών και ο προμηθευτής χρησιμοποιούν διαφορετικά λεξιλόγια. Η απαίτηση υπηρεσίας μεσολάβησης είναι πολύ ρεαλιστική, δεδομένου ότι χρησιμοποιούν ελεγχόμενα αντί των ειδικών λεξιλογίων. Παρατηρούμε ότι δεν χρησιμοποιούμε μια προσαρμοσμένη υπηρεσία μεσολάβησης για το συγκεκριμένο στόχο ανακαλύψεών μας, αλλά μάλλον για συμμορφώσεις της γενικής ορολογίας επιχειρήσεων που μπορούν να χρησιμοποιηθούν για άλλες ανταλλαγές πληροφοριών χρησιμοποιώντας τέτοιες ορολογίες.

Τα σύνορα μεταξύ των ελεγχόμενων λεξιλογίων και των οντολογιών είναι λεπτά και ανοικτά για ομαλή και επαυξητική ανάπτυξη. Οι οντολογίες είναι συναινετικές και επίσημες συλλήψεις ενός πεδίου (βλ. [Gru93b]). Τα ελεγχόμενα λεξιλόγια οργανώνονται στις ταξινομήσεις και με τα καθορισμένα χαρακτηριστικά όπως τα eCI@ss μοιάζουν με όλα τα απαραίτητα στοιχεία μιας οντολογίας. Οι οντολογίες μπορούν απλά να προσθέσουν μερικά λογικά αξιώματα περαιτέρω για να περιορίσουν τη σημασιολογία των στοιχείων τους. Παρατηρήστε ότι μια υπηρεσία αιτήσεων ή προμηθευτών παίρνει αυτούς τους λογικούς ορισμούς «δωρεάν». Μπορεί να επιλέξει τις έννοιες για το σχολιασμό της υπηρεσίας ή του στόχου του, αλλά όχι αναγκαστικά να γράψει τη λογική έκφραση εφ' όσον επαναχρησιμοποιεί μόνο αυτά που ήδη συμπεριλαμβάνονται στην οντολογία.

Αυτό το σενάριο φαίνεται αρκετά ελκυστικό δεδομένου ότι προσθέτει τις ευκολίες του Semantic Web στην ανακάλυψη υπηρεσιών Web που εξαλείφει δύο σημαντικούς παράγοντες κινδύνου των βασισμένων στη λογική τεχνικών:

- Ø Προσπάθεια στο γράψιμο των λογικών εκφράσεων. Να γραφούν οι σωστοί λογικοί τύποι είναι μια δύσκολη διαδικασία. Μια προσέγγιση ανακαλύψεων που υποθέτει αυτό σε μια μεγάλη κλίμακα για τα προβλήματα εξελιξιμότητας για τον αιτούντα και τον προμηθευτή.
- Ø Προσπάθεια στο συλλογισμό των λογικών εκφράσεων. Ο συλλογισμός πέρα από τη σύνθετες λογικές εντολές είναι υπολογιστικά δύσκολος σε περίπτωση που η επίσημη γλώσσα παρέχει ένα ορισμένο επίπεδο εκφραστικότητας. Επομένως, είναι σημαντικό να αποσυνδεθεί η διαδικασία συλλογισμού από την πραγματική διαδικασία ανακαλύψεων. Η λογική σχέση μεταξύ αυτών των εννοιών μπορεί να προέλθει ανεξάρτητα, από έναν συγκεκριμένο στόχο και τα υλοποιημένα συμπεράσματα μπορούν απλά να επαναχρησιμοποιηθούν κατά τη διάρκεια της διαδικασίας ανακαλύψεων.

Όσον αφορά τα ελεγχόμενα λεξιλόγια, η υποστήριξη μεσολάβησης απαιτείται σε περίπτωση που ο αιτών και ο προμηθευτής χρησιμοποιούν διαφορετικές οντολογίες. Εντούτοις, δεδομένου ότι οι γενικές οντολογίες μπορούν να χρησιμοποιηθούν, τέτοιες υπηρεσίες μεσολάβησης μπορούν να υποτεθούν. Τέλος, δεν πέρνουμε μια προσαρμοσμένη υπηρεσία μεσολάβησης για ανακάλυψη δικών μας υποθέσεων, αλλά μάλλον για γενικές συμμορφώσεις γενικών οντολογιών.

### **Ολοκληρωμένη λογική**

Οι απλοί, επαναχρησιμοποιήσιμοι, υπάρχοντες ορισμοί εννοιών, όπως περιγράφονται στο προηγούμενο υποκεφάλαιο έχουν το πλεονέκτημα της απλότητας στο σχολιασμό των υπηρεσιών και στο συλλογισμό αυτών.

Εντούτοις, αυτή η προσέγγιση έχει περιορίσει την ευελιξία, την εκφραστικότητα, και έχει μικρό μέγεθος της περιγραφής των υπηρεσιών και των αιτημάτων. Επομένως, είναι κατάλληλη μόνο για τα σενάρια όπου δεν απαιτείται μια ακριβέστερη περιγραφή από τους στόχους και τις υπηρεσίες. Επιπλέον, η περιγραφή της λειτουργίας μιας υπηρεσίας Web απαιτεί μια εκφραστικότητα που υπερβαίνει τις ικανότητες των τωρινών γλωσσών οντολογίας. Για αυτούς τους λόγους, όταν η υψηλότερη ακρίβεια είναι απαραίτητη για τη διαδικασία ανακαλύψεων, η ολοκληρωμένη λογική απαιτείται. Αυτή είναι η περίπτωση

για την κατασκευή υπηρεσιών, όπου οι ειδικές υπηρεσίες πρέπει να συμφωνηθούν και να κατασκευαστούν.

Η μεσολάβηση μπορεί μόνο να παρασχεθεί εάν η ορολογία που χρησιμοποιείται στις λογικές εκφράσεις είναι στηριγμένη στις οντολογίες. Διαφορετικά, είναι αδύνατο να αποδειχθεί η δεδομένη λογική σχέση μεταξύ των στόχων και της περιγραφής υπηρεσιών. Επομένως, η υποστήριξη μεσολάβησης που απαιτείται είναι ίδια όπως για τις οντολογίες.

#### **4.8 Επιτεύγματα**

Σε αυτό το κεφάλαιο, έχουμε εισαγάγει το πρόβλημα των αυτοματοποιημένων υπηρεσιών που εκπληρώνουν μια δεδομένη επιθυμία του αιτούντος. Προκειμένου να παρασχεθεί μια περιεκτική λύση στο πρόβλημα, έχουμε εισαγάγει σε προηγούμενη παράγραφο ένα εννοιολογικό πρότυπο για την ανακάλυψη και την συμφωνία από τις υπηρεσίες που λαμβάνει υπόψη τα πραγματικά ζητήματα όπως, ο βαθμός ακρίβειας που μπορεί να αναμένεται μέσα στις επίσημες περιγραφές των υπηρεσιών.

Σε άλλη παράγραφο έχουμε αναλύσει την εργασία που γίνεται μέχρι τώρα στον τομέα της αυτόματης υπηρεσίας ανακάλυψης και μια έντονα σχετιζόμενη πεδίο: την ανάκτηση τμημάτων λογισμικού.

Αφού εισαγάγαμε την περιγραφή των στόχων και των υπηρεσιών του Web, εισάγαμε την επίσημη σημασιολογία για τους ζητούμενους στόχους και τις αφηρημένες ικανότητες των υπηρεσιών, καθώς επίσης και την επίσημη σχέση μεταξύ εκείνων που πρέπει να ελεγχθούν με σκοπό να καθιερωθούν διαφορετικά είδη αντιστοιχίας. Επιπλέον, έχουμε συζητήσει την επιρροή από τις προθέσεις του αιτούντος και των υπηρεσιών κατά την περιγραφή των στόχων και των αφηρημένων ικανοτήτων τους, που έχουν αγνοηθεί μέχρι τώρα στη συγγραφή, και πώς έχουν επιπτώσεις στις σχέσεις που πρέπει να ελεγχθούν για την ανίχνευση των διαφορετικών τύπων αντιστοιχιών. Τέλος, έχουμε παράσχει μια διαμόρφωση βασισμένη στην πρωταρχική λογική και μια προσαρμογή που περιορίζει την εκφραστικότητα για την περιγραφή των στόχων και των ικανοτήτων και αυτό φέρνει τα επιθυμητά υπολογιστικά χαρακτηριστικά.

Σε άλλη παράγραφο έχουμε εξετάσει το τελευταίο βήμα στο εννοιολογικό πρότυπο δηλ. την συμφωνία μιας υπηρεσίας, παρέχοντας πάλι μια διαμόρφωση της έννοιας της αντιστοιχίας και εισάγαμε το είδος της απαραίτητης διαμόρφωσης για την συμφωνία

ικανοτήτων και για τους στόχων του αιτούντος. Τέτοια διαμόρφωση έχει παρασχεθεί χρησιμοποιώντας την πρωταρχική λογική και τη λογική συναλλαγής, και η τελευταία είναι εκτελέσιμη χρησιμοποιώντας FLORA-2.

Τέλος, έχουμε συζητήσει τη σχέση μεταξύ της ανακάλυψης υπηρεσιών και της μεσολάβησης, αναλύοντας τις υποθέσεις που πρέπει να κάνουμε στην υποστήριξη μεσολάβησης.

Εν συντομία, έχουμε παρουσιάσει μια περιεκτική ανάλυση του προβλήματος της αυτόματης ανακάλυψης υπηρεσιών και ένα εννοιολογικό πρότυπο για αυτό, παρέχοντας την επίσημη σημασιολογία για πιο σχετικά βήματα του προτύπου και την ανάλυση της χρήσης των διαφορετικών τύπων έτσι ώστε να επιτευχθεί. Επιπλέον, το πρότυπό μας είναι ανεξάρτητο στην βασική σημασιολογική περιγραφή του μοντέλου των υπηρεσιών Web, αν και βάζει μερικές απαιτήσεις στα εννοιολογικά στοιχεία που πρέπει να είναι σε ισχύ και όποια έκφραση απαιτείται για τη διαμόρφωση της γλώσσας χρησιμοποιείται.

#### **4.9 Ανοικτά σημεία**

Κατά τη διάρκεια της εργασίας μας, έχουμε βρει διάφορα σημεία που παραμένουν ανοικτά και πρέπει να λυθούν για μια εφαρμογή του προτύπου των ανακαλύψεών μας. Παρακάτω, απαριθμούμε τα πιο σχετικά από αυτά:

- Ø Η καθεμία διαδικασία των προκαθορισμένων στόχων που βασίζονται σε επιθυμίες χρηστών πρέπει να είναι αυτοματοποιημένη ή τουλάχιστον πιο εύκολη.
- Ø Ένα αποτελεσματικό πρότυπο ανακαλύψεων πρέπει να εκτελέσει το σύνθετο και ακριβή συλλογισμό μόνο για ένα μικρό υποσύνολο όλων των διαθέσιμων υπηρεσιών. Η DL διαμόρφωση θεωρείται καλή υποψήφια για την ανακάλυψη υπηρεσιών, καθώς αναμένεται να φιλτράρει αποτελεσματικά ένα μεγάλο αριθμό άσχετων υπηρεσιών. Εντούτοις, οι τρέχοντες DL αιτιολογητές παρουσιάζουν προβλήματα κατά την εξέταση των μικρότερων συνδέσεων για ορισμένες DL γλώσσες π.χ. SHOIN. Επομένως, ένας τρόπος για να υπερνικηθεί αυτό το πρόβλημα πρέπει να βρεθεί και, τελικά, οι δοκιμές απόδοσης να πραγματοποιηθούν για να επιλέξουν το καταλληλότερο DL αιτιολογητή.

- ∅ Δεδομένου ότι η DL διαμόρφωση δεν επιτρέπει τη διαφοροποίηση των διαφορετικών στόχων, ο σχολιασμός από τις DL έννοιες πρέπει να εισαχθεί σε επόμενο επίπεδο. Το πως θα εφαρμοστεί ακριβώς αυτό είναι ακόμα ανοικτό.
- ∅ Η δυνατότητες περιορισμού και διαχωρισμού υπηρεσιών απαιτούνται. Εντούτοις, η παροχή περιγραφών από την υπηρεσία που καλύπτει αυτές τις δύο πτυχές πρέπει να είναι συνεπής. Δεν έχει αποδειχτεί εάν η ικανότητα διαχωρισμού μπορεί να αφαιρεθεί αυτόματα από την ικανότητα περιορισμού. Εάν αυτό δεν ισχύει, απαιτείται ένας τρόπος για να εξασφαλιστεί η συνέπεια και των δύο περιγραφών. Επιπλέον, αυτές οι ικανότητες θέτουν διαφορετικές απαιτήσεις στη γλώσσα που χρησιμοποιείται για να τις περιγράψει. Το πως οι διαφορετικές γλώσσες αλληλεπιδρούν είναι ένα ανοικτό ζήτημα. Η οικογένεια των γλωσσών WSMML αναμένεται να είναι καλή υποψήφια για την επίλυση του. Προβλέπουμε ότι η παραλλαγή WSMML-DL θα απαιτηθεί για να περιγραφούν οι αφηρημένες ικανότητες, ενώ ο WSMML-Rule είναι πλέον πιθανός υποψήφιος για την συνένωση των ικανοτήτων. Όπως η WSMML-Full αναμένεται να παρέχει μια κοινή ενοποίηση της γλώσσας για την WSMML-DL και την WSMML-Rule, έτσι θα έλυne σχεδόν και το πρόβλημα. Η WSMML-Core μπορεί επίσης να χρησιμεύσει ως η γλώσσα για την περιγραφή των πεδίων των οντολογιών, δεδομένου ότι αυτό αντιστοιχεί στο μέγιστο κοινό υποσύνολο WSMML-DL και της WSMML-Rule. Η εκφραστικότητα που παρέχεται από την WSMML-Core αναμένεται να αρκεί για τις περισσότερες πεδία [Vol04], αλλά είναι κάτι που πρέπει να εξεταστεί. Ένα παρόμοιο πρόβλημα πρέπει να επιλυθεί για την περιγραφή των στόχων του αιτούντος.
- ∅ Οι ικανότητες περιορισμού στηρίζονται στην ύπαρξη των βεβαιώσεων που (διαφανώς) επικοινωνούν με το φορέα παροχής υπηρεσιών π.χ. εξετάζοντας τη βάση γνώσεων προμηθευτών. Εντούτοις, τα ζητήματα επικοινωνίας που περιλαμβάνονται και πρέπει να επιλυθούν π.χ. η δυναμική χρήση των υπηρεσιών χορογραφίας για την επικοινωνία με τον προμηθευτή. Ομοίως, οι πληροφορίες που ένας αιτών μπορεί να παρέχει υποτίθεται ότι έχουν παρασχεθεί διαφανώς, χωρίς να αντιμετωπίσει ζητήματα επικοινωνίας.

- Ø Η υποστήριξη μεσολάβησης υποτίθεται ότι έκανε την ανακάλυψη και τη συμβαλλόμενη εργασία σε ένα ετερογενή περιβάλλον. Η πραγματική υποστήριξη μεσολάβησης πρέπει να παρασχεθεί για μια πραγματική εφαρμογή από την ανακάλυψη και τις Συνενώσεις. Η WSMO παρέχει ένα κατάλληλο εννοιολογικό πλαίσιο για αυτό, και wgMediators, ggMediators και ooMediators [RLeditors04] μπορούν να υιοθετηθούν για αυτόν το λόγο.
- Ø Μια μηχανή ανακαλύψεων που εφαρμόζει τις προτεινόμενες προσεγγίσεις πρέπει να εφαρμοστεί και να δοκιμαστεί η αποδοτικότητά της.

#### **4.10 Μελλοντική εργασία**

Η μελλοντική εργασία μας θα επικεντρωθεί στην επίλυση των ανοικτών ζητημάτων που αναφέρονται στο προηγούμενο τμήμα, ιδιαίτερα στην αλληλεπίδραση των αφηρημένων και των επίσημων συμφωνιών των ικανοτήτων των υπηρεσιών και στις απαραίτητες γλώσσες και στην αξιολόγηση των αιτιών των υποψηφίων. Αυτό τελικά θα οδηγήσει στην εφαρμογή μιας μηχανής ανακαλύψεων βασισμένης σε αυτό το εννοιολογικό πρότυπο. Αυτό η εργασία έχει αρχίσει ήδη (δείτε [Leditors04b]).

Μια δεύτερη σημαντική γραμμή στη μελλοντική εργασία μας είναι να ενσωματωθεί η προγραμματισμένη μηχανή ανακαλύψεων με τις προσεγγίσεις για τη σύνθεση που θα παρουσιαστεί στα επόμενα κεφάλαια. Μια εννοιολογική ανάλυση τέτοιας ολοκλήρωσης θα παρασχεθεί σε επόμενο κεφάλαιο.

## Κεφάλαιο 5

### 5.1 Σημασιολογία για τη σύνθεση υπηρεσιών στο Web

Η αυτόματη σύνθεση των υπηρεσιών Web, δηλαδή το πρόβλημα «αυτόματης επιλογής, σύνθεσης, και λειτουργικότητας των [υπαρχόντων] υπηρεσιών Web που εκτελούν κάποιο σύνθετο στόχο, λαμβάνοντας υπόψη μια υψηλού επιπέδου περιγραφή ενός αντικειμενικού» [Coa04], είναι ένας από τους στόχους που παρακινούν την έρευνα στις σημασιολογικές υπηρεσίες Web. Σε αυτό το κεφάλαιο παρέχουμε ένα εννοιολογικό πρότυπο για την περιγραφή των προβλημάτων σύνθεσης υπηρεσιών Web και συζητάμε τεχνικές και εργαλεία που υποστηρίζουν την αυτόματη σύνθεση υπηρεσιών.

Διακρίνουμε τη σύνθεση δύο διαφορετικών μορφών υπηρεσιών του Web, δηλαδή μία «Λειτουργικού επιπέδου σύνθεση» και μια «ειδικού επιπέδου επεξεργασίας Σύνθεσης». Η **Λειτουργικού επιπέδου σύνθεση** εξετάζει το πρόβλημα ενός συνόλου υπηρεσιών, που συνδυάζονται με έναν κατάλληλο τρόπο, και είναι ικανή να τα ταιριάζει με μια δεδομένη ερώτηση. Κάθε υπάρχουσα υπηρεσία καθορίζεται από την άποψη της ατομικής αλληλεπίδρασης, δηλ., από την άποψη των παραμέτρων εισαγωγής και παραγωγής καθώς επίσης και των προϋποθέσεων και των αποτελεσμάτων. (Τέτοια περιγραφή παρέχεται για παράδειγμα στο σχεδιάγραμμα υπηρεσιών OWL-S, ή στο πρότυπο υπηρεσίας WSMO, που υπάρχει σε προηγούμενη παράγραφο). Η ερώτηση καθορίζει τη γενική λειτουργία που η αποτελούμενη υπηρεσία πρέπει να εφαρμόσει, πάλι από την άποψη των εισαγόμενων πληροφοριών, των εξαγόμενων πληροφοριών, των προϋποθέσεων, και των αποτελεσμάτων. Ένα παράδειγμα της λειτουργικού επιπέδου σύνθεσης στο παράδειγμα του εικονικού γραφείου ταξιδιών (όπως έχουμε αναφέρει σε προηγούμενη παράγραφο) είναι αυτή του προσδιορισμού των υπηρεσιών Web προκειμένου να απαντήσει σε ένα αίτημα πελατών. Το αποτέλεσμα της σύνθεσης μπορεί να είναι ο προσδιορισμός της υπηρεσίας «κράτησης πτήσης», της υπηρεσίας «κράτησης ξενοδοχείου», και μιας υπηρεσίας «ενοικίασης αυτοκινήτων» που είναι επαρκείς για την ιδιαίτερη αίτηση του πελάτη (π.χ., ο συγκεκριμένος προορισμός του ταξιδιού).

Η **ειδικού επιπέδου επεξεργασίας σύνθεση** καλύπτει μια πιο πρόσφατη φάση της γενικής σύνθεσης του στόχου. Εδώ υποθέτουμε ότι οι συνολικές υπηρεσίες του Web που είναι απαραίτητες για τον καθορισμό της σύνθεσης έχουν βρεθεί, και ότι πρέπει να

επεξεργαστούμε τις λεπτομέρειες για το πώς να αλληλεπιδρούμε με αυτές. Ο στόχος είναι να λάβουμε τον εκτελέσιμο κώδικα που εφαρμόζει τη σύνθεση. Σε αυτήν την φάση δεν είναι επαρκές να καθορίσουμε τις υπηρεσίες του Web από την άποψη των εισαγόμενων πληροφοριών, των εξαγόμενων πληροφοριών, των προϋποθέσεων και των αποτελεσμάτων. Μια πιο λεπτομερής περιγραφή κάθε υπηρεσίας Web είναι απαραίτητη προκειμένου να παραχθεί η ακριβής ακολουθία διαδικασιών για την αλληλεπίδραση με το φορέα παροχής υπηρεσιών. (Τέτοια περιγραφή παρέχεται για παράδειγμα στο πρότυπο διαδικασίας OWL-S, ή στη διεπαφή υπηρεσιών WSMO).

Στην περίπτωση της υπηρεσίας «κράτησης πτήσης», για παράδειγμα, αυτή η αλληλεπίδραση απαιτεί αρκετά βήματα συμπεριλαμβανομένης της επικύρωσης, της υποβολής μιας ιδιαίτερης αίτησης, της διαπραγμάτευσης μιας προσφοράς, της αποδοχής (ή άρνησης) της προσφοράς, και της πληρωμής. Επιπλέον, αυτά τα βήματα μπορούν να έχουν υπό όρους, ή μη-ονομαστικές εκβάσεις (π.χ., δεν μπορεί να υπάρξει καμία προσφορά διαθέσιμη από την ύπαρξη μιας υπηρεσίας...) αυτό μπορεί να έχει επιπτώσεις στα ακόλουθα βήματα (π.χ., εάν δεν υπάρχει καμία προσφορά διαθέσιμη, μια διαταγή δεν μπορεί να υποβληθεί...). Όλες αυτές οι λεπτομέρειες είναι επουσιώδεις για τον καθορισμό της σύνθεσης λειτουργικού επιπέδου, αλλά γίνονται σημαντικές για την παραγωγή του πραγματικού κώδικα στην εφαρμογή του. Η ανάγκη της σύνθεσης υπηρεσιών ειδικού επιπέδου επεξεργασίας αναγνωρίζεται σε αρκετές εργασίες, ειδικότερα στο [NM02], όπου μια προσέγγιση προτείνεται για την προσομοίωση, επαλήθευση και σύνθεση των σύνθετων υπηρεσιών που περιγράφονται στην OWL-S. Μια αυτοματοποιημένη προσέγγιση για ειδικού επιπέδου επεξεργασίας σύνθεση των υπηρεσιών Web περιγράφεται στο [TP04] επίσης σε αυτήν την περίπτωση η OWL-S χρησιμοποιείται σαν γλώσσα διαμόρφωσης.

Οι δύο μορφές αυτόματων συνθέσεων απαιτούν διαφορετικά εννοιολογικά πλαίσια και διαφορετικές τεχνικές. Για αυτόν τον λόγο, σε αυτό το κεφάλαιο, η λειτουργικού επιπέδου και η ειδικού επιπέδου επεξεργασίας σύνθεση συζητούνται χωριστά, στις ακόλουθες παραγράφους



## 5.2 Λειτουργικού επιπέδου σύνθεση

Εδώ συζητάμε τις προσεγγίσεις για την πλήρως αυτοματοποιημένη, λειτουργική σύνθεση των υπηρεσιών σύμφωνα με τους περιορισμούς των χρηστών. Αυτή η προσέγγιση στη σύνθεση υπηρεσιών συσχετίζεται πάρα πολύ με τον παραδοσιακό προγραμματισμό τεχνητής νοημοσύνης. Το πρόβλημα της σύνθεσης υπηρεσιών διευκρινίζεται από ένα σύνολο διαθέσιμων εισαγόμενων πληροφοριών και ένα σύνολο απαραίτητων εξαγόμενων πληροφοριών. Οι υπηρεσίες αντιστοιχούν στους χειριστές προγραμμάτων: Απαιτούν ορισμένες εισαγωγές (προϋποθέσεις) και παρέχουν τις εξαγωγές (αποτελέσματα). Τα αποτελέσματα του προγραμματισμού μιας ρύθμισης των υπηρεσιών σε μια ροή της δουλειάς για να ικανοποιήσει τις απαιτήσεις χρηστών. Μία σημαντική διαφορά στον προγραμματισμό είναι ότι το σύνολο της περιγραφής υπηρεσιών (δηλ., ο χειριστής προγραμμάτων) μπορεί να είναι πολύ μεγάλο και διατηρείται συνήθως στους καταλόγους υπηρεσιών. Ως εκ τούτου, αυτό είναι κρίσιμο για τους αλγορίθμους σύνθεσης υπηρεσιών για να αλληλεπιδράσουν με τους καταλόγους υπηρεσιών έτσι ώστε να ανακτήσουν δυναμικά τις σχετικές υπηρεσίες. Προκειμένου να επιτευχθεί η απόδοση λογικής σύνθεσης, η αλληλεπίδραση μεταξύ του αλγορίθμου σύνθεσης και του καταλόγου υπηρεσιών πρέπει να είναι προσεκτικά επεξεργασμένα. Σε αυτό το τμήμα, θα εξετάσουμε διάφορους τύπους ταιριασμάτων που χρησιμοποιούνται από τη ισόπεδη λειτουργική σύνθεση. Οι έννοιες της αντιστοιχίας που χρησιμοποιούνται δεν ακολουθούν συνολικά το εννοιολογικό πρότυπο που καθορίσαμε σε προηγούμενο κεφάλαιο, και περιορίζονται στην ανακάλυψη, μη περιλαμβάνοντας την συνένωση. Η σχέση μεταξύ του προτύπου που παρουσιάζεται σε προηγούμενο κεφάλαιο και της σύνθεσης, και μιας πλήρη ευθυγράμμισης μεταξύ της ανακάλυψης και της περιορισμού του προτύπου και της σύνθεσης θα είναι υπαγόμενα σε μελλοντική εργασία στο πρόγραμμα γνώσης του Web.

Το υπόλοιπο αυτού του τμήματος είναι δομημένο ως εξής: Στην παράγραφο 5.1.1 αναθεωρούμε εν συντομία τους υπάρχοντες φορμαλισμούς προγραμματισμού και τη σύνθεση υπηρεσιών βασισμένων στη μελέτη των συστημάτων στο προγραμματισμό. Στην παράγραφο 5.1.2 εισάγουμε το φορμαλισμό μας για να περιγράψουμε τις διαφημίσεις και τα αιτήματα υπηρεσιών. Τυποποιούμε επίσης τις διαφορετικές μορφές για το ταίριασμα των αιτημάτων και των διαφημίσεων. Στην παράγραφο 5.1.3 εισάγουμε

τον τύπο συμβατής σύνθεσης υπηρεσιών, δηλ., τη σύνθεση υπηρεσιών που λαμβάνει υπόψη τους περιορισμούς τύπων. Παρουσιάζουμε επίσης μια συγκεκριμένη σύνθεση αλγόριθμου υπηρεσιών που υποστηρίζει μερικές αντιστοιχίες τύπων. Στην παράγραφο 5.1.4 δίνουμε μια επισκόπηση από τις τεχνικές εφαρμογής των καταλόγων υπηρεσιών για να υποστηριχθεί εξελικτικά και αποδοτικά την αυτοματοποιημένη σύνθεση υπηρεσιών, συμπεριλαμβανομένων των τεχνικών για την πολυδιάστατη ευρετηρίαση, την υποστήριξη για τα μεγάλα σύνολα αποτελέσματος (επαυξητική ανάκτηση των εξαγόμενων πληροφοριών), τον αποδοτικό έλεγχο συναγωνισμού, και η υποστήριξη για τη καθορισμένη από το χρήστη αναζήτηση. Τέλος, στην παράγραφο 5.1.5 παρουσιάζουμε τη δοκιμή μας που προσφέρει διάφορα πρότυπα για να μιμηθεί τους μεγάλους καταλόγους υπηρεσιών. Παρουσιάζουμε πειραματικά τα αποτελέσματα που υπογραμμίζουν τα οφέλη μερικών τύπων που ταιριάζουν με τη διαδικασία για τη σύνθεση υπηρεσιών. Χάρη σε αυτήν την υποστήριξη για μερικές αντιστοιχίες τύπων, ένα πολύ μεγαλύτερο μέρος του συνόλου του προβλήματος μπορεί να λυθεί από την αυτοματοποιημένη σύνθεση. Παρουσιάζουμε επίσης τα αποτελέσματα που πείζουν τη σημαντικότητα υποστήριξης των καθορισμένων από το χρήστη καταλόγων αναζήτησης.

### **5.2.1 Ιστορικό και κατάσταση προόδου**

Σε αυτό το τμήμα ξαναβλέπουμε εν συντομία κάποια σχετική εργασία στον τομέα της σύνθεσης υπηρεσιών. Κατ' αρχάς, συζητάμε τον STRIPS προγραμματισμό, ο οποίος είναι η βάση για αρκετή ερευνητική εργασία στην πεδίο της ισόπεδης λειτουργικής σύνθεσης υπηρεσιών. Ξαναβλέπουμε επίσης το Golog, μια σύνθεση προσέγγισης με βάση τον υπολογισμό κατάστασης, και το SHOP-2, ένα ιεραρχικό σύστημα προγραμματισμού.

#### **Προγραμματισμός STRIPS**

Τα STRIPS υιοθετήθηκαν από την ερευνητική κοινότητα ως βάση για την πλειοψηφία του σχεδιασμού του φορμαλισμού που χρησιμοποιείται σήμερα. Διάφορες επεκτάσεις παρέχονταν και διάφορες γεύσεις STRIPS υπάρχουν [FN71, Lif87, McD03], αλλά η υψηλού επιπέδου περιγραφή που παρουσιάζεται έπειτα είναι κοινή σε όλες τις προσεγγίσεις. Στο STRIPS ο κόσμος περιγράφεται από άποψη των καταστάσεων του. Οι

ενέργειες περιγράφονται στο επίπεδο της προϋπόθεσης, «πρόσθεση» και «διαγραφή» των καταλόγων. Ο κατάλογος προϋπόθεσης μιας δράσης περιγράφει τις καταστάσεις που πρέπει να ισχύουν στον κόσμο προτού να μπορεί να εφαρμοστεί η δράση. Η «πρόσθεση» καταλόγου περιγράφει τις καταστάσεις στον κόσμο που θα είναι αληθινές μετά από την εφαρμογή του χειριστή προγραμμάτων. Η «διαγραφή» του καταλόγου περιγράφει την κατάσταση στον κόσμο που θα είναι ψεύτικη αφότου εφαρμοστεί ο χειριστής.

### **Συστήματα που υποστηρίζουν την αυτοματοποιημένη σύνθεση υπηρεσιών**

**Golog** Μια αρχική προσέγγιση στη σύνθεση διαδικασίας στα [MSZ01] και στα [MS02b] ήταν η χρησιμοποίηση μιας σχεδίασης φορμαλισμού βασισμένη στον υπολογισμό κατάστασης, μια πρωταρχικής λογικής γλώσσα για το συλλογισμό για τη δράση και την αλλαγή. Στον υπολογισμό, η κατάσταση του κόσμου είναι εκφρασμένη από την άποψη της λειτουργίας και των σχέσεων σε μια ιδιαίτερη κατάσταση. Το πλεονέκτημα αυτής της προσέγγισης είναι ότι τα σύνθετα κατασκευάσματα ελέγχου όπως οι βρόχοι μπορούν να διαμορφωθούν χρησιμοποιώντας αυτό το πλαίσιο. Το μειονέκτημα αυτής της προσέγγισης είναι η υψηλή υπολογιστική πολυπλοκότητά του.

Αυτή την εργασία στηρίζει και επεκτείνει η Golog, μια υψηλή επιπέδου, λογική γλώσσα προγραμματισμού, αναπτυγμένη στο πανεπιστήμιο του Τορόντο. Η Golog υποστηρίζει την προδιαγραφή και την εκτέλεση σύνθετων ενεργειών στις δυναμικές πεδία.

**SHOP-2** Στο [Wu, 03] οι συντάκτες περιγράφουν το SHOP2, ένα ιεραρχικό σχεδιασμό φορμαλισμού για την κωδικοποίηση των πεδίων σύνθεσης. Αυτή η προσέγγιση είναι αποδοτικότερη αλλά δεν υποστηρίζει σύνθετα κατασκευάσματα όπως τους βρόχους.

Το SHOP2 είναι ένα σύστημα σχεδιασμού ανεξάρτητων πεδίων HTN. Ο σχεδιασμός HTN είναι μια μεθοδολογία σχεδιασμού τεχνητής νοημοσύνης που δημιουργεί σχέδιο από το στόχο αποσύνθεσης. Αυτή είναι μια διαδικασία στην οποία το σύστημα προγραμματισμού αποσυνθέτει τους στόχους στις μικρότερες και μικρότερες δευτερεύουσες υποχρεώσεις, μέχρι οι πρωτόγονοι στόχοι να μπορούν να εκτελεστούν άμεσα. Η έννοια της αποσύνθεσης στόχου σε HTN είναι μοιάζει πολύ με την έννοια της διαδικασίας αποσύνθεσης στην OWL-S.

Μια διαφορά μεταξύ του SHOP2 και των περισσότερων άλλων συστημάτων σχεδιασμού HTN είναι ότι το SHOP2 σχεδιάζει τους στόχους με τον ίδιο τρόπο που θα εκτελεσθούν αργότερα. Ο σχεδιασμός των στόχων για να εκτελεσθούν το καθιστούν πιθανό να ξέρουν την τρέχουσα κατάσταση του κόσμου σε κάθε βήμα στη διαδικασία προγραμματισμού, η οποία καθιστά πιθανή τη προηγούμενη κατάσταση ελέγχου στο μηχανισμό του SHOP2 για να ενσωματώσει τη σημαντική και αιτιολογική δύναμη, συμπεριλαμβανομένου της δυνατότητα να κληθούν τα εξωτερικά προγράμματα. Αυτό επιτρέπει στο SHOP2 για να ενσωματώσει τον σχεδιασμό με εξωτερικές πηγές πληροφοριών όπως το περιβάλλον Web.

Προκειμένου να γίνει ο σχεδιασμός σε μια δεδομένη πεδίο σχεδιασμού, στο SHOP2 πρέπει να δοθεί γνώση για εκείνη την πεδίο. Στο SHOP2 η βάση γνώσεων περιέχει τους χειριστές και τις μεθόδους. Κάθε χειριστής είναι μια περιγραφή αυτών που πρέπει να γίνουν για να ολοκληρωθεί κάποιος πρωτόγονος στόχος, και κάθε μέθοδος λέει πώς να αποσυνθεθεί κάποιος σύνθετος στόχο σε μερικώς διαταγμένες δευτερεύουσες υποχρεώσεις.

### **5.2.2 Φορμαλισμός και Σημασιολογία**

Σε αυτό το τμήμα δίνουμε μερικούς βασικούς ορισμούς και εισάγουμε το φορμαλισμό μας για την περιγραφή υπηρεσιών διαφήμισης και αιτημάτων υπηρεσιών μαζί με τη σχετική σημασιολογία τους. Αυτό ο φορμαλισμός μοιάζει πολύ με την περιγραφή των υπηρεσιών που χρησιμοποιήθηκε σε προηγούμενο κεφάλαιο. Εντούτοις, μερικές διαφορές προκύπτουν και θα συζητηθούν. Επαναλαμβάνουμε εν συντομία κάποια κατάσταση τέχνης σχετικά με την αντιστοιχία που είναι ενδιαφέρουσα για την προσέγγιση της σύνθεσής μας. Εμείς εισάγουμε τους περιορισμούς διαστήματος, ένας ενισχυτικός φορμαλισμός που χρησιμοποιούμε για την περιγραφή και το ταίριασμα των υπηρεσιών.

#### **Διαφημίσεις και αιτήματα υπηρεσιών**

Οι λειτουργικές πτυχές των υπηρεσιών διαφήμισης και των αιτημάτων υπηρεσιών διευκρινίζονται σα παράμετροι και καταστάσεις του κόσμου [CCMW01, DS04]. Οι παράμετροι μπορούν είτε να εισαχθούν είτε να εξαχθούν, και οι καταστάσεις του κόσμου

μπορούν να είναι είτε προϋποθέσεις (απαραίτητα κράτη) είτε αποτελέσματα (που παράγονται από την εκτέλεση της υπηρεσίας). Θεωρούμε ότι οι όροι στην υπηρεσία περιγραφής καθορίζονται χρησιμοποιώντας μια κατηγορία/ μια γλώσσα οντολογίας όπως την OWL [DS04]. Πρωτόγονα στοιχεία-τύποι μπορούν να καθοριστούν χρησιμοποιώντας μια γλώσσα όπως η XSD [W3C]. Όπως διευκρινίζεται από την πιο πρόσφατη έκδοση της OWL-S [Coa04], στο φορμαλισμό μας κάθε παράμετρος έχει δύο στοιχεία:

- Ø Ένας ρόλος που περιγράφει την πραγματική σημασιολογία της παραμέτρου (π.χ., σε μια πεδίο ταξιδιού ο ρόλος μιας παραμέτρου θα μπορούσε να είναι αναχώρηση ή άφιξη).
- Ø Ένας τύπος που καθορίζει τα πραγματικά στοιχεία δεδομένων της παραμέτρου (π.χ., τα στοιχεία δεδομένων και για την αναχώρηση και για τη άφιξη θα μπορούσαν να είναι η τοποθεσία).
- Ø Καθορίζουμε τις καταστάσεις του κόσμου μέσω των προϋποθέσεων και των αποτελεσμάτων. Επεκτείνουμε την κανονική σημασιολογία των εννοιών που μπορεί να περιληφθεί στις προϋποθέσεις ή τα αποτελέσματα.
- Ø Στις υπηρεσίες διαφήμισης η εισαγωγή και εξαγωγή παραμέτρων, καθώς επίσης και οι προϋποθέσεις και τα αποτελέσματα, έχουν την ακόλουθη σημασιολογία:
- Ø Για να είναι μια υπηρεσία invocable, μια αξία πρέπει να μαθευτεί για κάθε μια από τις παραμέτρους που εισήγαγε η υπηρεσία και πρέπει να είναι σύμφωνη με τον αντίστοιχο σημασιολογικό ρόλο και το συντακτικός τύπος της παραμέτρου. Η παράμετρος που παρέχεται ως εισαγωγή πρέπει να είναι σημασιολογικά πιο συγκεκριμένη από ότι η υπηρεσία είναι σε θέση να δεχτεί. Όσον αφορά τον τύπο της παραμέτρου, στην περίπτωση των πρωτόγονων στοιχείων η αξία επίκλησης πρέπει να είναι μέσα στη σειρά των τιμών, ή στην περίπτωση των κατηγοριών που η αξία επίκλησης πρέπει να είναι ενταγμένη από τον τύπο της παραμέτρου. Οι προϋποθέσεις καθορίζουν σε ποιο κατάσταση ο κόσμος πρέπει να είναι προτού να μπορέσει να επικαλεσθεί η υπηρεσία. Όλες οι προϋποθέσεις πρέπει να συνεπάγονται τους όρους που διευκρινίζονται από την τρέχουσα κατάσταση του κόσμου.

- Ø Για κάθε επιτυχή επίκληση η υπηρεσία θα παράσχει μια αξία για κάθε μια από τις εξαγωγές των παραμέτρων και κάθε μια από αυτές τις τιμές θα είναι σύμφωνες με τον αντίστοιχο ρόλο των παραμέτρων και το ρόλο των τύπων δεδομένων. Μετά από την επίκληση η κατάσταση του κόσμου θα τροποποιηθεί έτσι ώστε όλα τα αποτελέσματα που απαριθμούνται στη διαφήμιση υπηρεσιών θα προστεθούν στη νέα παγκόσμια κατάσταση. Οι όροι στην αρχική κατάσταση που συγκρούεται με τους όρους στη νέα κατάσταση θα αφαιρεθούν από το νέα κατάσταση.

Η ανωτέρω σημασιολογία είναι σύμφωνη με τις περιγραφές που εισάγονται σε προηγούμενη παράγραφο. Τα αιτήματα υπηρεσιών αντιπροσωπεύονται κατά τρόπο παρόμοιο αλλά έχουν διαφορετική σημασιολογία:

- Ø Οι εισαγωγές αιτήματος υπηρεσιών αντιπροσωπεύουν τις διαθέσιμες παραμέτρους (π.χ., που παρέχονται από το χρήστη ή από μια άλλη υπηρεσία). Κάθε μια από αυτές τις παραμέτρους εισαγωγής έχει συνδέσει μια σημασιολογική περιγραφή ρόλου είτε κάποια περιγραφή του τύπου δεδομένων είτε μια συγκεκριμένη αξία. Οι προϋποθέσεις σε ένα αίτημα αντιπροσωπεύουν την κατάσταση του κόσμου που είναι διαθέσιμη για οποιοδήποτε ταίριασμα υπηρεσιών διαφήμισης. Είναι ισοδύναμα με τους αρχικούς όρους σε έναν κλασικό σχεδιαστικό περιβάλλον. Αυτή η κατάσταση πρέπει να συνεπάγεται τη κατάσταση που απαιτείται στην προϋπόθεση οποιασδήποτε συμβατής υπηρεσίας.
- Ø Οι εξαγωγές του αιτήματος υπηρεσιών αντιπροσωπεύουν τις παραμέτρους που μια συμβατή (αποτελούμενη) υπηρεσία πρέπει να παρέχει. Ο ρόλος της παραμέτρου καθορίζει την πραγματική σημασιολογία των απαραίτητων πληροφοριών και ο τύπος της παραμέτρου καθορίζει ποιες σειρές των τιμών μπορούν να αντιμετωπιστούν από τον αιτούντα. Η συμβατή (αποτελούμενη) υπηρεσία πρέπει να είναι σε θέση να παρέχει την αξία για κάθε μια από τις παραμέτρους στην παραγωγή του αιτήματος υπηρεσιών, σημασιολογικά πιο συγκεκριμένα από ότι ο ζητούμενος ρόλος, και έχοντας τις τιμές στη σειρά που καθορίζεται από το ζητούμενο τύπο της παραμέτρου. Τα αποτελέσματα αντιπροσωπεύουν την αλλαγή του κόσμου που επιδιώκεται από τον αιτών της υπηρεσίας ή των στόχων όπου το

αίτημα υπηρεσιών πρέπει να εκπληρωθεί. Για οποιονδήποτε από τους στόχους ή τα αποτελέσματα της υπηρεσίας χρειάζεται να θεωρηθούν εκπληρωμένα, η κατάσταση του κόσμου μετά από την επίκληση μιας δεδομένης υπηρεσίας θα πρέπει να περιέχει την επίδραση που συνεπάγεται τον αντίστοιχο στόχο.

Παρατηρήστε ότι στις περιγραφές που παρουσιάστηκαν σε προηγούμενη παράγραφο οι εισαγωγές δεν χρησιμοποιούνται για την ανακάλυψη αλλά μόνο για την συνένωση, και στην χρονική συνένωση το στοιχείο περίπτωσης αναμένεται για να γίνει διαθέσιμο από τον αιτούντα. Επιπλέον, οι όροι πέρα από την κατάσταση του κόσμου δεν αντιπροσωπεύονται στο στόχο.

#### **Αντιστοιχία - τρέχουσες προσεγγίσεις**

Όπως παρουσιάσαμε σε προηγούμενη παράγραφο, προηγούμενη εργασία σχετικά με το ταίριασμα των συστατικών λογισμικού [ZW97] έχουν εξετάσει διάφορους πιθανούς τύπους αντιστοιχιών βασισμένους στην επίπτωση σχέσεων μεταξύ των προϋποθέσεων και των μετέπειτα καταστάσεων ενός τμήματος βιβλιοθήκης  $S$  και μιας ερώτησης  $Q$ . Παραδείγματος χάριν η PlugIn match, ένας από τους πιο χρήσιμους τύπους αντιστοιχιών ορίζεται ως:

$$\text{match}_{\text{PlugIn}}(Q, S) = (\text{pre}_Q \Rightarrow \text{pres}) \wedge (\text{posts} \Rightarrow \text{post}_Q).$$

Στο LARKS [SWKL02] ο ανωτέρω όρος έχει προσαρμοστεί έτσι ώστε η επίπτωση ήταν αντικατεστημένο με μια πιο εύκολη λειτουργία, η  $\theta$  υποκατηγορία πέρα από τα σύνολα περιορισμών ( $\overset{\lambda}{\_}\theta$ ):

$$\text{match}_{\text{PlugIn}}(Q, S) = (\text{pre}_Q \overset{\lambda}{\_}\theta \text{pres}) \wedge (\text{posts} \overset{\lambda}{\_}\theta \text{post}_Q).$$

Ένα σύνολο περιορισμών  $\text{pres}$   $\theta$ -εντάσσει ένα σύνολο περιορισμών  $\text{pre}_Q$  ( $\text{pre}_Q \overset{\lambda}{\_}\theta \text{pres}$  ή διαφορετικά  $\text{pre}_Q.\text{pres}$  ή  $\text{pre}_Q \Rightarrow \text{pres}$ ), εάν κάθε περιορισμός στο  $\text{pre}_Q$  εντάσσεται από το περιορισμό στο  $\text{pres}$  (ομοίως για τις μετέπειτα καταστάσεις):

$$\text{pre}_Q \stackrel{\lambda}{\perp} \text{pre}_S \mathcal{O}$$

$$(\forall C_Q \in \text{pre}_Q) (\exists C_S \in \text{pre}_S) (C_Q \stackrel{\lambda}{\perp} C_S).$$

Η πιο πρόσφατη εργασία σχετικά με την αντιστοιχία [PKPS02, LH03, CF03] έχει επεκτείνει αυτές τις προσεγγίσεις με τη χρησιμοποίηση των βασισμένων στη λογική γλωσσών περιγραφής [BS01, DS04] για τον καθορισμό των όρων από τις διαφημίσεις ή τα αιτήματα υπηρεσιών.

### Περιορισμοί διαστήματος

Για την περιγραφή των διαφημίσεων και των αιτημάτων υπηρεσιών χρησιμοποιούμε τους περιορισμούς στα σύνολα διαστημάτων (ενδεχομένως παραγόμενα από τις περιγραφές κατηγορίας [CF03]). Ένας περιορισμός είναι μια πρόσθετη μορφή στο πρώτο κατηγορήμα διαταγής που ποσολογεί παγκοσμίως πέρα από τις τιμές των συνόλων διαστήματος η περίπτωση ότι ένα διάστημα αντιπροσωπεύει την κωδικοποίηση μιας κατηγορίας ο περιορισμός αντιστοιχεί στο προσδιορισμός της ποσότητας σε όλα τα άτομα στην κατηγορία:

$$P(C_1, C_2, \dots, C_N) \mathcal{O}$$

$$(\forall x_1 \in C_1) (\forall x_2 \in C_2) \dots (\forall x_N \in C_N) (x_1, x_2, \dots, x_N).$$

Καθορίζουμε διάφορες πιθανές σχέσεις μεταξύ δύο συνόλων διαστήματος  $C_1$  και  $C_2$ :

$$C_1 \sqsubseteq C_2 \Leftrightarrow (\forall i_1 \in C_1) (\exists i_2 \in C_2) (i_1 \subseteq i_2),$$

$$C_1 \equiv C_2 \Leftrightarrow C_1 \sqsubseteq C_2 \wedge C_2 \sqsubseteq C_1,$$

$$C_1 \dot{\sqcap} C_2 \Leftrightarrow (\exists i_1 \in C_1) (\exists i_2 \in C_2) (i_1 \cap i_2 \neq \emptyset).$$

Η σχέση  $\neg \dot{\sqcap}$  είναι η λογική άρνηση του  $\dot{\sqcap}$  και ισχύει όταν το σύνολο από το διάστημα επιχειρήματος χωρίζεται. Καθορίζουμε επίσης δύο πρόσθετες σχέσεις: κορυφαίο  $\dot{\sqsupset}$  που πάντα ισχύει και κατώτατο σημείο  $\dot{\sqsubseteq}$  που δεν ισχύει ποτέ. Υπάρχει μια



ομοιότητα μεταξύ της  $\theta$  σχέσης υποκατηγοριών και ανάμεσα στο σύνολο των προτάσεων και η καθορισμένη σχέση  $\sqsubseteq$ .

Υποθέτουμε ότι οι περιορισμοί έχουν τα μοναδικά arities – που είναι περιορισμοί με το ίδιο όνομα που έχουν πάντα τον ίδιο αριθμό όρων.

Καθορίζουμε *ent*, μια σύνθετη σχέση απαραίτητη μεταξύ δύο περιορισμών  $P_1$  ( $C_{11}, \dots, C_{1n}$ ) και  $P_2$  ( $C_{21}, \dots, C_{2n}$ ) έχοντας το ίδιο arity  $n$  αλλά τα ενδεχομένως διαφορετικά ονόματα  $P_1$  και  $P_2$ . Το κατηγορήμα *ent* ( $P_1, P_2$   $op_1, \dots, op_n$ ) ισχύει όταν κάθε ένας από τους όρους  $C_{1i}$  και  $C_{2i}$  των δύο οι περιορισμοί είναι στη σχέση που διευκρινίζεται από το αντίστοιχο  $op_i$  χειριστών:

$$\mathit{ent}(P_1, P_2, op_1, \dots, op_n) \Leftrightarrow \bigwedge_{i=1}^n C_{1i} \mathit{op}_i C_{2i}$$

**where**  $op_i \in \{\equiv, \sqsubseteq, \supseteq, \dot{\sqcap}, \neg\dot{\sqcap}, \dot{\sqcup}, \perp\}, i = 1..n.$

Καθορίζουμε *not Ent*, μια σχέση μη-απαραίτητη που έχει τη σημασιολογία στη συμφωνία με εκείνοι του *ent*- το κατηγορήμα που ισχύει όταν τουλάχιστον ένας από τους όρους  $C_{1i}$  και  $C_{2i}$  δεν είναι σχέση που διευκρινίζεται από το αντίστοιχο  $op_i$  χειριστών:

$$\mathit{notEnt}(P_1, P_2, op_1, \dots, op_n) \Leftrightarrow \bigvee_{i=1}^n \neg(C_{1i} \mathit{op}_i C_{2i})$$

**where**  $op_i \in \{\equiv, \sqsubseteq, \supseteq, \dot{\sqcap}, \neg\dot{\sqcap}, \dot{\sqcup}, \perp\}, i = 1..n.$

### 5.2.3 Αυτοματοποιημένη σύνθεση υπηρεσιών

Σε αυτό το τμήμα εισάγουμε ένα συμβατό τύπο για τη σύνθεση υπηρεσιών, δηλ., σύνθεση υπηρεσιών που λαμβάνει υπόψη τον τύπο περιορισμού. Παρουσιάζουμε επίσης μια συγκεκριμένη σύνθεση αλγόριθμου υπηρεσιών που υποστηρίζει μερικές αντιστοιχίες τύπων.

#### Συμβατός τύπος με την ανακάλυψη και την σύνθεση

Για τη σύνθεση εξετάζουμε δύο είδη πιθανών προσεγγίσεων: αλυσοδέοντας προς τα μπρος και προς τα πίσω αλυσοδέοντας. Ανεπίσημα, η ιδέα του αλυσοδέοντας προς τα εμπρός είναι να εφαρμοστεί iteratively μια πιθανή υπηρεσία  $S$  σε ένα σύνολο παραμέτρων εισαγωγής που παρέχονται από μια ερώτηση  $Q$  (δηλ., όλες οι εισαγωγές που

απαιτούνται από την  $S$  πρέπει να είναι διαθέσιμες). Εάν ισχύει η  $S$  το πρόβλημα δεν λύνεται (δηλ., όλα τα αποτελέσματα που απαιτούνται από την ερώτηση  $Q$  δεν είναι διαθέσιμα) έπειτα μια νέα ερώτηση  $Q'$  μπορεί να είναι υπολογισμένη από τη  $Q$  και την  $S$  και ολόκληρη η διαδικασία επαναλαμβάνεται. Αυτό το μέρος του πλαισίου μας αντιστοιχεί στις τεχνικές προγραμματισμού που χρησιμοποιούνται αυτήν την περίοδο για τη σύνθεση υπηρεσιών [TKAS02]. Στην περίπτωση αλυσοδένοντας προς τα πίσω αρχίζουμε από το σύνολο παραμέτρων που απαιτούνται από την ερώτηση  $Q$  και σε κάθε βήμα της διαδικασίας επιλέγουμε μια υπηρεσία  $S$  που θα παράσχει τουλάχιστον μια από τις απαραίτητες παραμέτρους. Εφαρμόζοντας το αποτέλεσμα της  $S$  στις νέες παραμέτρους που απαιτούνται μπορεί να τυποποιηθεί ως νέα ερώτηση  $Q'$ . Πάλι η διαδικασία επαναλαμβάνεται έως ότου να βρεθεί μια λύση.

Τώρα θεωρούμε τους όρους που απαιτούνται για μια υπηρεσία  $S$  για να απευθυνθούμε στις διαθέσιμες εισαγωγές από μια ερώτηση  $Q$  που χρησιμοποιεί το αλυσοδένοντας προς τα εμπρός: για όλες τις εισαγωγές που απαιτούνται από την υπηρεσία  $S$ , πρέπει να υπάρξει μια συμβατή παράμετρος στις εισαγωγές που παρέχονται από την ερώτηση  $Q$ . Η συμβατότητα πρέπει να επιτευχθεί και για τους δύο ρόλους, όπου ο ρόλος οποιασδήποτε παραμέτρου παρεχόμενης από την ερώτηση το  $Q$  πρέπει να είναι σημασιολογικά πιο συγκεκριμένη (.) από το ρόλο της απαιτούμενης παραμέτρου από την υπηρεσία  $S$ , και για τους τύπους, όπου η σειρά που παρέχεται από την ερώτηση  $Q$  πρέπει να είστε πιο συγκεκριμένο (.) από αυτό που γίνεται αποδεκτό από την υπηρεσία  $S$ . Στο φορμαλισμό που εισάγεται επάνω από το αλυσοδένοντας προς τα εμπρός ο όρος θα χαρτογραφούσε στο κατηγορημα  $\text{all}_{\text{some}_Q}$ :

$\text{fwdComp}(Q, S) =$

$\text{all}_{\text{some}_Q}(\text{IN}_Q, \text{IN}_S \cdot \text{vrole} \cdot \text{vtype}) \wedge \text{all}_{\text{some}_Q}(\text{PRE}_Q, \text{PRE}_S, \square)$ .

Ένα παρόμοιο είδος αντιστοιχίας PlugIn μεταξύ των εισαγωγών της ερώτησης  $Q$  και της υπηρεσίας  $S$  έχει προσδιοριστεί από το  $\text{Paolluci}$  [PKPS02] για την αντιστοιχία των υπηρεσιών DAML-S.

Το *forward complete matching* των τύπων είναι πάρα πολύ περιοριστικό και η μπορεί να μη λειτουργεί πάντα, επειδή οι τύποι που γίνονται αποδεκτοί από τις διαθέσιμες υπηρεσίες μπορεί μερικώς να επικαλύψουν τον τύπο που διευκρινίζεται στην ερώτηση. Παραδείγματος χάριν, μπορεί ένα VTA να προσφέρει κράτηση σε κάποιο συστημένο

εστιατόριο κατά την κράτηση ενός πλήρους ταξιδιού διακοπών. Κατά τη χρησιμοποίηση ενός προμηθευτή που συστήνει εστιατορίων, μια ερώτηση που δίνεται από το VTA για τις υπηρεσίες σύστασης εστιατορίων σε ολόκληρη την Ελβετία θα μπορούσε να το διευκρινίσει ένας γρήγορος κώδικας παραμέτρου ακέραιων αριθμών που θα μπορούσε να είναι στη σειρά [1000.9999] ενώ μια υπάρχουσα υπηρεσία για την παροχή των συστάσεων για το γαλλόφωνο μέρος της Ελβετίας θα μπορούσε να δεχτεί μόνο ακέραιοι αριθμοί στη σειρά [1000-2999] για τον γρήγορο κώδικα παραμέτρου.

Μια σημαντική καινοτομία της προσέγγισής μας σχετικά με τη σύνθεση είναι δεδομένου ότι η πιο πάνω κατάσταση για το προς τα εμπρός αλυσοδέοντα τροποποιείται έτσι ώστε οι υπηρεσίες με τις μερικές αντιστοιχίες τύπων να υποστηρίζονται. Για να το επιτύχουμε χαλαρώνουμε το συνυπολογισμό τύπων σε μια απλή επικάλυψη:

$$fwdPart(Q, S) = all_{some_Q}(IN_Q, IN_S, \sqsubseteq_{role}, \sqsupseteq_{type}) \wedge all_{some_Q}(PRE_Q, PRE_S, \sqsubseteq).$$

Αυτό το είδος ταιριάσματος μεταξύ των εισαγωγών της ερώτησης Q και της υπηρεσίας S αντιστοιχεί στην αντιστοιχία επικάλυψης ή διατομής που προσδιορίζεται από το Li[LH03] και το Constantinescu [CF03].

Θα εξετάσουμε επίσης τον όρο που απαιτείται για την προσέγγιση του προς τα πίσω αλυσοδέοντα. Η υπηρεσία S πρέπει να παρέχει τουλάχιστον μια παραγωγή που απαιτείται από την ερώτηση Q. Αυτό αντιστοιχεί στην αντιστοιχία plugIn για τα αποτελέσματα ερώτησης και υπηρεσιών. Μπορεί να διευκρινιστεί όπως:

$$backComp(Q, S) = some_Q some_S (OUT_Q, OUT_S, \sqsupseteq_{role}, \sqsupseteq_{type}) \vee some_Q some_S (EFF_Q, EFF_S, \sqsupseteq).$$

**Συμβατός τύπος στη σύνθεση υπηρεσιών εναντίον του σχεδιασμού**

Δεδομένου ότι η πλειοψηφία των προσεγγίσεων σύνθεσης υπηρεσιών στηρίζεται σήμερα στο σχεδιασμό θα αναλύσουμε την επικοινωνία μεταξύ του φορμαλισμού μας για τις περιγραφές υπηρεσιών με τους τύπους και ένα υποθετικό φορμαλισμό σχεδιασμού που χρησιμοποιεί τα ελεύθερα σύμβολα στη πρωταρχική λογική για τις προϋποθέσεις και αποτελέσματα.

Για παράδειγμα (δείτε τον πίνακα που εξετάζει την περιγραφή *S* υπηρεσιών που έχει δύο παραμέτρους εισαγωγής *A* και *B* και δύο παραμέτρους *C* και *D* παραγωγής. Οι τύποι τους αντιπροσωπεύονται από τα σύνολα αποδεκτών και παρεχόμενων τιμών και είναι *a1*, *a2* για το *A*, αντίστοιχα *b1*, *b2* για το *B*, *c1*, *c2* για το *C*, και *d1*, *d2* για το *D*. Αυτό αντιστοιχεί σε έναν χειριστή *S* που έχει προϋποθέσεις και αποτελέσματα. Η άρνηση δεν απαιτείται.

```

S = {
    :action S
    :precondition
      (and
        {or a1 a2}
        {or b1 b2})
    :effect
      (and
        {or c1 c2}
        {or d1 d2})
}

```

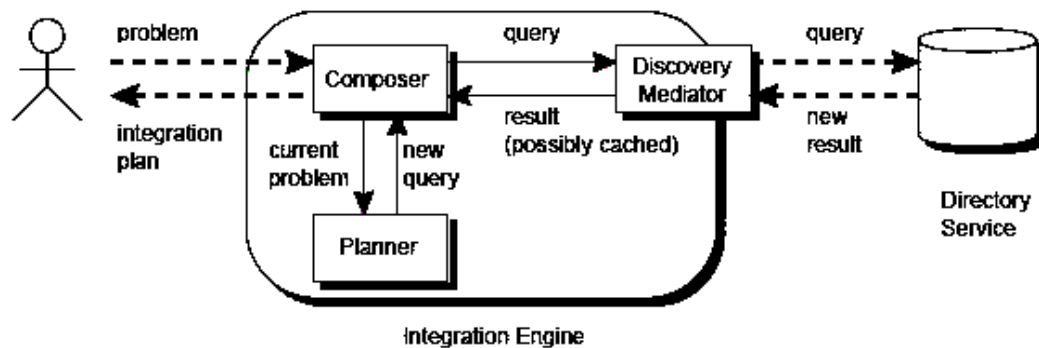
Σχήμα 5.1: Υπηρεσίες με τύπους και διάφορες αντιστοιχίες σχεδιασμού προγραμματισμού

### Υπολογίζοντας τους συμβατούς τύπους για τις συνθέσεις υπηρεσιών

Σε αυτό το τμήμα θα παρουσιάσουμε τους αλγορίθμους για τις συμβατούς τύπους για τις συνθέσεις υπηρεσιών. Το σχέδιό τους παρακινείται από δύο συγκεκριμένες πτυχές για τους καταλόγους υπηρεσιών μεγάλης κλίμακας λειτουργία στα ανοικτά περιβάλλοντα:

- Ø Μεγάλα σύνολα αποτελέσματος - για κάθε ερώτηση ο κατάλογος θα μπορούσε να επιστρέψει έναν μεγάλο αριθμό περιγραφών υπηρεσίας.
- Ø Δαπανηρές προσβάσεις καταλόγου - όντας ένας κοινός πόρος που έχει πρόσβαση στον κατάλογο (ενδεχομένως μακρινά) θα είναι ακριβό.

- ∅ Αντιμετωπίζουμε αυτά τα ζητήματα με την παρεμβολή λευκών σελίδων της ανακάλυψης και της σύνθεσης και με τον υπολογισμό η «σωστή» ερώτηση σε κάθε βήμα. Για αυτόν, η μηχανή ολοκλήρωσης χρησιμοποιεί τρία χωριστά συστατικά:
- ∅ Αρμόδιος για το σχεδιασμό - ένα συστατικό που υπολογίζει τι μπορεί να επιτευχθεί αυτήν την περίοδο από την τρέχουσα ερώτηση που χρησιμοποιεί το τρέχον σύνολο ανακαλυμμένων υπηρεσιών. Από αυτόν το πρόβλημα αυτό που παραμένει να λυθεί είναι να παραχθεί μια νέα ερώτηση και να επιστραφεί



Σχήμα 5.2: Αρχιτεκτονική των υπηρεσιών ενσωμάτωσης μηχανισμών

- ∅ Συνθέτης - ένα συστατικό που εφαρμόζει την παρεμβολή λευκών σελίδων μεταξύ του σχεδιασμού και της ανακάλυψης. Αποφασίζει ότι τι είδους οι ερωτήσεις (μερικές/πλήρεις) πρέπει να σταλούν στον κατάλογο και αυτό εξετάζει τα διακλαδωμένα σημεία και τη το να ξαναλυθούν τα υποπροβλήματα.
- ∅ Μεσολαβητής ανακαλύψεων - ένα συστατικό που μεσολαβεί τις προσβάσεις συνθετών στον κατάλογο με την εναποθήκευση των υπαρχόντων αποτελεσμάτων και το ταίριασμα των νέων ερωτήσεων των ήδη ανακαλυμμένων υπηρεσιών.

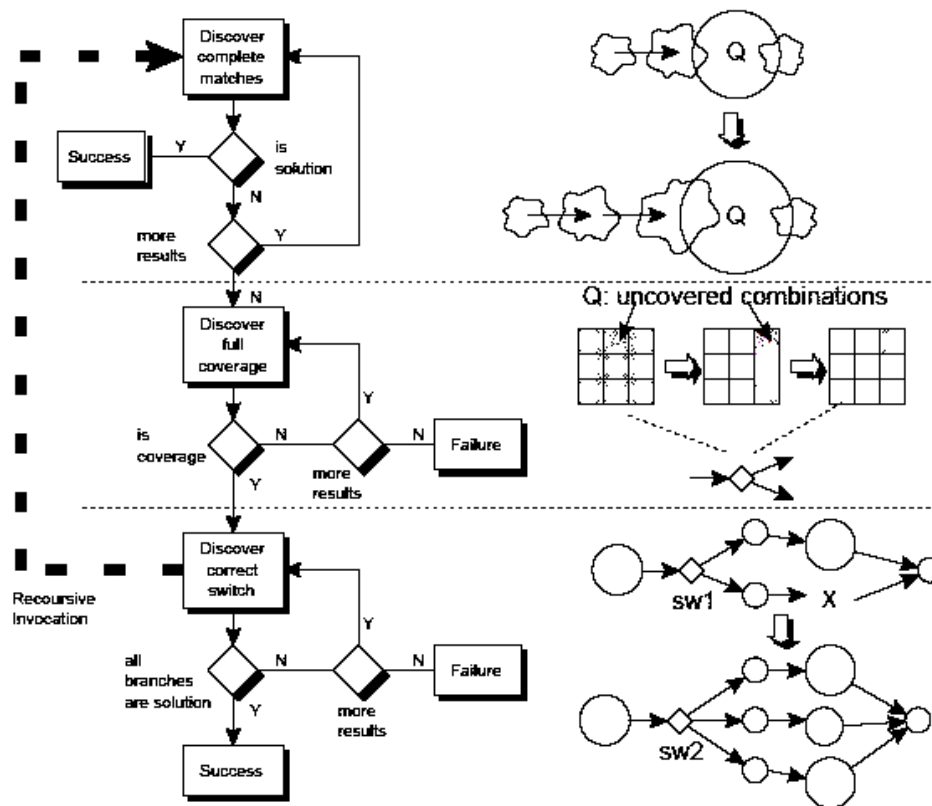
### Σύνθεση με τις πλήρεις αντιστοιχίες τύπων

Η σύνθεση των απόλυτα ταιριασμένων υπηρεσιών που χρησιμοποιούν το προς τα εμπρός αλυσοδένοντας είναι απλή: όταν η κατάσταση για τις πλήρεις αντιστοιχίες τύπων είναι ολοκληρωμένος (όλες οι εισαγωγές που απαιτούνται από την υπηρεσία  $S$  είναι παρών στην ερώτηση  $Q$  και οι τύποι στην ερώτηση είναι πιο συγκεκριμένα από τους τύπους που γίνονται αποδεκτοί από την υπηρεσία) μια νέα ερώτηση  $Q'$  μπορεί να υπολογιστεί με την προσθήκη συνόλου διαθέσιμων εισαγωγών της τρέχουσας ερώτησης  $Q$  όλα τα αποτελέσματα που παρέχονται από την υπηρεσία  $S$ .

### **Σύνθεση με τις μπροστινές μερικές αντιστοιχίες τύπων**

Εννοιολογικά ο αλγόριθμος που χρησιμοποιούμε για τη σύνθεση υπηρεσιών με μερικούς μπροστινούς τύπους οι αντιστοιχίες έχουν τρία βήματα (για περισσότερες λεπτομέρειες δείτε [CFB04b]):

- Ø Ανακάλυψη για να ταιριάζει εντελώς με τις υπηρεσίες.
- Ø Ανακάλυψη των υπηρεσιών για την πλήρη κάλυψη των διαθέσιμων εισαγωγών.
- Ø Ανακάλυψη των υπηρεσιών για το σωστό χειρισμό διακοπών



Σχήμα 5.3: Ροή αλγορίθμου με σύνθεση μερικών αντιστοιχίσεων τύπων

**Ανακαλύπτοντας την πλήρη κάλυψη εισαγωγής** Το δεύτερο βήμα του αλγορίθμου υποθέτει ότι δεν βρέθηκε ότι μια λύση χρησιμοποιείται μόνο στις πλήρεις αντιστοιχίες και ότι υπηρεσίες με το μερικό τύπο αντιστοιχίας χρησιμοποιούνται προκειμένου να λυθεί το πρόβλημα. Εξ ορισμού οποιαδήποτε από τις μερικά ταιριασμένες υπηρεσίες είναι σε θέση να χειριστούν μόνο περιορισμένο υπό-διάστημα των διαθέσιμων τιμών των εισαγωγών. Προκειμένου εξασφαλιστεί ότι οποιοσδήποτε συνδυασμός αξίας εισαγωγής μπορεί να χειρίζεται, το διάστημα των διαθέσιμων εισαγωγών discretized αρχικά στα κελιά αξίας της παραμέτρου. Ένα κελί είναι ένα ορθογώνιο υπερδιάστημα που περιέχει όλες τις διαστάσεις του διαστήματος των διαθέσιμων εισαγωγών αλλά μόνο ένα ενιαίο διάστημα για κάθε διάσταση. Ένα κελί αντιστοιχεί στον όρο φρουράς διακόπτης. Τα κελιά χτίζονται κατά τέτοιο τρόπο ώστε οποιοσδήποτε από τις απαραίτητες εισαγωγές για τις μερικώς ανακτημένες ταιριασμένες υπηρεσίες θα μπορούσαν να εκφραστούν ως συλλογή των κελιών. Κάθε μια από ανακτημένη μερικώς να ταιριάζει με τις υπηρεσίες ορίζεται στα κελιά που μπορεί να δεχτεί ως εισαγωγή. Η κάλυψη θεωρείται πλήρης όταν

διορίσουν όλα τα κελιά μια ή περισσότερες υπηρεσίες. Όταν όλοι τα κελιά καλύπτονται ο αλγόριθμος περνάει στο επόμενο βήμα. Εάν όχι άλλες μερικώς ταιριασμένες υπηρεσίες μπορούν να βρεθούν και μια πλήρης κάλυψη δεν επιτεύχθηκε ο αλγόριθμος επιστρέφει αποτυχία.

Ανακαλύπτοντας τη λύση μεταστρέψτε το τελευταίο βήμα του αλγορίθμου υποθέτει ότι μια κάλυψη ήταν βρήκε και ένας πρώτος διακόπτης μπορεί να δημιουργηθεί. Ο στόχος αυτού του βήματος είναι να εξασφαλιστεί ότι ο διακόπτης θα λειτουργήσει σωστά για κάθε έναν από τους κλάδους του. Για κάθε κύτταρο και το σύνολο ορισμένος του οι υπηρεσίες ο αλγόριθμος θα υπολογίσουν το σύνολο παραμέτρων παραγωγής που εκείνες οι υπηρεσίες παρέχετε. Κατόπιν μια νέα ερώτηση υπολογίζεται, έχοντας ως διαθέσιμες εισαγωγές τις παραμέτρους παραγωγής από το κύτταρο και όπως απαιτείται τα αποτελέσματα το σύνολο απαραίτητων εξαγόμενων πληροφοριών του πλήρους ταιριάσματος φάση. Ολόκληρη η διαδικασία σύνθεσης επικαλείται έπειτα κατ' επανάληψη. Στην περίπτωση που όλα τα κύτταρα επιστρέφουν ένα επιτυχές αποτέλεσμα που ο διακόπτης θεωρείται σωστός και ο αλγόριθμος επιστροφής επιτυχία. Διαφορετικά μια νέα υπηρεσία ανακτάται και η διαδικασία συνεχίζεται.

#### **5.2.4 Τεχνικές εφαρμογής (υποστήριξη καταλόγου για την αυτοματοποιημένη σύνθεση υπηρεσιών)**

Εδώ δίνουμε μια επισκόπηση των τεχνικών εφαρμογής καταλόγων υπηρεσιών στην εξελικτική υποστήριξη και στην αποδοτική αυτοματοποιημένη σύνθεση υπηρεσιών, συμπεριλαμβανομένων των τεχνικών για τη σύνταξη πολυδιάστατου ευρετηρίου, η υποστήριξη για τα μεγάλα σύνολα αποτελέσματος (επαυξητική ανάκτηση των αποτελεσμάτων), ο αποδοτικός έλεγχος συναγωνισμού, και η υποστήριξη για τη καθορισμένη από το χρήστη της αναζήτησης που ανακαλύφθηκε [CBF04b, CBF04a, BCF04].



### **Πολυδιάστατες μέθοδοι προσπέλασης – GiST**

Η ανάγκη για αποδοτική ανακάλυψη και αντιστοιχία οδηγεί σε ανάγκη για τις δομές αναζήτησης και δείκτες για τους καταλόγους. Εξετάζουμε τις αριθμητικά κωδικοποιημένες περιγραφές υπηρεσιών όπως είναι τα πολυδιάστατα στοιχεία και τη χρήση τεχνικών που αφορούσαν την εύρεση τέτοιου είδους πληροφοριών στον κατάλογο. Ο δείκτης καταλόγου είναι βασισμένος στο γενικευμένο δέντρο αναζήτησης (Gist), που προτείνεται ως πλαίσιο από το Hellerstein [HNP95]. Η αρχή του σχεδίου του GiST προκύπτει από την παρατήρηση που ψάχνει τα δέντρα που χρησιμοποιούνται στις βάσεις δεδομένων είναι ισορροπημένα δέντρα με υψηλό fanout στο οποίο οι εσωτερικοί κόμβοι χρησιμοποιούνται ως κατάλογος και οι κόμβοι φύλλων δείχνουν τα πραγματικά στοιχεία.

Κάθε εσωτερικός κόμβος κρατά ένα κλειδί υπό μορφή κατηγορήματος P και διάφορων δεικτών σε άλλους κόμβους (ανάλογα με τους περιορισμούς συστημάτων και υλικού, π.χ., το μέγεθος σελίδων συστήματος αρχείου). Για να ψάξουν για τα αρχεία που ικανοποιούν ένα κατηγορήμα Q, οι πορείες ερώτησης του δέντρου που έχουν τα κλειδιά P που ικανοποιεί το Q ακολουθούνται.

Πιο συγκεκριμένα, κάθε κόμβος φύλλων στην GiSP του καταλόγου κρατά τις αναφορές σε όλες περιγραφές υπηρεσιών με μια ορισμένη συμπεριφορά εισόδου-εξόδου. Οι απαραίτητες εισαγωγές της υπηρεσίας και τα παρεχόμενα αποτελέσματα (σύνολα ονομάτων παραμέτρου με τους σχετικούς τύπους) αποθηκεύονται στον κόμβο φύλλων. Για τους εσωτερικούς κόμβους του δέντρου, η ένωση όλων των εισαγόμενων πληροφοριών/ των εξαγόμενων πληροφοριών που βρίσκονται στο υποδέντρο αποθηκεύονται. Ακριβέστερα, κάθε εσωτερικός κόμβος I στην πορεία σε έναν κόμβο φύλλων L περιέχει όλες τις παραμέτρους που αποθηκεύουν όλες τις εισόδους-εξόδους στο L. Ο τύπος που συνδέεται με μια παράμετρο στο I εντάσσει τον τύπο της παραμέτρου στο L. Δηλαδή για έναν εσωτερικό κόμβο, οι παράμετροι εισόδου-εξόδου προσδιορίζουν ποιες συγκεκριμένες παράμετροι μπορούν να βρεθούν σε έναν κόμβο άδειας υποδέντρου. Εάν μια παράμετρος δεν είναι παρούσα σε έναν εσωτερικό κόμβο, δεν θα είναι παρούσα σε οποιοδήποτε κόμβο άδειας υποδέντρου.

**Σύνοδοι ολοκλήρωσης υπηρεσιών και έλεγχος συναγωνισμού** Δεδομένου ότι οι ερωτήσεις καταλόγου μπορούν να ανακτήσουν τους μεγάλους αριθμούς ταιριάσματος των καταχωρήσεων (ειδικά όταν οι μερικές αντιστοιχίες λαμβάνονται υπόψη), είναι σημαντικό να υποστηριχθεί η επαυξητική πρόσβαση στα αποτελέσματα μιας ερώτησης προκειμένου να αποφύγει το εύρος ζώνης δικτύων. Ο κατάλογος υπηρεσίας προσφέρει τις συνόδους που επιτρέπουν σε έναν χρήστη για να διανείμουν τις ερωτήσεις στον κατάλογο και για να ανακτήσουν τα αποτελέσματα ένα-ένα (ή στα χοντρά κομμάτια του περιορισμένου μεγέθους).

Η σύνοδος εγγυάται μια συνεπή άποψη του καταλόγου, δηλ., τη δομή του καταλόγου και το περιεχόμενο όπως φαίνεται από μια σύνοδο δεν αλλάζει. Ταυτόχρονες αναπροσαρμογές (εγγραφή υπηρεσιών, η αναπροσαρμογή, και η αφαίρεση) δεν έχουν επιπτώσεις στην ακολουθία αποτελεσμάτων ερώτησης που επιστρέφονται μέσα στη σύνοδο; οι σύνοδοι είναι απομονωμένες από τις ταυτόχρονες τροποποιήσεις.

Η προηγούμενη ερευνητική εργασία έχει εξετάσει τον έλεγχο συναγωνισμού στη γενικευμένη αναζήτηση στα δέντρα [ΚΜΗ97].Εντούτοις, αυτοί οι μηχανισμοί ελέγχου συναγωνισμού συγχρονίζουν μόνο ατομικές διαδικασίες στο δέντρο, ενώ ο κατάλογός μας υποστηρίζει τις μακράς διάρκειας συνόδους κατά τη διάρκεια των όποιων ορισμένα μέρη της δομής των δέντρων δεν πρέπει να αλλάζουν. Αυτό υπονοεί εκείνη την εισαγωγή και οι διαδικασίες διαγραφής δεν μπορούν να εκτελεστούν ταυτόχρονα με τις συνόδους ερώτησης, όπως αυτές οι διαδικασίες μπορούν σημαντικά να αλλάξουν τη δομή του δέντρου (διαχωρισμός ή ένωση των κόμβων, ισορροπώντας το δέντρο, κ.λπ.).

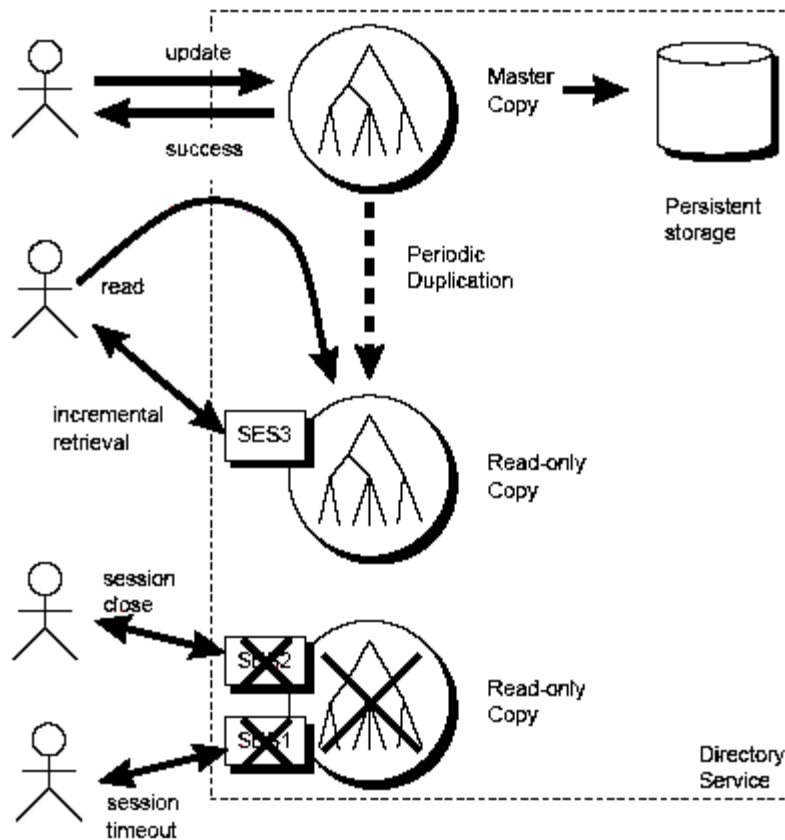
Οι ακόλουθες υποθέσεις underly το σχέδιο του μηχανισμού ελέγχου του συναγωνισμού:

1. Διαβασμένες προσβάσεις (δηλ., ερωτήσεις μέσα στις συνόδους και η επαυξητική ανάκτηση αποτελεσμάτων) θα είναι συχνότερος από τις αναπροσαρμογές.
2. Υψηλός συναγωνισμός για τις διαβασμένες προσβάσεις (υψηλός αριθμός ταυτόχρονων ερωτήσεων συνόδων).
3. Τις διαβασμένες προσβάσεις δεν πρέπει καθυστερούν.

4. Οι αναπροσαρμογές μπορούν να γίνουν ορατές με μια σημαντική καθυστέρηση, αλλά ανατροφοδοτούνται σχετικά με την αναπροσαρμογή (επιτυχία/ αποτυχία) που πρέπει να επιστραφούν αμέσως.

5. Η διάρκεια μιας συνόδου μπορεί να περιοριστεί (διάλειμμα).

Προκειμένου να συναντηθούν οι ανωτέρω υποθέσεις, έχουμε σχεδιάσει έναν μηχανισμό που εγγυάται ότι οι συνοδοί αναπτύσσουν δραστηριότητες μόνο στην ανάγνωση δομών δεδομένων που δεν υπόκεινται στις αλλαγές. Στην προσέγγισή μας η δομή -μνήμης του δέντρου καταλόγου (δηλ., ο δείκτης καταλόγου) ξαναδιπλώνεται μέχρι 3 φορές, ενώ οι πραγματικές περιγραφές υπηρεσιών μοιράζονται μεταξύ ξαναδιπλωμένων δέντρων.



Σχήμα 5.4 : Περίοδος κύκλου ζωής.

Όταν η υπηρεσία καταλόγου αρχίζει, η επίμονη αντιπροσώπευση του δέντρου καταλόγου φορτώνεται στη μνήμη. Αυτό το πρότυπο του δέντρου καταλόγου ενημερώνεται πάντα, δηλ., οι αναπροσαρμογές εφαρμόζονται αμέσως σε εκείνο το πρότυπο και γίνονται επίμονες, επίσης. Επάνω στην έναρξη της υπηρεσίας καταλόγου, ένα μόνο αντίγραφο ανάγνωσης του προτύπου μνήμης διατίθεται. Οι σύνοδοι αναπτύσσουν δραστηριότητες μόνο σε αυτό το αντίγραφο ανάγνωσης. Ως εκ τούτου, η διαχείριση συνόδου είναι τετριμμένος, δεν υπάρχει καμία ανάγκη συγχρονισμού. Περιοδικά, το πρότυπο αναπαράγεται δημιουργήστε ένα νέο αντίγραφο μόνο για ανάγνωση. Κατόπιν, οι νέες σύνοδοι επαναπροσανατολίζονται στο νέο μοναδικό αντίγραφο ανάγνωσης. Η συλλογή απορριμμάτων ελευθερώνει το παλαιό μοναδικό αντίγραφο ανάγνωσης κατά την τελευταία λειτουργία συνόδου το ολοκληρώνει (είτε από μια ρητή λήξη συνόδου από τον πελάτη είτε από ένα διάλειμμα).

Απαιτούμε το διάλειμμα συνόδου για να είναι μικρότερο από τη συχνότητα αναπροσαρμογών του μοναδικού αντιγράφου ανάγνωσης (η συχνότητα διπλασιασμού του προτύπου). Αυτός ο όρος εξασφαλίζει ότι θα υπάρξουν το πολύ-πολύ 3 αντίγραφα της αντιπροσώπευσης -μνήμης του καταλόγου στον ίδιο χρόνο: Ο κύριος όπου οι αναπροσαρμογές εφαρμόζονται αμέσως (αλλά που δεν είναι ακόμα ορατός σύνοδος), καθώς επίσης και τα προηγούμενα 2 αντίγραφα ανάγνωσης που χρησιμοποιούνται για τις συνόδους. Όταν νέο αντίγραφο που είναι μόνο για ανάγνωση δημιουργείται, το παλαιό αντίγραφο θα παραμείνει ενεργό μέχρι η τελευταία σύνοδος το ολοκληρώσει; αυτή τη φορά η έκταση είναι οριακή από το διάλειμμα συνόδου.

Στην προσέγγισή μας μόνο οι αναπροσαρμογές στο πρότυπο είναι συγχρονισμένες. Οι αναπροσαρμογές είναι αμέσως εφαρμοσμένες στο πρότυπο. Μόνο κατά τη διάρκεια της αντιγραφής ο κατάλογος εμποδίζεται για τις περαιτέρω αναπροσαρμογές. Στη συμφωνία με την τρίτη υπόθεση, η δημιουργία των συνόδων δεν απαιτεί κανέναν συγχρονισμό.

### **Περικοπή συνήθειας και ταξινόμηση των λειτουργιών**

Δεδομένου ότι οι ερωτήσεις καταλόγου μπορούν να ανακτήσουν τους μεγάλους αριθμούς ταιριάσματος των καταχωρήσεων (ειδικά όταν οι μερικές αντιστοιχίες λαμβάνουν υπόψη), οι σύνοδοι υποστήριξης καταλόγου μας προκειμένου να υπάρχει επαυξητική πρόσβαση στα αποτελέσματα μιας ερώτησης [CBF04b]. Εξ ορισμού, η

διαταγή στην οποία οι ταιριάζοντες περιγραφές υπηρεσιών επιστρέφονται εξαρτώνται από την πραγματική δομή του δείκτη καταλόγου(η δομή ουσίας που συζητείται πριν). Εντούτοις, ανάλογα με την υπηρεσία ο αλγόριθμος ολοκλήρωσης, που διατάζει τα αποτελέσματα μιας ερώτησης σύμφωνα με ορισμένο heuristics μπορεί σημαντικά να βελτιώσει την απόδοση της σύνθεσης υπηρεσιών. Προκειμένου να αποφευχθεί η μεταφορά από έναν μεγάλο αριθμό περιγραφών υπηρεσιών, της περικοπής, που ταξινομούν, και της ταξινομημένης χορήγησης εξαρτημένο από την εφαρμογή heuristics πρέπει να εμφανιστεί άμεσα μέσα στον κατάλογο.

**Το API για την περικοπή και την ταξινόμηση λειτουργιών** Η περικοπή και η ταξινόμηση των λειτουργιών λαμβάνονται σαν πληροφορίες επιχειρημάτων σχετικά με το ταίριασμα μιας περιγραφής υπηρεσιών στον κατάλογο με την τρέχουσα ερώτηση. Επιστρέφουν μια αξία που αντιπροσωπεύει την ποιότητα της αντιστοιχίας. Μεγαλύτερη η επιστροφής αξία, όσο καλύτερη η περιγραφή των υπηρεσιών ταιριάζει με τις απαιτήσεις ερώτηση. Η ακολουθία αποτελεσμάτων όπως επιστρέφεται από τον κατάλογο ταξινομείται με την κατιούσα σειρά από τις τιμές που υπολογίζονται από την περικοπή και η ταξινόμηση λειτουργεί (μια υψηλότερη αξία σημαίνει καλύτερη αντιστοιχία). Τα αποτελέσματα για τα οποία οι λειτουργίες αξιολογούν σε μηδέν έρχονται στο τέλος, μια αρνητική αξία δείχνει ότι η αντιστοιχία είναι πάρα πολύ φτωχή για να επιστραφεί, δηλ., το αποτέλεσμα απορρίπτεται και δεν περνάει στον πελάτη (περικοπή).

Σαν επιχειρήματα η καθορισμένη περικοπή του πελάτη και η ταξινόμηση των λειτουργιών παίρνουν τέσσερα ParamSet αντίθετα στα σύνολα παραμέτρων εισαγωγής και παραγωγής της ερώτησης, και αντίστοιχα από την υπηρεσία. Το αντικείμενο ParamSet παρέχει μεθόδους όπως το μέγεθος, ιδιότητα μέλους, ένωση, διατομή, και διαφορά. Οι μέθοδοι μεγέθους και ιδιότητας μέλους απαιτούν μόνο το τρέχον αντικείμενο ParamSet, ενώ η ένωση, η διατομή, και οι μέθοδοι διαφοράς χρησιμοποιούν δύο αντικείμενα ParamSet - το τρέχον αντικείμενο και ένα δεύτερο αντικείμενο ParamSet πέρασαν σαν επιχειρήματα.

Είναι σημαντικό να σημειωθεί ότι μερικές από τις ανωτέρω μεθόδους αντιμετωπίζουν δύο διαφορετικά ζητήματα στον ίδιο χρόνο:

1. Βασικές καθορισμένες διαδικασίες, όπου ένα καθορισμένο μέλος καθορίζεται από ένα όνομα παραμέτρου και τον τύπο του; για την απόφαση της ισότητας των παραμέτρων με το ίδιο όνομα και τους διαφορετικούς τύπους η καθορισμένη ως προς τον χρήστη έκφραση χρησιμοποιείται.

2. Ο υπολογισμός των νέων τύπων για μερικές παραμέτρους στα προκύπτοντα σύνολα; όταν μια παράμετρος είναι κοινή για δύο σύνολα επιχειρήματος ο τύπος τους στο προκύπτον αποτέλεσμα του συνόλου υπολογίζεται με μια καθορισμένη ως προς τον χρήστη έκφραση.

Η ρητή συμπεριφορά των μεθόδων ParamSet είναι η ακόλουθη:

**μέγεθος:** Επιστρέφει τον αριθμό παραμέτρων στο τρέχον σύνολο.

**containsParam:** Επιστρέφει αληθινό εάν το τρέχον σύνολο περιέχει μια παράμετρο με το ίδιο όνομα σαν επιχειρήμα μεθόδου (ανεξάρτητα από τον τύπο του).

**ένωση:** Επιστρέφει την ένωση των παραμέτρων στα δύο σύνολα. Για κάθε παράμετρο που είμαι κοινή για το επιχειρήμα τον τύπο στο προκύπτον σύνολο υπολογίζεται σύμφωνα με την καθορισμένη ως προς τον χρήστη έκφραση newTypeExpr.

**διατομή:** Επιστρέφει τις παράμετρος που είμαι κοινές για δύο σύνολα ΚΑΙ για που οι αντίστοιχοι τύποι προσαρμόζονται στη δοκιμή ισότητας που διευκρινίζεται από το eqTestExpr. Ο τύπος αυτών των παραμέτρων στο προκύπτον σύνολο υπολογίζεται σύμφωνα με την καθορισμένη ως προς τον χρήστη έκφραση newTypeExpr.

**αρνητικός:** Επιστρέφουν την παράμετρο που είμαι παρούσα μόνο στο τρέχων σύνολο και στην περίπτωση που οι κοινές παράμετροι δεν προσαρμόζονται στη δοκιμή ισότητας που διευκρινίζεται από το eqTestExpr. Για το τελευταίο είδος παραμέτρων ο τύπος στο προκύπτον σύνολο υπολογίζεται σύμφωνα με την καθορισμένη ως προς τον χρήστη έκφραση newTypeExpr.

Παράμετροι των οποίων ο newTypeExpr θα ήταν ο κενός τύπος (αποκαλούμενος ΤΙΠΟΤΑ(NOTHING) στον κατωτέρω πίνακα) αφαιρείται από το προκύπτον σύνολο.

Οι εκφράσεις που χρησιμοποιούνται στο eqTestExpr και newTypeExpr οι παράμετροι έχουν ίδιο σχήμα και εφαρμόζονται στις παραμέτρους που είναι κοινές για τα δύο αντικείμενα ParamSet που πέρασαν σε μια ένωση, μια διατομή, ή μια αρνητική μέθοδο. Για τέτοιο είδος παραμέτρου δείχνουμε τον τύπο στα δύο σύνολα επιχειρήματος ως A και B. Οι εκφράσεις δημιουργούνται από αυτούς τους δύο τύπους με τη χρησιμοποίηση

μερικών πρόσθετων κατασκευαστών βασισμένων στη λογική γλωσσική περιγραφή OWL [DS04] όπως  $\top$ ,  $\perp$ ,  $\neg$ ,  $\sqcap$ ,  $\sqcup$ ,  $\sqsubseteq$ ,  $\equiv$ . Οι εκφράσεις χτίζονται με τη διευκρίνιση ενός τύπου κατασκευαστών και οι τύποι A ή B πρέπει να αρνηθούν. Για ενιαίους κατασκευαστές τύπων  $\top$  και  $\perp$  η άρνηση δεν μπορεί να διευκρινιστεί και για τους κατασκευαστές A και το B η άρνηση επιτρέπεται μόνο για τον αντίστοιχο τύπο (π.χ., για τον τύπο κατασκευαστών A, μόνο: Το A μπορεί να τεθεί).

Constructor type	$\neg A?$	$\neg B?$	Possible expressions
<b>THING</b>	-	-	$\top$
<b>NOTHING</b>	-	-	$\perp$
<b>A</b>	Y/N	-	$A, \neg A$
<b>B</b>	-	Y/N	$B, \neg B$
<b>UNION</b>	Y/N	Y/N	$A \sqcup B, A \sqcup \neg B, \neg A \sqcup B, \neg A \sqcup \neg B$
<b>INTERSECTION</b>	Y/N	Y/N	$A \sqcap B, A \sqcap \neg B, \neg A \sqcap B, \neg A \sqcap \neg B$
<b>SUBCLASS</b>	Y/N	Y/N	$A \sqsupseteq B, A \sqsupseteq \neg B, \neg A \sqsupseteq B, \neg A \sqsupseteq \neg B$
<b>SUPERCLASS</b>	Y/N	Y/N	$A \sqsubseteq B, A \sqsubseteq \neg B, \neg A \sqsubseteq B, \neg A \sqsubseteq \neg B$
<b>SAMECLASS</b>	Y/N	Y/N	$A \equiv B, A \equiv \neg B, \neg A \equiv B, \neg A \equiv \neg B$

### Οι υποδειγματικές λειτουργίες ταξινόμησης

Στο παράδειγμα του πιο κάτω σχήματος παρουσιάζεται η λειτουργία της βασικής ταξινόμησης δύο λειτουργούν, το πρώτο πιο κατάλληλο για τη χρησιμοποίηση αλγορίθμων στη σύνθεση υπηρεσιών είναι το αλυσοδένοντας προς τα εμπρός(που εξετάζει μόνο τις πλήρεις αντιστοιχίες), ο δεύτερος για τη χρησιμοποίηση αλγορίθμων τη προς τα πίσω αλυσοδένοντας. Παρατηρείστε την αποδυνάμωση των όρων περικοπής για τους εσωτερικούς κόμβους.

**Ασφαλής και αποδοτική εκτέλεση της ταξινόμησης των λειτουργιών**  
Χρησιμοποιώντας ένα υποσύνολο της Java σε γλώσσα προγραμματισμού για την περικοπή και την ταξινόμηση των λειτουργιών έχει διάφορα πλεονεκτήματα: Η Java είναι γνωστή σε πολλούς προγραμματιστές, υπάρχουν μέρη των εργαλείων προγραμματισμού για την Java, και, προ πάντων, ενσωματώνει πολύ καλά με την υπηρεσία καταλόγου μας, η οποία γράφεται στην Java.

Λειτουργίες σύνταξης και καθορισμένες από το χρήστη ταξινόμησης ενσωματώνονται στις δυνάμεις καταλόγου βελτιώνοντας τις καταστάσεις προόδου στις πρόσφατες εφαρμογές JVM. Για παράδειγμα, η δυναμική ζώνη VM [Sun] αρχικά ερμηνεύει το JVM κώδικα των bit [LY99] και συλλέγει τις στατιστικές εκτέλεσης. Εάν κώδικας εκτελείται αρκετά συχνά, συντάσσεται στο βελτιωμένο εγγενή κώδικα για τη γρήγορη εκτέλεση. Κατά αυτόν τον τρόπο, η συχνά χρησιμοποιημένη περικοπή και η ταξινόμηση των λειτουργιών εκτελούνται τόσο αποτελεσματικά όπως αλγόριθμοι που χτίζονται άμεσα στον κατάλογο.

Η κατηγορία που περιέχει τη λειτουργία ταξινόμησης αναλύεται από το πρόσθετο ελεγκτή του κώδικα byte μας ο οποίος εξασφαλίζει ότι η καθορισμένη από το χρήστη ταξινομημένη λειτουργία πάντα ολοκληρώνω μέσα σε μια καθορισμένη με σαφήνεια χρονική έκταση και δεν παρεμποδίζει την εφαρμογή καταλόγου. Η αποδοτικά εκτεταμένη επαλήθευση του κώδικα byte για να επιβάλει τους περιορισμούς σε για την ασφαλή εκτέλεση από τον μη έμπιστο κινητό κώδικα έχει μελετηθεί στο JavaSeal [VBB98] και J-SEAL2 [Bin01, BHVV01] κινητοί πυρήνες αντικειμένου. Ο ελεγκτής του κώδικα byte εξασφαλίζει μετά από τους όρους:

- Ø Να εφαρμόζεται η ταξινομημένη διεπαφή.
- Ø Να παρέχονται μόνο οι μέθοδοι των ταξινομημένων διεπαφών.
- Ø Οι γραφικές παραστάσεις ελέγχου ροής του rankLeaf () και οι μέθοδοι rankInner () είναι άκυκλικοι. Οι γραφικές παραστάσεις έλεγχου ροής δημιουργούνται από έναν αποδοτικό αλγόριθμο με την εκτέλεση γραμμικού χρόνου με τις οδηγίες JVM στη μέθοδο.
- Ø Κανένας χειριστής εξαίρεσης (που χρησιμοποιεί τους όχι καλά σχηματισμένους χειριστές εξαίρεσης, ορισμένοι άπειροι βρόχοι μπορεί να κατασκευαστούν που δεν ανιχνεύονται από τον τυποποιημένο ελεγκτή της Java, όπως παρουσιάζεται [BR02]). Εάν η λειτουργία ταξινόμησης έχει μια εξαίρεση (π.χ., λόγω ενός τμήματος κοντά μηδέν), το αποτέλεσμα του πρόκειται να θεωρηθούν μηδέν από τον κατάλογο.
- Ø Καμία υπορουτίνα JVM (μπορεί να προκύψουν από τη σύνταξη των προτάσεων finallyfg).



- Ø Καμία ρητή κατανομή αντικειμένου. Δεδομένου ότι υπάρχει κάποια υπονοούμενη κατανομή αντικειμένου στο σύνολο διαδικασιών σε ParamSet, ο αριθμός καθορισμένων διαδικασιών και το μέγιστο μέγεθος από τα προκύπτοντα σύνολα είναι περιορισμένος.
- Ø Μόνο οι μέθοδοι διεπαφών ParamSet μπορούν να επικαλεσθούν, όπως επίσης και τα καθορισμένα με σαφήνεια σύνολα μεθόδων από την τυποποιημένη συσκευασία μαθηματικών.

```

public final class RankingForward implements Ranking {
    public double rankLeaf( ParamSet qin, ParamSet qout,
                           ParamSet sin, ParamSet sout ) {
        // discard service if it requires parameters that are not in the query;
        // the provided input has to be more specific than the required one
        if (sin.minus(qin, ParamSet.SUBCLASS, ParamSet.A).size() > 0) return -1.0d;

        // services that provide more required parameters are better;
        // the provided output has to be more specific than the required one
        return (double)
            sout.intersection(qout, ParamSet.SUPERCLASS, ParamSet.A).size();
    }

    public double rankInner( ParamSet qin, ParamSet qout,
                            ParamSet sin, ParamSet sout ) {
        // for forward chaining, pruning inner nodes is not possible,
        // but an upper bound of the overlap of the outputs can be easily computed
        return (double)
            sout.intersection(qout, ParamSet.INTERSECTION, ParamSet.A).size();
    }
}

public final class RankingBackward implements Ranking {
    public double rankLeaf( ParamSet qin, ParamSet qout,
                           ParamSet sin, ParamSet sout ) {
        // discard service if it does not provide any required output
        if (sout.intersection(qout, ParamSet.SUPERCLASS, ParamSet.A).size() == 0)
            return -1.0d;

        // services that reduce most the number of required outputs are better
        ParamSet remaining = qout.minus(sout, ParamSet.SUBCLASS, ParamSet.A);
        ParamSet newRequired = sin.minus(qin, ParamSet.SUBCLASS, ParamSet.A);
        ParamSet required = remaining.union(newRequired, ParamSet.INTERSECTION);
        return 1 / (double) (1+required.size());
    }

    public double rankInner( ParamSet qin, ParamSet qout,
                            ParamSet sin, ParamSet sout ) {
        if (sout.intersection(qout, ParamSet.INTERSECTION, ParamSet.A).size() == 0)
            return -1.0d;

        ParamSet remaining = qout.minus(sout, ParamSet.INTERSECTION, ParamSet.A);
        return 1 / (double) (1+remaining.size());
    }
}

```

Σχήμα 5.5: Υποδειγματική περικοπή και την ταξινόμηση των λειτουργιών

- Ø Μόνο οι στατικοί τομείς που καθορίζονται στη διεπαφή ParamSet μπορούν να προσεγγιστούν.
- Ø Κανένας τομέας δεν καθορίζεται.
- Ø Καμία οδηγία συγχρονισμού.

Αυτοί οι περιορισμοί εξασφαλίζουν ότι ο χρόνος εκτέλεσης της περικοπής και της λειτουργίας ταξινόμησης είναι οριακή από το μέγεθος του κώδικά της. Ως εκ τούτου, ένας επιτιθέμενος δεν μπορεί να συντρίψει τον κατάλογο με την παροχή, παραδείγματος χάριν, μιας περικοπής και την ταξινόμηση της λειτουργίας που περιέχει έναν ατελείωτο βρόχο. Επιπλέον, αυτές οι λειτουργίες δεν μπορούν να διαθέσουν τη μνήμη. Η εκτεταμένη επαλήθευσή του κώδικα byte του αλγόριθμου μας είναι ιδιαίτερα αποδοτική, η απόδοσή της είναι γραμμική με το μέγεθος της περικοπής και ταξινόμηση των μεθόδων. Σαν πρόληψη ενάντια στις επιθέσεις άρνηση υπηρεσιών, η υπηρεσία καταλόγου μας επιτρέπει να θέσει ένα όριο για το μέγεθος των λειτουργιών συνήθειας.

Η περικοπή και η ταξινόμηση των λειτουργιών φορτώνονται από τα χωριστά classloaders, προκειμένου να υποστηρίξουν πολλαπλάσιες εκδόσεις των κατηγοριών με το ίδιο όνομα (που αποφεύγει τις διαφωνίες ονόματος μεταξύ των πολλαπλάσιων πελατών) και για να επιτρέψει τη συλλογή απορριμμάτων των δομών κατηγορίας. Η εφαρμογή καταλόγου (που φορτώνεται από το σύστημα classloader) έχει πρόσβαση καθορισμένες λειτουργίες από το χρήστη μόνο μέσω της ταξινομημένης διεπαφής.

### 5.2.5 Αξιολόγηση

Σε αυτό το τμήμα παρουσιάζουμε τις δοκιμές μας, που προσφέρουν διάφορα πρότυπα για να προσομοιώσουμε μεγάλες υπηρεσίες καταλόγων [CFB0β]. Παρουσιάζουμε τα αποτελέσματα των πειραμάτων που προσδιορίζουν τα οφέλη μερικών τύπων αντιστοίχισης στο στάδιο της σύνθεσης υπηρεσιών. Χάρη σε αυτήν την υποστήριξη μερικών τύπων αντιστοίχισης, ένα πολύ μεγαλύτερο μέρος του συνόλου του προβλήματος μπορεί να λυθεί από αυτοματοποιημένη σύνθεση. Παρουσιάζουμε επίσης τα αποτελέσματα που τονίζουν τη σημασία ενός καθορισμένου χρήστη που είναι ικανός για λύση αλλά χωρίς κρίση στην αναζήτηση καταλόγου.

### **Πρότυπα δοκιμών και προσομοίωσης**

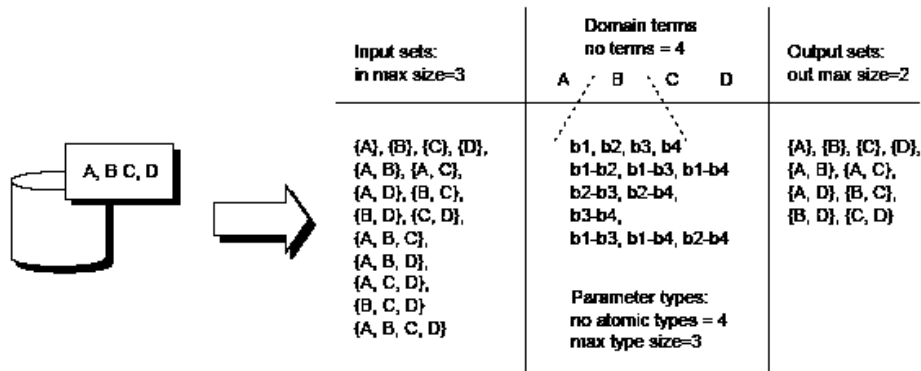
Εδώ παρουσιάζουμε τις δοκιμές μας για τη σύνθεση υπηρεσιών μεγάλης κλίμακας. Η δοκιμή αυτή βασίστηκε στην υπόθεση ότι η πλειοψηφία των μελλοντικών υπηρεσιών Web θα δημιουργηθεί εκθέτοντας αναγνώσιμες από μηχανές εφαρμογές και συστήματα που είναι αυτήν την περίοδο προσιτές μέσω του ανθρώπινου επιπέδου διεπαφών.

Από την προοπτική τεχνολογίας υπάρχουν διάφορα είδη επιλογών για τη δημιουργία Ιστοσελίδων που κυμαίνονται από στατικές ιστοσελίδες σε δυναμικότερες και ανέπτυξαν τη χρησιμοποίηση συγκεκριμένων γλωσσών για να εξυπηρετήσουν πλάγια κείμενα (π.χ., ASP, PHP ή JSP). Για την παραγωγή της παρουσίασης του περιεχομένου, οι δυναμικές σελίδες μπορούν να έχουν άμεση πρόσβαση σε ένα στρώμα στοιχείων (π.χ., μια σχετική βάση δεδομένων) ή μπορούν να χρησιμοποιήσουν και ενδιάμεσο objectual (διαφορετικό) στρώμα που να εφαρμόζει την επιχειρησιακή λογική και τη συμπύκνωση των στοιχείων (π.χ., ένας κεντρικός υπολογιστής εφαρμογής).

Σε πιο υψηλό επίπεδο, οι διαφορετικές εφαρμογές οργανώνονται συνήθως αναλόγως τη πεδίο τους (π.χ., ταξίδι, ψυχαγωγία, ακίνητα ή ιατρικά). Η κατανόηση, η ανάπτυξη και η χρησιμοποίηση της ορολογίας που απαιτείται για ένα συγκεκριμένο πεδίο απαιτεί συνήθως μια σημαντική προσπάθεια εφαρμοσμένης μηχανικής γνώσης.

Η δοκιμή μας στηρίζεται στις έννοιες της διανομής στοιχείων και της διανομής πεδίων χρησιμοποιώντας πεδία εφαρμογής σαν κεντρική ιδέα. Στο πλαίσιο μας ένα πεδίο εφαρμογής αντιπροσωπεύει μια συλλογή όρων και σχετικών τύπων δεδομένων. Κατόπιν οι υπηρεσίες καθορίζονται σαν μετασχηματισμοί μεταξύ των συνόλων όρων σε δύο πεδία εφαρμογής. Τυπικά αυτό είναι εξηγείται ως μια κατευθυνόμενη δομή γραφικών παραστάσεων, όπου κάθε κόμβος αντιπροσωπεύει μια πεδίο εφαρμογής και κάθε άκρη αντιπροσωπεύει μια ή περισσότερες υπηρεσίες Web. Κάθε υπηρεσία Web εκτελεί έναν μετασχηματισμό μεταξύ δύο δεδομένων συνόλων όρων από τα δύο πεδία εφαρμογής που συνδέονται με τις δύο άκρες του τέλους της γραφικής παράστασης.

Θα παρουσιάσουμε έπειτα λεπτομερέστερα τους κόμβους των πεδίων εφαρμογής και τη δομή της γραφικής παράστασης υπηρεσιών.



Σχήμα 5.6: Ο κόμβος πεδίων εφαρμογής.

### Κόμβοι πεδίων εφαρμογής

Στους όρους του πλαισίου εργασίας μας είναι μια πρωταρχική έννοια που χρησιμοποιείται για τη προδιαγραφή ενός πεδίου εφαρμογής. Κάθε όρος σχεδίασης, στη σχεδίαση μιας περιγραφής υπηρεσίας καθορίζει είτε μια εισαγόμενη είτε μια εξαγόμενη παράμετρο. Για κάθε όρο η πεδίο εφαρμογής καθορίζει ένα σύνολο από πιθανούς τύπους δεδομένων.

Για την επιτάχυνση της παραγωγής ειδικής επεξεργασίας παράγουμε εξαντλητικά όλα τα πιθανά εισαγόμενα ή εξαγόμενα σύνολα παραμέτρων για την παραγωγή συνόλων παραγωγής από τους όρους πεδίων. Δεδομένου ότι εξετάζουμε ότι ο αριθμός των όρων σε μια πεδίο θα είναι μεγαλύτερος από τον αριθμό των υπηρεσιών που χρησιμοποιούνται συνήθως ως εισαγόμενες ή εξαγόμενες πληροφορίες, θα καθιερώσουμε ένα ανώτερο αριθμό συνόλων παραμέτρων και θα φιλτράρουμε το αρχικό σύνολο παραγωγής ανάλογα. Παραδείγματος χάριν στην περίπτωση ενός πεδίου με τους όρους A,B,C,D με μέγιστο αριθμό παραμέτρων για μια υπηρεσία το 2, θα έχουμε ως πιθανές παραμέτρους τα σύνολα A, B, C, D, A,

B, A, C, A, D, B, C, B, D και C, D. Θα φιλτράρουμε τα σύνολα A, B, C, A, B, D, A, C, D, B, C, D και A, B, C, D μέχρι ο αριθμός των στοιχείων του συνόλου να ξεπεράσει το 2.

Δεδομένου ότι μια υπηρεσία μπόρεσε να χρησιμοποιήσει ένα δεδομένο σύνολο όρων σε μια πεδίο, είτε τον έναν σαν εισαγόμενες παραμέτρους και τον άλλο σαν παραμέτρους

αποτελεσμάτων, κάνουμε την ίδια διαφοροποίηση σχετικά με τα σύνολα πιθανών όρων έτσι ώστε από την κατοχή των διαφορετικών μέγιστων μεγεθών για τις πιθανές εισαγόμενες πληροφορίες και τα πιθανά αποτελέσματα, θα έχουμε διαφορετικά fan-in και fan-out σχετικά με τον αριθμό υπηρεσιών που θα μπορούσαν να κάνουν μετασχηματισμό από ή σε μια πεδίο εφαρμογής. Παραδείγματος χάριν το πεδίο προηγουμένως αναφέρθηκε, ότι το πεδίο θα μπορούσε να έχει 2 ως μέγιστο μέγεθος εξαγόμενων υπηρεσιών αλλά θα μπορούσε να έχει 3 ως μέγιστο μέγεθος εισαγόμενων πληροφοριών, το οποίο θα οδηγούσε σε μία 3/2 σχέση.

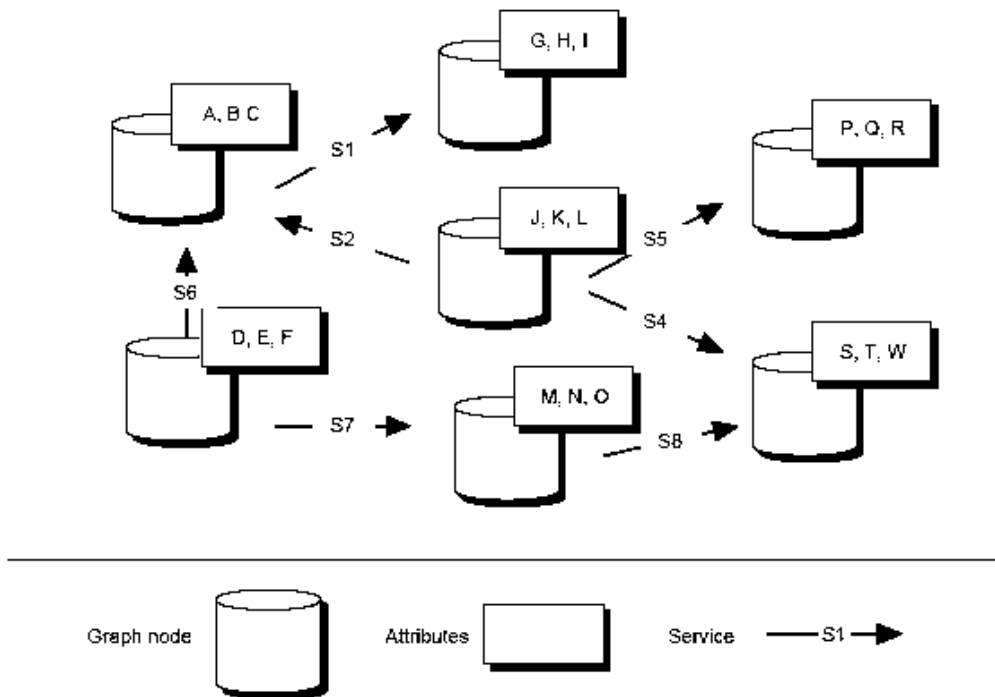
Για κάθε όρο σε ένα πεδίο καθορίζουμε διάφορους πιθανούς τύπους στοιχείων. Κατ' αρχάς, για κάθε έναν όρο, ένας αριθμός από "ατομικούς" τύπους διευκρινίζεται. Θεωρούμε ότι ο αριθμός περιστατικών για κάθε έναν από τους ατομικούς τύπους υπακούει μια διανομή του νόμου της δύναμης της μορφής  $1/i^a$  όπου  $i$  ο δείκτης του τύπου και το  $a$  είναι κοντά στη μονάδα. Χρησιμοποιώντας αυτό υπολογίζουμε τις συχνότητες περιστατικού για κάθε έναν από τους ατομικούς τύπους. Δημιουργούμε έπειτα ένα "ατομικό σύνολο zipf" ενός δεδομένου μεγέθους όπου οι ατομικοί τύποι εμφανίζονται μία φορά ή περισσότερες, αναλόγως στις συχνότητές τους. Κατόπιν από το "ατομικό σύνολο zipf" δημιουργούμε όπως παραπάνω, το σύνολο δύναμης των ατομικών τύπων ενώ θα κρατήσουμε μια υψηλότερη σύνδεση για τον αριθμό στοιχείων συνόλου των αποκτηθέντων συνόλων. Αυτά τα σύνολα θα αντιπροσωπεύσουν "σύνθετοι τύποι" που αποκτώνται από την αλληλουχία ενός ή περισσότερων "ατομικών τύπων". Οι "σύνθετοι τύποι" είναι ομαλοποιημένοι έτσι ώστε τα διπλά περιστατικά του ίδιου ατομικού τύπου να απορρίπτονται ( π.χ  $\{b1, b1, b2\} \rightarrow \{fb1, b2\}$  ) και αυτοί οι διαδοχικοί τύποι συγχωνεύονται (π.χ.,  $\{b1, b2, b3\} \rightarrow \{b1 - b3\}$ ).

### **Δομή γραφικών παραστάσεων υπηρεσιών**

Παράγουμε τις υπηρεσίες στις δοκιμές μας ως μετασχηματιστές μεταξύ συνόλων όρων σε δύο πεδία εφαρμογής. Για να κάνουμε αυτό για κάθε υπηρεσία πρέπει πρώτα να επιλέξουμε τυχαία δύο πεδία εφαρμογής. Κατόπιν επιλέγουμε τυχαία ένα σύνολο παραμέτρων από το σύνολο παραμέτρων εισαγωγής του πρώτου πεδίου εφαρμογής και ένα σύνολο παραμέτρων από το σύνολο παραμέτρων παραγωγής του δεύτερου πεδίου

εφαρμογής. Κατόπιν για κάθε μια από τις παραμέτρους στα δύο σύνολα επιλέγουμε τυχαία έναν τύπο δεδομένων παραμέτρου από τα αντίστοιχα πεδία εφαρμογής.

Ο κύριος περιορισμός που επιβάλλουμε κάνοντας τις επιλογές πρόκειται ανωτέρω να επιλέξει διαφορετικά πεδία εφαρμογής, δεδομένου ότι ενδιαφερόμαστε κυρίως για



Σχήμα 5.7: Η δομή γραφικών παραστάσεων υπηρεσιών.

τον περιορισμό των πεδίων που διασταυρώνονται. Ένας άλλος περιορισμός που επιβάλλουμε είναι το φιλτράρισμα των αντιγράφων, υπηρεσίες δηλαδή που έχουν ακριβώς τις ίδιες εισαγόμενες πληροφορίες και τις ίδιες εξαγόμενες πληροφορίες.

Για την παραγωγή δοκιμαστικών προβλημάτων χρησιμοποιούμε έναν παρόμοιο αλγόριθμο όπως αυτός για τη δημιουργία των υπηρεσιών, επιλέγοντας τυχαία τα πεδία, τα σύνολα παραμέτρων και τύπους δεδομένων παραμέτρων. Η σημαντικότερη διαφορά στέκεται στη διαφορετική ερμηνεία των παραμέτρων εισαγωγής και παραγωγής στην περίπτωση των ερωτήσεων (βλ. το τμήμα “συμβατός τύπος σύνθεσης υπηρεσιών”). Οι

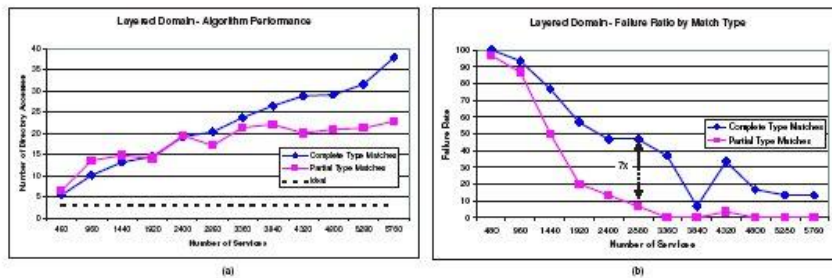
παράμετροι που επιλέγονται από το πρώτο πεδίο των συνόλων εισαγωγής είναι "διαθέσιμες εισαγωγές" και οι παράμετροι που επιλέγονται από το δεύτερο πεδίο των συνόλων παραγωγής είναι "απαραίτητα αποτελέσματα".

### Αποτελεσματικότητα και απόδοση σύνθεσης

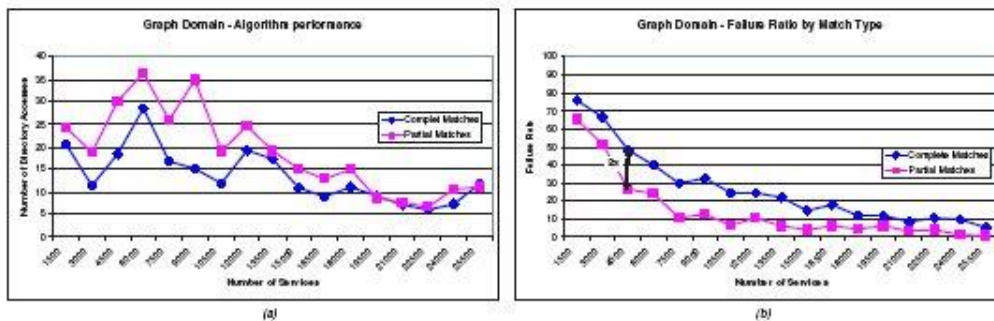
Και για τις δύο πεδία, έχουμε παραγάγει τυχαία τις υπηρεσίες και τις ερωτήσεις. Λύσαμε έπειτα τις ερωτήσεις που χρησιμοποιούν πρώτα έναν αλγόριθμο που χειρίζεται μόνο τις πλήρεις αντιστοιχίες τύπων και έπειτα έναν αλγόριθμο που χειρίζεται τις μερικές αντιστοιχίες τύπων (και περιλαμβάνει προφανώς τις πλήρεις αντιστοιχίες).

Έχουμε μετρήσει τον αριθμό προσβάσεων καταλόγου και την αναλογία αποτυχίας των αλγορίθμων ολοκλήρωσης.

Το σχέδιο 5.8 (α) και το σχέδιο 5.9 (α) παρουσιάζει μέσο αριθμό προσβάσεων καταλόγου για αλγόριθμο που χρησιμοποιεί το πλήρες ταίριασμα τύπων εναντίον του



Σχήμα 5.8: Η βαλμένη σε στρώσεις πεδίο.



Σχήμα 5.9: Η πεδίο γραφικών παραστάσεων.

μέσου αριθμού προσβάσεων καταλόγου για αλγόριθμο που χρησιμοποιεί επίσης το μερικό ταίριασμα τύπων. Οι δύο αλγόριθμοι αυξάνονται διαδοχικά καλά, αφού είναι στα περισσότερα μια αργή αύξηση στον αριθμό προσβάσεων καταλόγου όσο ο αριθμός υπηρεσιών στον κατάλογο αυξάνεται. Όπως προκύπτει από τα πειραματικά στοιχεία και για τα δύο πεδία η αύξηση που προκαλείται από τη χρήση των μερικών αντιστοιχιών, δεν είναι πολύ σημαντική και μειώνεται όταν στον κατάλογο επέρχεται κορεσμός από τις υπηρεσίες. Αυτό οφείλεται στο γεγονός ότι όταν έχει περισσότερες επιλογές, καθιστά το πρόβλημα κάλυψης πληροφόρησης ευκολότερο στο μερικό αλγόριθμο. Εκτός από αυτό, το πεδίο, σε διαδοχικά επίπεδα, από μια μεριά αποδίδει καλύτερα στον μερικό αλγόριθμο παρά στον πλήρη (σχέδιο 5.8 (α) μετά από 3000 υπηρεσίες). Αυτό οφείλεται στο γεγονός ότι ο αλγόριθμος που χρησιμοποιεί τις μερικές αντιστοιχίες αποτυγχάνει λιγότερο στα υποπροβλήματα και κάνει υπό αυτήν τη μορφή την καλύτερη χρήση των ήδη ανακτημένων υπηρεσιών.

Το σημαντικότερο αποτέλεσμα αφορά τον αριθμό πρόσθετων προβλημάτων που μπορούν να λυθούν με τη χρησιμοποίηση των μερικών αντιστοιχιών και μπορεί να φανεί στο σχέδιο 5.8 (β) και το σχέδιο 5.9 (β). Η γραφική παράσταση δείχνει ότι το ποσοστό αποτυχίας στην περίπτωση της χρησιμοποίησης μόνο των πλήρων αντιστοιχιών είναι πολύ μεγαλύτερο από το ποσοστό αποτυχίας όταν χρησιμοποιούνται οι μερικές αντιστοιχίες: μέχρι **7 φορές** στην περίπτωση του πεδίου των διαδοχικών επιπέδων και **2 φορές** στην περίπτωση του πεδίου των γραφικών παραστάσεων. Αυτό δείχνει ότι χρησιμοποιώντας τις μερικές αντιστοιχίες ανοίγει η πόρτα για την επίλυση πολλών προβλημάτων που ήταν αδιάλυτα από τον πλήρη τύπο αντιστοιχίας αλγόριθμων.

### **Αποτελέσματα σύνθεσης υπηρεσιών ικανής για λύση αλλά χωρίς κρίση**

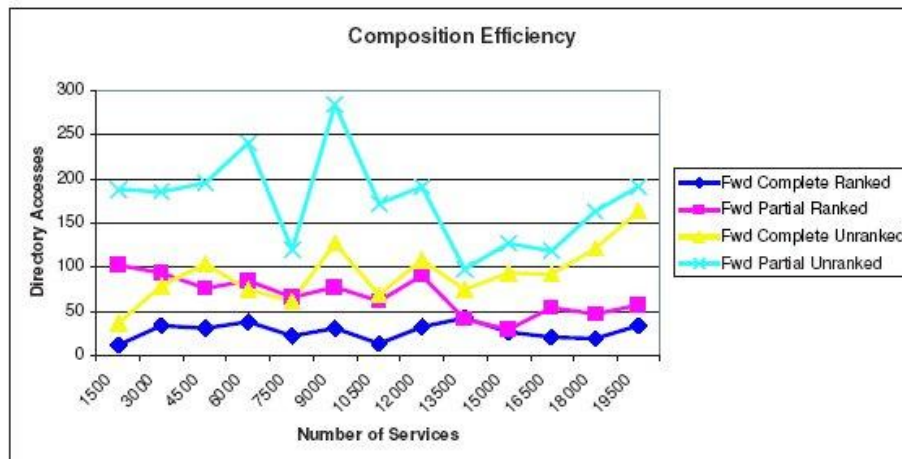
Έχουμε αξιολογήσει τον αντίκτυπο της καθορισμένης από το χρήστη περικοπής και της ταξινόμησης των λειτουργιών στην απόδοση σύνθεσης. Δεδομένου ότι θεωρούμε τις προσβάσεις καταλόγου υπολογιστικά ακριβείς στη λειτουργία τους, τις χρησιμοποιούμε ως μέτρο της αποδοτικότητας.

Τα προβλήματα έχουν λυθεί χρησιμοποιώντας δύο μπροστινούς αλγόριθμους σύνθεσης περιορισμών:



Ένα που χειρίζεται μόνο τις πλήρεις αντιστοιχίες τύπων και ένα άλλο που μπορούν να συνθέσουν υπηρεσίες μερικής αντιστοίχισης, [CFB04b]. Όταν τρέξουν οι αλγόριθμοι θα έχουμε χρησιμοποιήσει δύο διαφορετικές διαμορφώσεις καταλόγου: Η πρώτη διαμόρφωση χρησιμοποιεί τον εκτεταμένο κατάλογο που περιγράφεται εδώ και που υποστηρίζει την περικοπή συνήθειας και την ταξινόμηση των λειτουργιών, ειδικότερα χρησιμοποιώντας την μπροστινή λειτουργία ταξινόμησης περιορισμού, που περιγράφεται στο σχέδιο 5.8. Στη δεύτερη διαμόρφωση χρησιμοποιήσαμε έναν κατάλογο που δεν γνωρίζει τον αλγόριθμο σύνθεσης υπηρεσιών (π.χ., μπροστινοί πλήρης, οπίσθιος, κ.λ.π.) και δεν μπορεί να επεκταθεί από τον κώδικα πελατών. Αυτός ο κατάλογος εφαρμόζει μια γενική διαταγή ικανή για λύση αλλά χωρίς κρίση με την εξέταση ενός αριθμού υπέρθετων εισαγόμενων πληροφοριών στην ερώτηση και στην υπηρεσία, συν τον αριθμό υπέρθετων αποτελεσμάτων στην ερώτηση και στην υπηρεσία.

Και για τους δύο καταλόγους έχουμε χρησιμοποιήσει ακριβώς το ίδιο σύνολο περιγραφών υπηρεσιών και για κάθε επανάληψη έχουμε τρέξει τους αλγόριθμους ακριβώς στα ίδια τυχαία προβλήματα. Όπως μπορεί να φαίνεται στο σχέδιο 5.13, η χρησιμοποίηση της περικοπής συνήθειας και της ταξινόμησης των λειτουργιών βελτιώνει την απόδοση των αλγορίθμων μας. Στην περίπτωση των πλήρων αντιστοιχιών η βελτίωση είναι μέχρι έναν παράγοντα 5 (για έναν κατάλογο 10500 υπηρεσιών) και στη περίπτωση των μερικών αντιστοιχιών η βελτίωση είναι ενός παράγοντα 3 (για έναν κατάλογο 9000 των υπηρεσιών).



Σχήμα 5.10: Αντίκτυπος της ταξινόμησης των λειτουργιών στους αλγορίθμους σύνθεσης.

### 5.3 Σύνθεση ειδικού επιπέδου επεξεργασίας

Λαμβάνοντας υπόψη ένα σύνολο υπαρχουσών υπηρεσιών Web  $W_1, \dots, W_n$ , το πρόβλημα της δημιουργίας μιας σύνθεσης ειδικού επιπέδου επεξεργασίας βρίσκεται στο ότι πρέπει να βρεθεί ένα πρόγραμμα που να αλληλεπιδρά με αυτές τις υπηρεσίες Web με έναν κατάλληλο τρόπο, προκειμένου να επιτευχθεί μια δεδομένη απαίτηση σύνθεσης.

Ας εξετάσουμε παραδείγματος χάριν την περίπτωση του εικονικού γραφείου ταξιδιών που παρουσιάζεται στο τμήμα 3.2, και υποθέτουμε ότι ένα σύνολο φορέων παροχής υπηρεσιών τουρισμού έχει προσδιοριστεί για την επίλυση ενός αιτήματος πελατών. Αυτές οι υπηρεσίες μπορούν αποτελούνται, παραδείγματος χάριν, από μια υπηρεσία "κράτησης πτήσης" και μια υπηρεσία "κράτησης ξενοδοχείων" που είναι επαρκείς για μια ιδιαίτερη αίτηση του πελάτη, π.χ., ένας συγκεκριμένος προορισμός. Ο στόχος της σύνθεσης ειδικού επιπέδου επεξεργασίας είναι να πετύχει τον εκτελέσιμο κώδικα που επικαλείται αυτές τις υπηρεσίες Web, προκειμένου να ληφθεί μια προσφορά για το αίτημα του πελάτη.

Στον καθορισμό του εκτελέσιμου κώδικα που εκτελεί την εφαρμογή της σύνθεσης, χρειάζεται να λάβουμε υπόψη το γεγονός ότι, σε πραγματικές περιπτώσεις, η κράτηση ενός ξενοδοχείου δεν είναι ένα ατομικό βήμα, αλλά απαιτεί άντ' αυτού μια ακολουθία

διαδικασιών, συμπεριλαμβανομένης της επικύρωσης, της υποβολής ενός συγκεκριμένου αιτήματος, τη διαπραγμάτευση μιας προσφοράς, την αποδοχή (ή άρνηση) της προσφοράς, και την πληρωμή. Δηλαδή οι υπηρεσίες Web  $W_1, \dots, W_n$  είναι συνήθως σύνθετες, δηλ., η αλληλεπίδραση τους δεν αποτελεί ένα ενιαίο βήμα αιτήματος-απάντησης, αλλά απαιτεί να ακολουθηθεί ένα σύνθετο πρωτόκολλο προκειμένου να επιτευχθεί το απαραίτητο αποτέλεσμα. Επιπλέον, τα βήματα που καθορίζουν τη σύνθετη αλληλεπίδραση δεν καθορίζουν απαραίτητα μια ακολουθία. Πράγματι, αυτά τα βήματα μπορούν να έχουν υποθετικά, ή μη-ονομαστικά αποτελέσματα (π.χ., η επικύρωση μπορεί να αποτύχει, μπορεί να μην υπάρξει κάποια προσφορά διαθέσιμη από μια υπάρχουσα υπηρεσία.) που να επηρεάσει τα ακόλουθα βήματα (κανένα αίτημα δεν μπορεί να υποβληθεί εάν η επικύρωση αποτύχει, εάν δεν υπάρχει καμία προσφορά διαθέσιμη, μια διαταγή μπορεί να μην υποβληθεί.). Μπορεί επίσης να συμβεί η περίπτωση, η ίδια λειτουργία να επαναληφθεί, π.χ., προκειμένου να καθοριστεί ένα αίτημα ή για να διαπραγματευτεί τους όρους της προσφοράς.

Ενώ οι λεπτομέρειες στην ακριβή ακολουθία διαδικασιών που απαιτούνται για να αλληλεπιδράσουν με μια υπάρχουσα υπηρεσία δεν είναι σημαντικές στην ανακάλυψη και στην ειδικού επιπέδου επεξεργασία σύνθεσης, γίνονται αναπόφευκτες όταν ο εκτελέσιμος κώδικας για την εφαρμογή της σύνθεσης πρέπει να παραχθεί.

Για αυτόν τον λόγο, στην ειδικού επιπέδου επεξεργασία σύνθεσης οι υπάρχουσες υπηρεσίες Web πρέπει να περιγράφονται από την άποψη των πολύπλοκων, σύνθετων διαδικασιών, οι οποίες αποτελούνται από αυθαίρετους (υποθετικούς και επαναληπτικούς) συνδυασμούς ατομικών αλληλεπιδράσεων, και εκεί οι ατομικές αλληλεπιδράσεις μπορούν να έχουν υποθετικά αποτελέσματα. Κατά συνέπεια, και ο εκτελέσιμος κώδικας που έχει παραχθεί, πρέπει να είναι ένα σύνθετο πρόγραμμα, δεδομένου ότι πρέπει να λάβει υπόψη όλα τα πιθανά απρόβλεπτα έξοδα που εμφανίζονται μέσα στην αλληλεπίδραση των υπηρεσιών Web.

### **"On the fly" σύνθεση εναντίον "παραγωγής υπηρεσιών"**

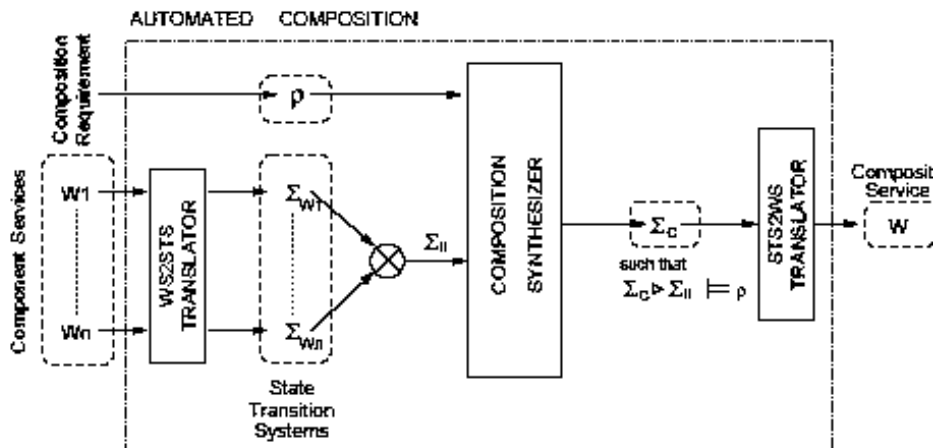
Δύο διαφορετικά σενάρια της ειδικού επιπέδου επεξεργασίας σύνθεσης είναι δυνατά. Στο πρώτο σενάριο, η αποκαλούμενη **"On the fly σύνθεση"**, είναι η σύνθεση που

δημιουργείται προκειμένου να ικανοποιηθεί ένα συγκεκριμένο αίτημα του πελάτη (π.χ., ένα ταξίδι σε μια συγκεκριμένη θέση σε μια συγκεκριμένη χρονική περίοδο).

Σε αυτό το σενάριο, ένας στόχος σύνθεσης εκτελείται για κάθε υποβληθέν αίτημα: κάθε φορά που το εικονικό γραφείο ταξιδιών λαμβάνει το αίτημα ενός πελάτη, παράγει και εκτελεί τη σύνθεση, λαμβάνοντας κατά συνέπεια μια προσφορά που μπορεί να δοθεί πίσω στον πελάτη. Στο δεύτερο σενάριο, που καλείται "**παραγωγή υπηρεσιών**", ο στόχος της σύνθεσης είναι να καθορίσει έναν γενικό πρόγραμμα που είναι σε θέση να απαντήσει στα γενικά αιτήματα που τίθενται από τους πελάτες. Δηλαδή αντί της σύνθεσης υπηρεσιών της κράτησης ξενοδοχείων και της κράτησης πτήσης για μια *ιδιαίτερη αίτηση*, η σύνθεση αυτών των υπηρεσιών καθορίζεται έτσι ώστε να μπορεί να απαντήσει σε ένα γενικό αίτημα. Το τελευταίο σενάριο καλείται "παραγωγή υπηρεσιών" δεδομένου ότι ο δημιουργημένος εκτελέσιμος κώδικας εφαρμόζει μία υπηρεσία που παρέχεται από το εικονικό γραφείο ταξιδιών για να απαντήσει στα γενικά αιτήματα των πελατών, και μπορεί να επεκταθεί και να εκτελεσθεί μέσα στο σύστημα πληροφοριών του γραφείου ταξιδιών.

Θα εξετάσουμε και τα δύο σενάρια της ειδικού επιπέδου επεξεργασίας σύνθεσης, τονίζοντας τις ομοιότητες και τις διαφορές. Παρατηρούμε εδώ μια πρώτη διαφορά μεταξύ τους, δηλαδή το είδος των αλληλεπιδράσεων που υποστηρίζουν με το χρήστη της συγκροτημένης υπηρεσίας. Στην περίπτωση της On the fly σύνθεσης, οι αλληλεπιδράσεις με το χρήστη αναγκάζονται να είναι ατομικές: στην περίπτωση του εικονικού γραφείου ταξιδιών, παραδείγματος χάριν, αφότου έχει στείλει ο πελάτης ένα αίτημα, η αντιπροσωπεία αλληλεπιδρά με έναν κατάλληλο τρόπο με τις υπηρεσίες του ξενοδοχείου και της πτήσης και στέλνουν μια προσφορά πίσω στον πελάτη. Στη περίπτωση της παραγωγής υπηρεσιών, γενικότερες αλληλεπιδράσεις με τον πελάτη μπορεί να εξεταστούν. Παραδείγματος χάριν, είναι δυνατό να παραχθεί μια σύνθεση που, αφού λάβει ένα αίτημα από τον πελάτη, ζητά από τις υπηρεσίες πτήσης και τις υπηρεσίες ξενοδοχείων τις πιθανές προσφορές που θα συνδυαστούν και θα σταλούν πίσω στον πελάτη, ο πελάτης μπορεί τώρα να ελέγξει τις προσφορές και να αποφασίσει εάν θα τις δεχτεί ή αν θα τις απορρίψει, το εικονικό γραφείο ταξιδιών μπορεί να επιβεβαιώσει την ακύρωση των προσφορών του ξενοδοχείου και της πτήσης αφού λάβει την απάντηση

από τον πελάτη. Αυτός ο τρόπος, μαζί με την αλληλεπίδραση με τον πελάτη γίνεται μια σύνθετη ακολουθία διαδικασιών, ενδεχομένως με υποθετικές και



Σχήμα 5.11: Διαδικασία-ισόπεδος σύνθεση υπηρεσιών: Μια επισκόπηση

επαναληπτικές συμπεριφορές. Από τεχνικής άποψης, ο πελάτης γίνεται μια από τις υπάρχουσες υπηρεσίες Web  $W_1, \dots, W_n$  όπου ο δημιουργημένος εκτελέσιμος κώδικας πρέπει να αλληλεπιδράσει προκειμένου να φέρει έξω τη σύνθεση.

### Αυτοματοποιημένη σύνθεση υπηρεσιών: Μια επισκόπηση

Καθορίζοντας την ειδικού επιπέδου επεξεργασία σύνθεσης, ενώ συνήθως δεν είναι πολύ σύνθετη, είναι μια δραστηριότητα χρονοβόρα και επιρρεπής στα λάθη, δεδομένου ότι απαιτεί μια ανάλυση του καθορισμού πρωτοκόλλων των υπάρχουσών υπηρεσιών, και τη παραγωγή ενός κώδικα που λαμβάνει υπόψη όλο τα πιθανά διαφορετικά σενάρια που μπορούν να εμφανιστούν κατά τη διάρκεια της αλληλεπίδρασης με αυτές τις υπηρεσίες. Για αυτόν τον λόγο, η αυτοματοποιημένη υποστήριξη είναι επιθυμητή για να μειώσει το χρόνο και τα λάθη στην ειδικού επιπέδου επεξεργασία σύνθεση των υπηρεσιών Web. Περιγράφουμε τώρα μια προσέγγιση στην αυτοματοποιημένη σύνθεση των υπηρεσιών Web η οποία είναι σε θέση να εξετάσει τα χαρακτηριστικά γνωρίσματα που περιγράφονται ανωτέρω. Μια επισκόπηση της προσέγγισης καθορίζεται παρακάτω, ενώ οι περαιτέρω λεπτομέρειες περιγράφονται στο υπόλοιπο του κεφαλαίου.

Ο στόχος μας είναι να παραγάγουμε αυτόματα μια νέα υπηρεσία  $W$  (αποκαλούμενη *σύνθετη υπηρεσία*) που να αλληλεπιδρά με ένα σύνολο δημοσιευμένων υπηρεσιών Web  $W_1, \dots, W_n$  (που αποκαλούνται συστατικές υπηρεσίες) και να ικανοποιεί ένα δεδομένο αίτημα σύνθεσης. Πιο συγκεκριμένα (σχήμα 5.13), αρχίζουμε από  $n$  περιγραφές ειδικού επιπέδου επεξεργασίας σύνθεσης υπηρεσιών Web  $W_1, \dots, W_n$ , η ενότητα WS2STS αυτόματα μεταφράζει κάθε ένα από αυτά σε ένα *σύστημα αλλαγής κατάστασης* (STS από εδώ και πέρα)  $\Sigma_{W_1}, \dots, \Sigma_{W_n}$ . Διαισθητικά, κάθε μια  $\Sigma_{W_i}$  είναι μια συμπαγής αντιπροσώπευση όλων των πιθανών συμπεριφορών, που εξελίσσεται της συνθετημένης υπηρεσίας  $W_i$ . Κάθε ένας  $\Sigma_{W_i}$  περιγράφεται από την άποψη της κατάστασης, των ενεργειών εισαγωγής και παραγωγής, και εσωτερικών ενεργειών.

Κατασκευάζουμε έπειτα ένα *παράλληλο STS*  $\Sigma_{\Pi}$  που συνδυάζει  $\Sigma_{W_1}, \dots, \Sigma_{W_n}$ . Τυπικά, αυτός ο συνδυασμός είναι ένα παράλληλο προϊόν, το οποίο επιτρέπει στις  $n$  υπηρεσίες να εξελιχθούν ταυτόχρονα. Το  $\Sigma_{\Pi}$  αντιπροσωπεύει επομένως όλες τις πιθανές συμπεριφορές, εξελίξεις του διαφορετικού συνθετημένων υπηρεσιών, χωρίς οποιοδήποτε έλεγχο και αλληλεπίδραση με τη σύνθετη υπηρεσία που θα είναι παραγόμενη, δηλ.  $W$ . Το δεύτερο είδος εισαγωγής στην αυτοματοποιημένη σύνθεση αποτελείται από την απαίτηση  $\frac{1}{2}$  για τη σύνθετη υπηρεσία, η οποία περιγράφει ποια είναι η λειτουργία αυτή της αποτελούμενης υπηρεσίας που θεωρείται υποτιθέμενη για να επιτύχει. Λαμβάνοντας υπόψη  $\Sigma_{\Pi}$  και  $\frac{1}{2}$ , παράγουμε αυτόματα ένα STS  $\Sigma_c$  που κωδικοποιεί τη νέα υπηρεσία  $W$  που πρέπει να παραχθεί, το οποίο δυναμικά λαμβάνει και στέλνει τις επικλήσεις από/προς τις σύνθετες υπηρεσίες  $W_1, \dots, W_n$  και συμπεριφέρεται ανάλογα με τις απαντήσεις που παραλαμβάνονται από τις εξωτερικές υπηρεσίες. Το  $\Sigma_c$  είναι έτσι ώστε  $\Sigma_c \triangleright \Sigma_{\Pi}$  να ικανοποιεί την απαίτηση  $\frac{1}{2}$ , όπου  $\Sigma_c \triangleright \Sigma_{\Pi}$  αντιπροσωπεύει όλες τις εξελίξεις των συνθετικών υπηρεσιών όπως ελέγχονται από τη σύνθετη υπηρεσία. Το STS  $\Sigma_c$  δίνεται έπειτα μέσα στην εισαγωγή της ενότητας STS2WS που την μεταφράζει στο συγκεκριμένο, εκτελέσιμο κώδικα που εφαρμόζει την επιθυμητή σύνθετη υπηρεσία Web.

### 5.3.1 Ιστορικό και κατάσταση προόδου

Σε αυτό το τμήμα δίνουμε μια επισκόπηση των υπάρχουσών τεχνικών για ένα ειδικού επιπέδου επεξεργασίας σύνθεσης των υπηρεσιών Web. Διακρίνουμε τις προσεγγίσεις που βασίζονται στον προγραμματισμό των τεχνικών (η πλειοψηφία) και τις προσεγγίσεις που βασίζονται στα αυτόματα.

#### Τεχνικές προγραμματισμού

Υπάρχει ένας μεγάλος αριθμός εργασιών όπου οι διαφορετικές τεχνικές προγραμματισμού χρησιμοποιούνται για την επίτευξη αυτοματοποιημένης σύνθεσης των υπηρεσιών Web, όλες μέσα στη σημασιολογική έρευνα υπηρεσιών Web αλλά και έξω από αυτές (βλ., π.χ., [BDG03, WPS+ 03, Der98, SdF03, MS0a, MF02, PBB+ 04, TP04]). Σε αυτές τις εργασίες, οι υπάρχουσες υπηρεσίες μπορούν να χρησιμοποιηθούν για να κατασκευάσουν το πεδίο προγραμματισμού, οι απαιτήσεις σύνθεσης μπορούν να τυποποιηθούν σαν στόχοι προγραμματισμού, και ο προγραμματισμός των αλγόριθμων σχεδιασμού μπορεί να χρησιμοποιηθεί για να παράγει τα σχέδια που θα συνθέτουν τις υπάρχουσες υπηρεσίες.

Οι περισσότερες από αυτές τις εργασίες, εντούτοις, περιορίζονται στην περίπτωση ότι οι υπάρχουσες υπηρεσίες Web είναι ατομικές (δηλ. περιγράφονται σε σχέση με τις εισαγόμενων/εξαγόμενων πληροφοριών και των προϋποθέσεων/ postconditions), και ως εκ τούτου εκτελούν μια λειτουργικού επιπέδου σύνθεση. Αναφερόμαστε στο τμήμα 5.1.1 για μια συζήτηση σε ένα ανώτερο επίπεδο τεχνικής στην λειτουργικού επιπέδου σύνθεση, και εμείς εστιάζουμε εδώ στις λίγες προσεγγίσεις που εξετάζουν τις ειδικού επιπέδου επεξεργασίας περιγραφές των υπάρχουσών υπηρεσιών.

Στο [NM02], οι συντάκτες προτείνουν μια προσέγγιση στην προσομοιωμένη, επαληθευμένη, και αυτοματοποιημένη σύνθεση των υπηρεσιών Web που βασίζεται σε μια μετάφραση του υπολογισμού κατάστασης DAML-S και των δικτύων petri. Εντούτοις, ενώ το πλαίσιο επιτρέπει τη σύνθετη διαμόρφωση, non-atomic υπηρεσίες Web, η αυτοματοποιημένη σύνθεση περιορίζει στη διαδοχική σύνθεση των ατομικών υπηρεσιών.

Οι μόνες προσεγγίσεις προγραμματισμού που γνωρίζουμε, οι οποίες επιτρέπουν αυτοματοποιημένη σύνθεση ειδικού επιπέδου επεξεργασίας των υπηρεσιών Web είναι

αυτές που προτείνονται στα [PBB+ 04, TP04]. Αυτή η προσέγγιση θα περιγραφεί λεπτομερώς στις παραγράφους 5.2.2 και 5.2.3.

### **Αυτοματοποιημένες τεχνικές**

Στο [HBCS03], ένα επίσημο πλαίσιο καθορίζεται για τη σύνθεση των e-υπηρεσιών από περιγραφές συμπεριφορών που δίνονται σε σχέση με τα αυτόματα. Το πρόβλημα που εξετάζεται σε αυτήν την εργασία είναι στο επίπεδο διαδικασίας, από τότε που εξετάζουμε τις υπηρεσίες που μπορούν να εκτελέσουν τις πιο σύνθετες αλληλεπιδράσεις από μια απλή λειτουργία ερώτησης-απάντησης. Εντούτοις, το πρόβλημα σύνθεσης που λύνεται στο [HBCS03] είναι διαφορετικό από το πρόβλημα που εξετάζουμε σε αυτό το τμήμα. Πράγματι, σε αυτή τη προσέγγιση, το πρόβλημα σύνθεσης θεωρείται ως πρόβλημα συντονισμού των εκτελέσεων από ένα δεδομένο σύνολο διαθέσιμων υπηρεσιών, και όχι ως πρόβλημα που δημιουργεί μια νέα σύνθετη υπηρεσία Web που αλληλεπιδρά με τις διαθέσιμες. Οι λύσεις στο προηγούμενο πρόβλημα μπορούν να χρησιμοποιηθούν για να συναγάγουν τους περιορισμούς σε μια υπαρκτή (αντιπροσώπευση αυτομάτων σύνθεσης) υπηρεσία σύνθεσης, αλλά να μη παράγουν τον εκτελέσιμο κώδικα για την εφαρμογή της σύνθεσης. Αυτό είναι η κύρια εννοιολογική διαφορά μαζί με την εργασία που περιγράφεται μέσα στο [BCG+ 03], όπου οι αυτοματοποιημένες τεχνικές συλλογισμού, βασισμένες στη λογική περιγραφής, χρησιμοποιούνται για να εξετάσουν το πρόβλημα της αυτοματοποιημένης σύνθεσης των e-υπηρεσιών που περιγράφονται ως μηχανές πεπερασμένης κατάστασης.

Γενικά, η σύνθεση υπηρεσιών Web μοιράζεται μερικές ιδέες με την εργασία που βασίζεται στην αυτόματη σύνθεση των ελεγκτών (βλ. π.χ., [PR89, VAR]).

Πράγματι, η υπηρεσία σύνθεσης μπορεί να θεωρηθεί ως ενότητα που ελέγχει ένα περιβάλλον που αποτελείται από συνθετικές υπηρεσίες. Εντούτοις, το μεγαλύτερο μέρος της εργασίας σε αυτό το πεδίο εστιάζει στα θεωρητικά θεμέλια, χωρίς παροχή των πρακτικών εφαρμογών. Επιπλέον, είναι βασισμένο μάλλον σε διαφορετικές τεχνικές υποθέσεις για την αλληλεπίδραση με το περιβάλλον.



### 5.3.2 Επίσημος καθορισμός του προβλήματος

Παρουσιάζουμε τώρα ένα επίσημο πλαίσιο για τη σύνθεση υπηρεσιών Web, που λαμβάνεται από το [PTB04].

Η κεντρική έννοια είναι αυτή των *συστημάτων μεταβολής κατάστασης*, είναι ένα γενικό πρότυπο για την περιγραφή δυναμικών συστημάτων που χρησιμοποιείται για την αντιπροσώπευση των υπαρχουσών υπηρεσιών Web και του εκτελέσιμου κώδικα που καθορίζει την ειδικού επιπέδου επεξεργασίας σύνθεση.

#### Συστήματα μεταβολής κατάστασης

Ένα σύστημα μεταβολής κατάστασης καθορίζει ένα δυναμικό σύστημα που μπορεί να είναι σε διάφορες πιθανές καταστάσεις (μερικές εκ των οποίων είναι χαρακτηρισμένες ως *αρχικές καταστάσεις*) και μπορεί να εξελιχθούν σε νέες καταστάσεις ως αποτέλεσμα εκτέλεσης μερικών *ενεργειών*. Οι ενέργειες διακρίνονται στις *ενέργειες εισαγωγής*, οι οποίες αντιπροσωπεύουν την υποδοχή των μηνυμάτων, *ενέργειες εξαγωγής*, που αντιπροσωπεύουν τα μηνύματα που στέλνονται στις εξωτερικές υπηρεσίες, και μια ειδική ενέργεια  $\tau$ , αποκαλούμενη ως *εσωτερική δράση*. Η δράση  $\tau$  χρησιμοποιείται για να αντιπροσωπεύσει εσωτερικές εξελίξεις που δεν είναι ορατές στις εξωτερικές υπηρεσίες, δηλ., το γεγονός ότι η κατάσταση του συστήματος μπόρεσε να εξελιχθεί χωρίς παραγωγή οποιασδήποτε πληροφορίας, και ανεξάρτητα από την υποδοχή των εισαγόμενων μηνυμάτων.

Μια *σχέση μεταβολής* περιγράφει πώς η κατάσταση μπορεί να εξελιχθεί βάσει των εισαγόμενων πληροφοριών, των εξαγόμενων πληροφοριών, ή της εσωτερικής ενέργειας  $\tau$ . Τέλος, η ονομασία μιας λειτουργίας συνδέει την κάθε κατάσταση με το σύνολο των ιδιοτήτων *Prop* που κρατούν τη κατάσταση. Αυτές οι ιδιότητες θα χρησιμοποιηθούν για να καθορίσουν τις απαιτήσεις σύνθεσης.

#### Καθορισμός 1 [σύστημα μεταβολής κατάστασης (State Transition System (STS))]

Ένα σύστημα μεταβολής κατάστασης  $\Sigma$  ανήκει στο  $(S, S^0, I, O, R, L)$  όπου:

- $S$  είναι το σύνολο των καταστάσεων
- $S^0 \subseteq S$  είναι το σύνολο αρχικών καταστάσεων
- $I$  είναι το σύνολο ενεργειών εισαγωγής

- $\mathcal{R} \subseteq \mathcal{S} \times (\mathcal{I} \cup \mathcal{O} \cup \{\tau\}) \times \mathcal{S}$  είναι η σχέση μεταβολής
- $\mathcal{O}$  είναι το σύνολο ενεργειών παραγωγής
- $\mathcal{L} : \mathcal{S} \rightarrow 2^{\mathcal{P}^{\tau\sigma\phi}}$  είναι η ονομαστική λειτουργία.

Υποθέτουμε ότι ατέλειωτες συνδέσεις των τ-μεταβολών δεν μπορούν να εμφανιστούν στο σύστημα. Πράγματι, άπειρες τ -συνδέσεις θα περιέγραφαν μια αποκλίνουσα συμπεριφορά του συστήματος, δηλ., μια συμπεριφορά όπου η υπηρεσία δεν αλληλεπιδρά με το περιβάλλον.

Το σχήμα 5.12 παρουσιάζει μια αντίστοιχη STS σε μια πολύ απλουστευμένη υπηρεσία Web κράτησης ξενοδοχείων, που αντιστοιχεί σε ένα σύστημα κράτησης αλυσίδων ξενοδοχείων. Σε αυτό το απλουστευμένο πρότυπο, η υπηρεσία περιμένει ένα αίτημα από το χρήστη, καθορίζοντας την περίοδο του ταξιδιού και το μέρος. Η υπηρεσία ελέγχει έπειτα τη διαθεσιμότητα των δωματίων. Εάν δεν υπάρχει κανένας χώρος διαθέσιμος, μια αρνητική (not avail) απάντηση στέλνεται στο χρήστη. Εάν τα δωμάτια είναι διαθέσιμα, άντ' αυτού, η υπηρεσία στέλνει μια προσφορά στο χρήστη, που παρέχει τις αναλυτικές πληροφορίες για το ξενοδοχείο και ένα κόστος για την παραμονή. Ο χρήστης μπορεί τώρα να επιβεβαιώσει (ack) ή να ακυρώσει (nack) την κράτηση των δωματίων.

Το σύνολο των  $S$  καταστάσεων διαμορφώνει τα βήματα της εξέλιξης της διαδικασίας και των τιμών των μεταβλητών του. Το ειδικό μεταβλητό PC εκτελεί ένα "πρόγραμμα αντίθετο" που κρατά το ισχύον βήμα εκτέλεσης της υπηρεσίας (π.χ., το PC έχει την τιμή getRequest όταν περιμένει η διαδικασία για να λάβει ένα αίτημα κράτησης, και την τιμή checkAvailable όταν είναι έτοιμο να ελέγξει εάν η κράτηση είναι δυνατή). Άλλες μεταβλητές όπως το προσφορά ξενοδοχείου ή το κόστος προσφοράς αντιστοιχούν σε εκείνες που χρησιμοποιούνται στη διαδικασία αποθήκευσης των σημαντικότερων πληροφοριών. Τέλος, σειρές όπως AvailableHotel ή CostOfRoom περιγράφουν δηλώσεις και εκφραστικές ιδιότητες λειτουργιών της υπηρεσίας Web (π.χ., το γεγονός ότι υπάρχει ένα ξενοδοχείο διαθέσιμο για μια περίοδο και σε μια δεδομένη θέση, ή το κόστος ενός δωματίου σε ένα δεδομένο ξενοδοχείο για μια δεδομένη περίοδο).

Στις αρχικές καταστάσεις  $S^0$  το PC βάζει σαν στόχο το START ενώ όλες οι άλλες βασικές μεταβλητές είναι απροσδιόριστες. Οι αρχικές τιμές των σειρών AvailableHotel

και CostOfRoom είναι απροσδιόριστες, δεδομένου ότι μπορούν να υπολογίσουν οποιαδήποτε αξία στην πεδίο.

Η εξέλιξη της διαδικασίας διαμορφώνεται μέσω ενός συνόλου πιθανών μεταβάσεων.

Κάθε μετάβαση καθορίζει τις προϋποθέσεις καταλληλότητας στην πηγή της κατάστασης, στην κύρια ενέργεια του και στην κατάσταση προορισμού. Παραδείγματος χάριν, "PC = checkAvailable & AvailableHotel[req\_period, req\_LOC] ≠ UNDEF -[tau] - > PC = isAvailable" δηλώνει ότι μια δράση  $\tau$  μπορεί να εκτελεσθεί στη κατάσταση checkAvailable,

```

PROCESS
    HotelBooking;
TYPE
    Period, Location: Hotel: Cost;
STATE
    pc: { START, getRequest, checkAvailable, isAvailable, isNotAvailable,
        prepareOffer, sendOffer, waitAnswer, prepareNotAvail, sendNotAvail, SUCC, FAIL };
    req_period: Period ∪ { UNDEF };
    req_loc: Location ∪ { UNDEF };
    offer_hotel: Hotel ∪ { UNDEF };
    offer_cost: Cost ∪ { UNDEF };
    AvailableHotel[Period,Location]: Hotel ∪ { UNDEF };
    CostOfRoom[Period,Hotel]: Cost;
INIT
    pc = START;
    req_period = UNDEF;
    req_loc = UNDEF;
    offer_hotel = UNDEF;
    offer_cost = UNDEF;
INPUT
    request(p: Period, l: Location);
    ack();
    nack();
OUTPUT
    offer(h: Hotel, c: Cost);
    not avail();
TRANS
    pc = START -[TAU]-> pc = getRequest;
    pc = getRequest -[INPUT request(p,l)]-> pc = checkAvailable, req_period = p, req_loc = l;
    pc = checkAvailable & AvailableHotel(req_period, req_loc) ≠ UNDEF -[TAU]-> pc = isAvailable;
    pc = checkAvailable & AvailableHotel(req_period, req_loc) = UNDEF -[TAU]-> pc = isNotAvailable;
    pc = isAvailable -[TAU]-> pc = prepareOffer;
    pc = prepareOffer -[TAU]-> jk = sendOffer;
        offer_hotel = AvailableHotel(req_period, req_location);
        offer_cost = CostOfRoom(req_period, offer_hotel);
    pc = sendOffer -[OUTPUT offer(offer_cost, offer_hotel)]-> pc = waitAnswer;
    pc = waitAnswer -[INPUT nack]-> pc = FAIL;
    pc = waitAnswer -[INPUT ack]-> pc = SUCC;
    pc = isNotAvailable -[TAU]-> pc = prepareNotAvail;
    pc = prepareNotAvail -[TAU]-> pc = sendNotAvail;
    pc = sendNotAvail -[OUTPUT not avail]-> pc = FAIL;

```

Σχήμα 5.12: Το STS για τη διαδικασία κράτησης ξενοδοχείων.

οδηγώντας στη κατάσταση `isAvailable`, εάν υπάρχει κάποιο ξενοδοχείο με τα διαθέσιμα δωμάτια για την καθορισμένη περίοδο και τοποθεσία. Παρατηρούμε ότι κάθε πρόταση της TRANS του σχήματος 3.14 αντιστοιχεί στα διαφορετικά στοιχεία στη σχέση μεταβολής  $R$  : π.χ., "PC = checkAvailable & AvailableHotel[req\_period, req\_LOC,] ≠

UNDEF -[tau] -> PC = isAvailable" δημιουργεί διαφορετικά στοιχεία της  $R$ , ανάλογα με οι τιμές των μεταβλητών req\_period και req\_LOC.

Σύμφωνα με το επίσημο πρότυπο, διακρίνουμε μεταξύ τριών διαφορετικών ειδών ενεργειών.

Οι ενέργειες εισαγωγής  $I$  διαμορφώνουν όλα τα εισερχόμενα αιτήματα στη διαδικασία και στις πληροφορίες που φέρνουν (δηλ., το requrst χρησιμοποιείται για τη λήψη του αιτήματος κράτησης, ενώ το ack διαμορφώνει την επιβεβαίωση της διαταγής και το pack την ακύρωσή του). Οι ενέργειες παραγωγής  $O$  αντιπροσωπεύουν τα εξερχόμενα μηνύματα (δηλ., το not\_avail χρησιμοποιείται όταν δεν είναι η κράτηση πιθανή, ενώ το offer χρησιμοποιείται για να παραδώσει μια προσφορά στο χρήστη της υπηρεσίας). Η δράση  $\tau$  χρησιμοποιείται για να διαμορφώσει τις εσωτερικές εξελίξεις της διαδικασίας, σαν παραδείγματα αναθέσεων και διαδικασίας αποφάσεων (π.χ., όταν είναι η διαδικασία HotelBooking στη κατάσταση checkAvailable και εκτελεί τις εσωτερικές δραστηριότητες για να αποφασίσει εάν υπάρχουν διαθέσιμοι χώροι ή όχι, ή όταν είναι, στη κατάσταση prepareOffer, πρέπει να αποφασίσει το ξενοδοχείο και το κόστος του δωματίου).

Τέλος, οι ιδιότητες του STS είναι εκφράσεις της μορφής  $\langle \text{variable} \rangle = \langle \text{value} \rangle$  ή  $\langle \text{array} \rangle [\text{indx}_1, \dots, \text{indx}_n] = \langle \text{value} \rangle$  και η ονομαστική διαδικασία είναι προφανής.

Ο ορισμός STS που παρουσιάζεται στο σχήμα 5.14 είναι παραμετρικός w.r.t. των τύπων Period, Location, Hotel, και Cost που χρησιμοποιούνται στα μηνύματα. Προκειμένου να ληφθεί ένα συγκεκριμένο STS και για να εφαρμόσει τις αυτοματοποιημένες τεχνικές σύνθεσης που περιγράφονται αργότερα σε αυτό το έγγραφο, συγκεκριμένα πεδία πρέπει να οριστούν σε αυτούς τους τύπους. Οι διαφορετικές προσεγγίσεις είναι δυνατές για τον καθορισμό αυτών των πεδίων. Η πρώτη, η απλούστερη προσέγγιση είναι να συνδεθούν τα πεπερασμένα (και ενδεχομένως μικρά) πεδία σε κάθε τύπο. Αυτή η προσέγγιση, που χρησιμοποιείται μέσα στο [PTB04], καθιστά τον καθορισμό του STS εύκολο (και, όπως θα δούμε, επιτρέπει μια αποδοτική αυτοματοποιημένη σύνθεση), εντούτοις, έχει το μειονέκτημα της επιβολής των μη ρεαλιστικών υποθέσεων στους τύπους δεδομένων που αντιμετωπίζονται από τις υπηρεσίες Web. Μια ρεαλιστικότερη (αλλά πιο σύνθετη) προσέγγιση αποτελείται από τη χρησιμοποίηση των αφηρημένων προτύπων για τους τύπους δεδομένων, που αποφεύγουν

την απαρίθμηση όλων των συγκεκριμένων τιμών που αυτοί οι τύποι μπορούν να υποθέσουν, και αντιπροσωπεύοντας ρητά μόνο εκείνες τις πτυχές του τύπου δεδομένων που είναι σχετικές για το κοντινό στόχο.

### Ελεγκτής για μια ειδικού επιπέδου επεξεργασία σύνθεσης

Το πρόβλημα αυτοματοποιημένης σύνθεσης έχει δύο εισαγωγές (σχήμα 5.13): η επίσημη απαίτηση σύνθεσης  $\rho$  και το παράλληλο STS  $\Sigma_{||}$ , το οποίο αντιπροσωπεύει τις υπηρεσίες  $\Sigma_{w_1}, \dots, \Sigma_{w_n}$ . Τώρα τυπικά καθορίζουμε το *παράλληλο προϊόν* δύο STSs, το οποίο διαμορφώνει το γεγονός ότι και τα δύο συστήματα μπορούν να εξελιχθούν ανεξάρτητα, και το οποίο χρησιμοποιείται για να παραγάγει το  $\Sigma_{||}$  από τις συνθετημένες υπηρεσίες Web.

### Καθορισμός 2 (παράλληλο προϊόν)

Έστω  $\Sigma_1 = \langle \mathcal{S}_1, \mathcal{S}_1^0, \mathcal{I}_1, \mathcal{O}_1, \mathcal{R}_1, \mathcal{L}_1 \rangle$  και  $\Sigma_2 = \langle \mathcal{S}_2, \mathcal{S}_2^0, \mathcal{I}_2, \mathcal{O}_2, \mathcal{R}_2, \mathcal{L}_2 \rangle$  να είναι δύο STSs με  $(\mathcal{I}_1 \cup \mathcal{O}_1) \cap (\mathcal{I}_2 \cup \mathcal{O}_2) = \emptyset$ . Το παράλληλο προϊόν  $\Sigma_1 \parallel \Sigma_2$  από τα  $\Sigma_1$  και  $\Sigma_2$  ορίζεται ως

$$\Sigma_1 \parallel \Sigma_2 = \langle \mathcal{S}_1 \times \mathcal{S}_2, \mathcal{S}_1^0 \times \mathcal{S}_2^0, \mathcal{I}_1 \cup \mathcal{I}_2, \mathcal{O}_1 \cup \mathcal{O}_2, \mathcal{R}_1 \parallel \mathcal{R}_2, \mathcal{L}_1 \parallel \mathcal{L}_2 \rangle$$

όπου:

- $\langle (s_1, s_2), a, (s'_1, s'_2) \rangle \in (\mathcal{R}_1 \parallel \mathcal{R}_2)$  if  $\langle s_1, a, s'_1 \rangle \in \mathcal{R}_1$ ;
- $\langle (s_1, s_2), a, (s_1, s'_2) \rangle \in (\mathcal{R}_1 \parallel \mathcal{R}_2)$  if  $\langle s_2, a, s'_2 \rangle \in \mathcal{R}_2$ ;

και

$$(\mathcal{L}_1 \parallel \mathcal{L}_2)(s_1, s_2) = \mathcal{L}_1(s_1) \cup \mathcal{L}_2(s_2).$$

Το σύστημα που αντιπροσωπεύει (οι παράλληλες εξελίξεις) τις συνθετημένες υπηρεσίες  $W_1, \dots, W_n$  του σχήματος 5.13 ορίζονται τυπικά ως  $\Sigma_{||} = \Sigma_{w_1} \parallel \dots \parallel \Sigma_{w_n}$ .

Παρατηρούμε ότι αυτός ο καθορισμός ισχύει μόνο για τη συγκεκριμένη περίπτωση όπου οι εισαγόμενες/εξαγόμενες πληροφορίες της  $\Sigma_1$  και εκείνες της  $\Sigma_2$  είναι ξεχωριστές.

Αυτό είναι μια λογική υπόθεση στην περίπτωση της σύνθεσης των υπηρεσιών Web, όπου τα διαφορετικά συστατικά είναι ανεξάρτητα (π.χ., στη πεδίο του εικονικού ταξιδιωτικού γραφείου, δεν υπάρχει καμία άμεση επικοινωνία μεταξύ της κράτησης ξενοδοχείων και της υπηρεσίας κράτησης πτήσης). Είναι εντούτοις δυνατό να επεκταθεί η προσέγγιση σε μια πιο γενικότερη περίπτωση όπου τα  $\Sigma_1$  και  $\Sigma_2$  μπορούν να στείλουν μηνύματα ο ένας στον άλλο [δηλ.  $(\mathcal{I}_1 \cup \mathcal{O}_1) \cap (\mathcal{I}_2 \cup \mathcal{O}_2) \neq \emptyset$ ] τροποποιώντας με έναν κατάλληλο τρόπο τον καθορισμό του παράλληλου προϊόντος.

Το πρόβλημα αυτοματοποιημένης σύνθεσης συνίσταται στην παραγωγή ενός STS  $\Sigma_c$  που ελέγχει το  $\Sigma_{//}$  ικανοποιώντας το  $\rho$ . Καθορίζουμε τώρα τυπικά το STS περιγράφοντας τις συμπεριφορές ενός STS  $\Sigma$  όταν ελέγχεται από το  $\Sigma_c$ .

### Καθορισμός 3 (ελεγχόμενο σύστημα)

Έστω  $\Sigma = \langle \mathcal{S}, \mathcal{S}^0, \mathcal{I}, \mathcal{O}, \mathcal{R}, \mathcal{L} \rangle$  και  $\Sigma_c = \langle \mathcal{S}_c, \mathcal{S}_c^0, \mathcal{O}, \mathcal{I}, \mathcal{R}_c, \mathcal{L}_\emptyset \rangle$  να είναι δύο συστήματα μεταβολής κατάστασης όπου  $\mathcal{L}_\emptyset(s_c) = \emptyset$  για όλα τα  $s_c \in \mathcal{S}_c$ . Το STS  $\Sigma_c \triangleright \Sigma$ , που περιγράφει τις συμπεριφορές του συστήματος  $\Sigma$  όταν ελέγχεται από το  $\Sigma_c$ , ορίζεται ως:

$$\Sigma_c \triangleright \Sigma = \langle \mathcal{S}_c \times \mathcal{S}, \mathcal{S}_c^0 \times \mathcal{S}^0, \mathcal{I}, \mathcal{O}, \mathcal{R}_c \triangleright \mathcal{R}, \mathcal{L} \rangle$$

όπου:

- $\langle (s_c, s), \tau, (s'_c, s') \rangle \in (\mathcal{R}_c \triangleright \mathcal{R})$  if  $\langle s_c, \tau, s'_c \rangle \in \mathcal{R}_c$ ;
- $\langle (s_c, s), \tau, (s_c, s') \rangle \in (\mathcal{R}_c \triangleright \mathcal{R})$  if  $\langle s, \tau, s' \rangle \in \mathcal{R}$ ;
- $\langle (s_c, s), a, (s'_c, s') \rangle \in (\mathcal{R}_c \triangleright \mathcal{R})$ , with  $a \neq \tau$ , if  $\langle s_c, a, s'_c \rangle \in \mathcal{R}_c$  and  $\langle s, a, s' \rangle \in \mathcal{R}$ .

Παρατηρούμε ότι απαιτούμε οι εισαγωγές  $\Sigma_c$  να συμπίπτουν με τα αποτελέσματα  $\Sigma$  και αντίστροφα. Παρατηρούμε επίσης ότι, αν και τα συστήματα συνδέονται έτσι ώστε η παραγωγή του ενός να συνδέεται στην εισαγωγή του άλλου, οι προκύπτουσες μεταβολές

$\mathcal{R}_c \triangleright \mathcal{R}$  ονομάζονται από τις ενέργειες εισαγωγής/παραγωγής. Αυτό επιτρέπει σε μας να διακρίνουμε τις μεταβάσεις που αντιστοιχούν σε  $\tau$  ενέργειες  $\Sigma_c$  ή  $\Sigma$  από εκείνες που προέρχονται από τις επικοινωνίες μεταξύ  $\Sigma_c$  και  $\Sigma$ . Τέλος, παρατηρούμε ότι εμείς υποθέτουμε ότι το σχέδιο δεν ονομασίες που να συνδέονται με καταστάσεις.

Ένα STS  $\Sigma_c$  μπορεί να μην είναι επαρκές στο να ελέγξει ένα σύστημα  $\Sigma$ . Πράγματι, πρέπει να εγγυηθούμε ότι, όποτε ένα  $\Sigma_c$  εκτελεί μια μεταβολή παραγωγής, κατόπιν ένα  $\Sigma$  είναι σε θέση να το δεχτεί, και αντίστροφα. Καθορίζουμε τον όρο κάτω από τον οποίο μια κατάσταση  $s$  του  $\Sigma$  είναι σε θέση να δεχτεί μια χορήγηση μηνυμάτων σύμφωνα με το ασύγχρονο πρότυπό μας, το οποίο αφαιρεί μακριά τις σειρές αναμονής. Υποθέτουμε ότι ένα  $s$  μπορεί να δεχτεί ένα μήνυμα  $a$  εάν υπάρχει κάποιος διάδοχος  $s'$  του  $s$  μέσα στο  $\Sigma$ , παίρνοντας από το  $s$  απευθείας ένα περιορισμό  $\tau$  μεταβολών, έτσι ώστε το  $s$  να μπορεί να εκτελέσει μια μετάβαση εισαγωγής που ονομάζεται  $a$ . Αντίστροφα, εάν η κατάσταση  $s$  δεν έχει κανέναν τέτοιο διάδοχο  $s'$ , και το μήνυμα  $a$  στέλνεται στο  $\Sigma$ , κατόπιν μια κατάσταση αδιεξόδου επιτυγχάνεται.

Στον ακόλουθο καθορισμό, και στο υπόλοιπο του εγγράφου, δείχνουμε με  $\tau$ -τερματισμό(-ούς) το σύνολο εκτεταμένων καταστάσεων από  $s$  μέσω μιας ακολουθίας  $\tau$  μεταβάσεων, και κοντά σε  $\tau$ -τερματισμό με  $\mathcal{S} \subseteq \mathcal{S}$  την ένωση  $\tau$ -τερματισμού σε όλους  $s \in \mathcal{S}$ .

#### Καθορισμός 4 (ελεγκτής μη αδιέξοδου)

Έστω  $\Sigma = \langle \mathcal{S}, \mathcal{S}^0, \mathcal{I}, \mathcal{O}, \mathcal{R}, \mathcal{L} \rangle$  να είναι ένα STS και ένα  $\Sigma_c = \langle \mathcal{S}_c, \mathcal{S}_c^0, \mathcal{O}, \mathcal{I}, \mathcal{R}_c, \mathcal{L}_\emptyset \rangle$  να είναι ελεγκτής για το  $\Sigma$ . Το  $\Sigma_c$  λέγεται ότι είναι μη αδιέξοδου για  $\Sigma$  εάν όλες οι καταστάσεις  $(s_c, s) \in \mathcal{S}_c \times \mathcal{S}$  που είναι εφικτές από τις αρχικές καταστάσεις  $(s, a, s') \in \mathcal{R}$  ικανοποιούν τους ακόλουθους όρους:

$\emptyset$  Εάν  $(s, a, s') \in \mathcal{R}$  ανήκει στο  $\mathcal{R}$  με  $a \in \mathcal{I}$  τότε υπάρχουν μερικά  $s'_c \in \tau$   
 - τερματισμού έτσι ώστε  $(s'_c, a, s''_c) \in \mathcal{R}$  για μερικά  $s''_c \in \mathcal{S}_c$ ;

και



$\emptyset$  Εάν  $\langle s_c, a, s'_c \rangle \in \mathcal{R}_{εμε}$   $a \in \mathcal{O}$  τότε υπάρχουν μερικά  $s' \in \tau$ -τερματισμού  
 έτσι ώστε  $\langle s', a, s'' \rangle \in \tilde{\mathcal{R}}$  για μερικά  $s'' \in \mathcal{S}$ .

### Προβλήματα σύνθεσης ειδικού επιπέδου επεξεργασίας

Σε ένα πρόβλημα σύνθεσης υπηρεσιών Web, πρέπει να παραγάγουμε  $\Sigma_c$  που να εγγυείται την ικανοποίηση μιας απαίτησης σύνθεσης  $\rho$  (σχήμα 5.13). Αυτό τυποποιείται απαιτώντας ότι το ελεγχόμενο σύστημα  $\Sigma_c \triangleright \Sigma_{||}$  πρέπει να ικανοποιεί το  $\rho$ , το οποίο καθορίζεται στα σύνολα των εκτελέσεων όπου το  $\Sigma_c \triangleright \Sigma_{||}$  μπορεί να εκτελεστεί.

Προκειμένου να καθορίσουμε τυπικά όταν το  $\Sigma_c \triangleright \Sigma_{||}$  ικανοποιεί το  $\rho$ , πρέπει να καθορίσουμε πρώτα τις εκτελέσεις των  $\Sigma_c \triangleright \Sigma_{||}$ . Για να κάνουμε αυτό, πρέπει να λάβουμε υπόψη ότι το σύστημα μεταβολής κατάστασης  $\Sigma_{||}$  διαμορφώνει μια πεδίο που είναι μόνο μερικώς αισθητή από το  $\Sigma_c$ . Δηλαδή στο χρόνο εκτέλεσης, η σύνθετη υπηρεσία  $\Sigma_c$  δεν μπορεί γενικά να γνωρίσει ακριβώς ποια είναι η επικρατούσα κατάσταση από τις συνθετιμένες υπηρεσίες που διαμορφώνονται από το  $\Sigma_{w_1}, \dots, \Sigma_{w_n}$ . Εξετάζουμε παραδείγματος χάριν την STS αντιστοιχία στην υπηρεσία HotelBooking (σχήμα 5.14). Η σύνθετη υπηρεσία εικονικού γραφείου ταξιδιών δεν έχει καμία πρόσβαση στις τιμές των εσωτερικών μεταβλητών των υπηρεσιών ξενοδοχείων, και μπορεί μόνο να συναγάγει τις τιμές τους από τα μηνύματα που ανταλλάσσονται με τη διαδικασία HotelBooking. Αυτή η αβεβαιότητα έχει δύο διαφορετικές πηγές. Η πρώτη είναι η τυποποιημένη πηγή από την αβεβαιότητα στον σχεδιασμό, δηλαδή η παρουσία μη-αποφασιστικών μεταβολών (π.χ., δύο  $\tau$  μεταβάσεις του HotelBooking από "PC = checkAvailable", οι οποίες διαμορφώνουν το γεγονός ότι τα δωμάτια μπορούν να είναι διαθέσιμα ή όχι για το αίτημα ενός δεδομένου πελάτη). Η δεύτερη πηγή αβεβαιότητας οφείλεται στο γεγονός ότι διαμορφώνουμε ένα ασύγχρονο πλαίσιο και αυτό, επομένως, δεν είναι δυνατό για το εικονικό γραφείο ταξιδιών να ξέρει πότε οι εσωτερικές  $\tau$  μεταβολές εκτελούνται στην υπηρεσία HotelBooking. Λόγω αυτής της αβεβαιότητας, μετά από ένα μήνυμα "request (s, l)" που έχει σταλεί στην υπηρεσία HotelBooking, είναι

αδύνατο να διακρίνει εάν η υπηρεσία κράτησης ακόμα ελέγχει εάν η παράδοση είναι δυνατή (PC = checkAvailable), ή εάν αυτός ο στόχος έχει ολοκληρωθεί θετικά (το PC είναι isAvailable, prepareOffer, ή sendOffer) ή αρνητικά ( το PC είναι isNotAvailable, prepareNotAvail, ή sendNotAvail). Αυτή η αβεβαιότητα εξαφανίζεται μόνο όταν ένα μήνυμα "offer " ή ένα " not\_avail " λαμβάνεται από τον εικονικό γραφείο ταξιδιών.

### Καθορισμός 5 (εξέλιξη πεποίθησης)

Έστω  $B \subseteq \mathcal{S}$  να είναι μια πεποίθηση σε κάποιο σύστημα κατάστασης μεταβολής  $\Sigma$ . Καθορίζουμε την εξέλιξη του  $B$  στο πλαίσιο της ενέργειας  $a$  ως πεποίθηση  $B' = \text{εξέλιξη}(B, a)$ , όπου :

$$\text{εξέλιξη}(B, a) = \{s' : \exists s \in \tau\text{-closure}(B). \langle s, a, s' \rangle \in \mathcal{R}\}.$$

Στον καθορισμό της ικανοποίησης στόχου, χρησιμοποιούμε τις πεποιθήσεις για να περιγράψουμε τις διαφορετικές "διαμορφώσεις" που επιτεύχθηκαν κατά τη διάρκεια της εκτέλεσης. Προκειμένου να χαρακτηρίσουμε την ικανοποίηση στόχου, χρειάζεται να καθορίσουμε πότε μια πεποίθηση  $B$  ικανοποιεί μια δεδομένη κατάσταση ιδιοκτησίας. Στον προγραμματισμό μερικής αίσθησης, το  $B$  μπορεί να ικανοποιήσει το  $p$  απλά, εάν όλα οι καταστάσεις  $s \in B$  ικανοποιούν το  $p$ . Ο καθορισμός γίνεται πιο σύνθετος στην ασύγχρονη ρύθμισή μας, λόγω της παρουσίας  $\tau$  μεταβολών.

Ας εξετάσουμε πάλι την υπηρεσία HotelBooking. Μετά από ένα αίτημα το μήνυμα έχει σταλεί στην υπηρεσία, δεν είναι ακόμα δυνατό να προβλέψει εάν η κράτηση μπορεί να εκπληρωθεί ή όχι επομένως, αναμένουμε ότι όροι PC = prepareOffer και PC = notAvailable είναι και οι δύο ψεύτικοι μέσα στο  $B$ . Αφ' ετέρου, αφότου αναγνωριστεί ότι το μήνυμα έχει παραληφθεί από την υπηρεσία κράτησης, είναι αναπόφευκτο να παραδοθεί μια επιτυχής κατάσταση επομένως, θέλουμε να είναι σε θέση να καταλήξει στο συμπέρασμα ότι ο όρος PC = SUCC είναι αληθινός στην αντίστοιχη πεποίθηση. Ανεπίσημα, υποθέτουμε ότι η εκτέλεση των  $\tau$  μεταβολών δεν μπορεί να αναβληθεί για πάντα.

Επομένως, εάν η εκτέλεση των  $\tau$  μεταβολών είναι εγγυημένη για να παραδώσει μια επιτυχημένη κατάσταση που ικανοποιεί τον όρο, έπειτα μπορούμε να υποθέσουμε ότι ο όρος κρατά επίσης σε πεποίθηση τον όρο  $B$ . Αντιθέτως, εάν υπάρχει κάποια ακολουθία  $\tau$  μεταβολών που δεν περιέχουν την ικανοποίηση της κατάστασης  $p$  και αυτή δεν μπορεί να επεκταθεί περαιτέρω με άλλες  $\tau$  μεταβολές έτσι ώστε η  $p$  να επιτυγχάνεται, έπειτα η ιδιότητα  $p$  δεν είναι ικανοποιημένη μέσα στο  $B$ .

### **5.3.3 Αυτοματοποιημένη υπηρεσία σύνθεσης ειδικού επιπέδου επεξεργασίας**

Εξετάζουμε τώρα το πρόβλημα της αυτόματης παραγωγής μιας υπηρεσίας Web σύνθεσης ειδικού επιπέδου επεξεργασίας. Σύμφωνα με το σχήμα 5.13, διακρίνουμε τέσσερα ξεχωριστά βήματα:

1. **Από τις υπηρεσίες Web στα συστήματα μεταβολής κατάστασης.** Αυτό το βήμα αποτελείται από την απόκτηση των περιγραφών της ειδικού επιπέδου επεξεργασίας των υπαρχουσών υπηρεσιών Web που συνθέτονται, και από τη μετατροπή της περιγραφής κάθε μιας από αυτές τις υπηρεσίες σε ένα σύστημα μεταβολής κατάστασης.
2. **Έκφραση των απαιτήσεων σύνθεσης.** Αυτό το βήμα αφορά τον καθορισμό της απαίτησης  $p$  που καθορίζει τη λειτουργία που πρέπει να παρέχει η συντεθειμένη υπηρεσία Web.
3. **Σύνθεση της σύνθεσης.** Κατά τη διάρκεια αυτού του βήματος, το σύστημα μεταβολής κατάστασης εκτελεί τη σύνθεση των υπηρεσιών Web που αυτόματα δημιουργεί την έναρξη από το STS που δημιουργείται στο βήμα 1 και από την απαίτηση που καθορίζεται στο βήμα 2.
4. **Ανάπτυξη και εκτέλεση της συνθετημένης υπηρεσίας.** Σε αυτό το βήμα, το STS που παράγεται στο βήμα 3 είναι μεταφρασμένο στον εκτελέσιμο κώδικα.

Παρακάτω, απαριθμούμε τα τέσσερα βήματα που περιγράφονται ανωτέρω.

### **Από τις υπηρεσίες Web στα συστήματα μεταβολής κατάστασης**

Αυτή η φάση εκτελείται χωριστά για κάθε υπάρχουσα υπηρεσία Web που συμμετέχει στη σύνθεση. Μπορεί να θεωρηθεί ως συνδυασμός δύο βημάτων: η απόκτηση της ειδικού επιπέδου επεξεργασίας περιγραφή της υπηρεσίας Web και η μετατροπή αυτής της περιγραφής σε ένα σύστημα μεταβολής κατάστασης.

Δεν περιγράφουμε τις τεχνικές λεπτομέρειες στην απόκτηση της ειδικού επιπέδου επεξεργασίας περιγραφής από μια υπηρεσία Web, δεδομένου ότι είναι έντονη τεχνολογία και εξαρτημένη γλώσσα. Παρατηρούμε ότι, δεδομένου ότι μια τέτοια περιγραφή είναι απαραίτητη για την άδεια των αλληλεπιδράσεων με την υπηρεσία Web, κάθε τεχνολογία θα πρέπει να παρέχει έναν μηχανισμό για τη δημοσίευση των περιγραφών ειδικού επιπέδου επεξεργασίας. Ενώ μια περιγραφή φυσικής γλώσσας του πρωτοκόλλου που πρέπει να ακολουθείται για την αλληλεπίδραση με μια υπηρεσία Web είναι ικανοποιητική για τη χειρόγραφη σύνθεση, είναι σαφές ότι μια επίσημη περιγραφή τέτοιου πρωτοκόλλου είναι απαραίτητη για την υποστήριξη αυτής της σύνθεσης με τα αυτόματα εργαλεία. Πράγματι, γλώσσες που παρέχουν μια ειδικού επιπέδου επεξεργασίας περιγραφή από τις υπηρεσίες Web και υποστηρίζουν την αυτοματοποιημένη σύνθεση έχουν προταθεί από τις κύριες προσεγγίσεις υπηρεσιών Semantic Web, που βλέπουν παραδείγματος χάριν τα *πρότυπα διαδικασίας OWL-S* ή το *WSMO μοντέλο διεπαφής των υπηρεσιών Web*. Η ανάγκη της περιγραφής γλώσσας ειδικού επιπέδου επεξεργασίας αναγνωρίζεται επίσης και εκτός από την κοινότητα υπηρεσιών του SemanticWeb. Παραδείγματος χάριν, η BPEL4WS γλώσσα, σαν μια γλώσσα βιομηχανικών προτύπων για τη διαμόρφωση και εκτέλεση υπηρεσιών Web, επιτρέπει τη δημοσίευση της αποκαλούμενης *αφηρημένης* περιγραφής της διαδικασίας υπηρεσίας Web, δηλ., η περιγραφή της ακολουθίας αλληλεπιδράσεων που είναι απαραίτητες προκειμένου να εκτελεστεί μια μακροπρόθεσμη αλληλεπίδραση με έναν δεδομένο φορέα παροχής υπηρεσιών.

Μόλις ληφθεί η περιγραφή ειδικού επιπέδου επεξεργασίας μιας υπάρχουσας υπηρεσίας Web, αυτή η περιγραφή είναι μετετραμμένη στα συστήματα μεταβολής κατάστασης. Αυτή η μετατροπή εξαρτάται από το συγκεκριμένη γλώσσα ειδικού επιπέδου επεξεργασίας. Οι συγκεκριμένες μετατροπές έχουν καθοριστεί μέσα στο [TP04] για την περίπτωση των

*OWL-S* διαμόρφωσης διαδικασιών και μέσα στο [PBB+ 04] για την περίπτωση των περιγραφών ειδικού επιπέδου επεξεργασίας που εκφράζονται σε BPEL4WS. Οι μετατροπές από άλλες γλώσσες είναι εύκολο να καθοριστούν, λόγω της γενικότητας και της εκφραστικότητας των συστημάτων μεταβολής κατάστασης.

### Έκφραση των απαιτήσεων σύνθεσης

Αυτό είναι ένα κρίσιμο βήμα, δεδομένου ότι είναι ευθύνη της απαίτησης σύνθεσης να καθορίσει ακριβώς ποιες είναι οι αναμενόμενες συμπεριφορές της σύνθεσης. Στο εισαγωγικό μέρος από την παράγραφο 5.2 διακρίναμε δύο διαφορετικά σενάρια της εφαρμογής για την ειδικού επιπέδου επεξεργασίας σύνθεση, δηλαδή ένα **on-the-fly** σενάριο και ένα σενάριο παραγωγής υπηρεσιών. Αυτά τα δύο σενάρια απαιτούν τα διαφορετικά είδη απαιτήσεων σύνθεσης, έτσι τα εξετάζουμε χωριστά.

Στην περίπτωση της **on-the-fly** σύνθεσης των υπηρεσιών Web, πρέπει να παραγάγουμε εκτελέσιμο κώδικα που να ικανοποιεί το αίτημα ενός συγκεκριμένου πελάτη. Στην περίπτωση του εικονικού ταξιδιωτικού γραφείου, παραδείγματος χάριν, το αίτημα του πελάτη διευκρινίζει μια δεδομένη τοποθεσία  $l$  που επισκέπτεται σε μια δεδομένη περίοδο  $p$ . Προκειμένου να ικανοποιηθεί αυτό το αίτημα, το γραφείο ταξιδιών πρέπει να βρει την πτήση και το δωμάτιο ξενοδοχείου που να είναι συμβατό με το αίτημα του πελάτη. Η απαίτηση σύνθεσης θα μπορούσε να είναι κάτι σαν:

εάν μια προσφορά ταξιδιού είναι δυνατή,  
τότε πουλήστε μια προσφορά ταξιδιού στον πελάτη.

Στο πλαίσιο της εργασίας μας, η προσφορά είναι δυνατή εάν είναι δυνατό να κρατηθεί η πτήση ή/και το ξενοδοχείο για την περίοδο και για τη τοποθεσία που διευκρινίζεται από τον πελάτη. Το γεγονός ότι η προσφορά πωλείται περιγράφεται με την απαίτηση ότι οι διαδικασίες HotelBooking και οι FlightBooking φθάνουν σε κατάσταση επιτυχίας. Ο όρος του στόχου μπορεί ως εκ τούτου να διευκρινιστεί από τον τύπο:

HotelBooking.AvailableHotel[p, l] ≠ UNDEF &  
FlightBooking.AvailableFlight[p, l] ≠ UNDEF

→ HotelBooking.pc = SUCC & FlightBooking.pc = SUCC &  
 h = HotelBooking.AvailableHotel[p, l] &  
 f = FlightBooking.AvailableFlight[p, l] &  
 c = HotelBooking.CostOfRoom[p, h] +  
     FlightBooking.CostOfFlight[f]

όπου το c, το f, και το h περιγράφουν, αντίστοιχα, το κόστος της προσφοράς, και της συγκεκριμένης πτήσης και πληροφορίες ξενοδοχείων. Αυτή η απαίτηση πρέπει να ερμηνευθεί ως όρος που πρέπει να κρατηθεί στο τέλος της εκτέλεσης του κώδικα για την εφαρμογή της σύνθεσης.

Η περίπτωση της **παραγωγής υπηρεσιών** είναι πιο σύνθετη. Εξετάζουμε το ακόλουθο παράδειγμα, όπου θέλουμε να παραγάγουμε αυτόματα τη σύνθετη υπηρεσία για την εφαρμογή τον εικονικού γραφείου ταξιδιών. Ο στόχος του γραφείου ταξιδιών είναι "να πουλήσει μια προσφορά ταξιδιού στον πελάτη".

Αυτό σημαίνει ότι θέλουμε την εικονική υπηρεσία γραφείου ταξιδιών για να φθάσουμε στην κατάσταση όπου η προσφορά έχει υποβληθεί στον πελάτη, ο πελάτης έχει επιβεβαιώσει αυτήν την προσφορά, και την υπηρεσία έχει επιβεβαιώσει την αντιστοιχία (sub-)offers στις υπηρεσίες HotelBooking και FlightBooking. Εντούτοις, το ξενοδοχείο μπορεί να μην έχει κανένα διαθέσιμο δωμάτιο, η πτήση μπορεί να μην είναι δυνατή, ο χρήστης δεν μπορεί να δεχτεί την προσφορά λόγω του κόστους του. Δεν μπορούμε να αποφύγουμε αυτές τις καταστάσεις, και εμείς επομένως δεν μπορούμε να ζητήσουμε από τη σύνθετη υπηρεσία να εγγυηθεί αυτήν την απαίτηση. Εντούτοις, θα επιθυμούσαμε από την εικονική υπηρεσία γραφείου ταξιδιών να προσπαθήσει (να κάνει οτιδήποτε είναι δυνατό) για να την ικανοποιήσει. Επιπλέον, στην περίπτωση "πουλήστε μια προσφορά ταξιδιού στον πελάτη" η απαίτηση δεν είναι ικανοποιημένη, θα θέλαμε το εικονικό γραφείο ταξιδιών να μη δεσμεύει μια διαταγή για ένα δωμάτιο ή για μια πτήση, δεδομένου ότι δεν θέλουμε η υπηρεσία να κρατά και να πληρώνει τα δωμάτια και τις πτήσεις εάν αυτό δεν γίνει αποδεκτό από τον πελάτη. Ας καλέσουμε αυτήν την απαίτηση "ποτέ χωριστή δέσμευση". Η γενική απαίτησή μας επομένως θα ήταν κάτι σαν:

προσπαθήστε "να πουλήσετε μια προσφορά ταξιδιού στον πελάτη"

επάνω στην αποτυχία,  
"ποτέ χωριστή δέσμευση".

Παρατηρούμε ότι η δευτερεύουσα απαίτηση ("ποτέ χωριστή δέσμευση") έχει μια διαφορετική ισχύ w.r.t. από την αρχική ("πωλείστε μια προσφορά ταξιδιού στον πελάτη"). Γράφουμε ότι "κάνει" την ικανοποίηση, παρά "να προσπαθήσει " να ικανοποιηθεί. Πράγματι, σε αυτήν την περίπτωση η πρωταρχική απαίτηση δεν είναι ικανοποιημένη, θέλουμε τη δευτερεύουσα απαίτηση για να εγγυηθούμε.

Χρειαζόμαστε μια επίσημη γλώσσα που μπορεί να εκφράσει τις απαιτήσεις όπως εκείνες του προηγούμενου παραδείγματος, συμπεριλαμβανομένων των όρων των διαφορετικών ισχύων (όπως "προσπαθώ" και "κάνω"), και τις προτιμήσεις μεταξύ των διαφορετικών (π.χ., αρχική και δευτερεύουσα) απαιτήσεων. Για αυτόν τον λόγο, δεν μπορούμε απλά να χρησιμοποιούμε έναν συγκεκριμένο τύπο όπως κάναμε για την περίπτωση της **on-the-fly** σύνθεσης. Χρησιμοποιούμε άντ' αυτού τη γλώσσα EAGLE, η οποία έχει σχεδιαστεί με σκοπό να ικανοποιήσει τέτοια εκφραστικότητα. Ένας λεπτομερής καθορισμός και μια επίσημη σημασιολογία για τη γλώσσα EAGLE βρίσκεται μέσα στο [DLPT02]. Εδώ εξηγούμε ακριβώς πώς η EAGLE μπορεί να εκφράσει τη σύνθεση απαίτησης του τρέχοντος παραδείγματος.

Η διαμόρφωση EAGLE της απαίτησης είναι η ακόλουθη:

### **TryReach**

```
HotelBooking.pc = SUCC & FlightBooking.pc = SUCC &  
Customer.pc = SUCC &  
Customer.h = HotelBooking.AvailableHotel[Customer.p, Customer.l] &  
Customer.f = FlightBooking.AvailableFlight[Customer.p, Customer.l] &  
Customer.c = HotelBooking.CostOfRoom[Customer.p, Customer.h] +  
FlightBooking.CostOfFlight[Customer.f]
```

### **Fail DoReach**

```
HotelBooking.pc = Fail & FlightBooking.pc = Fail &  
Customer.pc = Fail
```

Ο στόχος είναι του τύπου " **TryReach**  $c$  **Fail DoReach**  $d$  ". Το **TryReach**  $c$  απαιτεί μια υπηρεσία που προσπαθεί να φθάσει στον όρο  $c$ , στην περίπτωση μας ο όρος "πούλησε μια προσφορά ταξιδιού στον πελάτη ". Κατά τη διάρκεια της εκτέλεσης της υπηρεσίας, μια κατάσταση μπορεί να επιτευχθεί από εκεί όπου δεν είναι πιθανό να παραδοθεί το  $c$ , π.χ., δεδομένου ότι η πτήση δεν είναι διαθέσιμη. Όταν μια τέτοια κατάσταση επιτυγχάνεται, η απαίτηση **TryReach**  $c$  αποτυγχάνει και ο ανακτημένος όρος **DoReach**  $d$ , στην περίπτωση μας "ποτέ χωριστή δέσμευση" εξετάζεται.

### Σύνθεση της σύνθεσης

Αυτό είναι το κύριο βήμα της αυτοματοποιημένης σύνθεσης: ξεκινάμε από ένα σύνολο STS διαμορφώνοντας τις υπάρχουσες υπηρεσίες Web και από μια απαίτηση σύνθεσης, ένα νέο STS παράγεται που εφαρμόζει την αποτελούμενη υπηρεσία. (Επίσημος καθορισμός της λειτουργίας που διενεργείται σε αυτό το βήμα δίνεται στον καθορισμό 8.) Η προσέγγιση που υιοθετείται μέσα στα [PBB+ 04, TP04, PTB04] για την επίτευξη αυτής της αυτοματοποιημένης σύνθεσης είναι βασισμένη στο "σχεδιασμό σαν πρότυπο ελέγχου" [BCPT03, BCP+ 01] το πλαισίου εργασίας.

Όπως συζητείται στην παράγραφο 5.2.1, υπάρχει ένας μεγάλος αριθμός εργασιών όπου ο διαφορετικός σχεδιασμός τεχνικών χρησιμοποιούνται για την επίτευξη της αυτοματοποιημένης σύνθεσης των υπηρεσιών Web. Οι περισσότερες από αυτές τις εργασίες, εντούτοις, περιορίζονται στην περίπτωση που οι υπάρχουσες υπηρεσίες Web είναι ατομικές, δηλ., εκτελούν την λειτουργικού επιπέδου σύνθεση. Αυτός ο περιορισμός επιβάλλεται επίσης από το γεγονός ότι οι τεχνικές προγραμματισμού που χρησιμοποιούνται σε εκείνες τις προσεγγίσεις δεν είναι ικανές να εξετάσουν προηγμένα χαρακτηριστικά γνωρίσματα που απαιτούνται για να εκτελέσουν την ειδικού επιπέδου επεξεργασία σύνθεση.

Ακριβέστερα, μια τεχνική προγραμματισμού ικανή για ειδικού επιπέδου επεξεργασία σύνθεση πρέπει να παρέχει τους τρόπους συμπεριφοράς με τις ακόλουθες δυσκολίες.

Ø **Nondeterminism:** Ο αρμόδιος για το σχεδιασμό δεν μπορεί να προβλέψει την πραγματική αλληλεπίδραση που θα πάρει μέρος με τις εξωτερικές



διαδικασίες, π.χ., αυτό δεν μπορεί να προβλέψει a priori εάν η απάντηση ενός αιτήματος για τη διαθεσιμότητα θα είναι θετικό ή αρνητικό, εάν ένας χρήστης θα επιβεβαιώσει ή **δεν** θα αποδεχτεί μια υπηρεσία, κ.λπ....

Ø **Μερικό observability**: Ο αρμόδιος για το σχεδιασμό μπορεί μόνο να παρατηρήσει τις επικοινωνίες με τις εξωτερικές διαδικασίες δηλαδή δεν έχει καμία πρόσβαση στην εσωτερική κατάσταση και στις μεταβλητές της. Για παράδειγμα, ο αρμόδιος για το σχεδιασμό δεν μπορεί να ξέρει a priori τον κατάλογο διαθέσιμων στοιχείων για πώληση από μια υπηρεσία.

Ø **Εκτεταμένοι στόχοι**: Δεδομένου ότι έχουμε συζητήσει προηγουμένως, οι απαιτήσεις σύνθεσης συχνά περιλαμβάνουν σύνθετους όρους στη συμπεριφορά της διαδικασίας, και όχι μόνο στη τελική κατάσταση. Παραδείγματος χάριν, να απαιτήσουμε ότι η διαδικασία δεν φτάνει ποτέ σε κατάσταση να αγοράζει ένα αντικείμενο που κοστίζει περισσότερο από το διαθέσιμο προϋπολογισμό. Επιπλέον, οι απαιτήσεις χρειάζεται να εκφράσουν όρους προτιμήσεων στους διαφορετικούς στόχους που επιτυγχάνουν. Για παράδειγμα, μια διαδικασία πρέπει να προσπαθήσει πρώτα να κρατήσει και να επιβεβαιώσει και μια πτήση και ένα ξενοδοχείο από δύο διαφορετικούς φορείς παροχής υπηρεσιών, και μόνο εάν μια από τις δύο υπηρεσίες δεν είναι διαθέσιμη, πρέπει να ξαναγυρίσει πίσω και να ακυρώσει και τις δύο κρατήσεις.

Ø Το πλαίσιο εργασίας "σχεδιασμός σαν πρότυπο ελέγχου" εκμεταλλεύεται μια οικογένεια nondeterministic συστημάτων μεταβολής για την διαμόρφωση μιας περιοχής σχεδιασμού καθώς επίσης και τα παραχθέντα σχέδια. Το STS Σ II που χρησιμοποιήθηκε περιέγραψε την πεδίο "σύνθεσης" μπορεί να μεταβληθεί σε μια πεδίο προγραμματισμού που διαμορφώνεται ως nondeterministic συστήματα μετάβασης και, αντίστροφα, μια αντίστροφη σχεδίαση μπορεί να καθοριστεί σχεδιάζοντας το παραχθέν σχέδιο σε έναν STS που καθορίζει την η συγκροτημένη υπηρεσία.

Ø Τέλος, το πλαίσιο εργασίας "σχεδιασμός σαν πρότυπο ελέγχου" έχει παρουσιάσει να προσφέρει μια πολύ αποδοτική προσέγγιση για την παραγωγή σχεδίων, δεδομένου ότι εκμεταλλεύεται τους βασισμένους σε BDD συμβολικούς μηχανισμούς που αναπτύσσονται αρχικά για την επίλυση του προτύπου που

ελέγχει τα προβλήματα [bcm+ 92]. Αυτοί οι συμβολικοί μηχανισμοί επιτρέπουν μια συμπαγή αντιπροσώπευση και για μια αποδοτική εξερεύνηση του διαστήματος αναζήτησης nondeterministic και μερικώς αισθητές πεδία προγραμματισμού. Επιτρέπουν μια αποδοτική παραγωγή όταν εφαρμόζονται επίσης στη σύνθεση των συγκροτημένων υπηρεσιών Web.

Μια λεπτομερής περιγραφή της ακριβούς αντιστοιχίας μεταξύ της σύνθεσης υπηρεσιών και του σχεδιασμού προβλημάτων, και η απόδειξη της ακρίβειας της υιοθέτησης των τεχνικών σχεδιασμού για τα προβλήματα σύνθεσης υπηρεσιών, μπορούν να βρεθούν μέσα στο [PTB04].

### **Ανάπτυξη και εκτέλεση της συνθετημένης υπηρεσίας**

Σε αυτό το βήμα, το STS που καθόρισε την αποτελούμενη υπηρεσία μετασχηματίζεται στον εκτελέσιμο κώδικα. Σε αυτή τη περίπτωση ενδιαφερόμαστε για μια **on-the-fly** σύνθεση, ο κώδικας εκτελείται αμέσως. Στην περίπτωση της παραγωγής υπηρεσιών, ο κώδικας επεκτείνεται, και είναι έτοιμος να απαντήσει στα αιτήματα των χρηστών.

Οι διαφορετικές γλώσσες μπορούν να χρησιμοποιηθούν σαν γλώσσες-στόχοι αυτής της μεταβολής. Μια δυνατότητα είναι να εκμεταλλευτεί τις γενικές γλώσσες όπως την Java για την εφαρμογή των κατασκευών ελέγχου των σχεδίων και οι υπηρεσίες Web καλούν για την εφαρμογή των αλληλεπιδράσεων με τις συνθετημένες υπηρεσίες Web.

Μια πιο ενδιαφέρουσα προσέγγιση συνίσταται στην παραγωγή του κώδικα BPEL4WS.

Η BPEL4WS είναι μια γλώσσα που σχεδιάζεται για την περιγραφή και την ανάπτυξη εκτελέσιμων Υπηρεσιών Web που εκτελούν τους σύνθετους στόχους μέσω των μακροπρόθεσμων αλληλεπιδράσεων με μια από τις περισσότερες εξωτερικές υπηρεσίες. Είναι ως εκ τούτου μια φυσική γλώσσα-στόχος στην περίπτωση μιας σύνθεσης "παραγωγής υπηρεσιών". Η μετάφραση ενός STS σε BPEL4WS είναι εύκολη, δεδομένου ότι BPEL4WS παρέχει τα τυποποιημένα προϊόντα ροής ελέγχου (if-then-else, switch, while...) και ειδικά προϊόντα για την αλληλεπίδραση με τις εξωτερικές υπηρεσίες Web.

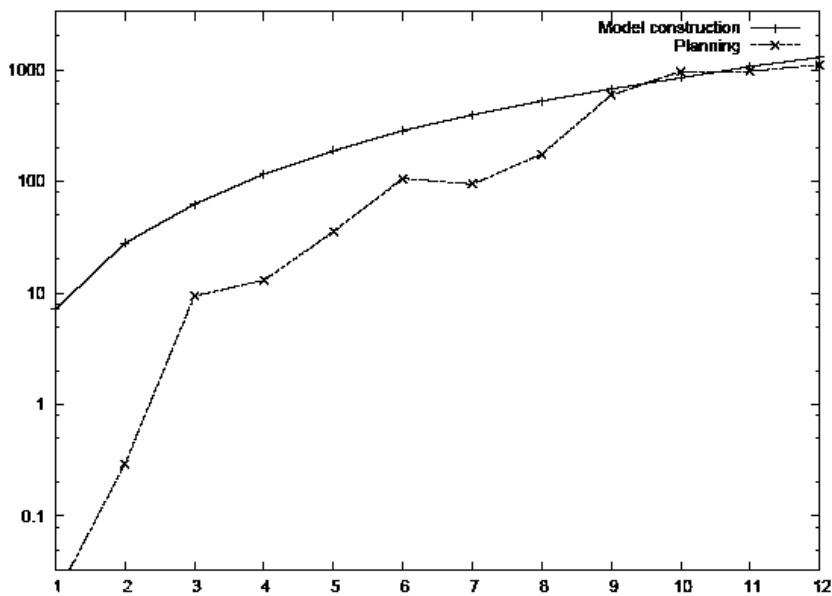
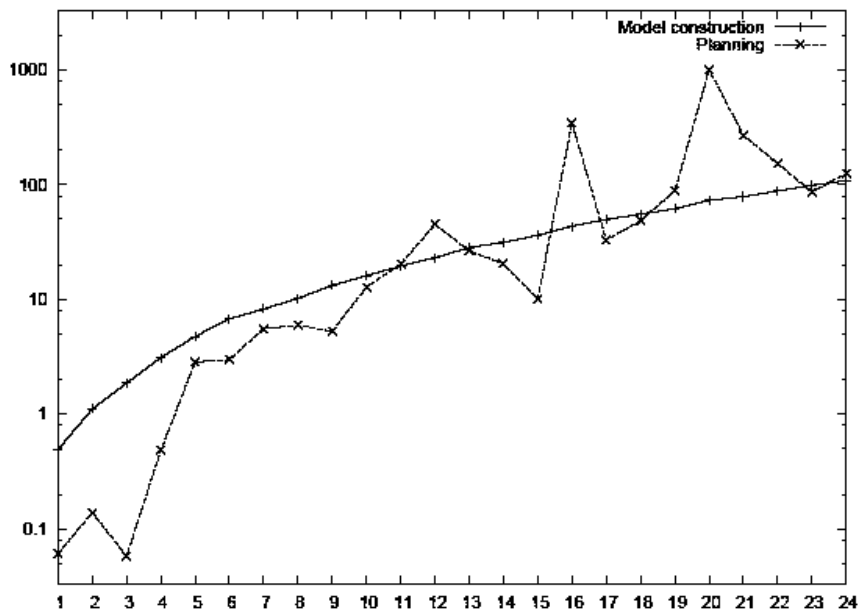
Τέλος, το παραχθέν STS μπορεί να μεταφραστεί σε σημασιολογικές γλώσσες όπως οι OWL-S ή WSMO. Στην περίπτωση των OWL-S ειδικότερα, ένα *πρότυπο διαδικασίας*

μπορεί να χρησιμοποιηθεί για να καθορίσει την ακολουθία διαδικασιών που καθόρισε την αποτελούμενη υπηρεσία. Παρατηρούμε, εντούτοις, ότι αυτή η μεταβολή είναι δυνατή μόνο στην περίπτωση της **on-the-fly** σύνθεσης. Πράγματι, το πρότυπο διαδικασία OWL-S επιτρέπει τον καθορισμό ενός κατάλληλου συνδυασμού επικλήσεων των ατομικών διαδικασιών που παρέχονται από τις υπηρεσίες συστατικού Web (π.χ., οι υπηρεσίες κράτησης ξενοδοχείων και κράτησης πτήσης). Εντούτοις, δεν επιτρέπει το ανακάτεμα αυτών των επικλήσεων με τις αλληλεπιδράσεις με το "χρήστη" (δηλ., ο πελάτης του εικονικού γραφείου ταξιδιών), δεδομένου ότι οι OWL-S δεν επιτρέπει να διαμορφώσει μια διαδικασία που και επικαλείται και ζητά βοήθεια από εξωτερικές υπηρεσίες

#### **5.3.4 Αξιολόγηση και εκτίμηση**

Προκειμένου να εξεταστεί η απόδοση των αυτοματοποιημένων τεχνικών σύνθεσης που περιγράφονται μέσα το προηγούμενο τμήμα, έχουμε πραγματοποιήσει μερικά πειράματα. Έχουμε εξετάσει δύο διαφορετικές κατηγορίες πειραμάτων, τις οποίες καλέσαμε συνθετικές πεδία και πραγματικές πεδία.

Οι πρώτες είναι τεχνητές πεδία, και χρησιμοποιούνται για να εξετάσουν την απόδοση αυτοματοποιημένης σύνθεσης στα εξελικτικά προβλήματα. Τα τελευταία περιγράφουν τα ρεαλιστικά σενάρια για αυτοματοποιημένη σύνθεση και χρησιμοποιούνται για να επικυρώσουν τα αποτελέσματα των πειραμάτων στις συνθετικές πεδία.



Σχήμα 5.13: Τα πειράματα με τα πεδία

### Συνθετικά πεδία

Εξετάζουμε δύο διαφορετικά σύνολα συνθετικών πεδίων. Στο πρώτο σύνολο πειραμάτων, εξετάζουμε την αυτοματοποιημένη σύνθεση w.r.t. ο αριθμός υπηρεσιών που

συντίθενται. Κάθε συστατικό είναι αντιπροσωπευμένο με μια πολύ απλή διαδικασία που απαιτείται για να παρέχει μια υπηρεσία και μπορεί να αποκριθεί είτε θετικά είτε αρνητικά. Η απαίτηση σύνθεσης είναι επίσης πολύ απλή: είτε όλες οι υπηρεσίες τελειώνουν επιτυχώς ή μια αποτυχία αναφέρεται στον υλοποιητή της αποτελούμενης υπηρεσίας. Τα αποτελέσματα παρουσιάζονται στην αριστερή πλευρά του σχήματος 5.13. Στον οριζόντιο άξονα έχουμε τον αριθμό συστατικών. Στον κάθετο άξονα, εκθέτουμε σε δευτερόλεπτα το χρόνο για το πρότυπο κατασκευής (δηλ., ο χρόνος για να χτιστεί το παράλληλο σύστημα μεταβολής κατάστασης  $\Sigma_{\parallel}$  στο σχήμα 5.11 που αρχίζει από τις συνθετημένες υπηρεσίες Web) και ο χρόνος για την αυτοματοποιημένη σύνθεση (δηλ., ο χρόνος που ξοδεύεται για να παραγάγει το σύνθετο STS  $\Sigma_c$  και για να εκπέμψει ως υπηρεσία Web).

Όπως αναμένεται, ο χρόνος για την πρότυπη κατασκευή αυξάνεται ομαλά με τον αριθμό συνθέσεων. Επίσης ο χρόνος σύνθεσης αυξάνεται με μια παρόμοια τάση, αλλά λιγότερο ομαλά. Αυτό εξαρτάται από το βασισμένο σε BDD συμβολικό μηχανισμό που υιοθετείται στη σύνθεση της σύνθεσης, οι οποίοι είναι αρμόδιοι της κατασκευής μιας συμπαγής εσωτερικής αντιπροσώπευσης του διαστήματος αναζήτησης για τη φάση σύνθεσης. Αυτή η εσωτερική αντιπροσώπευση μπορεί να είναι περισσότερο ή λιγότερο αποδοτική, ανάλογα με τη συγκεκριμένη περίπτωση προβλήματος, με τις ισχυρές επιδράσεις της απόδοσης της σύνθεσης. Με αυτά τα παραδείγματα, ο χρόνος που απαιτείται για την αυτοματοποιημένη σύνθεση αυξάνεται λιγότερο από ότι αναμενόταν και κατορθώνει να εξετάσει έναν μάλλον υψηλό αριθμό συστατικών σε μάλλον σύντομο χρονικό διάστημα. Η περίπτωση με τη χειρότερη απόδοση μεταξύ των εξεταζόμενων πειραμάτων είναι αυτή των 20 συστατικών, όπου η πρότυπη κατασκευή παίρνει περίπου 70 δευτερόλεπτα, ενώ η αυτοματοποιημένη σύνθεση διαρκεί περίπου 1000 δευτερόλεπτα.

Παρατηρούμε ότι οι σύνθετες υπηρεσίες Web που χρησιμοποιούνται στο προηγούμενο πείραμα είναι πολύ στοιχειώδης, δεδομένου ότι εφαρμόζουν ουσιαστικά ένα πρωτόκολλο απαίτησης-απάντησης. Στο δεύτερο σύνολο από τα συνθετικά πειράματα, έχουμε εξετάσει την περίπτωση μιας σύνθεσης των υπηρεσιών Web οι οποίες απαιτούν μια σύνθετη αλληλεπίδραση. Ακριβέστερα, έχουμε περιπλέξει την παραμετρική πεδίο με την επιβολή μιας σύνθεσης που απαιτεί έναν υψηλό βαθμό διαστρωμάτωσης μεταξύ των

συστατικών. Εδώ, οι αλληλεπιδράσεις με κάθε συστατικό είναι πιο σύνθετες από ένα ενιαίο βήμα απαίτησης-απάντησης, και, για να επιτύχουν τη σύνθεση, είναι απαραίτητο να πραγματοποιήσουν αλληλεπιδράσεις με όλα τα συστατικά με κάποιο τρόπο. Τέτοια διαστρωμάτωση είναι κοινή μέσα το εικονικό παράδειγμα γραφείου ταξιδιών όπου, π.χ., μια προσφορά μπορεί να σταλεί στον πελάτη μόνο μετά την επιβεβαίωση από τις υπηρεσίες κράτησης ξενοδοχείων και πτήσης ότι υπάρχουν διαθέσιμες θέσεις, και οι κρατήσεις πτήσης και ξενοδοχείων επιβεβαιώνονται μόνο εάν ο πελάτης δέχεται την προσφορά γραφείου ταξιδιών.

Όπως φαίνεται στη γραφική παράσταση στη δεξιά πλευρά του σχήματος 5.15, η αυτοματοποιημένη σύνθεση σε αυτήν την περίπτωση είναι δυσκολότερη απ' ό,τι στο προηγούμενο σύνολο πειραμάτων. Ενώ στο προηγούμενο πείραμα η πρότυπη κατασκευή και η αυτοματοποιημένη σύνθεση με 12 συστατικά πήραν, αντίστοιχα, 25 και 45 δευτερόλεπτα, τώρα παίρνουν και τα δύο περίπου 1200 δευτερόλεπτα. Παρά το γεγονός ότι η απαραίτητη διαστρωμάτωση ελαττώνει τις αποδόσεις, η τεχνική κατορθώνει ακόμα να εξετάσει έναν μάλλον μεγάλο αριθμό συστατικών, δεδομένου ότι αναμένουμε ότι οι ρεαλιστικές συνθέσεις δεν θα περιλαμβάνουν περισσότερα από 12 συστατικά.

### **Πραγματικά πεδία**

Για να επικυρώσουμε τα αποτελέσματα της πειραματικής αξιολόγησης σχετικά με τις συνθετικές πεδία, πρέπει επίσης να πραγματοποιήσουμε μερικά πειράματα της αυτοματοποιημένης σύνθεσης στα προβλήματα που εξήχθησαν από ρεαλιστικές πεδία υπηρεσιών Web. Τα αποτελέσματα αναφέρονται στο σχήμα 5.16. Οι δύο πρώτες περιπτώσεις αντιστοιχούν σε μια πεδίο αγοράς και μεταφοράς επίπλων ( P&S). Συνδυάζει δύο χωριστές, ανεξάρτητες, και υπάρχουσες υπηρεσίες, ένας PRODUCER (ΠΑΡΑΓΩΓΟΣ )επίπλων και μια υπηρεσία παράδοσης (SHIPPER), έτσι ώστε ο χρήστης μπορεί άμεσα να ζητήσει από την αποτελούμενη υπηρεσία P&S να αγοράσει και να παραδώσει ένα δεδομένο στοιχείο σε μια δεδομένη θέση. Στην πρώτη παραλλαγή του πεδίου P&S, η αυτοματοποιημένη σύνθεση είναι πολύ γρήγορη, δεδομένου ότι παρά την διαστρωμάτωση που απαιτείται, έχουμε ακριβώς τρία συστατικά (Shipper, Produced παραγωγός, και χρήστης του P&S ). Έχουμε πειραματιστεί έπειτα με μια πιο σύνθετη έκδοση P&S. Έχουμε προσθέσει μια περαιτέρω υπηρεσία, λαμβάνοντας υπόψη αυτό, σε

πραλιστικές περιπτώσεις, η σύνθετη υπηρεσία μπορεί να απαιτήσει την πληρωμή που να εξετάζεται από το κάποιον τρίτο, δηλ., μια τράπεζα, η οποία μεταβιβάζει τη λήψη χρημάτων από τον πελάτη. Σε αυτό το παράδειγμα, η παρεμβολή διαστρωμάτωσης των αλληλεπιδράσεων αυξάνεται από την ανάγκη μιας επιβεβαίωσης πληρωμής από την τράπεζα πριν η εντολή να μπορεί να επιβεβαιωθεί στον παραγωγό και στον Shipper. Το πειραματικό αποτέλεσμα επιβεβαιώνει ότι το πρόβλημα είναι δυσκολότερο από ότι στο προηγούμενο παράδειγμα P&S, και οι αυτοματοποιημένες συνθέσεις χρόνου αυξάνονται με μια εντολή ποσότητας.

	number of components	model construction	composition
P&S	3	8.4 sec.	1.0 sec.
P&S + BANK	4	39.6 sec.	35.4 sec.
WMO1	5	187.5 sec.	31.6 sec.
WMO2	5	173.1 sec.	48.6 sec.
WMO3	5	174.9 sec.	120.6 sec.

Σχήμα 5.14: Πειράματα με διαφορετικές εφαρμογές.

Τέλος, πειραματιζόμαστε με μια περιπτωσιολογική μελέτη που λαμβάνεται από μια πραγματική e-government εφαρμογή που αναπτύσσουμε για μια ιδιωτική επιχείρηση. Στοχεύουμε στην παροχή μιας υπηρεσίας που εφαρμόζει ένα (δημόσιο) διοικητικό γραφείο των αποβλήτων ( WMO ), δηλ. μια υπηρεσία που διαχειρίζεται τα αιτήματα χρηστών για να δημιουργηθούν πεδία για τη διάθεση των επικίνδυνων αποβλήτων. Σύμφωνα με τους ιταλικούς νόμους που υπάρχουν, ένα τέτοιο αίτημα περιλαμβάνει την αλληλεπίδραση των διαφορετικών γραφείων της δημόσιας διοίκησης. Συγκεκριμένα, ένα γραφείο πρωτοκόλλου είναι υπεύθυνο για την παροχή ενός μοναδικού προσδιοριστικού (το "πρωτόκολλο") στο αίτημα, η Τεχνική Επιτροπή πρέπει να επικαλεσθεί για να παραγάγει μια αξιολόγηση σχετικά με η τεχνική δυνατότητα πραγματοποίησης και του οικολογικού αντίκτυπου της πεδίου, και ένας πίνακας επαρχιών είναι αρμόδιος για να πάρει μια τελική απόφαση. Στην εφαρμογή εξετάζουμε επίσης μια υπηρεσία πολιτών που αλληλεπιδρά με τον πολίτη ή την επιχείρηση που υποβάλλει το αίτημα, και έναν γραμματέα Υπηρεσίας που αλληλεπιδρά με ένα γραφείο γραμματειών. Η απαίτηση σύνθεσης μπορεί εδώ να περιγραφεται από ένα σύνολο περιορισμών για την εκτέλεση

των διαφορετικών διαδικαστικών σταδίων που διενεργείται από διαφορετικά γραφεία. Εξετάζουμε τρεις παραλλαγές αυτής του πεδίου, αντίστοιχα σε μια αύξηση που παρεμβάλλει διαστρωμάτωση μεταξύ των διαφορετικών υπηρεσιών, που παίρνουν σε όλες τις περιπτώσεις μια πολύ καλή εκτέλεση.

Σε όλα τα ρεαλιστικά παραδείγματα, η αυτοματοποιημένη σύνθεση παρουσιάζεται εφικτή και παίρνει ένα μάλλον χαμηλό χρονικό διάστημα, σίγουρα πολύ γρηγορότερο από τη χειρωνακτική ανάπτυξη σύνθετων διαδικασιών. Επιπλέον, οι χρόνοι που απαιτούνται για τη σύνθεση επιβεβαιώνουν τις τάσεις από τα πειράματα που αναφέρονται στο σχήμα 5.13.



## Κεφάλαιο 6

### 6.1 Ενοποίηση της υπηρεσίας ανακάλυψης και της σύνθεσης

Σε αυτό το κεφάλαιο μελετώνται πιθανές προσεγγίσεις για το συνδυασμό της υπηρεσίας ανακάλυψης(discovery service), της λειτουργικού επιπέδου (functional level) υπηρεσίας σύνθεσης, και της διαδικαστικού επιπέδου(process level) υπηρεσίας σύνθεσης. Δίνεται μια προκαταρκτική περιγραφή των πιθανών τρόπων σύνθεσης αυτών των λειτουργιών χωρίς πολλές λεπτομέρειες.

Το υπόλοιπο κεφάλαιο οργανώνεται σε διάφορα τμήματα, κάθε ένα από τα οποία αναλύει ένα διαφορετικό σενάριο του πιθανού συνδυασμού των υπηρεσιών ανακάλυψης και σύνθεσης.

### 6.2 Ανακάλυψη μέσα στη σύνθεση

Αυτό το σενάριο αντιστοιχεί στην επίκληση των λειτουργιών ανακάλυψης μέσα σε μία λειτουργία σύνθεσης. Αυτή η master-slave ενοποίηση των δύο λειτουργιών υιοθετείται ήδη από διαφορετικές λειτουργικού επιπέδου προσεγγίσεις σύνθεσης. Ειδικότερα υιοθετείται από την προσέγγιση που περιγράφηκε στην παράγραφο 5.1.3, όπου ένας μεσολαβητής ανακαλύψεων (discovery mediator) χρησιμοποιείται για να βρει υπάρχουσες υπηρεσίες που ταιριάζουν με έναν νεοσυσταθέν στόχο σύνθεσης (βλ. παραδείγματος χάριν την αρχιτεκτονική της μηχανής της υπηρεσίας ενοποίησης στο σχήμα 5.2).

Μια τέτοια ενοποίηση της ανακάλυψης μέσα στη σύνθεση είναι λιγότερο φυσική στην περίπτωση της επεξεργαστικού επιπέδου σύνθεσης. Πράγματι, θεωρούμε σε αυτήν την περίπτωση ότι οι υπηρεσίες που συμμετέχουν στη σύνθεση είναι ήδη γνωστές όταν αρχίζει η σύνθεση. Παράλληλα, μπορούμε να υποθέσουμε ότι οι επεξεργαστικού επιπέδου υπηρεσίες που συμμετέχουν στη σύνθεση είναι "γενικής χρήσης", υπό την έννοια ότι περιγράφουν τις διαδικασίες που απαιτούνται για να αλληλεπιδράσουν με μια γενικής χρήσης υπηρεσία που παρέχει μία δοθείσα λειτουργία (π.χ., μια γενικής χρήσης υπηρεσία κρατήσεων ξενοδοχείων). Όταν η επεξεργαστικού επιπέδου σύνθεση έχει επιτευχθεί, χρησιμοποιούνται λειτουργίες ανακάλυψης για να βρεθούν "συγκεκριμένες"

περιπτώσεις αυτής της διαδικασίας (δηλ., η υπηρεσία κρατήσεων ξενοδοχείων για μια συγκεκριμένη αλυσίδα ξενοδοχείων). Αυτό είναι ιδιαίτερα ενδιαφέρον στην περίπτωση της "παραγωγής υπηρεσιών" της επεξεργαστικού επιπέδου σύνθεσης: σ' αυτή την περίπτωση, είναι λογικό να προβλέψει ότι διαφορετικές εφαρμογές των γενικής χρήσης υπηρεσιών σύνθεσης, πρέπει να ανακαλυφθούν για κάθε μία από τις αιτήσεις του κάθε πελάτη (π.χ., διαφορετικές υπηρεσίες κράτησης πτήσης πρέπει να χρησιμοποιηθούν ανάλογα με το αν ο πελάτης ταξιδεύει στην Ευρώπη ή εκτός Ευρώπης).

### **6.3 Επαυξητική προσέγγιση**

Σε αυτό το σενάριο η ανακάλυψη, η λειτουργικού επιπέδου σύνθεση και η διαδικαστικού επιπέδου σύνθεση, εφαρμόζονται στη σειρά, προκειμένου να βρεθούν οι λύσεις διαφορετικής πολυπλοκότητας για μια ερώτηση ανακάλυψης. Ακριβέστερα, λαμβάνοντας υπόψη μια ερώτηση που καθορίζει μια υπηρεσία, εκτελούνται τα ακόλουθα βήματα:

1. πρώτα, ψάχνουμε μια απλή, ατομική υπηρεσία Web(atomic Web service) που ταιριάζει με την ερώτηση (υπηρεσία ανακάλυψης)
2. εάν καμία κατάλληλη υπηρεσία δεν βρίσκεται στο βήμα 1, ψάχνουμε για σύνθεση ατομικών Υπηρεσιών Web που ταιριάζουν με την ερώτηση (λειτουργικού επιπέδου σύνθεση)
3. εάν καμία κατάλληλη σύνθεση δεν βρίσκεται στο βήμα 2, ψάχνουμε για επεξεργαστικού επιπέδου σύνθεση των υπηρεσιών Web που ταιριάζουν με την ερώτηση (επεξεργαστικού επιπέδου σύνθεση).

Παρατηρούμε ότι η σύνθεση στο τρίτο βήμα είναι γενικότερη από αυτή στο δεύτερο βήμα, δεδομένου ότι επιτρέπει μια γενική διαστρωμάτωση(general interleaving) μεταξύ των υπηρεσιών της σύνθεσης. Παραδείγματος χάριν, εάν συνθέτουμε μια υπηρεσία κρατήσεων ξενοδοχείων και μια υπηρεσία κρατήσεων πτήσεων, μπορούμε να διαχωρίσουμε τις αλληλεπιδράσεις με τις δύο υπηρεσίες, πράγμα που δεν μπορούμε να κάνουμε στην περίπτωση της λειτουργικού επιπέδου σύνθεσης.

Ένα σημαντικό πλεονέκτημα αυτού του σεναρίου είναι ότι επιτρέπει μία black box ενοποίηση των τριών λειτουργιών. Ο μόνος περιορισμός είναι ότι μια κοινή ερώτηση (ή σύνθεση στόχου) πρέπει να χρησιμοποιηθεί και για τις τρεις λειτουργίες.

## 6.4 Επαναληπτική προσέγγιση

Ο στόχος αυτού του σεναρίου είναι να βελτιωθεί επαυξητικά μια σύνθεση υπηρεσιών. Αρχικά, η σύνθεση ορίζεται ως μια ενιαία αφηρημένη υπηρεσία που αντιστοιχεί στο στόχο της σύνθεσης. Κατά τη διάρκεια της επαυξητικής σύνθεσης, η μερικώς αναπτυγμένη σύνθεση θα περιλάβει συγκεκριμένες υπηρεσίες καθώς επίσης και αφηρημένες υπηρεσίες που αντιπροσωπεύουν τους στόχους για τις περαιτέρω βελτιώσεις.

Κατά τη διάρκεια κάθε βήματος βελτίωσης, μια αφηρημένη υπηρεσία της μερικώς αναπτυγμένης σύνθεσης επιλέγεται και βελτιώνεται περαιτέρω. Η επαυξητική σύνθεση ολοκληρώνεται όταν η παραχθείσα διαδικασία δεν περιέχει καμία αφηρημένη υπηρεσία.

Οι διαφορετικές μορφές βημάτων βελτίωσης είναι δυνατές κατά τη διάρκεια της επαυξητικής παραγωγής της σύνθεσης:

- ∅ η αφηρημένη υπηρεσία αντικαθίσταται από μια ενιαία συγκεκριμένη υπηρεσία (πλήρες ταίριασμα)

- ∅ η αφηρημένη υπηρεσία αντικαθίσταται από μια συγκεκριμένη υπηρεσία και από μια νέα αφηρημένη υπηρεσία περιγράφοντας τη "παραμονή" της αρχικής αφηρημένης διαδικασίας (πλήρης τύπος ταιριάσματος)

- ∅ η αφηρημένη υπηρεσία αντικαθίσταται από έναν διακόπτη, όπου κάθε διακλάδωση αποτελείται από μια συγκεκριμένη υπηρεσία και την "παραμονή" της αφηρημένης υπηρεσίας (ταίριασμα διακόπτη)

- ∅ μία ειδικού επιπέδου επεξεργασία γίνεται, δηλ. η αφηρημένη διαδικασία βελτιώνεται μέσα σε μια γενική σύνθεση άλλων (αφηρημένων και συγκεκριμένων) υπηρεσιών Web.

Οι διαφορετικές στρατηγικές μπορούν να υιοθετηθούν για την επιλογή του μετασχηματισμού που εκτελείται κατά τη διάρκεια ενός βήματος βελτίωσης, και είναι κατάλληλες και ικανές για λύση χωρίς κρίση και μπορούν να χρησιμοποιηθούν για να καθοδηγήσουν αυτήν την διαδικασία.

Η επαυξητική σύνθεση μπορεί να θεωρηθεί ως γενίκευση του αλγορίθμου για τη σύνθεση με το μερικό ταίριασμα τύπων που περιγράφεται στο σχήμα 3.3, το οποίο περιλαμβάνει ήδη τα πρώτα τρία είδη βημάτων βελτίωσης.

Παρατηρούμε ότι αυτό το σενάριο μπορεί επίσης να θεωρηθεί ως γενίκευση της επαυξητικής προσέγγισης που καθορίζεται προηγουμένως. Σε αυτήν την περίπτωση, εντούτοις, οι διαφορετικές λειτουργίες ανακάλυψης και σύνθεσης δεν είναι συνδυασμένες σαν μαύρο κουτί, αλλά είναι ενσωματωμένες σε έναν ενιαίο αλγόριθμο σύνθεσης.

### **6.5 Σύνθεση δύο επιπέδων**

Σε αυτό το σενάριο, τα δύο είδη σύνθεσης εκτελούνται στη σειρά, προκειμένου να προσδιορίσουν τη σύνθεση στα δύο επίπεδα αφηρημένων εννοιών. Η λειτουργικού επιπέδου σύνθεση εκτελείται πρώτα, προκειμένου να προσδιοριστεί ένα σύνολο υπηρεσιών Web που πρέπει να συντεθούν, και για να λάβει μια υψηλού επιπέδου περιγραφή της σύνθεσής τους. Κατόπιν, η ειδικού επιπέδου διαδικασία σύνθεσης

εκτελείται προκειμένου να βελτιώσει αυτή την υψηλού επιπέδου περιγραφή της αποτελούμενης υπηρεσίας στη χαμηλού επιπέδου, εκτελέσιμη σύνθεση.

Σε αυτό το σενάριο, ένας ενιαίος στόχος σύνθεσης πρέπει να διευκρινιστεί, δηλαδή ο στόχος για λειτουργικού επιπέδου σύνθεση. Πράγματι, μετά την απόκτηση υψηλού επιπέδου προδιαγραφής ο στόχος τέθηκε στην ειδικά επεξεργασμένη διαδικασία σύνθεσης. Ένα πρόβλημα που πρέπει να λυθεί για την εφαρμογή αυτού του σεναρίου είναι να καταλάβουμε πώς να χρησιμοποιήσουμε το υψηλό επίπεδο υπηρεσιών για να καθοδηγήσουμε τη χαμηλή σύνθεση επιπέδων.

### **6.6 Ανακάλυψη ειδικού επιπέδου επεξεργασίας**

Αυτό το σενάριο είναι διαφορετικό από τα προηγούμενα όπως αντιστοιχεί σε ένα διαφορετικό είδος "ενοποίησης" μεταξύ της ανακάλυψης και της σύνθεσης. Η ανακάλυψη που περιγράφεται στο κεφάλαιο 2 μπορεί να εξεταστεί σε λειτουργικό επίπεδο. Πράγματι, οι υπηρεσίες Web που ανακαλύπτονται περιγράφονται από την άποψη των εισαγόμενων πληροφοριών, των εξαγόμενων πληροφοριών, των προϋποθέσεων, και των αποτελεσμάτων τους. Ομοίως σε αυτά που συμβαίνουν για τη σύνθεση, μπορούμε να σκεφτούμε έναν χαμηλότερο τύπο ανακάλυψης, δηλαδή την ανακάλυψη ειδικού επιπέδου επεξεργασίας. Σε αυτήν την περίπτωση, στην ερώτηση δεν διευκρινίζουμε μόνο τη λειτουργική παράμετρο από μια υπηρεσία που ανακαλύπτεται,

αλλά διευκρινίζουμε επίσης τους όρους σε ολόκληρο το πρωτόκολλο που είναι ανάγκη να ακολουθηθεί για να αλληλεπιδράσει με την υπηρεσία.

Η ειδικού επιπέδου επεξεργασία ανακάλυψης είναι χρήσιμη να εκφράσει στην ερώτηση τους όρους αναζήτησης στην αλληλεπίδραση που ρέουν με τις υπηρεσίες (π.χ., μπορούμε να διευκρινίσουμε ότι θέλουμε μια υπηρεσία κράτησης ξενοδοχείων που να επιτρέπει σε μας να ακυρώσουμε την κράτηση μας οποιαδήποτε στιγμή ή μια υπηρεσία που να πιστώνει την κάρτα μας μόνο εάν ενδιαφερόμαστε να δεχτούμε μια προσφορά). Η ειδικού επιπέδου επεξεργασία ανακάλυψης μπορεί επίσης να χρησιμοποιείται για να βελτιώσει και να περιορίσει το σύνολο υπηρεσιών που εμφανίζονται μετά από ένα λειτουργικού επιπέδου βήμα ανακαλύψεων.

## Βιβλιογραφία

### Προτεινόμενα βιβλία για ανάγνωση

- [ACD+03] T. Andrews, F. Curbera, H. Dholakia, Y. Goland, J. Klein, F. Leymann, K. Liu, D. Roller, D. Smith, S. Thatte, I. Trickovic, and S. Weerawarana. Business Process Execution Language for Web Services Version 1.1. <http://www.ibm.com/developerworks/library/ws-bpel/>, May 2003.
- [ACKM03] G. Alonso, F. Casati, H. Kuno, and V. Machiraju. *Web Services*. Springer, 2003.
- [Ame91] P. America. *Designing an object-oriented programming language with behavioural subtyping*, volume 489 of *Lecture Notes in Computer Science*, pages 60–90. Springer-Verlag, 1991
- [Baa96] F. Baader. Using automata theory for characterizing the semantics of terminological cycles. *Annals of Mathematics and Artificial Intelligence*, 18(2{4}):175-219,1996
- [BCE+02] T. Bellwood, L. Clement, D. Ehnebuske, A. Hately, Maryann Hondo, Y.L. Husband, K. Januszewski, S. Lee, B. McKee, J. Munter, and C. von Riegen. UDDI version 3.0. <http://uddi.org/pubs/uddi-v3.00-published-20020719.htm>, July 2002.
- [BCF04] Walter Binder, Ion Constantinescu, and Boi Faltings. A directory for web service integration supporting custom query pruning and ranking. In *European Conference on Web Services (ECOWS 2004)*, Erfurt, Germany, September 2004.
- [BCG+03] D. Berardi, D. Calvanese, G. De Giacomo, M. Lenzerini, and M. Mecella. Automatic composition of E-Services that export their behaviour. In *Proc. ICSOC'03*, 2003.
- [BCM+92] J. R. Burch, E. M. Clarke, K. L. McMillan, D. L. Dill, and L. J. Hwang. Symbolic Model Checking: 10<sup>20</sup> States and Beyond. *Information and Computation*, 98(2):142–170, June 1992.
- [BCP+01] P. Bertoli, A. Cimatti, M. Pistore, M. Roveri, and P. Traverso. MBP: a Model Based Planner. In *Proc. of IJCAI-2001 workshop on Planning under Uncertainty and Incomplete Information*, 2001.
- [BCPT03] P. Bertoli, A. Cimatti, M. Pistore, and P. Traverso. A Framework for Planning with Extended Goals under Partial Observability. In *Proc. ICAPS'03*, 2003.

- [BDG03] J. Blythe, E. Deelman, and Y. Gil. Planning for workflow construction and maintenance on the grid. In *Proc. of ICAPS'03 Workshop on Planning for Web Services*, 2003.
- [Bec02] Dave Beckett. RDF/XML syntax specification (revised). W3C Working Draft, 2002. Available at <http://www.w3.org/TR/2002/WD-rdf-syntax-grammar-20021108>
- [Ber98] T. Berners-Lee. What the Semantic Web Can Represent. September 1998. <<http://www.w3.org/DesignIssues/RDFnot.html>>.
- [Ber00] Tim Berners-Lee, "Semantic Web on XML", Keynote presentation for XML 2000. Slides available at: <http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide1-0.html>. Reporting available at: <http://www.xml.com/pub/a/2000/12/xml2000/timbl.html>
- [BF99] T. Berners-Lee, with M. Fischetti. *Weaving the Web*. San Francisco: Harper, 1999.
- [BG00] Dan Brickley & R.V. Guha, "Resource Description Framework (RDF) Schema Specification 1.0", W3C Candidate Recommendation 27 March 2000, World Wide Web Consortium, Cambridge (MA); available on-line as <http://www.w3.org/TR/rdf-schema/>.
- [BH91] F. Baader and P. Hanschke. A schema for integrating concrete domains into concept languages. In *Proc. of the 12th Int. Joint Conf. on Artificial Intelligence (IJCAI-91)*, pages 452-457, Sydney, 1991.
- [BHL01] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American* 284 (May 2001): 34-43.
- [BHRT03] B. Benatallah, M-S. Hacid, C. Rey, and F. Toumani. Request rewriting based Web service discovery. In *The Semantic Web - ISWC 2003*, pages 242-257, October 2003.
- [BHS02] Franz Baader, Ian Horrocks, and Ulrike Sattler. Description Logics for the Semantic Web. *KI - Künstliche Intelligenz*, 16(4):57-59, 2002.
- [BHV01] Walter Binder, Jarle Hulaas, Alex Villazón, and Rory Vidal. Portable resource control in Java: The J-SEAL2 approach. In *ACM Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA-2001)*, Tampa Bay, Florida, USA, October 2001.
- [Bin01] Walter Binder. Design and implementation of the J-SEAL2 mobile agent kernel. In *The 2001 Symposium on Applications and the Internet (SAINT-2001)*, San Diego, CA, USA, January 2001.

[BK98] A.J. Bonner and M. Kifer. A logic for programming database transactions. In J. Chomicki and G. Saake, editors, *Logics for Databases and Information Systems*, chapter 5, pages 117–166. Kluwer Academic Publishers, March 1998.

[BMNPS02] F. Baader, D. L. McGuinness, D. Nardi, and P. Patel-Schneider, editors. *Description Logic Handbook: Theory, implementation and applications*. Cambridge University Press, 2002.

[BN03] F. Baader and W. Nutt. Basic description logics. In Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors, *The Description Logic Handbook: Theory, Implementation, and Applications*, pages 43–95. Cambridge University Press, 2003.

[BPM-98] V. R. Benjamins, E. Plaza, E. Motta, D. Fensel, R. Studer, B. Wielinga, G. Schreiber, Z. Zdrahal, and S. Decker. IBROW3: An intelligent brokering service for knowledge-component reuse on the world-wide web. In *Proceedings of the 11th Banff Knowledge Acquisition for Knowledge-Based System Workshop (KAW98)*, Banff, Canada, 1998.

[BG03] Dan Brickley and R. V. Guha. RDF Vocabulary Description Language 1.0: RDF Schema. W3C Working Draft, 23 January 2003. Available at <http://www.w3.org/TR/2003/WD-rdf-schema-20030123/>.

[BR02] Walter Binder and Volker Roth. Secure mobile agent systems using Java: Where are we heading? In *Seventeenth ACM Symposium on Applied Computing (SAC-2002)*, Madrid, Spain, March 2002.

[BS00] F. Baader and U. Sattler. An overview of tableau algorithms for description logics. *Studia Logica*, 2001. To appear. An abridged version appeared in *Tableaux 2000*, volume 1847 of LNAI, 2000. Springer-Verlag.

[BS01] F. Baader and U. Sattler. An overview of tableau algorithms for description logics. *Studia Logica*, 69:5–40, 2001.

[BvHH-04] Sean Bechhofer, Frank van Harmelen, James Hendler, Ian Horrocks, Deborah L. McGuinness, Peter F. Patel-Schneider, and Lynn Andrea Stein eds. OWL Web Ontology Language Reference. URL <http://www.w3.org/TR/owl-ref/>, Feb 2004.

[CBF04a] Ion Constantinescu, Walter Binder, and Boi Faltings. An Extensible Directory Enabling Efficient Semantic Web Service Integration. In *3rd International Semantic Web Conference (ISWC04)*, Hiroshima, Japan, November 2004.



- [CBF04b] Ion Constantinescu, Walter Binder, and Boi Faltings. Directory services for incremental service integration. In *First European Semantic Web Symposium (ESWS-2004)*, Heraklion, Greece, May 2004.
- [CC00] Y. Chen and B.H.C. Cheng. *A Semantic Foundation for Specification Matching*. Cambridge University Press, Eds. M. Sitaraman and G. Leavens, eds. m. edition, 2000.
- [CCMW01] E. Christensen, F. Curbera, G. Meredith, and S. Weerawarana. Web Services Description Language (WSDL) 1.1. <http://www.w3.org/TR/wsdl>, March 2001.
- [CF03] Ion Constantinescu and Boi Faltings. Efficient matchmaking and directory services. In *The 2003 IEEE/WIC International Conference on WebIntelligence*, 2003.
- [CFarFik+98] Vinay K. Chaudhri, Adam Farquhar, Richard Fikes, Peter D. Karp, and James Rice. OKBC: A programmatic foundation for knowledge base interoperability. In Proc. of the 15th Nat. Conf. on Artificial Intelligence (AAAI'98), pages 600–607, 1998.
- [CFB04a] Ion Constantinescu, Boi Faltings, and Walter Binder. Large scale testbed for type compatible service composition. In *ICAPS 04 workshop on planning and scheduling for Web and grid services*, 2004.
- [CFB04b] Ion Constantinescu, Boi Faltings, and Walter Binder. Large scale, typecompatible service composition. In *IEEE International Conference on Web Services (ICWS-2004)*, San Diego, CA, USA, July 2004.
- [CGL98] D. Calvanese, G. De Giacomo, and M. Lenzerini. On the decidability of query containment under constraints. In Proc. of the Seventeenth ACM SIGACT SIGMOD Sym. on Principles of Database Systems (PODS-98), pages 149-158, 1998
- [CKW93] W. Chen, M. Kifer, and D.S. Warren. Hilog: A foundation for higherorder logic programming. *Journal of Logic Programming*, 15(3):187–230, 1993.
- [Coa04] The OWL Services Coalition. OWL-S 1.1 beta release. Available at <http://www.daml.org/services/owl-s/1.1B/>, July 2004.
- [CPRT03] A. Cimatti, M. Pistore, M. Roveri, and P. Traverso. Weak, Strong, and Strong Cyclic Planning via Symbolic Model Checking. *Artificial Intelligence*, 147(1-2):35–84, 2003.
- [CvHarHor+01] Dan Connolly, Frank van Harmelen, Ian Horrocks, Deborah L. McGuinness, Peter F. Patel-Schneider, and Lynn Andrea Stein. DAML+OIL (March2001) reference description. W3C Note, 18 December 2001. Available at <http://www.w3.org/TR/2001/NOTE-daml+oil-reference-20011218>.

[Darr98] Tim Darr, Mark Fox, and Deborah L. McGuinness, editors. Special Configuration Issue of the Artificial Intelligence for Engineering Design, Analysis, and Manufacturing Journal 1998.

[De04] Jos de Bruijn (*editor*). The WSML Family of Representation Languages. Working draft, Digital Enterprise Research Institute (DERI), March 2004. Look at <http://www.wsmo.org/2004/d16/d16.1/>.

[Der98] D. Mc Dermott. The Planning Domain Definition Language Manual. Technical Report 1165, Yale Computer Science University, 1998. CVC Report 98-003.

[DFvF02] J. Davies, D. Fensel and F. van Harmelen, eds. *Towards the Semantic Web: Ontology-Driven Knowledge Management*. New York: Wiley, 2002

[DKO+02] Ying Ding, M. Korotkiy, Boris Omelayenko, V. Kartseva, V. Zykov, Michael Klein, Ellen Schulten, and Dieter Fensel. Goldenbullet: Automated classification of product data in e-commerce. In *Proceedings of Business Information Systems Conference (BIS 2002)*, Poznan, Poland, 2002.

[DL96] K.K. Dhara and G.T. Leavens. Forcing behavioral subtyping through specification inheritance. In *Proceedings of the 18th International Conference on Software Engineering (ICSE'18)*, March 1996.

[DLNS98] Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, and Andrea Schaerf. Al-log: integrating datalog and description logics. *Journal of Intelligent Information Systems*, 10:227–252, 1998.

[DLPT02] U. Dal Lago, M. Pistore, and P. Traverso. Planning with a Language for Extended Goals. In *Proc. AAI'02*, 2002.

[dMRME04] Jos de Bruijn, Francisco Martin-Recuerda, Dimitar Manov, and MarcEhrig. State-of-the-art survey on ontology merging and aligning v1. Technical report, SEKT project IST-2003-506826, 2004.

[Don03] Francesco M. Donini. Complexity of Reasoning. In Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors, *The Description Logic Handbook: Theory, Implementation, and Applications*, pages 96–136. Cambridge University Press, 2003.

[DCvH+03] Mike Dean, Dan Connolly, Frank van Harmelen, James Hendler, Ian Horrocks, Deborah L. McGuinness, Peter F. Patel-Schneider, and Lynn Andrea Stein.

OWL web ontology language reference. W3C Working Draft, 31 March 2003. Available at <http://www.w3.org/TR/2003/WD-owl-ref-20030331>

[DS04] Mike Dean and Guus Schreiber, editors. *OWL Web Ontology Language Reference*. 2004. W3C Recommendation 10 February 2004.

[Dum00] E. Dumbill. The Semantic Web: A Primer. November 1, 2000. <<http://www.xml.com/pub/a/2000/11/01/semanticweb/>>.

[EGW94] Oren Etzioni, Keith Golden, and Daniel Weld. Tractable closed world reasoning with updates. In *KR'94: Principles of Knowledge Representation and Reasoning*, 1994.

[Eme90] E. A. Emerson. Temporal and modal logic. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science, Volume B: Formal Models and Semantics*. Elsevier, 1990.

[FB02] D. Fensel and C. Bussler. The Web Service Modeling Framework WSMF. *Electronic Commerce Research and Applications*, 1(2), 2002. [Fel98] Christiane Fellbaum, editor. *WordNet: An Electronic Lexical Database*. MIT Press, May 1998.

[Fen03] D. Fensel. *Ontologies: Silver Bullet for Knowledge Management and Electronic Commerce, 2nd edition*. Springer-Verlag, Berlin, 2003.

[FMB+03] D. Fensel, E. Motta, V.R. Benjamins, S. Decker, M. Gaspari, R. Groenboom, W. Grosso, M. Musen, E. Plaza, G. Schreiber, R. Studer, and B. Wielinga. The Unified Problem-solving Method development Language UPML. *Knowledge and Information Systems (KAIS): An international journal*, 5(1), 2003.

[FN71] Richard Fikes and Nils J. Nilsson. Strips: A new approach to the application of theorem proving to problem solving. In *IJCAI*, pages 608–620, 1971.

[FPV98] Alfonso Fuggetta, Gian Pietro Picco, and Giovanni Vigna. Understanding Code Mobility. *IEEE Transactions on Software Engineering*, 24(5):342–361, May 1998.

[GCTB01] J. Gonzalez-Castillo, D. Trastour, and C. Bartolini. Description logics for matchmaking of services. In *KI-2001 Workshop on Applications of Description Logics*, September 2001.

[GMP04] Stephan Grimm, Boris Motik, and Chris Preist. Variance in e-business service discovery. *Semantic Web Services: Preparing to Meet the World of Business Applications, Workshop at ISWC 2004*, November 2004.

[Gru93a] T. R. Gruber. Towards Principles for the Design of Ontologies Used for Knowledge Sharing. In N. Guarino and R. Poli, editors, *Formal Ontology in*

*Conceptual Analysis and Knowledge Representation*, Deventer, The Netherlands, 1993. Kluwer Academic Publishers.

[Gru93b] T. R. Gruber. A translation approach to portable ontology specification. *Knowledge Acquisition*, 5(2):199–220, 1993.

[Gua95] N. Guarino. Formal ontology, conceptual analysis and knowledge representation. *Int. Journal of Human-Computer Studies*, 43(5/6):625-640, 1995

[Gua98] Nicola Guarino, "Formal Ontology and Information Systems". In the Proceedings of Formal Ontology in Information Systems, June 1998. Also in *Frontiers in Artificial Intelligence and Applications*, IOS-Press, Washington, DC, 1998.

[HBCS03] R. Hull, M. Benedikt, V. Christophides, and J. Su. E-Services: A Look Behind the Curtain. In *Proc. PODS'03*, 2003.

[Hef03] Jeff Heflin. Web ontology language (owl) use cases and requirements. W3C Working Draft, 31 March 2003. Available at <http://www.w3.org/TR/2003/WD-webontreq-20030331/>.

[Hen01] J. Hendler. Agents and the Semantic Web. *IEEE Intelligent Systems* 16 (March-April 2001): 30-37.

[HFB+00] I. Horrocks, D. Fensel, J. Broekstra, S. Decker, M. Erdmann, C. Goble, F. van Harmelen, M. Klein, S. Staab, R. Studer, and E. Motta. OIL: The Ontology Inference Layer. Technical Report IR-479, Vrije Universiteit Amsterdam, Faculty of Sciences, September 2000

[HHO04] H. He, H. Haas, and D. Orchard. Web services architecture usage scenarios. W3C working group note, W3C, February 2004.

[HM01] Volker Haarslev and Ralf Møller. RACER System Description. volume 2083, 2001.

[HMA02] J. Heflin and H. Munoz-Avila. LCW-based agent planning for the semantic web. In AAAI Press, editor, *Ontologies and the Semantic Web. Papers from the 2002 AAAI Workshop WS-02-11*, pages 63–70, 2002.

[HMcG00] James Hendler and Deborah McGuinness. "The DARPA Agent Markup Language". In *IEEE Intelligent Systems Trends and Controversies*, November/December 2000. Available from <http://www.ksl.stanford.edu/people/dlm/papers/ieee-daml01-abstract.html>.

- [HNP95] Joseph M. Hellerstein, Jeffrey F. Naughton, and Avi Pfeffer. Generalized search trees for database systems. In Umeshwar Dayal, Peter M. D. Gray, and Shojiro Nishio, editors, *Proc. 21st Int. Conf. Very Large Data Bases, VLDB*, pages 562–573. Morgan Kaufmann, 11–15 1995.
- [Hoa69] C.A.R. Hoare. An axiomatic basis for computer programming. *Communications of the ACM*, 12(10), October 1969.
- [Hor97] I. Horrocks. *Optimising Tableaux Decision Procedures for Description Logics*. PhD thesis, The University of Manchester, 1997. URL <http://www.cs.man.ac.uk/horrocks/Publications/download/1997/phd-2sss.ps.gz>.
- [Hor98] I. Horrocks. Using an Expressive Description Logic: FaCT or Fiction? pages 636–647, 1998.
- [Hor03] I. Horrocks. Implementation and Optimisation Techniques. In Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors, *The Description Logic Handbook: Theory, Implementation, and Applications*, pages 306–346. Cambridge University Press, 2003.
- [HPS98] I. Horrocks and P. F. Patel-Schneider. Comparing subsumption optimizations. 1998.
- [HPSB+04] I. Horrocks, P.F. Patel-Schneider, H. Boley, S. Tabet, B. Grosz, and M. Dean. SWRL: A semantic web rule language combining OWL and RuleML. Available at <http://www.w3.org/Submission/2004/SUBMSWRL-20040521/>, May 2004.
- [HS01] Ian Horrocks and Ulrike Sattler. Ontology reasoning in the *SHOQ(D)* description logic. pages 199–204, 2001.
- [HS02] Ian Horrocks and Ulrike Sattler. Optimised Reasoning for *SHIQ*. In *Proc. of the 15th Eur. Conf. on Artificial Intelligence (ECAI 2002)*, pages 277–281, July 2002.
- [HST99] I. Horrocks, U. Sattler, and S. Tobies. Practical reasoning for expressive description logics. In H. Ganzinger, D. McAllester, and A. Voronkov, editors, *Proc. of the 6th Int. Conf. on Logic for Programming and Automated Reasoning (LPAR'99)*, number 1705 in *Lecture Notes In Artificial Intelligence*, pages 161–180. Springer-Verlag, 1999.
- [JC92] J.J. Jeng and B.H.C. Cheng. Using automated reasoning techniques to determine software reuse. *International Journal of Software Engineering and Knowledge Engineering*, 2(4), December 1992.

- [JC93] J.J Jeng and B.H.C. Cheng. *Using Formal Methods to Construct a Software Component Library*, volume 717 of *Lecture Notes in Computer Science*, pages 397–417. Springer Verlag, September 1993.
- [JC94] J.J Jeng and B.H.C. Cheng. A formal approach to reusing more general components. In *IEEE 9th Knowledge-Based Software Engineering*, September 1994.
- [JC95] J.J Jeng and B.H.C. Cheng. Specification matching for software reuse: A foundation. In *SSR'95. ACM SIGSOFT*. ACM Press, April 1995.
- [KC03] Graham Klyne and Jeremy J. Carroll. Resource Description Framework (RDF): Concepts and abstract syntax. W3C Working Draft, 2003. Available at <http://www.w3.org/TR/2003/WD-rdf-concepts-20030123>
- [KHW95] Nicholas Kushmerick, Steve Hanks, and Daniel S. Weld. An algorithm for probabilistic planning. *Artificial Intelligence*, 76(1–2):239–286, 1995.
- [kif98] KIF. Knowledge Interchange Format: Draft proposed american national standard (dpan). Technical Report 2/98-004, available at <http://www.daml.org/services/owl-s/1.1B/DRSguide.pdf>, 1998.
- [KLeDitors04] Uwe Keller, Rubén Lara, and Axel Polleres (editors). WSMO web service discovery. Technical report, DERI, November 2004.
- [KLP+04] Michael Kifer, Rubén Lara, Axel Polleres, Chang Zhao, Uwe Keller, Holger Lausen, and Dieter Fensel. A logical framework for web service discovery. In *"Semantic Web Services: Preparing to Meet the World of Business Applications" workshop at ISWC 2004*, 2004.
- [KMH97] Marcel Kornacker, C. Mohan, and Joseph M. Hellerstein. Concurrency and recovery in generalized search trees. In Joan M. Peckman, editor, *Proceedings, ACM SIGMOD International Conference on Management of Data: SIGMOD 1997: May 13–15, 1997, Tucson, Arizona, USA*, 1997.
- [KSF04] Uwe Keller, Michael Stollberg, and Dieter Fensel. WOOGLÉ meets Semantic Web Fred. In *Proceedings of the Workshop on WSMO Implementations (WIW 2004)*, CEUR-WS.org/Vol-113/, September 2004.
- [Las98] Ora Lassila, "Web Metadata: A Matter of Semantics", *IEEE Internet Computing* 2(4): 30-37 (1998).

- [Leditors04a] Rub'en Lara and Axel Polleres (*editors*). Formal mapping and tool to owls. Technical report, DERI, December 2004.
- [Leditors04b] Rub'en Lara and Holger Lausen (*editors*). WSMO discovery engine. Technical report, DERI, November 2004.
- [LH03] L. Li and I. Horrocks. A software framework for matchmaking based on semantic web technology. In *Proceedings of the 12th International Conference on the World Wide Web*, Budapest, Hungary, May 2003.
- [Lif87] V. Lifschitz. On the semantics of strips. In M. P. Georgeff and A. L. Lansky, editors, *Reasoning about Actions and Plans*, pages 1–9. Kaufmann, Los Altos, CA, 1987.
- [LPL+] Rub'en Lara, Axel Polleres, Holger Lausen, Dumitru Roman, Jos de Bruijn, and Dieter Fensel. A comparison of WSMO and OWL-S. *World Wide Web Journal, Special issue on Web Services: Theory and Practice*. Submitted.
- [Lut02] Carsten Lutz. The Complexity of Reasoning with Concrete Domains. PhD thesis, LuFG Theoretical Computer Science, RWTH Aachen, Germany, 2002.
- [LW94] B.H. Liskov and J.M. Wing. A behavioral notion of subtyping. *ACM Transactions on Programming Languages*, 16(10), November 1994.
- [LY99] Tim Lindholm and Frank Yellin. *The Java Virtual Machine Specification*. Addison-Wesley, Reading, MA, USA, second edition, 1999.
- [McD98] Drew McDermott. The planning domain definition language manual. Technical Report 1165, Yale Computer Science, 1998.
- [McD03] Drew V. McDermott. Pddl2.1 - the art of the possible? commentary on fox and long. *J. Artif. Intell. Res. (JAIR)*, 20:145–148, 2003.
- [McD04] D. McDermott. DRS: A set of conventions for representing logical languages in RDF. Available at <http://www.daml.org/services/owls/1.1B/DRSguide.pdf>, January 2004.
- [MDCG03] E. Motta, J. Domingue, L. Cabral, and M. Gaspari. IRS-II: A framework and infrastructure for semantic web services. In *2nd International Semantic Web Conference (ISWC2003)*. Springer Verlag, October 2003.
- [MF02] S. McIlraith and R. Fadel. Planning with Complex Actions. In *Proc. NMR'02*, 2002.
- [Min81] Marvin Minsky. A Framework for Representing Knowledge. In J. Haugeland, editor, *Mind Design*. The MIT Press, 1981.

- [MS02a] S. McIlraith and S. Son. Adapting Golog for composition of semantic Web Services. In *Proc. KR'02*, 2002.
- [MS02b] Sheila A. McIlraith and Tran Cao Son. Adapting golog for composition of semantic web services. In Dieter Fensel, Fausto Giunchiglia, Deborah McGuinness, and Mary-Anne Williams, editors, *Proceedings of the 8<sup>th</sup> International Conference on Principles and Knowledge Representation and Reasoning (KR-02)*, pages 482–496, San Francisco, CA, April 22–25 2002. Morgan Kaufmann Publishers.
- [MSZ01] S. McIlraith, T.C. Son, and H. Zeng. Mobilizing the semantic web with daml-enabled web services. In *Proc. Second International Workshop on the Semantic Web (SemWeb-2001)*, Hongkong, 2001.
- [NM02] S. Narayanan and S. McIlraith. Simulation, Verification and Automated Composition of Web Services. In *Proc. WWW2002*, 2002.
- [OIPL04] Daniel Olmedilla, Rubén Lara, Axel Polleres, and Holger Lausen. Trust negotiation for semantic web services. In *First International Workshop on Semantic Web Services and Web Process Composition (SWSWPC 2004) at ICWS 2004*, July 2004.
- [PA97] J. Penix and P. Alexander. Towards automated component adaptation. In *Proceedings of the 9th International Conference on Software Engineering and Knowledge Engineering*, June 1997.
- [Pan04a] Jeff Z. Pan. *Description Logics: Reasoning Support for the Semantic Web*. PhD thesis, School of Computer Science, The University of Manchester, Oxford Rd, Manchester M13 9PL, UK, Sept 2004.
- [Pan04b] Jeff Z. Pan. Reasoning Support for OWL-E (Extended Abstract). In *Proc. of Doctoral Programme in the 2004 International Joint Conference of Automated Reasoning (IJCAR2004)*, July 2004.
- [PBB+04] M. Pistore, P. Bertoli, F. Barbon, D. Shaparau, and P. Traverso. Planning and Monitoring Web Service Composition. In *Proc. AIMS'04*, 2004.
- [Ped89] Edwin P. D. Pednault. Adl: Exploring the middle ground between strips and the situation calculus. In *Proceedings of the First International Conference on Principles of Knowledge Representation and Reasoning (KR'89)*, pages 324–332, Morgan Kaufmann Publishers, 1989.
- [PH04] Jeff Z. Pan and Ian Horrocks. OWL-E: Extending OWL with Expressive Datatype Expressions. Technical report, School of Computer Science, the University of Manchester, April 2004.



- [PKPS02] M. Paolucci, T. Kawamura, T. Payne, and K. Sycara. Semantic matching of web services capabilities. In I. Horrocks and J. Handler, editors, *1<sup>st</sup> Int. Semantic Web Conference (ISWC)*, pages 333–347. Springer Verlag, 2002.
- [PR89] A. Pnueli and R. Rosner. On the synthesis of an asynchronous reactive module. In *Proc. ICALP'89*, 1989.
- [Pre04] Chris Preist. A conceptual architecture for semantic web services. In *Proceedings of the International Semantic Web Conference 2004 (ISWC 2004)*, November 2004.
- [PS99] Peter F. Patel-Schneider. DLP. In *Description Logics*, 1999.
- [PSHH04] Peter F. Patel-Schneider, Patrick Hayes, and Ian Horrocks. OWLWeb Ontology Language Semantics and Abstract Syntax. Technical report, W3C, Feb. 2004. W3C Recommendation, URL <http://www.w3.org/TR/2004/REC-owl-semantics-20040210/>.
- [PTB04] M. Pistore, P. Traverso, and P. Bertoli. Automated composition of web services by planning in asynchronous domains, 2004. Submitted for publication.
- [Qui67] M. Ross Quillian. Word Concepts: A Theory and Simulation of Some Basic Capabilities. *Behavioral Science*, 12:410–430, 1967.
- [RLeditors04] D. Roman, H. Lausen, and U. Keller (*editors*). Web service modeling ontology - standard (WSMO-Standard). Working draft, Digital Enterprise Research Institute (DERI), September 2004. <http://www.wsmo.org/2004/d2/v1.0/>.
- [Ros94] R. Rosenfield. *Adaptive statistic language model*. PhD thesis, Carnegie Mellon University, 1994.
- [RS95] Ray Richardson and Alan F. Smeaton. Using WordNet in a knowledge based approach to information retrieval. Technical Report CA-0395, Dublin, Ireland, 1995.
- [RW91] E.J. Rollings and J.M. Wing. Specifications as search keys for software libraries. In *Proceedings of the Eighth International Conference on Logic Programming*, June 1991.
- [Sat00] U. Sattler. Description logics for the representation of aggregated objects. In W.Horn, editor, *Proceedings of the 14th European Conference on Artificial Intelligence* IOS Press, Amsterdam, 2000.
- [SdF03] M. Sheshagiri, M. desJardins, and T. Finin. A Planner for Composing Services Described in DAML-S. In *Proc. AAMAS'03*, 2003. [*Seditors*] M. Stollberg and R. Lara (*editors*). D3.3 v0.1 wsmo use case. Technicalreport.

[Sch91] K. Schild. A correspondence theory for terminological logics: Preliminary report. In Proc. of the 12th Int. Joint Conf. on Artificial Intelligence (IJCAI-91), pages 466–471, Sydney, 1991.

[SF97] J. Schumann and B. Ficher. Nora/hammr: Making deduction-based software component retrieval practical. In *Proceedings of the 12th IEEE International Automated Software Engineering Conference (ASE97)*, November 1997.

[Smi98] Barry Smith, “Basic Concepts of Formal Ontologies”, in N. Guarino (Ed.) *Formal Ontology in Information Systems*, IOS Press, 1998.

[Sun] Sun Microsystems, Inc. Java HotSpot Technology. Web pages at <http://java.sun.com/products/hotspot/>.

[SVSM03] Kaarthik Sivashanmugam, Kunal Verma, Amit Sheth, and John Miller. Adding semantics to web services standards. In *Proceedings of the International Conference on Web Services (ICWS'03)*, June 2003.

[SWKL02] K. Sycara, S. Widoff, M. Klusch, and J. Lu. LARKS: Dynamic matchmaking among heterogeneous software agents in cyberspace. *Autonomous Agents and Multi-Agent Systems*, pages 173–203, 2002.

[Tee94] G. Teege. Making the difference: A substraction operation for description logics. In *KR'94*, 1994.

[TKAS02] S. Thakkar, Craig A. Knoblock, Jose Luis Ambite, and Cyrus Shahabi. Dynamically composing web services from on-line sources. In *Proceeding of the AAAI-2002 Workshop on Intelligent Service Integration*, pages 1–7, Edmonton, Alberta, Canada, July 2002.

[Tob01] S. Tobies. Complexity Results and Practical Algorithms for Logics in Knowledge Representation. PhD thesis, RWTH Aachen, 2001. electronically available at <http://www.bth.rwth-aachen.de/ediss/ediss.html>.

[TP04] P. Traverso and M. Pistore. Automated Composition of Semantic Web Services into Executable Processes. In *Proc. ISWC'04*, 2004.

[UG96] M. Uschold and M. Gruninger. *Ontologies: Principles, Methods and Applications*. *The Knowledge Engineering Review*, 1996.

[Var95] M. Y. Vardi. An automata-theoretic approach to fair realizability and synthesis. In *Proc. CAV'95*. 1995.

[VBB98] Jan Vitek, Ciarán Bryce, and Walter Binder. Designing JavaSeal or how to make Java safe for agents. Technical report, University of Geneva, July 1998.

- [Vol04] Raphael Volz. *Web Ontology Reasoning with Logic Databases*. PhD thesis, AIFB, Karlsruhe, 2004.
- [VSSP04] K. Verma, K. Sivashanmugam, A. Sheth, and A. Patil. METEOR-S WSDI: A scalable P2P infrastructure of registries for semantic publication and discovery of web services. *Journal of Information Technology and Management*, 2004.
- [W3C] W3C. XML Schema Part 2: Datatypes, <http://www.w3.org/tr/xmlschema-2/>.
- [W3C03] W3C. SOAP version 1.2 part 0: Primer. <http://www.w3.org/TR/2003/REC-soap12-part0-20030624/>, June 2003.
- [Wie92] G. Wiederhold. Mediators in the architecture of future information systems. *IEEE Computer*, 25(3):38–49, March 1992.
- [WPS+03] D. Wu, B. Parsia, E. Sirin, J. Hendler, and D. Nau. Automating DAML-S Web Services Composition using SHOP2. In *Proc. ISWC'03*, 2003.
- [Wu,03] Wu, Dan and Parsia, Bijan and Sirin, Evren and Hendler, James and Nau, Dana. Automating DAML-S Web Services Composition Using SHOP2. In *Proceedings of 2nd International Semantic Web Conference (ISWC2003)*, 2003.
- [ZK04] O.K. Zein and Y. Kermarrec. An approach for describing/discovering services and for adapting them to the needs of users in distributed systems. In *Semantic Web Services. Papers from 2004 AAAI Spring Symposium*, March 2004.
- [ZW95] A.M. Zaremski and J.M. Wing. Specification matching of software components. In *3rd ACM SIGSOFT Symposium on the Foundations of Software Engineering*, October 1995.
- [ZW97] A.M. Zaremski and J.M. Wing. Specification matching of software components. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 6:333–369, 1997.