



**ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ**

**ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ**

**ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ**

**ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ**

**ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ**

**Ανάπτυξη μιας γλώσσας και του απαραίτητου  
λογισμικού εκτέλεσής της, για την αναζήτηση  
ετερογενών υπηρεσιών με ενιαίο τρόπο**

**Μιχαήλ Κ. Πανταζόγλου**

**ΑΘΗΝΑ**

**ΝΟΕΜΒΡΙΟΣ 2009**



## **ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ**

Ανάπτυξη μιας γλώσσας και του απαραίτητου λογισμικού εκτέλεσής της, για την αναζήτηση ετερογενών υπηρεσιών με ενιαίο τρόπο

**Μιχαήλ Κ. Πανταζόγλου**

### **ΕΠΙΒΛΕΠΟΥΣΑ ΚΑΘΗΓΗΤΡΙΑ:**

**Αφροδίτη Τσαλαγατίδου**, Επίκουρη Καθηγήτρια ΕΚΠΑ

### **ΤΡΙΜΕΛΗΣ ΕΠΙΤΡΟΠΗ ΠΑΡΑΚΟΛΟΥΘΗΣΗΣ:**

**Αφροδίτη Τσαλαγατίδου**, Επίκουρη Καθηγήτρια ΕΚΠΑ

**Ιωάννης Κοτρώνης**, Επίκουρος Καθηγητής ΕΚΠΑ

**Αλέξης Δελής**, Καθηγητής ΕΚΠΑ

### **ΕΠΤΑΜΕΛΗΣ ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ**

**Αφροδίτη Τσαλαγατίδου**  
Επίκουρη Καθηγήτρια ΕΚΠΑ

**Ιωάννης Κοτρώνης**  
Επίκουρος Καθηγητής ΕΚΠΑ

**Αλέξης Δελής**  
Καθηγητής ΕΚΠΑ

**Ιωάννης Ιωαννίδης**  
Καθηγητής ΕΚΠΑ

**Δημήτριος Γουνόπουλος**  
Αναπληρωτής Καθηγητής ΕΚΠΑ

**Μανόλης Κουμπάρκης**  
Αναπληρωτής Καθηγητής ΕΚΠΑ

**Ευστάθιος Χατζηευθυμιάδης**  
Επίκουρος Καθηγητής ΕΚΠΑ

Ημερομηνία εξέτασης 5/11/2009



## ΠΕΡΙΛΗΨΗ

Η Υπηρεσιοστρεφής Υπολογιστική προσφέρει ένα υψηλού επιπέδου αφαίρεσης μοντέλο, για την ανάπτυξη κατανεμημένων εφαρμογών λογισμικού. Σύμφωνα με τις αρχές του μοντέλου αυτού, η σχεδίαση και υλοποίηση μιας εφαρμογής πραγματοποιείται μέσω της αναζήτησης, σύνθεσης και επαναχρησιμοποίησης υπηρεσιών, οι οποίες προσφέρονται στο διαδίκτυο. Η ευρεία αποδοχή του υπηρεσιοστρεφούς μοντέλου τα τελευταία χρόνια και η εμφάνιση πολλαπλών υπηρεσιοστρεφών τεχνολογιών είχαν ως αποτέλεσμα την ανάπτυξη και διαθεσιμότητα προς αναζήτηση και χρήση διαφόρων τύπων υπηρεσιών επιπέδου εφαρμογής, όπως υπηρεσίες ιστού, υπηρεσίες ομότιμου-προς-ομότιμο και υπηρεσίες πλέγματος. Ωστόσο, οι υπηρεσίες αυτές χαρακτηρίζονται από πολυδιάστατη ετερογένεια, που διέπει τα πρωτόκολλα και τις υποδομές τους, και ως εκ τούτου δεν είναι μεταξύ τους διαλειτουργικές. Παρόλα αυτά, σε πολλά πεδία εφαρμογών, οι απαιτήσεις καθιστούν επιθυμητή, ή ακόμα και απαραίτητη, τη συνδυαστική χρήση διαφόρων τύπων υπηρεσιών, ανεξάρτητα από τις σχετικές με αυτές τεχνολογίες. Το γεγονός αυτό καθιστά επιτακτική την ανάγκη για μια νέα γενιά εργαλείων ανάπτυξης υπηρεσιοστρεφών εφαρμογών, ικανών να υποστηρίξουν ετερογενείς υπηρεσίες.

Η παρούσα διατριβή προτείνει μια νέα προσέγγιση στην αντιμετώπιση της προαναφερθείσας πρόκλησης διαλειτουργικότητας στο επίπεδο της αναζήτησης υπηρεσιών. Συγκεκριμένα, συνεισφέρει ένα πλαίσιο *ενιαίας* αναζήτησης ετερογενών υπηρεσιών ιστού, υπηρεσιών ομότιμου-προς-ομότιμο και υπηρεσιών πλέγματος, αποτελούμενο από (α) μια γλώσσα ενιαίων επερωτήσεων για υπηρεσίες, την USQL, και (β) μια ανοικτή μηχανή αναζήτησης υπηρεσιών, με την ονομασία Proteus. Συνολικά, το προτεινόμενο πλαίσιο χαρακτηρίζεται από ευελιξία, μέσω κατάλληλων μηχανισμών επεκτάσεων. Η γλώσσα USQL υποστηρίζει τη σύνταξη περιεκτικών επερωτήσεων, στις οποίες καταγράφονται οι λειτουργικές και μη λειτουργικές απαιτήσεις των χρηστών, με τρόπο ανεξάρτητο από τις διάφορες υπηρεσιοστρεφείς τεχνολογίες. Εξάλλου, υλοποιώντας και συνδυάζοντας διαδεδομένους αλγορίθμους αντιστοίχισης, η μηχανή Proteus παράγει ακριβή αποτελέσματα, ως προς τα κριτήρια αναζήτησης του χρήστη. Επιπλέον, χάρη στην αρχιτεκτονική και τις δυνατότητες κλιμάκωσης που διαθέτει, η μηχανή Proteus αποτελεί μια αποδοτική λύση στην αναζήτηση ετερογενών υπηρεσιών, γεγονός που επιβεβαιώνεται από τα αποτελέσματα πειραματικών μετρήσεων.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Κατανεμημένες Εφαρμογές

ΛΕΞΕΙΣ-ΚΛΕΙΔΙΑ: Αναζήτηση υπηρεσιών, διαλειτουργικότητα, ετερογενείς υπηρεσίες



# ABSTRACT

Service-Oriented Computing offers a high-level model for the development of distributed software applications. According to the principles of that model, the design and implementation of an application is realized through discovery, composition and reuse of services, which are available on the Internet. The wide acceptance of the service-oriented model within the last years, along with the emergence of multiple service-oriented technologies, yielded the development and availability for discovery and reuse of various types of application-level services, such as web services, peer-to-peer services, and grid services. Still, these services are characterized by multi-dimensional heterogeneity, in terms of their underlying protocols and infrastructure; thus, they cannot easily interoperate. Despite this, in multiple application domains, requirements suggest, or even impose, the need for a combined use of various types of services, irrespectively of their related technologies. This fact brings forth a demand for a new generation of service-oriented development tools, capable of supporting heterogeneous services.

This dissertation proposes a new approach to address the aforementioned interoperability challenge at the level of service discovery. Specifically, it contributes a framework supporting the *unified* discovery of heterogeneous web services, peer-to-peer services, and grid services, which comprises (a) a unified service query language named USQL; and (b) an open service discovery engine called Proteus. Overall, the proposed framework is characterized by flexibility, which is ensured by appropriate extension mechanisms. The USQL language supports the formulation of comprehensive queries, which capture both functional and non-functional user requirements in a service technology-agnostic manner. Besides, by implementing and combining renown matching algorithms, the Proteus engine produces accurate results that correspond to the user-specified search criteria. Further, thanks to its architecture and scalability potential, the Proteus engine constitutes a well-performing solution in the discovery of heterogeneous services. Viability and performance effectiveness of the proposed approach are asserted by experimental measurements.

SUBJECT AREA: Distributed Applications

KEYWORDS: Service discovery, interoperability, heterogeneous services





# ΕΥΧΑΡΙΣΤΙΕΣ

Πρωτίστως, θα ήθελα να ευχαριστήσω την επιβλέπουσα της παρούσης εργασίας, καθ. Αφροδίτη Τσαλαγατίδου, για τη συνολική της υποστήριξη στην προσπάθειά μου. Θεωρώ τον εαυτό μου προνομιούχο, που είχα την ευκαιρία να συνεργαστώ μαζί της κατά την εκπόνηση της διδακτορικής μου διατριβής. Οι προερχόμενες από την ακαδημαϊκή και ερευνητική της εμπειρία παρατηρήσεις και η γενικότερη καθοδήγηση που μου προσέφερε, μέσα από εποικοδομητικές συζητήσεις επιστημονικού και μή περιεχομένου, υπήρξαν καθοριστικές για την επιτυχή ολοκλήρωση της εργασίας μου. Επιπλέον, όλα αυτά τα χρόνια της συνεργασίας μας, μου έδωσε απλόχερα τη δυνατότητα να γίνω μέτοχος και κοινωνός ενός ευρέως φάσματος ακαδημαϊκών και ερευνητικών δραστηριοτήτων, καθιστώντας με ενεργό μέλος της διεθνούς ερευνητικής και επιστημονικής κοινότητας. Για όλα αυτά, την ευχαριστώ θερμά.

Θα ήθελα επίσης να ευχαριστήσω τα υπόλοιπα μέλη της τριμελούς επιτροπής παρακολούθησης της διδακτορικής μου διατριβής, καθ. Γιάννη Κοτρώνη και καθ. Αλέξη Δελή. Οι εύστοχες παρατηρήσεις και χρήσιμες συμβουλές τους αποτέλεσαν ταυτόχρονα κίνητρο και γνώμονα, στην προσπάθεια βελτίωσης και καλύτερης παρουσίασης των ερευνητικών αποτελεσμάτων της εργασίας μου.

Η εκπόνηση μιας διδακτορικής διατριβής, αν και ολοκληρώνεται τελικώς από ένα άτομο, αποτελεί στην ουσία προϊόν αλληλεπιδράσεων με τις ιδέες και εργασίες πολλών ανθρώπων. Σε αυτό το πλαίσιο, θα ήθελα να ευχαριστήσω όλους εκείνους τους ανθρώπους, με τους οποίους είχα την τύχη να έρθω σε επαφή και να συζητήσω για τα θέματα της εργασίας μου. Ειδικά, ευχαριστώ όλους τους συναδέλφους και συνοδοιπόρους μου στο εργαστήριο Ανάλυσης Συστημάτων και Ανάπτυξης Εφαρμογών, του Τμήματος Πληροφορικής και Τηλεπικοινωνιών. Τα αποτελέσματα της παρούσης εργασίας θα ήταν διαφορετικά, χωρίς την ουσιαστική τους βοήθεια και συμβολή, προερχόμενη από συζήτηση, ανταλλαγή απόψεων και εποικοδομητική κριτική.

Τέλος, θα ήθελα να ευχαριστήσω την οικογένειά μου, για την άνευ όρων υποστήριξη που μου παρείχε όλα αυτά τα χρόνια. Τίποτα δεν θα ήταν δυνατό χωρίς την αγάπη τους. Τους αφιερώνω το παρόν κείμενο.

Αθήνα, Νοέμβριος 2009



# ΠΕΡΙΕΧΟΜΕΝΑ

<b>ΠΡΟΛΟΓΟΣ</b>	<b>19</b>
<b>ΚΕΦΑΛΑΙΟ 1. ΕΙΣΑΓΩΓΗ</b>	<b>21</b>
1.1 Κίνητρο της Διατριβής . . . . .	22
1.1.1 Σενάριο 1: Εφαρμογή στον Τομέα της Υγείας . . . . .	24
1.1.2 Σενάριο 2: Εφαρμογή στον Τομέα της Διαχείρισης Κρίσεων . . . . .	25
1.2 Προκλήσεις και Συνεισφορά της Διατριβής . . . . .	26
1.3 Δομή της Διατριβής . . . . .	28
<b>ΚΕΦΑΛΑΙΟ 2. ΤΕΧΝΟΛΟΓΙΚΟ ΥΠΟΒΑΘΡΟ</b>	<b>31</b>
2.1 Το Υπηρεσιοστρεφές Μοντέλο Ανάπτυξης Εφαρμογών . . . . .	31
2.2 Ετερογενείς Τύποι Υπηρεσιών . . . . .	33
2.2.1 Υπηρεσίες Ιστού (Web Services) . . . . .	33
2.2.2 Υπηρεσίες Ομότιμου-προς-Ομότιμο (Peer-to-Peer Services) . . . . .	35
2.2.3 Υπηρεσίες Πλέγματος (Grid Services) . . . . .	37
2.2.4 Προς Ένα Γενικό Μοντέλο Υπηρεσιών . . . . .	39
2.3 Ετερογενείς Τεχνολογίες Περιγραφής Υπηρεσιών . . . . .	42
2.4 Ετερογενείς Τεχνολογίες Αναζήτησης Υπηρεσιών . . . . .	45
2.5 Γλώσσες Αναζήτησης Υπηρεσιών . . . . .	47
2.6 Μηχανές Αναζήτησης Υπηρεσιών . . . . .	50
2.7 Συζήτηση . . . . .	52
<b>ΚΕΦΑΛΑΙΟ 3. Η ΓΛΩΣΣΑ ΑΝΑΖΗΤΗΣΗΣ ΕΤΕΡΟΓΕΝΩΝ ΥΠΗΡΕΣΙΩΝ USQL</b>	<b>55</b>
3.1 Κίνητρο . . . . .	55
3.2 Επισκόπηση της USQL . . . . .	56
3.3 Μετα-Μοντέλο της USQL . . . . .	58
3.3.1 Στοιχεία Απαιτήσεων . . . . .	58
3.3.2 Το Έγγραφο Επερωτήσεων USQLRequest . . . . .	61
3.3.3 Το Έγγραφο Αποκρίσεων USQLResponse . . . . .	63
3.4 Συντακτικό της USQL . . . . .	65
3.4.1 Επερώτηση Αναζήτησης Υπηρεσίας Ιστού . . . . .	66
3.4.2 Επερώτηση Αναζήτησης Υπηρεσίας Ομότιμου-προς-Ομότιμο . . . . .	68
3.4.3 Επερώτηση Αναζήτησης Υπηρεσίας Πλέγματος . . . . .	69
3.5 Μαθηματικό Μοντέλο της USQL . . . . .	70
3.5.1 Βασικοί Ορισμοί . . . . .	72
3.5.2 Αντιστοίχιση Κριτηρίων Αναζήτησης . . . . .	74
3.6 Συμβατότητα και Επεκτασιμότητα . . . . .	78

3.7	Συζήτηση	80
<b>ΚΕΦΑΛΑΙΟ 4. Η ΜΗΧΑΝΗ ΑΝΑΖΗΤΗΣΗΣ ΕΤΕΡΟΓΕΝΩΝ ΥΠΗΡΕΣΙΩΝ Proteus</b>		<b>83</b>
4.1	Κίνητρο	83
4.2	Αρχιτεκτονική Συστήματος	85
4.3	Το Υποσύστημα Αναζήτησης	85
4.3.1	Χρονοπρογραμματισμός Επισκέψεων	86
4.3.2	Πραγματοποίηση Επισκέψεων	87
4.3.3	Ανάλυση Περιγραφών Υπηρεσιών	90
4.3.4	Αποθήκευση Διαφημίσεων Υπηρεσιών	92
4.4	Ο Τοπικός Κατάλογος Υπηρεσιών	92
4.5	Το Υποσύστημα Επερωτήσεων	95
4.5.1	Αρχικοποίηση και Επέκταση Επερωτήσεων USQL	96
4.5.2	Ανάκτηση Διαφημίσεων Υπηρεσιών	96
4.5.3	Μηχανισμοί Αντιστοίχισης	98
4.6	Μηχανισμός Επεκτάσεων	101
4.7	Μηχανισμός Κλιμάκωσης	105
4.8	Συζήτηση	107
<b>ΚΕΦΑΛΑΙΟ 5. ΑΞΙΟΛΟΓΗΣΗ</b>		<b>109</b>
5.1	Αξιολόγηση της Γλώσσας USQL	109
5.1.1	Μελέτη Διεπαφών Αναζήτησης Υπηρεσιών	110
5.1.2	Μελέτη Γλωσσών Περιγραφής Υπηρεσιών	112
5.1.3	Συζήτηση	116
5.2	Απόδοση των Μηχανισμών Αντιστοίχισης	118
5.2.1	Αποτελέσματα Μετρήσεων	120
5.2.2	Συζήτηση	122
5.3	Απόδοση της Μηχανής Αναζήτησης Proteus	123
5.3.1	Αποτελέσματα Μετρήσεων	125
5.3.2	Συζήτηση	128
<b>ΚΕΦΑΛΑΙΟ 6. ΣΥΜΠΕΡΑΣΜΑΤΑ</b>		<b>131</b>
6.1	Σύνοψη	131
6.2	Συνεισφορά της Διατριβής	131
6.3	Περιβάλλον Εφαρμογής	133
6.4	Κατευθύνσεις Μελλοντικής Έρευνας	134
6.4.1	Υποστήριξη Επιπρόσθετων Τύπων Υπηρεσιών	134
6.4.2	Αναζήτηση Υπηρεσιών Βάσει Πληροφορίας Περιβάλλοντος Πλαισίου	135
6.4.3	Αναζήτηση Σύνθετων Υπηρεσιών Βάσει Κριτηρίων Συμπεριφοράς	135

6.4.4	Δυναμική Σύνθεση Υπηρεσιών . . . . .	136
6.4.5	Υποστήριξη Συστατικών Σημασιολογικών Μεσολαβητών . . . . .	136
6.4.6	Αξιοποίηση Τεχνικών Κοινωνικής Δικτύωσης . . . . .	136
6.5	Επίλογος . . . . .	137
<b>ΠΑΡΑΡΤΗΜΑ. ΣΧΗΜΑ XML ΤΗΣ ΓΛΩΣΣΑΣ USQL</b>		<b>139</b>
<b>ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΩΝ</b>		<b>145</b>
<b>ΑΚΡΩΝΥΜΙΑ – ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ</b>		<b>147</b>



# ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

1	Το υπηρεσιοστρεφές μοντέλο . . . . .	32
2	Εννοιολογικό μοντέλο υπηρεσιών ιστού . . . . .	33
3	Εννοιολογικό μοντέλο υπηρεσιών ομότιμου-προς-ομότιμο . . . . .	35
4	Εννοιολογικό μοντέλο υπηρεσιών πλέγματος . . . . .	37
5	Αρχιτεκτονική του μοντέλου GeSMO . . . . .	40
6	Ενιαίο εννοιολογικό μοντέλο υπηρεσιών . . . . .	41
7	Διαδικασία ορισμού της USQL . . . . .	57
8	Ιεραρχία στοιχείων απαιτήσεων της USQL . . . . .	59
9	Δομή του εγγράφου USQLRequest . . . . .	62
10	Δομή του εγγράφου USQLResponse . . . . .	64
11	Γραφική απεικόνιση της συμπεριφοράς του Υπολογιστή Αντιστοιχίας . . . . .	73
12	Αρχιτεκτονική της μηχανής αναζήτησης Proteus . . . . .	84
13	Αυξομείωση της περιόδου επισκέψεων . . . . .	89
14	Δομή των εγγράφων διαφήμισης λειτουργιών υπηρεσιών . . . . .	91
15	Σχήμα δεδομένων του τοπικού καταλόγου υπηρεσιών . . . . .	93
16	Σχισιακό σχήμα δεδομένων του τοπικού καταλόγου υπηρεσιών . . . . .	95
17	Αρχιτεκτονική του συστατικού αντιστοίχισης . . . . .	99
18	Ιεραρχικές σχέσεις στο γράφο μιας οντολογίας . . . . .	101
19	Διεργασία του μηχανισμού επεκτάσεων . . . . .	103
20	Συνδυασμός πολλαπλών γλωσσών για την περιγραφή μιας υπηρεσίας ιστού. . . . .	117
21	Ενιαία επερώτηση USQL για μια υπηρεσία ιστού. . . . .	118
22	Απόδοση των αλγορίθμων αντιστοίχισης . . . . .	121
23	Μέσος χρόνος επεξεργασίας των αλγορίθμων αντιστοίχισης . . . . .	121
24	Συνολικός χρόνος επεξεργασίας του Αναζητητή . . . . .	124
25	Μέσος χρόνος επεξεργασίας του Αναζητητή . . . . .	126
26	Κλιμάκωση του Αναζητητή . . . . .	126
27	Απόδοση του Επεξεργαστή Επερωτήσεων . . . . .	127
28	Κλιμάκωση του Επεξεργαστή Επερωτήσεων . . . . .	128





## ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

1	Στοιχεία απαιτήσεων της USQL . . . . .	60
2	Διαλειτουργικότητα USQL - τεχνολογιών αναζήτησης υπηρεσιών . . . . .	79
3	Συμβατότητα USQL - τεχνολογιών περιγραφής υπηρεσιών . . . . .	79
4	Παράδειγμα αντίγραφων του υποσυστήματος αναζήτησης . . . . .	106
5	Αντιστοιχία περιγραφών WSDL και επερωτήσεων USQL . . . . .	113
6	Αντιστοιχία περιγραφών OWL-S και επερωτήσεων USQL . . . . .	113
7	Αντιστοιχία στοιχείων WS-QoS και επερωτήσεων USQL . . . . .	114
8	Αντιστοιχία στοιχείων WSLA και USQL . . . . .	115
9	Επερωτήσεις για τη μέτρηση της απόδοσης των αλγορίθμων αντιστοίχισης	120
10	Στοιχεία πειράματος μέτρησης της κλιμάκωσης του Proteus . . . . .	123
11	Ανάπτυξη της πειραματικής διάταξης . . . . .	124
12	Υλοποιημένα συνδεδεμένα συστατικά . . . . .	133



## ΠΡΟΛΟΓΟΣ

Η παρούσα διδακτορική διατριβή εκπονήθηκε από τον Ιούνιο του 2005 έως τον Οκτώβριο του 2009, στο Εργαστήριο Ανάλυσης Συστημάτων και Τεχνολογίας Λογισμικού του Τμήματος Πληροφορικής και Τηλεπικοινωνιών, που υπάγεται στη Σχολή Θετικών Επιστημών του Εθνικού και Καποδιστριακού Πανεπιστημίου Αθηνών. Την κεντρική επίβλεψη της διατριβής ανέλαβε η επιστημονική υπεύθυνη του εργαστηρίου, Αφροδίτη Τσαλαγατίδου, η οποία διατελεί επίκουρη καθηγήτρια του Τμήματος, ενώ στην τριμελή επιτροπή παρακολούθησης συμμετείχαν επίσης οι Ιωάννης Κοτρώνης, επίκουρος καθηγητής του Τμήματος, και Αλέξης Δελής, καθηγητής του Τμήματος.

Το ερευνητικό αντικείμενο της διατριβής εντάσσεται στην ευρύτερη θεματική περιοχή των κατανεμημένων εφαρμογών λογισμικού και αφορά στις υπηρεσιοστρεφείς τεχνολογίες. Συγκεκριμένα, η έρευνα της διατριβής εστίασε στη μελέτη προβλημάτων, που εμφανίζονται κατά τη διαδικασία αναζήτησης υπηρεσιών διαφόρων τύπων, όπως υπηρεσίες ιστού, υπηρεσίες πλέγματος και υπηρεσίες ομότιμου-προς-ομότιμο, και τα οποία οφείλονται στην πολυδιάστατη ετερογένεια, που διέπει τις τεχνολογίες αυτές. Η μελέτη της βιβλιογραφίας έδειξε, ότι οι υπάρχουσες λύσεις αναζήτησης υπηρεσιών δεν αντιμετωπίζουν επαρκώς τις προκλήσεις, που η ετερογένεια αυτή δημιουργεί.

Αναγνωρίζοντας το τεχνολογικό αυτό κενό και καθοδηγούμενη από απαιτήσεις προερχόμενες από πραγματικές εφαρμογές, η διατριβή προτείνει ένα πλαίσιο αποτελούμενο από μια γλώσσα επερωτήσεων για υπηρεσίες και την αντίστοιχη μηχανή εκτέλεσής της. Συνολικά, μέσω του πλαισίου, προδιαγράφεται μια *ενιαία* προσέγγιση στην αναζήτηση ετερογενών υπηρεσιών, σύμφωνα με την οποία επιτυγχάνεται πλήρης διαχωρισμός των επερωτήσεων και της μηχανής που τις εκτελεί, από τις διάφορες ετερογενείς υπηρεσιοστρεφείς τεχνολογίες. Για την επίτευξη αυτού του στόχου, η προτεινόμενη γλώσσα επερωτήσεων ορίζεται σε ένα υψηλό επίπεδο αφαίρεσης, ενώ, τόσο αυτή, όσο και η μηχανή εκτέλεσής της χαρακτηρίζονται από ευελιξία, που υποστηρίζεται από κατάλληλους μηχανισμούς επεκτάσεων. Το υψηλό επίπεδο αφαίρεσης και η ευελιξία του πλαισίου δεν έχουν αρνητικές επιπτώσεις στην χρησιμότητα και απόδοσή του, όπως επιβεβαιώνουν μια σειρά πειραματικών μετρήσεων.

Ανάπτυξη μιας γλώσσας και του απαραίτητου λογισμικού εκτέλεσής της, για την αναζήτηση ετερογενών υπηρεσιών με ενιαίο τρόπο

# ΚΕΦΑΛΑΙΟ 1

## ΕΙΣΑΓΩΓΗ

Στις μέρες μας, οι περισσότερες εταιρίες και οργανισμοί επεκτείνουν τις δραστηριότητές τους στο διαδίκτυο, αποσκοπώντας στην επίτευξη αυτοματισμού και βελτίωσης του τρόπου λειτουργίας τους, καθιστώντας με τον τρόπο αυτό έντονη την παρουσία τους στο λεγόμενο ψηφιακό κόσμο και τη συνεπαγόμενη ψηφιακή οικονομία. Στο πλαίσιο αυτό, η υιοθέτηση κατάλληλων, εύρωστων και ανταγωνιστικών τεχνολογιών αποτελεί σημαντικό παράγοντα επιβίωσης. Προκειμένου να ανταπεξέλθουν στις συνεχείς αλλαγές, στο υψηλό επίπεδο ανταγωνισμού, αλλά και να προσαρμόσουν τις ιδιαίτερες απαιτήσεις και ανάγκες τους στο νέο αυτό μοντέλο λειτουργίας, οι σημερινές εταιρίες και οργανισμοί στρέφονται σε νέες, δυναμικές τεχνολογίες γρήγορης υλοποίησης και ολοκλήρωσης εφαρμογών λογισμικού, οι οποίες επιτρέπουν την επαναχρησιμοποίηση και διαλειτουργικότητα υπαρχουσών υποδομών. Έτσι, τα τελευταία χρόνια, παρατηρείται αυξημένη αποδοχή και χρήση κατανεμημένων τεχνολογιών βασισμένων στη χρήση συστατικών (component-based technologies) (π.χ. Enterprise Java Beans, EJB [41], COM/COM+ [31], CORBA [32], κ.ά.). Η *υπηρεσιοστρεφής υπολογιστική (Service-Oriented Computing, SOC)* αποτελεί την σύγχρονη έκφανση της τάσης αυτής, προσφέροντας ένα προγραμματιστικό μοντέλο για την ανάπτυξη κατανεμημένων, διαλειτουργικών εφαρμογών λογισμικού.

Σε μια *υπηρεσιοστρεφή αρχιτεκτονική (service-oriented architecture, SOA)*, οι *πάροχοι υπηρεσιών (service providers)* καθιστούν τις υπηρεσίες τους προσβάσιμες μέσω καλά ορισμένων και βασισμένων σε πρότυπα διεπαφών, των οποίων οι περιγραφές δημοσιεύονται σε κατάλληλες υποδομές *μεσαζόντων υπηρεσιών (service brokers)*. Ακολουθώντας, οι ενδιαφερόμενοι *καταναλωτές υπηρεσιών (service consumers)* πραγματοποιούν αναζητήσεις στα περιεχόμενα των μεσαζόντων, προκειμένου να εντοπίσουν τις υπηρεσίες που τους ενδιαφέρουν και ικανοποιούν τις λειτουργικές ή/και μη λειτουργικές τους απαιτήσεις. Εφόσον μια υπηρεσία επιλεγεί από τον καταναλωτή, ακολουθεί μια διαπραγματευτική διαδικασία μεταξύ αυτού και του αντίστοιχου παρόχου, με το πέρας της οποίας η υπηρεσία καθίσταται διαθέσιμη προς χρήση, στο πλαίσιο της εκάστοτε εφαρμογής.

Ήδη από τις αρχές της τρέχουσας δεκαετίας, η υπηρεσιοστρεφής προσέγγιση στη σχεδίαση και υλοποίηση κατανεμημένων εφαρμογών κατέστη ιδιαίτερα δημοφιλής, ενώ ο αριθμός των διαθέσιμων υπηρεσιών, που προσφέρουν μη τετριμμένη λειτουργικότητα στο επίπεδο εφαρμογής, αυξάνεται επίσης με γοργούς ρυθμούς. Υπό αυτές τις συνθήκες, η διαδικασία της *αναζήτησης υπηρεσιών (service discovery)* αποκτά ιδιαίτερη σημασία, καθώς βοηθά στον εντοπισμό της κατάλληλης υπηρεσίας από ένα εν δυνάμει μεγάλο σύνολο πιθανών εναλλακτικών λύσεων. Επιπλέον, αφορά σε όλα τα στάδια του κύκλου ζωής μιας υπηρεσιοστρεφούς εφαρμογής και πραγματοποιείται επαναληπτικά, έτσι ώστε

να βελτιώσει τη λειτουργικότητα και απόδοσή της. Πράγματι, στο πλαίσιο ανάπτυξης και χρήσης μιας υπηρεσιοστρεφούς εφαρμογής, η αναζήτηση υπηρεσιών εφαρμόζεται στις ακόλουθες περιπτώσεις:

- Κατά τη φάση του προσδιορισμού και της εκτίμησης των απαιτήσεων, που προηγείται της φάσης σχεδίασης της εφαρμογής
- Κατά τη σχεδίαση της εφαρμογής, όπου οι επιθυμητές λειτουργίες και διεπαφές καθορίζονται μέσω αποσύνθεσης, ή ακολουθώντας μια “από κάτω προς τα πάνω” προσέγγιση
- Κατά την εκτέλεση της εφαρμογής, προκειμένου να καταστεί δυνατή η δυναμική αντικατάσταση υπηρεσιών που είτε δε λειτουργούν ή παρουσιάζουν ποιοτικά προβλήματα, με άλλες, ισοδύναμες λειτουργικά και ποιοτικά καλύτερες υπηρεσίες

Εξάλλου, η αναζήτηση μιας υπηρεσίας μπορεί να πραγματοποιείται με ad hoc τρόπο από απλούς χρήστες, που επιθυμούν να ενσωματώσουν μια συγκεκριμένη λειτουργία στην επιφάνεια εργασίας του υπολογιστή τους, ή στον προσωπικό τους βοηθό δεδομένων (PDA, Personal Data Assistant). Σε κάθε περίπτωση, η δυνατότητα αποτελεσματικής αναζήτησης υπηρεσιών αποτελεί σημαντικό παράγοντα στη διάδοση και αποδοχή του υπηρεσιοστρεφούς μοντέλου, καθώς επιτρέπει σε επιχειρήσεις, οργανισμούς και άλλες οντότητες, να καθιστούν τους εαυτούς τους γνωστούς και να αλληλεπιδρούν, τόσο μεταξύ τους, όσο και με οποιονδήποτε άλλο χρήστη, μέσω των υπηρεσιών που προσφέρουν, ή χρειάζονται.

## 1.1 Κίνητρο της Διατριβής

Χάρη στην υποσχόμενη διαλειτουργικότητα μεταξύ ετερογενών συστημάτων μέσω κοινά αποδεκτών προτύπων, η υπηρεσιοστρέφεια αποτελεί σήμερα τη σύγχρονη τάση στην περιοχή των κατανεμημένων εφαρμογών. Η πλέον δημοφιλής υπηρεσιοστρεφής τεχνολογία είναι αναμφισβήτητα οι *υπηρεσίες ιστού* (*web services*), οι οποίες βασίζονται σε ευρείας αποδοχής πρότυπα, που καθορίζουν τον τρόπο περιγραφής τους (WSDL, Web Services Description Language) [26], τους μηχανισμούς δημοσίευσης και αναζήτησης των περιγραφών τους (UDDI, Universal Description, Discovery and Integration) [28], καθώς και τη μορφοποίηση των μηνυμάτων κλήσης τους (SOAP, Simple Object Access Protocol) [80].

Ακολουθώντας την επιτυχία και αποδοχή των υπηρεσιών ιστού, το υπηρεσιοστρεφές μοντέλο υλοποιήθηκε στο πλαίσιο διαφόρων ετερογενών τεχνολογιών, οι οποίες ακολουθούν διαφορετικές αρχιτεκτονικές και βασίζονται σε εξειδικευμένα εννοιολογικά μοντέλα. Στην περιοχή της *Υπολογιστικής Πλέγματος* (*Grid Computing*), η αρχιτεκτονική OGSA (Open Grid Service Architecture) [46] και τελικώς το πλαίσιο WSRF (Web Services Resource

Framework) [128], μαζί με τις διάφορες πλατφόρμες υλοποίησης (π.χ. Globus [45], gLite [74]), σηματοδότησαν την ήδη διαφαινόμενη τάση χρήσης υπηρεσιών για την ανάπτυξη σύγχρονων εφαρμογών Πλέγματος. Βασικό χαρακτηριστικό και ειδοποιός διαφορά των *υπηρεσιών πλέγματος (grid services)*, σε σχέση με άλλους τύπους υπηρεσιών, είναι η συσχέτιση και στενή εξάρτησή τους με τον πόρο που διαχειρίζονται.

Εξάλλου, μια νέα γενιά τεχνολογιών ομότιμου-προς-ομότιμο (p2p technologies) εμφανίστηκαν τα τελευταία χρόνια, με κύριους αντιπροσώπους τις τεχνολογίες JXTA [53] και Edutella [83], ορίζοντας μια εναλλακτική μορφή υπηρεσιοστρεφών αρχιτεκτονικών και χρησιμοποιώντας *υπηρεσίες ομότιμου-προς-ομότιμο (peer-to-peer services)*, ως μέσο επικοινωνίας των ομότιμων κόμβων, αλλά και ως βασικά συστατικά της λειτουργίας του δικτύου. Έτσι, σε ένα υπηρεσιοστρεφές δίκτυο ομότιμων κόμβων, κάθε κόμβος μπορεί να διαδραματίζει ταυτόχρονα τους ρόλους του παρόχου, μεσάζοντα και καταναλωτή υπηρεσιών.

Η ετερογένεια που διέπει τις προαναφερθείσες τεχνολογίες υπηρεσιών καθιστά την αναζήτησή τους, ως πρώτο και βασικό βήμα στην υπηρεσιοστρεφή προσέγγιση ανάπτυξης λογισμικού, ιδιαίτερα δύσκολη και απαιτητική σε πόρους. Πράγματι, καθεμία από τις τεχνολογίες αυτές χρησιμοποιεί τους δικούς της μηχανισμούς για την υποστήριξη της αναζήτησης υπηρεσιών: οι υπηρεσίες ιστού βασίζονται στη χρήση ειδικών μητρώων, όπως αυτά που προσδιορίζουν τα πρότυπα UDDI [28] και ebXML [38], οι υπηρεσίες πλέγματος χρησιμοποιούν ειδικούς, εσωτερικούς μηχανισμούς, όπως για παράδειγμα καταλόγους βασισμένους στο σχήμα LDAP [70], ενώ οι υπηρεσίες ομότιμου-προς-ομότιμο δημοσιεύονται μέσω ειδικών εγγράφων-διαφημίσεων, τα οποία διαχέονται στο δίκτυο και αναζητούνται με τη χρήση εγγενών μηχανισμών, που αυτό προσφέρει.

Παρόμοια ετερογενής χαρακτηρίζεται η κατάσταση που επικρατεί στις τεχνολογίες περιγραφής των υπηρεσιών ιστού, πλέγματος και ομότιμου-προς-ομότιμο, όπου μια πληθώρα προτύπων, σχημάτων και προδιαγραφών χρησιμοποιούνται για την περιγραφή των συντακτικών, σημασιολογικών και ποιοτικών χαρακτηριστικών τους. Αν και η γλώσσα WSDL αποτελεί το κοινά αποδεκτό πρότυπο για την περιγραφή της διεπαφής μιας υπηρεσίας, υπάρχουν διάφορες εναλλακτικές προσεγγίσεις για τη σημασιολογική της περιγραφή (π.χ. OWL-S [77], WSMO [101], SAWSDL [43]), ενώ για την περιγραφή των ποιοτικών χαρακτηριστικών επιπέδου εφαρμογής, όπως για παράδειγμα η διαθεσιμότητα, αξιοπιστία, ή η απόδοση της υπηρεσίας, ο αριθμός των προτεινόμενων σχημάτων είναι ακόμη μεγαλύτερος (π.χ. WSLA [63], WS-QoS [116], DAML-QoS [134], QoSOnt [35], WSOL [117], κτλ.).

Καθώς, στις μέρες μας, η ανάγκη συνδυασμένης χρήσης υπηρεσιών ιστού, υπηρεσιών πλέγματος και υπηρεσιών ομότιμου-προς-ομότιμο στο πλαίσιο ολοκληρωμένων λύσεων λογισμικού παρατηρείται έντονα σε αρκετές περιοχές εφαρμογών, η εξεύρεση εφαρμόσιμων λύσεων στο πρόβλημα της παρατηρούμενης, πολύπλευρης ετερογένειας παρουσιάζεται επιτακτική. Ακολουθώντας, τεκμηριώνουμε τον ισχυρισμό αυτό και απεικονίζουμε το

πρόβλημα της ετερογένειας περιγράφοντας δύο σενάρια πραγματικών εφαρμογών, των οποίων οι λειτουργικές απαιτήσεις επιβάλλουν τη σύνθεση και χρήση υπηρεσιών ιστού, υπηρεσιών πλέγματος και υπηρεσιών ομότιμου-προς-ομότιμο. Τα δύο σενάρια προέρχονται από τις πιλοτικές εφαρμογές που αναπτύχθηκαν στο πλαίσιο του ερευνητικού έργου SODIUM [119].

### 1.1.1 Σενάριο 1: Εφαρμογή στον Τομέα της Υγείας

Μια ιδιωτική κλινική αναπτύσσει μια σειρά εφαρμογών εσωτερικής χρήσεως βάσει του υπηρεσιοστρεφούς μοντέλου, προκειμένου να υποστηρίξει το ιατρικό προσωπικό κατά την καθημερινή τους εργασία. Μια από τις συχνότερες δραστηριότητες στο πλαίσιο λειτουργίας της κλινικής είναι ο καθορισμός της θεραπευτικής αγωγής των ασθενών της, όπου απαιτείται ο συνδυασμός διαφόρων πληροφοριών (π.χ. φαρμακευτικό ιστορικό του ασθενούς, λεπτομέρειες σχετικά το ιατρικό επεισόδιο που οδήγησε στην εισαγωγή του ασθενούς, κτλ.) καθώς και η συνεργασία εξειδικευμένων ιατρών.

Η διεργασία καθορισμού της θεραπευτικής αγωγής ξεκινά με την ανάκτηση του φακέλου του ασθενούς, στον οποίο καταγράφεται το ιστορικό του επεισοδίου. Βάσει των κωδικών του ασθενούς και του συγκεκριμένου επεισοδίου, ανακτάται η τρέχουσα φαρμακευτική αγωγή, ενώ ταυτόχρονα εκτελείται μια διαδικασία εντοπισμού κάποιου διαθέσιμου, εξειδικευμένου ιατρού, προκειμένου να αναλάβει τον καθορισμό της θεραπευτικής αγωγής. Ο πρώτος διαθέσιμος ιατρός που θα βρεθεί μετά την ανάκτηση της λίστας όλων των συμβεβλημένων με την κλινική ιατρών, αξιοποιεί τις πληροφορίες σχετικά με την τρέχουσα φαρμακευτική αγωγή του ασθενούς, ενώ λαμβάνει μια στατιστική ανάλυση των φαρμάκων που αυτή συμπεριλαμβάνει. Επιπλέον, για την εξασφάλιση της ορθότητας της γνώματευσης κατά το μέγιστο δυνατό, ο ιατρός έρχεται σε επαφή με κάποιο συνάδελφό του, προκειμένου να πάρει μια δεύτερη ιατρική γνώμη για το επεισόδιο. Τέλος, βάσει όλων των πληροφοριών που συνέλεξε, ο ιατρός καθορίζει τη θεραπευτική αγωγή του ασθενούς, καταγράφοντάς τη στο αρχείο του και ενημερώνοντας το φάκελο του ασθενούς.

Για την ανάπτυξη της διεργασίας καθορισμού θεραπευτικής αγωγής ως υπηρεσιοστρεφή εφαρμογή, θα πρέπει να αναζητηθούν οι κατάλληλες υπηρεσίες, για καθεμιά από τις προδιαγεγραμμένες επιμέρους λειτουργίες. Με βάση τις επιθυμητές λειτουργίες, οι υπηρεσίες αυτές μπορεί να είναι διαφόρων τύπων. Για παράδειγμα, η ανάκτηση πληροφοριών από τη βάση δεδομένων της κλινικής, όπως ο ιατρικός φάκελος του ασθενούς και η λίστα εξειδικευμένων ιατρών, μπορεί να πραγματοποιηθεί από ασφαλείς υπηρεσίες ιστού υψηλής διαθεσιμότητας, που έχουν δημοσιευτεί στο ιδιωτικό μητρώο της κλινικής. Ο έλεγχος της διαθεσιμότητας ενός ιατρού, καθώς και η επικοινωνία μεταξύ δύο ιατρών μπορεί να υποστηρίζονται από εξειδικευμένες, αξιόπιστες υπηρεσίες αποστολής μηνυμάτων ομότιμου-προς-ομότιμο, που είναι διαθέσιμες στο κλειστό δίκτυο ομότιμων κόμβων της



κλινικής. Τέλος, η στατιστική ανάλυση των φαρμάκων μπορεί να υλοποιηθεί από εξεζητημένες, υψηλής απόδοσης και ακρίβειας υπηρεσίες πλέγματος, που είναι διαθέσιμες στην υποδομή Πλέγματος, στην οποία συμμετέχει η κλινική στο πλαίσιο συνεργασίας με άλλους οργανισμούς από το χώρο της Υγείας.

Προκειμένου να υλοποιηθεί η διεργασία καθορισμού θεραπευτικής αγωγής, οι μηχανικοί λογισμικού της κλινικής θα πρέπει πρωτίστως να αναζητήσουν διαθέσιμες υπηρεσίες, που ικανοποιούν τις λειτουργικές και μη λειτουργικές απαιτήσεις των επιμέρους λειτουργιών. Καθώς οι υπηρεσίες ιστού, πλέγματος και ομότιμου-προς-ομότιμο είναι δημοσιευμένες σε ετερογενή μητρώα, καταλόγους και δίκτυα, η διαδικασία αναζήτησης καθίσταται ακριβή και χρονοβόρα. Πράγματι, οι μηχανικοί λογισμικού θα πρέπει να αναζητήσουν ξεχωριστά τις διάφορες υπηρεσίες, χρησιμοποιώντας πολλαπλούς, εξειδικευμένους μηχανισμούς αναζήτησης και συντάσσοντας κάθε φορά τις επερωτήσεις τους σύμφωνα με την εκάστοτε μορφοποίηση των δημοσιευμένων περιγραφών υπηρεσιών, γεγονός που προϋποθέτει εξοικείωση με τεχνικές λεπτομέρειες χαμηλού επιπέδου για καθεμιά υπηρεσιοστρεφή τεχνολογία. Μια τέτοια κατάσταση έρχεται σε πλήρη αντιδιαστολή με τις βασικές αρχές της υπηρεσιοστρέφειας, σύμφωνα με τις οποίες η ανάπτυξη μιας εφαρμογής διευκολύνεται από την αναζήτηση και επαναχρησιμοποίηση διαθέσιμων υπηρεσιών.

### 1.1.2 Σενάριο 2: Εφαρμογή στον Τομέα της Διαχείρισης Κρίσεων

Το τμήμα μηχανογράφησης μιας εταιρίας, που δραστηριοποιείται στην ανάπτυξη εφαρμογών λογισμικού διαχείρισης κρίσεων, αποφάσισε να υιοθετήσει το υπηρεσιοστρεφές μοντέλο, για την ανάπτυξη και διάθεση των εφαρμογών της. Μια τυπική και ταυτόχρονα σημαντική λειτουργία, που απαιτείται από τα διάφορα συστήματά της, αφορά στην έγκαιρη πρόσβαση ασθενοφόρων στην περιοχή, όπου έχει συμβεί ένα ατύχημα ή μια φυσική καταστροφή, προκειμένου να περισυλλέξει και να μεταφέρει τους τραυματίες στο πλησιέστερο νοσοκομείο.

Η αντίστοιχη διεργασία ξεκινά να εκτελείται με τη λήψη ενός τηλεφωνήματος από το τηλεφωνικό κέντρο της μονάδας αντιμετώπισης κρίσεων. Βάσει του αριθμού της κλήσης, εντοπίζεται η περιοχή του ατυχήματος και η διεργασία προχωρά στην παράλληλη εκτέλεση των ακόλουθων δραστηριοτήτων:

- Το σύστημα ανακτά μια λίστα με τα ασθενοφόρα που βρίσκονται κοντά στην πληγείσα περιοχή. Καθένα από αυτά τα ασθενοφόρα ελέγχεται ως προς τη διαθεσιμότητά του και, σε περίπτωση που είναι διαθέσιμο, του αποστέλλονται οι κατάλληλες οδηγίες πρόσβασης. Προκειμένου το ασθενοφόρο να φτάσει στο σημείο του ατυχήματος όσο το δυνατό πιο γρήγορα, το σύστημα υπολογίζει τη βέλτιστη διαδρομή, δεδομένης της παρούσας θέσης του ασθενοφόρου.

- Το σύστημα λαμβάνει την τρέχουσα κατεύθυνση του ανέμου στην περιοχή όπου συνέβη το ατύχημα και, βάσει αυτής της πληροφορίας, υπολογίζει τη διάδοση του χημικού νέφους, που προέκυψε από την έκρηξη στο εργοστάσιο. Τελικά, αποστέλλεται ειδοποίηση προς τις τοπικές αρχές προκειμένου να προετοιμαστούν για πιθανή εκκένωση της περιοχής.

Σύμφωνα με το υπηρεσιοστρεφές μοντέλο ανάπτυξης εφαρμογών, κάθε μια από τις επιμέρους δραστηριότητες της παραπάνω διεργασίας μπορεί να υλοποιηθεί από μια κατάλληλη υπηρεσία, η οποία προσφέρει τη συγκεκριμένη λειτουργία. Κατ' αυτόν τον τρόπο, αποφεύγεται η εξ ολοκλήρου υλοποίηση της εφαρμογής από μηδενική βάση. Έτσι, οι μηχανικοί λογισμικού της εταιρίας καλούνται να ανακαλύψουν τις κατάλληλες υπηρεσίες, πριν προχωρήσουν στην υλοποίησή της. Αυτές οι υπηρεσίες, βάσει των ζητούμενων λειτουργιών, μπορεί να είναι:

- *Υπηρεσίες ιστού (web services)*, που παρέχουν πληροφορίες σχετικά με την τοποθεσία από την οποία έγινε το τηλεφώνημα, τα πλησιέστερα ασθενοφόρα, την κατεύθυνση του ανέμου στην περιοχή, κτλ.
- *Υπηρεσίες πλέγματος (grid services)*, οι οποίες αναλαμβάνουν τους πολύπλοκους υπολογισμούς σχετικά με τις βέλτιστες διαδρομές των ασθενοφόρων, τη διάδοση του χημικού νέφους, κτλ.
- *Υπηρεσίες ομότιμου-προς-ομότιμο (peer-to-peer services)*, οι οποίες επιτρέπουν την απευθείας επικοινωνία του κέντρου αντιμετώπισης κρίσεων με κάθε ασθενοφόρο, προκειμένου να μάθει την τοποθεσία του, την κατάστασή του και να του παρέχει εντολές μετάβασης, οδηγίες πρόσβασης, κτλ.

Όπως και στην περίπτωση του σεναρίου, που περιγράφηκε στην προηγούμενως, (Παράγραφος 1.1.1), οι διαφορετικοί τύποι των απαιτούμενων υπηρεσιών, σε συνδυασμό με την ετερογένεια των πρωτοκόλλων περιγραφής τους και των υποδομών στις οποίες αυτές έχουν δημοσιευτεί, καθιστούν την αναζήτησή τους ιδιαίτερα επίπονη.

## 1.2 Προκλήσεις και Συνεισφορά της Διατριβής

Συνοψίζοντας τα συμπεράσματα της προηγούμενης ενότητας παρατηρούμε, ότι, για την ανάπτυξη σύγχρονων, εξεζητημένων υπηρεσιοστρεφών εφαρμογών, απαιτείται συχνά η εύρεση και χρήση διαφόρων ειδών υπηρεσιών. Ωστόσο, προκειμένου να καταστεί δυνατή και αποτελεσματική η αναζήτησή τους, κρίνεται αναγκαία η αντιμετώπιση μιας σειράς προκλήσεων:

- **Ετερογένεια των τύπων και περιγραφών υπηρεσιών.** Ο αιτών μια υπηρεσία χρειάζεται υποστήριξη στο σχηματισμό επερωτήσεων, με τρόπο ανεξάρτητο από τα μοντέλα των διαφόρων τύπων υπηρεσιών και τη μορφοποίηση των δημοσιευμένων περιγραφών τους.
- **Ετερογένεια των υποδομών μεσαζόντων.** Η επεξεργασία και εκτέλεση των επερωτήσεων αναζήτησης υπηρεσιών πρέπει να πραγματοποιείται ανεξάρτητα από τα ετερογενή περιβάλλοντα, στα οποία έχουν δημοσιευτεί οι περιγραφές τους, ενώ οι τεχνικές λεπτομέρειες των μηχανισμών αναζήτησης, που προσφέρουν οι διάφορες υποδομές μεσαζόντων, πρέπει να αποκρύπτονται από το χρήστη.
- **Πολυ-διάστατη μορφοποίηση και αντιστοίχιση επερωτήσεων.** Κατά την αναζήτηση υπηρεσιών, ο αιτών χρειάζεται υποστήριξη στο σχηματισμό επερωτήσεων, που αντικατοπτρίζουν τις λειτουργικές και μη λειτουργικές απαιτήσεις του προς διάφορες ιδιότητες και χαρακτηριστικά της ζητούμενης υπηρεσίας, καθώς και στην αξιολόγηση των διαθέσιμων υπηρεσιών με βάση τα προσδιορισμένα κριτήρια αναζήτησης.
- **Τεχνολογική ρευστότητα.** Οποιαδήποτε λύση στην αναζήτηση ετερογενών υπηρεσιών χρειάζεται να μπορεί να ανταπεξέλθει σε ένα ρευστό τεχνολογικά περιβάλλον, όπου οι εξελίξεις τρέχουν, προκειμένου να καταστεί βιώσιμη σε βάθος χρόνου.

Θεωρούμε, ότι μια *ενοποιημένη* προσέγγιση στην αναζήτηση υπηρεσιών μπορεί να αντιμετωπίσει με επιτυχία τις ανωτέρω προκλήσεις. Κινούμενοι προς αυτή την κατεύθυνση, παρουσιάζουμε με την διατριβή αυτή μια ολοκληρωμένη λύση για την ενιαία αναζήτηση ετερογενών υπηρεσιών, υπό τη μορφή ενός επεκτάσιμου πλαισίου. Το προτεινόμενο πλαίσιο περιλαμβάνει μια υψηλού επιπέδου αφαίρεσης γλώσσα επερωτήσεων για υπηρεσίες, την USQL (Unified Service Query Language), καθώς και μια ευέλικτη μηχανή αναζήτησης υπηρεσιών, την οποία ονομάζουμε Proteus. Συνολικά, η προσέγγισή μας:

- Υποστηρίζει τον ενιαίο σχηματισμό επερωτήσεων αναζήτησης υπηρεσιών, καθώς και των αντίστοιχων αποκρίσεων, ανεξαρτήτως του τύπου της επιθυμητής υπηρεσίας, του τρόπου περιγραφής της, ή/και της υποδομής στην οποία έχει δημοσιευτεί.
- Προσφέρει ένα επαρκές συντακτικό, για την έκφραση των λειτουργικών αλλά και μη λειτουργικών απαιτήσεων του χρήστη υπό τη μορφή κριτηρίων αναζήτησης, που μπορούν να αξιολογηθούν έναντι μιας ευρείας γκάμας μορφοποιήσεων περιγραφών υπηρεσιών.
- Είναι διαλειτουργική με ένα ευρύ φάσμα υπηρεσιοστρεφών τεχνολογιών και τύπων υπηρεσιών και συνεπώς εφαρμόσιμη στην αναζήτηση υπηρεσιών σε διάφορα περιβάλλοντα, αποκρύπτοντας τις χαμηλού επιπέδου τεχνικές λεπτομέρειές τους από το χρήστη, χάρη στον ευέλικτο, αφαιρετικό της σχεδιασμό.

- Μπορεί εύκολα να επεκταθεί με την προσθήκη επιπλέον χαρακτηριστικών και λειτουργιοτήτων, έτσι ώστε να συμβαδίζει με τις εξελίξεις στην περιοχή των υπηρεσιοστρεφών τεχνολογιών.

Συνοψίζοντας, σύμφωνα με την υπάρχουσα κατάσταση στην αναζήτηση υπηρεσιών ιστού, υπηρεσιών ομότιμου-προς-ομότιμο και υπηρεσιών πλέγματος, ο αιτών είναι υποχρεωμένος να συντάξει τις αντίστοιχες επερωτήσεις χρησιμοποιώντας ετερογενή πρωτόκολλα μορφοποίησης και μηχανισμούς αναζήτησης. Αντιθέτως, με τη χρήση της γλώσσας USQL και της μηχανής αναζήτησης Proteus, επιτυγχάνεται η ενοποίηση των επιμέρους διαδικασιών αναζήτησης, μέσω μιας ενιαίας διεπαφής για τη σύνταξη και εκτέλεση των επερωτήσεων, χάρη στην οποία αποκρύπτονται τόσο η ετερογένεια, όσο και οι χαμηλού επιπέδου τεχνικές λεπτομέρειες των διαφόρων υπηρεσιοστρεφών τεχνολογιών.

### 1.3 Δομή της Διατριβής

Το υπόλοιπο κείμενο της παρούσης διατριβής είναι οργανωμένο ως εξής:

- Το Κεφάλαιο 2 παρουσιάζει το υπηρεσιοστρεφές μοντέλο και περιγράφει τρεις διακριτές τεχνολογίες που το υλοποιούν: τις υπηρεσίες ιστού, τις υπηρεσίες ομότιμου-προς-ομότιμο και τις υπηρεσίες πλέγματος. Στη συνέχεια, αναφέρονται ένας αριθμός προτύπων που χρησιμοποιούνται για την περιγραφή των υπηρεσιών αυτών, καθώς και για την αναζήτησή τους. Από την παρουσίαση γίνεται σαφές, ότι οι διάφορες υπηρεσιοστρεφείς τεχνολογίες χαρακτηρίζονται από ετερογένεια, γεγονός που καθιστά δύσκολη τη συνδυασμένη αναζήτηση και χρήση τους. Το κεφάλαιο κλείνει με μια παρουσίαση της υπάρχουσας κατάστασης σχετικά με γλώσσες και μηχανισμούς αναζήτησης υπηρεσιών, όπου και διαπιστώνεται η έλλειψη μιας ολοκληρωμένης προσέγγισης για την ενιαία αναζήτηση ετερογενών υπηρεσιών.
- Το Κεφάλαιο 3 παρουσιάζει τη γλώσσα USQL, η οποία αποτελεί τον ένα από τους δύο πυλώνες συνεισφοράς της παρούσης διατριβής. Η USQL προσφέρει ένα πλούσιο σε στοιχεία και δομές συντακτικό, εκφρασμένο σε XML, για τη σύνταξη επερωτήσεων, οι οποίες περιλαμβάνουν κριτήρια αναζήτησης τόσο προς τις λειτουργικές όσο και προς τις ποιοτικές ιδιότητες μιας υπηρεσίας, ανεξάρτητα από τον τύπο της, το πρότυπο περιγραφής της και την υποδομή στην οποία έχει δημοσιευτεί. Χάρη στο υψηλό επίπεδο αφαίρεσης του μετα-μοντέλου της και την επεκτασιμότητά της, η USQL μπορεί να προσαρμοστεί σε διάφορα ετερογενή υπηρεσιοστρεφή περιβάλλοντα χρήσης, ενώ μπορεί εύκολα να ενσωματώσει πρόσθετα χαρακτηριστικά, εφόσον αυτό καταστεί αναγκαίο. Το μαθηματικό μοντέλο της γλώσσας παρέχει υποστήριξη στη σύνθετη διαδικασία αντιστοίχισης των κριτηρίων αναζήτησης με τις αντίστοιχες ιδιότητες της κάθε μιας υπηρεσίας. Χάρη στον αφαιρετικό του ορισμό, το μοντέλο αυτό είναι

ανεξάρτητο από τους επιμέρους αλγορίθμους, που χρησιμοποιούνται για την ποσοτικοποιημένη σύγκριση διαφόρων ειδών τιμών, όπως ελεύθερο κείμενο, οντολογικές έννοιες, αριθμητικές τιμές κτλ.

- Το Κεφάλαιο 4 παρουσιάζει μια νέα, ευέλικτη μηχανή αναζήτησης ετερογενών υπηρεσιών, η οποία ονομάζεται Proteus και αποτελεί τον δεύτερο πυλώνα συνεισφοράς της διατριβής. Η μηχανή Proteus υλοποιεί τη γλώσσα USQL, ενώ η αρχιτεκτονική της προσομοιάζει τις παραδοσιακές μηχανές αναζήτησης ιστού, διαχωρίζοντας σαφώς τη διαδικασία ανάκτησης των περιγραφών ετερογενών υπηρεσιών, από τα διάφορα ετερογενή μητρώα, αποθετήρια, δίκτυα, κτλ., στα οποία έχουν δημοσιευτεί, από τη διαδικασία αξιολόγησης μιας επερώτησης και αναζήτησης της καταλληλότερης υπηρεσίας, βάσει των κριτηρίων αναζήτησης. Χάρη σε αυτό το διαχωρισμό, η μηχανή αναζήτησης βελτιώνει σημαντικά τους χρόνους απόκρισης, μειώνοντας ταυτόχρονα τις απαιτήσεις σε πόρους συστήματος. Όπως περιγράφεται στο κεφάλαιο, η ετερογένεια που χαρακτηρίζει τις διάφορες υπηρεσιοστρεφείς τεχνολογίες αντιμετωπίζεται αποτελεσματικά από τη προτεινόμενη αρχιτεκτονική, μέσω ενός ευέλικτου μηχανισμού επεκτάσεων, ο οποίος επιτρέπει τη δυναμική σύνδεση πρόσθετων συστατικών σε πολλαπλά σημεία του συστήματος.
- Το Κεφάλαιο 5 αναφέρει μια σειρά πειραματικών αποτελεσμάτων και μετρήσεων, που πραγματοποιήθηκαν στη βάση μιας πρωτότυπης υλοποίησης του συστήματος Proteus και παρουσιάζουν ενδιαφέρον. Συγκεκριμένα, το κεφάλαιο περιλαμβάνει μια αξιολόγηση της γλώσσας USQL, μετρήσεις της απόδοσης των αλγορίθμων αντιστοίχισης ως προς την ακρίβεια και ανάκληση των αποτελεσμάτων αναζήτησης, καθώς και μια σειρά μετρήσεων της απόδοσης της μηχανής αναζήτησης Proteus.
- Το Κεφάλαιο 6 ολοκληρώνει τη διατριβή, συνοψίζοντας τη συνεισφορά της και παρέχοντας μια συζήτηση σχετικά με διάφορες μελλοντικές κατευθύνσεις έρευνας.

Ανάπτυξη μιας γλώσσας και του απαραίτητου λογισμικού εκτέλεσής της, για την αναζήτηση ετερογενών υπηρεσιών με ενιαίο τρόπο

## ΚΕΦΑΛΑΙΟ 2

### ΤΕΧΝΟΛΟΓΙΚΟ ΥΠΟΒΑΘΡΟ

Σε αυτό το κεφάλαιο, αναπτύσσονται οι περιγραφές των διαφόρων υπηρεσιοστρεφών τεχνολογιών, στις οποίες αναφερθήκαμε στην εισαγωγή, διαμορφώνοντας έτσι το τεχνολογικό υπόβαθρο, που θεωρείται απαραίτητο για την κατανόηση της παρούσης διατριβής. Συγκεκριμένα, παρουσιάζουμε το υπηρεσιοστρεφές μοντέλο (Ενότητα 2.1) και περιγράφουμε τρεις αντιπροσωπευτικές και διαδεδομένες υλοποιήσεις, στις οποίες επικεντρώσαμε την έρευνά μας: τις υπηρεσίες ιστού (Παράγραφος 2.2.1), τις υπηρεσίες ομότιμου-προς-ομότιμο (Παράγραφος 2.2.2) και τις υπηρεσίες πλέγματος (Παράγραφος 2.2.3). Οι ετερογενείς αυτοί τύποι υπηρεσιών διαθέτουν ομοιότητες, αλλά και αρκετές διαφορές, οι οποίες καταγράφονται σε ένα γενικό εννοιολογικό μοντέλο υπηρεσιών (Παράγραφος 2.2.4), που παρουσιάζεται εν συντομία και προσφέρει υποστήριξη για την ενιαία μοντελοποίησή τους. Στη συνέχεια του κεφαλαίου, εστιάζουμε στην ετερογένεια των υπηρεσιών αυτών των τύπων και παρουσιάζουμε την υπάρχουσα κατάσταση αναφορικά με τα πρότυπα περιγραφής τους και τους διάφορους μηχανισμούς αναζήτησής τους (Ενότητες 2.3 και 2.4). Τέλος, περιγράφουμε μια σειρά ερευνητικών προσπαθειών για την ανάπτυξη γλωσσών και μηχανισμών αναζήτησης (Ενότητες 2.5 και 2.6), καταδεικνύοντας τις ελλείψεις και την αδυναμία τους να παρέχουν μια ολοκληρωμένη λύση στο πρόβλημα της αναζήτησης ετερογενών υπηρεσιών.

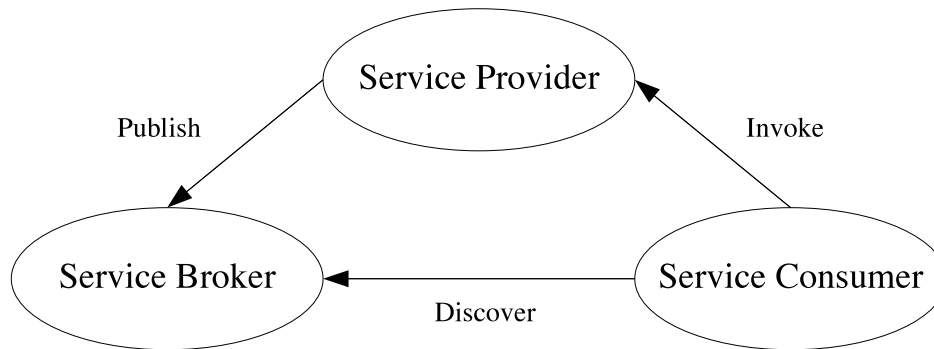
#### 2.1 Το Υπηρεσιοστρεφές Μοντέλο Ανάπτυξης Εφαρμογών

Η υπηρεσιοστρέφεια (service-orientation) προσφέρει ένα μοντέλο σχεδίασης κατανεμημένων εφαρμογών λογισμικού, σύμφωνα με το οποίο οι διάφορες λειτουργικότητες προσφέρονται ως *υπηρεσίες*. Σε μια υπηρεσιοστρεφή αρχιτεκτονική (service-oriented architecture, SOA), οι υπηρεσίες χρησιμοποιούνται ως δομικά συστατικά τα οποία συνδυάζονται μέσω διαδικασιών σύνθεσης, προκειμένου να προσφέρουν νέες, εξεζητημένες λειτουργικότητες. Το αποτέλεσμα αυτών των συνθέσεων μπορεί επίσης να προσφερθεί ως μια νέα, σύνθετη υπηρεσία.

Κάθε υπηρεσιοστρεφής αρχιτεκτονική προσφέρει υποστήριξη σε μια σειρά βασικών λειτουργιών, οι οποίες υπαγορεύονται από το υπηρεσιοστρεφές μοντέλο (Σχήμα 1).

Συγκεκριμένα, το υπηρεσιοστρεφές μοντέλο προσδιορίζει τους ακόλουθους βασικούς ρόλους:

- **Πάροχος Υπηρεσίας (Service Provider).** Ο ρόλος αυτός αντιπροσωπεύει οντότητες (οργανισμούς, εταιρίες, κτλ.), οι οποίες προσφέρουν στους πελάτες τους εφαρμογές λογισμικού υπό τη μορφή υπηρεσιών. Ένας πάροχος δημοσιεύει τις περιγρα-



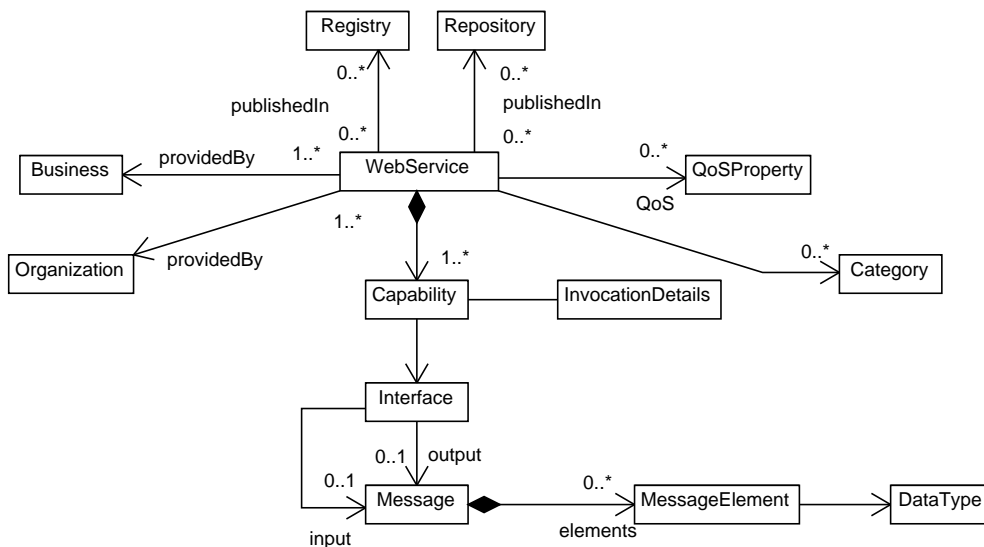
Σχήμα 1: Το υπηρεσιοστρεφές μοντέλο

φές των υπηρεσιών που προσφέρει στο διαδίκτυο, προκειμένου να τις καταστήσει γνωστές στους (υποψηφίους) πελάτες. Ο πάροχος θεωρείται ιδιοκτήτης των υπηρεσιών που προσφέρει καθώς και της πλατφόρμας, στην οποία αυτές υλοποιούνται και διατίθενται προς χρήση.

- **Καταναλωτής Υπηρεσίας (Service Consumer).** Κάθε οντότητα, της οποίας οι ανάγκες ικανοποιούνται από τη χρήση υπηρεσιών, διαδραματίζει το ρόλο του καταναλωτή στο υπηρεσιοστρεφές μοντέλο. Συνήθως, πρόκειται για επιχειρήσεις των οποίων οι εφαρμογές απαιτούν συγκεκριμένα είδη λειτουργικότητας. Σε επίπεδο λογισμικού, ο ρόλος του καταναλωτή αντιστοιχεί σε εφαρμογές, οι οποίες αναζητούν και εν τέλει καλούν διάφορες υπηρεσίες. Γενικότερα, ωστόσο, καταναλωτής μιας υπηρεσίας μπορεί να είναι κάθε απλός χρήστης, που έχει πρόσβαση σε αυτή μέσω του υπολογιστή του, ή οποιασδήποτε άλλης συσκευής συνδεδεμένης στο διαδίκτυο, οποιαδήποτε εφαρμογή λογισμικού, ή ακόμα και μια άλλη υπηρεσία.
- **Μεσάζων Υπηρεσίας (Service Broker).** Ο ρόλος αυτός αφορά σε οντότητες, οι οποίες προσφέρουν υποδομές και μηχανισμούς για τη δημοσίευση και αναζήτηση υπηρεσιών. Οι πάροχοι δημοσιεύουν τις περιγραφές των υπηρεσιών που προσφέρουν σε κάποιο μεσάζοντα, ενώ οι υποψήφιοι καταναλωτές αναζητούν τις επιθυμητές υπηρεσίες χρησιμοποιώντας τους μηχανισμούς αναζήτησης του μεσάζοντος, μέσω κατάλληλων διεπαφών. Ο μεσάζων επιστρέφει όλα τα απαραίτητα στοιχεία μιας υπηρεσίας στον καταναλωτή, έτσι ώστε αυτός να μπορεί να τη χρησιμοποιήσει πραγματοποιώντας απομακρυσμένη κλήση.

Δεδομένου ότι εφόσον οι οντότητες που αναλαμβάνουν τους παραπάνω ρόλους σε μια υπηρεσιοστρεφή αρχιτεκτονική αλληλεπιδρούν μεταξύ τους, απαιτείται η χρήση κοινώς αποδεκτών προτύπων και πρωτοκόλλων προκειμένου να εξασφαλιστεί η διαλειτουργικότητά τους. Ωστόσο, όπως περιγράφουμε στην επόμενη ενότητα, η ύπαρξη πολλών διαφορετικών υλοποιήσεων του υπηρεσιοστρεφούς μοντέλου καθιστά την προτυποποίηση των





Σχήμα 2: Εννοιολογικό μοντέλο υπηρεσιών ιστού

βασικών λειτουργιών του ιδιαίτερα δύσκολη, δυσχεραίνοντας την ανάπτυξη εφαρμογών στις οποίες εμπεριέχονται και διαλειτουργούν υπηρεσίες διαφόρων τύπων.

## 2.2 Ετερογενείς Τύποι Υπηρεσιών

Τα πλεονεκτήματα του υπηρεσιοστρεφούς μοντέλου και η υποσχόμενη από αυτό διαλειτουργικότητα οδήγησε στην εμφάνιση διαφόρων τεχνολογιών, οι οποίες ορίζουν και χρησιμοποιούν διαφορετικά πρότυπα για την περιγραφή, αναζήτηση και κλήση των υπηρεσιών. Στη συνέχεια, εξετάζουμε τρεις δημοφιλείς, ετερογενείς υπηρεσιοστρεφείς τεχνολογίες, τις υπηρεσίες ιστού, τις υπηρεσίες ομότιμου-προς-ομότιμου και τις υπηρεσίες πλέγματος.

### 2.2.1 Υπηρεσίες Ιστού (Web Services)

Η τεχνολογία των υπηρεσιών ιστού (web services) αποτελεί την πλέον δημοφιλή υλοποίηση του υπηρεσιοστρεφούς μοντέλου. Ως υπηρεσίες ιστού ορίζονται επιχειρησιακές εφαρμογές οι οποίες είναι αυτάρκες και αρθρωτές, ενώ προσφέρουν βασισμένες σε πρότυπα διαδικτυακές διεπαφές. Εξάλλου, σύμφωνα με τον οργανισμό διαδικτυακών προτύπων W3C (World Wide Web Consortium), υπηρεσία ιστού είναι “κάθε εφαρμογή, η οποία ταυτοποιείται από ένα μοναδικό URI (Uniform Resource Identifier), και της οποίας η διεπαφή μπορεί να οριστεί, να περιγραφεί και να αναζητηθεί με τη χρήση τεχνολογιών XML. Επιπλέον, μια υπηρεσία ιστού υποστηρίζει απευθείας αλληλεπιδράσεις με άλλους πράκτορες λογισμικού (software agents) ανταλλάσσοντας μηνύματα XML σύμφωνα με διαδικτυακά πρωτόκολλα” [16].

Βασικός στόχος της τεχνολογίας των υπηρεσιών ιστού είναι η επίτευξη της διαλειτουργικότητας μεταξύ ετερογενών συστημάτων λογισμικού, αλλά και η διευκόλυνση της ολο-

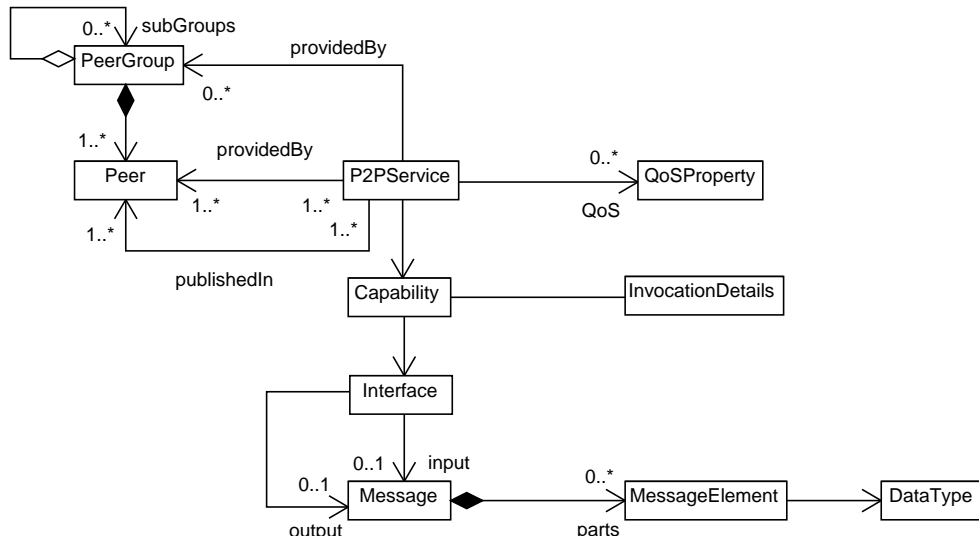
κλήρωσης επιχειρησιακών εφαρμογών (Enterprise Application Integration, EAI) [3]. Αν και οι υπηρεσίες ιστού θεωρούνται από πολλούς ως μια ακόμα τεχνολογία καταναμημένων συστατικών, υπάρχουν αρκετές διαφορές μεταξύ τους, οι οποίες καταδεικνύονται στη βιβλιογραφία [111] [126]. Μια ίσως από τις βασικότερες διαφορές μεταξύ των υπηρεσιών ιστού και των συστατικών έγκειται στα είδη λειτουργικότητας που καθιστώνται διαθέσιμα προς χρήση μέσω αυτών. Οι υπηρεσίες ιστού προσφέρουν ως επί το πλείστον επιχειρησιακές λειτουργίες υψηλού επιπέδου αφαίρεσης, σε αντίθεση με τα συστατικά, τα οποία χρησιμοποιούνται για την έκθεση λειτουργιών και αντικειμένων χαμηλότερου επιπέδου αφαίρεσης.

Στο μοντέλο του Σχήματος 2 απεικονίζονται οι βασικές έννοιες, που συναντώνται στις υπηρεσίες ιστού. Οι υπηρεσίες ιστού παρέχονται ως επί το πλείστον από οργανισμούς και επιχειρήσεις, ενώ δημοσιεύονται σε εξειδικευμένα μητρώα και αποθετήρια. Μια υπηρεσία ιστού καθιστά προσβάσιμες στο χρήστη μια σειρά από λειτουργίες, ή, ισοδύναμα, ικανότητες (capabilities), της επιχείρησης, ή του οργανισμού, που την παρέχει. Βάσει των λειτουργιών αυτών, η υπηρεσία μπορεί να εμπίπτει σε διάφορες κατηγορίες. Κάθε μια λειτουργία χαρακτηρίζεται από μια διεπαφή, η οποία προσδιορίζει τα μηνύματα εισόδου και εξόδου, που χρησιμοποιούνται για την επικοινωνία μεταξύ αυτής και του χρήστη. Αναμένεται, ότι μια λειτουργία υπηρεσίας ιστού δέχεται το πολύ ένα μήνυμα εισόδου και επιστρέφει το πολύ ένα μήνυμα εξόδου. Τόσο το μήνυμα εισόδου, όσο και το μήνυμα εξόδου μιας λειτουργίας υπηρεσίας ιστού αποτελούνται από μηδέν ή περισσότερα στοιχεία, τα οποία έχουν συγκεκριμένο τύπο δεδομένων.

Εκτός από τις προαναφερθείσες λειτουργικές ιδιότητες, μια υπηρεσία ιστού χαρακτηρίζεται από μια σειρά ποιοτικών χαρακτηριστικών (QoS), όπως για παράδειγμα η διαθεσιμότητα, αξιοπιστία, απόδοση, το κόστος χρήσης της, κτλ. Τέλος, προκειμένου να χρησιμοποιηθεί μια συγκεκριμένη λειτουργία, που παρέχεται από μια υπηρεσία ιστού, είναι απαραίτητη η γνώση των στοιχείων εκείνων, που την ταυτοποιούν και την καθιστούν απομακρυσμένα προσβάσιμη.

Η πρώιμη εγκαθίδρυση μιας σειράς προτύπων γύρω από τις βασικές λειτουργίες που απαιτούνται για την ανάπτυξη εφαρμογών βασισμένων σε υπηρεσίες ιστού, καθώς και η υποστήριξη που τα πρότυπα αυτά έλαβαν από μεγάλες εταιρίες στο χώρο της Πληροφορικής, συνετέλεσε στην ευρεία αποδοχή της τεχνολογίας αυτής από τον επιχειρηματικό κόσμο, αλλά και την ερευνητική κοινότητα. Συγκεκριμένα, η γλώσσα WSDL (Web Service Description Language) [26] χρησιμοποιείται για την περιγραφή μιας υπηρεσίας ιστού, η προδιαγραφή UDDI (Universal Description, Discovery and Integration) [28] για τη δημοσίευση και αναζήτησή της, ενώ το πρωτόκολλο SOAP (Simple Object Access Protocol) [80] για την πραγματοποίηση μιας κλήσης προς κάποια από τις προσφερόμενες λειτουργίες της.

Πέρα από τα προαναφερθέντα πρότυπα για την υποστήριξη των βασικών λειτουργιών



Σχήμα 3: Εννοιολογικό μοντέλο υπηρεσιών ομότιμου-προς-ομότιμο

του υπηρεσιοστρεφούς μοντέλου από την τεχνολογία των υπηρεσιών ιστού, ο οργανισμός προτυποποίησης στο διαδίκτυο, W3C, προσδιόρισε μια βασισμένη σε επίπεδα αρχιτεκτονική στοίβα πρωτοκόλλων, για την υποστήριξη πιο εξεζητημένων λειτουργιών, όπως η σύνθεση υπηρεσιών, η διαχείριση δοσοληψιών κ.ά. Η στοίβα αυτή αποτελεί σημείο αναφοράς για τα διάφορα πρωτόκολλα που κάνουν τα τελευταία χρόνια την εμφάνισή τους, καθώς ωριμάζει η τεχνολογία των υπηρεσιών ιστού.

### 2.2.2 Υπηρεσίες Ομότιμου-προς-Ομότιμο (Peer-to-Peer Services)

Σε εννοιολογικό επίπεδο, η υπηρεσιοστρέφεια αποτελεί δομικό στοιχείο του αρχιτεκτονικού μοντέλου ομότιμων κόμβων, καθώς στην πλειοψηφία των σχετικών τεχνολογιών (π.χ. JXTA [53], Edutella [83], Gnutella [51], FreeNet [27]), οι συμμετέχοντες σε ένα δίκτυο ομότιμων κόμβων προσφέρουν, αναζητούν και καταναλώνουν υπηρεσίες ανταλλάσσοντας μηνύματα. Έτσι, οι κόμβοι ενός δικτύου ομότιμων κόμβων υποστηρίζουν καί τους τρεις βασικούς ρόλους του υπηρεσιοστρεφούς μοντέλου. Παρόλα αυτά, η έννοια της υπηρεσίας ομότιμου-προς-ομότιμο είναι κάπως συγκεχυμένη και, σε πολλές περιπτώσεις, μη ορισμένη. Γενικά, σε ένα δίκτυο ομότιμων κόμβων, οι διαθέσιμες υπηρεσίες μπορούν να κατηγοριοποιηθούν ως εξής:

- *Βασικές υπηρεσίες υποδομής*, οι οποίες χρησιμοποιούνται για το σχηματισμό και την υποστήριξη βασικών λειτουργιών του δικτύου ομότιμων κόμβων
- *Υπηρεσίες εφαρμογών*, οι οποίες προσφέρουν εξεζητημένη λειτουργικότητα στους χρήστες, όπως διαμοιρασμό αρχείων, επικοινωνία μέσω άμεσης αποστολής μηνυμάτων, κτλ.

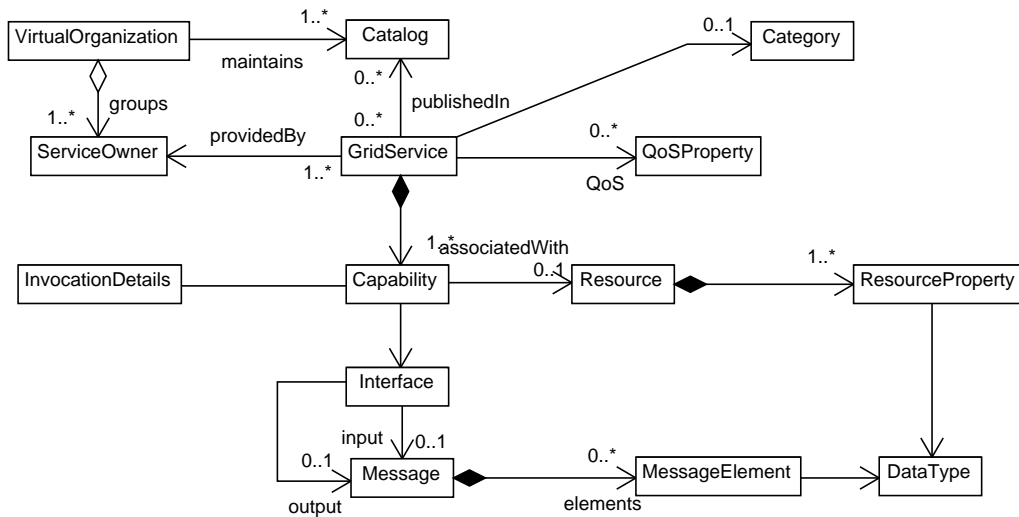
Με βάση την κατηγοριοποίηση αυτή, ιδιαίτερο ενδιαφέρον παρουσιάζουν οι υπηρεσίες της δεύτερης κατηγορίας, καθώς προσφέρουν λειτουργικότητα, η οποία μπορεί να αξιοποιηθεί σε πολλά πεδία εφαρμογών, ενώ μπορεί να συνδυαστεί με λειτουργικότητες άλλων τύπων υπηρεσιών. Εξάλλου, από τοπολογική άποψη, οι υπηρεσίες ομότιμων κόμβων εμπίπτουν στις ακόλουθες κατηγορίες:

- *Υπηρεσίες κόμβων*, που προσφέρονται και χρησιμοποιούνται από κόμβους του δικτύου.
- *Υπηρεσίες δικτύου*, που προσφέρονται από ολόκληρο το δίκτυο ομότιμων κόμβων (π.χ. διαμοιρασμός χώρου αποθήκευσης δεδομένων).
- *Υπηρεσίες ενδιάμεσου επιπέδου*, που προσφέρονται από κόμβους, ή ομάδες κόμβων, σε χρήστες εκτός του δικτύου ομότιμων κόμβων.

Το διάγραμμα του Σχήματος 3 παρουσιάζει εννοιολογικά τις ιδιότητες και τα χαρακτηριστικά των υπηρεσιών ομότιμου-προς-ομότιμο. Κάθε μια υπηρεσία ομότιμου-προς-ομότιμο προσφέρει μια συγκεκριμένη λειτουργικότητα, ενώ μπορεί να παρέχεται από έναν ή περισσότερους ομότιμους κόμβους. Οι κόμβοι ενός δικτύου μπορούν να οργανώνονται σε ομάδες, οι οποίες με τη σειρά τους συσχετίζονται με ιεραρχικό τρόπο. Έτσι, μια ομάδα ομότιμων κόμβων μπορεί να περιλαμβάνει επιμέρους υπο-ομάδες, κ.ο.κ. Μια υπηρεσία μπορεί να προσφέρεται από μια ομάδα ομότιμων κόμβων, σε περιπτώσεις όπου όλοι οι κόμβοι-μέλη της ομάδας παρέχουν την αντίστοιχη λειτουργικότητα. Σε αντίθεση με τις υπηρεσίες ιστού, οι υπηρεσίες ομότιμου-προς-ομότιμο δε χρησιμοποιούν κάποια τρίτη οντότητα για τη δημοσίευσή τους. Αντ' αυτού, η δημοσίευση πραγματοποιείται σε έναν ή περισσότερους κόμβους του δικτύου, οι οποίοι αναλαμβάνουν το ρόλο του μεσάζοντα υπηρεσιών.

Παρατηρούμε, ότι, εκτός από τις προαναφερθείσες διαφορές με τις υπηρεσίες ιστού, τα εννοιολογικά μοντέλα των δύο ειδών υπηρεσιών παρουσιάζουν αρκετές ομοιότητες. Η λειτουργικότητα μιας υπηρεσίας ομότιμου-προς-ομότιμο χρησιμοποιείται μέσω μιας διεπαφής, η οποία προσδιορίζει τα μηνύματα εισόδου/εξόδου, που δέχεται/στέλνει η υπηρεσία κατά την κλήση της. Επίσης, μια υπηρεσία ομότιμου-προς-ομότιμο μπορεί να χαρακτηρίζεται από μια σειρά ιδιοτήτων σχετικών με την ποιότητά της. Τέλος, για την πραγματοποίηση μιας κλήσης προς μια υπηρεσία ομότιμου-προς-ομότιμο, είναι απαραίτητη από τον καταναλωτή της υπηρεσίας μιας σειράς λεπτομερειών, το περιεχόμενο των οποίων καθορίζεται από την εκάστοτε τεχνολογία ομότιμων κόμβων.

Τα πλεονεκτήματα και η χρησιμότητα των υπηρεσιών ομότιμου-προς-ομότιμο έχει διερευνηθεί σε αρκετές εργασίες, στις οποίες παρουσιάζονται εξεζητημένες λύσεις προς διάφορα εγγενή προβλήματα. Για παράδειγμα, οι Gerke et al. [50] προτείνουν ένα αποκεντρωμένο σύστημα αγοράς υπηρεσιών ομότιμου-προς-ομότιμο, υποστηρίζοντας επίσης τη δυναμική τους σύνθεση. Σε μια προσπάθεια να βελτιώσουν τη διαδικασία αναζήτησης



Σχήμα 4: Εννοιολογικό μοντέλο υπηρεσιών πλέγματος

υπηρεσιών ομότιμου-προς-ομότιμο, οι Elenius και Ingmarsson [42] προτείνουν το πλαίσιο Oden, το οποίο βασίζεται στη χρήση οντολογιών και τεχνολογιών του Σημασιολογικού Ιστού (Semantic Web) προκειμένου να εμπλουτίσει τις περιγραφές των υπηρεσιών και να διευκολύνει έτσι την αναζήτηση και επιλογή τους. Εξάλλου, οι Amoretti et al. [4] παρουσίασαν την πλατφόρμα SP2A, για την υλοποίηση υποδομών Πλέγματος βάσει του αρχιτεκτονικού μοντέλου ομότιμων κόμβων, υποστηρίζοντας το διαμοιρασμό πόρων και σημασιολογικών υπηρεσιών ομότιμου-προς-ομότιμο.

### 2.2.3 Υπηρεσίες Πλέγματος (Grid Services)

Η επιτυχία του υπηρεσιοστρεφούς μοντέλου στο πρόσωπο της τεχνολογίας των υπηρεσιών ιστού, καθώς και η υποσχόμενη από αυτό διαλειτουργικότητα μεταξύ ετερογενών εφαρμογών, οδήγησε στη σταδιακή εφαρμογή του στην περιοχή των εφαρμογών Πλέγματος. Σύμφωνα με τους πρωτοπόρους της περιοχής [47], ο όρος “Πλέγμα” (“Grid”) χρησιμοποιείται για να περιγράψει την απαραίτητη υποδομή λογισμικού για την ολοκλήρωση (integration), εικονικοποίηση (virtualization) και διαχείριση υπηρεσιών και πόρων σε ένα κατακεντρωμένο περιβάλλον. Η υποδομή αυτή επιτρέπει την οργάνωση ομάδων χρηστών και πόρων ανεξάρτητα από τη διαχειριστική και οργανωτική περιοχή του διαδικτύου, στην οποία αυτοί υπάγονται. Έτσι, σε αντιδιαστολή με τους οργανισμούς του πραγματικού κόσμου, το Πλέγμα οδηγεί στη δημιουργία εικονικών οργανισμών (virtual organizations), με κύριο στόχο το διαμοιρασμό υπηρεσιών και πόρων.

Η στροφή της κοινότητας της τεχνολογίας Πλέγματος προς μια υπηρεσιοστρεφή προσέγγιση ανάπτυξης εφαρμογών σηματοδοτήθηκε με την έλευση της αρχιτεκτονικής OGSA (Open Grid Services Architecture) [46], καθώς και την υλοποίηση αυτής, OGSi (Open Grid Services Infrastructure) [122]. Η υλοποίηση OGSi ουσιαστικά επέκτεινε την τεχνολογία

των υπηρεσιών ιστού, προκειμένου να παρέχει υποστήριξη σε μια σειρά από λειτουργίες, που απαιτούνται από τις εφαρμογές Πλέγματος, όπως οι μεταβατικές (transient) και με διατήρηση της κατάστασης (stateful) αλληλεπιδράσεις, η διαχείριση κύκλου ζωής (lifecycle management), η αποστολή και λήψη ειδοποιήσεων (notifications), κτλ. Ωστόσο, ο τρόπος με τον οποίο πραγματοποιήθηκαν οι επεκτάσεις αυτές προκάλεσε έντονη κριτική, καθώς καταστρατηγήθηκαν βασικές αρχές του υπηρεσιοστρεφούς μοντέλου, αλλά και της τεχνολογίας των υπηρεσιών ιστού.

Ως αποτέλεσμα της διαμάχης που ακολούθησε μεταξύ της ερευνητικής κοινότητας των υπηρεσιών ιστού και αυτής των εφαρμογών Πλέγματος, το πλαίσιο WS-RF (Web Service - Resource Framework) [128] έκανε την εμφάνισή του, αποτελώντας μετεξέλιξη των OGSA και OGSF, υποσχόμενο να γεφυρώσει τις διαφορές μεταξύ των δύο τεχνολογιών. Στο πλαίσιο WS-RF, μια υπηρεσία πλέγματος (grid service) ορίζεται ως η σύνθεση μιας υπηρεσίας ιστού και του πόρου (resource) στον οποίο επενεργεί, ή τον οποίο χρειάζεται για την εκτέλεσή της. Οι πόροι του Πλέγματος καθιστώνται προσβάσιμοι αποκλειστικά μέσω συγκεκριμένων διεπαφών, οι οποίες υλοποιούνται από υπηρεσίες ιστού, και η αλληλεπίδραση του χρήστη με αυτούς πραγματοποιείται σύμφωνα με το πρότυπο υπονοούμενου πόρου (implied resource pattern). Εν ολίγοις, το πρότυπο αυτό ορίζει ότι ένας πόρος αρχικοποιείται, χρησιμοποιείται και τελικά καταστρέφεται ως αποτέλεσμα της κλήσης αντίστοιχων λειτουργιών μιας υπηρεσίας ιστού. Ο προγραμματιστής της υπηρεσίας αποφασίζει το βαθμό στον οποίο η σχέση μεταξύ της υπηρεσίας ιστού και του πόρου εκτίθεται στο χρήστη.

Σύμφωνα με τις προδιαγραφές WS-Resource και WS-ResourceProperties του πλαισίου WS-RF, η κατάσταση ενός πόρου προσδιορίζεται με τη χρήση ενός εγγράφου ιδιοτήτων πόρου (resource properties), το οποίο συσχετίζεται με το έγγραφο WSDL της υπηρεσίας ιστού, που χρησιμοποιείται για το χειρισμό του πόρου. Το έγγραφο ιδιοτήτων πόρου ορίζει τους τύπους δεδομένων των ιδιοτήτων του πόρου, που είναι ορατές και μπορούν να τροποποιηθούν από το χρήστη μέσω της διεπαφής που υλοποιεί η υπηρεσία ιστού.

Η οφειλόμενη στο πλαίσιο WS-RF εννοιολογική εγγύτητα των υπηρεσιών πλέγματος με τις υπηρεσίες ιστού μπορεί να διαπιστωθεί μέσω των εννοιολογικών μοντέλων τους, που απεικονίζονται στα Σχήματα 4 και 2, αντίστοιχα. Παρατηρώντας κανείς τα δύο αυτά διαγράμματα, οι διαφορές μεταξύ των υπηρεσιών πλέγματος και των υπηρεσιών ιστού εστιάζονται στα ακόλουθα σημεία:

- Οι υπηρεσίες πλέγματος δε δημοσιεύονται σε μητρώα και αποθετήρια μεσαζόντων, αλλά σε εξειδικευμένους καταλόγους (κλάση Catalog), οι οποίοι περιλαμβάνουν επιπλέον πληροφορίες σχετικά με την εκάστοτε υποδομή Πλέγματος. Κάθε κατάλογος αποτελεί μέρος της υποδομής ενός Ιδεατού Οργανισμού (κλάση VirtualOrganization), ο οποίος δημιουργεί μια λογική ομαδοποίηση των παρόχων υπηρεσιών πλέγματος (κλάση ServiceOwner).

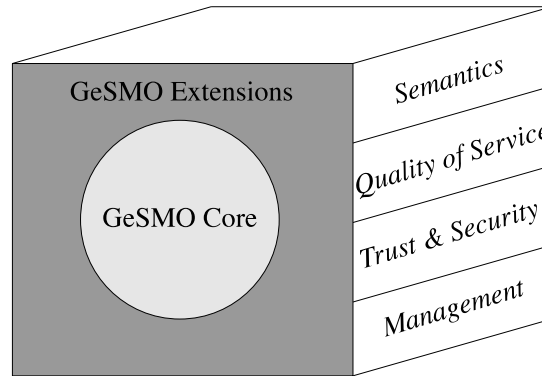
- Λαμβάνοντας υπόψη τις προδιαγραφές του πλαισίου WS-RF, κάθε μια λειτουργικότητα, που προσφέρεται από μια υπηρεσία πλέγματος, συσχετίζεται με ένα συγκεκριμένο πόρο (κλάση Resource). Ένας πόρος αποτελεί ουσιαστικά μια συλλογή ιδιοτήτων (κλάση ResourceProperty), με συγκεκριμένο τύπο δεδομένων.

Οι υπηρεσίες πλέγματος και ειδικά το πλαίσιο WS-RF έτυχαν ευρείας αποδοχής και ήδη εφαρμόζονται επιτυχώς στην ανάπτυξη εφαρμογών Πλέγματος. Για παράδειγμα, οι Chen et al. προτείνουν ένα πλαίσιο για την ανάπτυξη και εκτέλεση κατανεμημένων προσομοιώσεων, οι οποίες βασίζονται στη χρήση συστατικών [24]. Σε μια άλλη προσπάθεια [93], τα πρωτόκολλα του WS-RF χρησιμοποιήθηκαν για την ενίσχυση και βελτίωση ενός αριθμού υπάρχουσών μαθηματικών υπηρεσιών ιστού. Επίσης, το πλαίσιο Weka4WS, που προτάθηκε από τους Talia et al. [115], υιοθέτησε τις τεχνολογίες του WS-RF για την εκτέλεση αλγορίθμων απομακρυσμένης εξόρυξης δεδομένων και για τη διαχείριση κατανεμημένων υπολογισμών. Παράλληλα με την παρουσίαση εργασιών σχετικά με την χρήση υπηρεσιών πλέγματος, εμφανίστηκαν επίσης εργαλεία για τη διευκόλυνση της ανάπτυξής τους, όπως για παράδειγμα το εργαλείο SGSDesigner [52], που επιτρέπει τη δημιουργία και σημασιολογική περιγραφή υπηρεσιών πλέγματος με γραφικό τρόπο. Εξάλλου, μέσω του πλαισίου WS-RF, οι υπηρεσίες πλέγματος βρήκαν εφαρμογή και στο βιομηχανικό τομέα, όπως π.χ. αναφέρεται στις εργασίες των Wu et al. [129], [130].

Όλες οι προαναφερθείσες προσπάθειες υποδηλώνουν τη χρησιμότητα των υπηρεσιών πλέγματος και ταυτόχρονα υποδεικνύουν, ότι ένας δυνητικά μεγάλος αριθμός υπηρεσιών αυτού του είδους είναι διαθέσιμες προς χρήση στο πλαίσιο των εικονικών οργανισμών, όπου ανήκουν.

#### **2.2.4 Προς Ένα Γενικό Μοντέλο Υπηρεσιών**

Από την παρουσίαση των υπηρεσιών ιστού, ομότιμου-προς-ομότιμο και πλέγματος προκύπτει, ότι, υλοποιώντας το υπηρεσιοστρεφές μοντέλο, οι τεχνολογίες αυτές παρουσιάζουν ομοιότητες, καθώς και διαφορές. Σε μια προσπάθεια να επιλύσουμε αυτή την ετερογένεια σε εννοιολογικό επίπεδο, έτσι ώστε να καταστεί δυνατή η ενιαία αναζήτηση, σύνθεση και γενικότερη χρήση τέτοιων υπηρεσιών, αναπτύξαμε ένα Γενικό Μοντέλο Υπηρεσιών (Generic Service Model, GeSMO) [9], [118], στο πλαίσιο του ερευνητικού έργου SODIUM [119]. Στόχος του μοντέλου αυτού ήταν ο σαφής εννοιολογικός προσδιορισμός των κοινών χαρακτηριστικών, που διέπουν τις τεχνολογίες υπηρεσιών ιστού, πλέγματος και ομότιμου-προς-ομότιμου, καθώς και η μοντελοποίηση των επιμέρους χαρακτηριστικών καθεμιάς τεχνολογίας. Για την επίτευξη αυτού του στόχου, το μοντέλο ορίστηκε και σχεδιάστηκε σε επίπεδα, όπως φαίνεται στο Σχήμα 5. Συγκεκριμένα στο βασικό επίπεδο (GeSMO Core), ορίστηκαν όλες οι κοινές έννοιες, ενώ χρησιμοποιούνται επιπλέον επίπεδα επέκτασης (GeSMO Extensions), για την περιγραφή των συγκεκριμένων για κάθε



Σχήμα 5: Αρχιτεκτονική του μοντέλου GeSMO

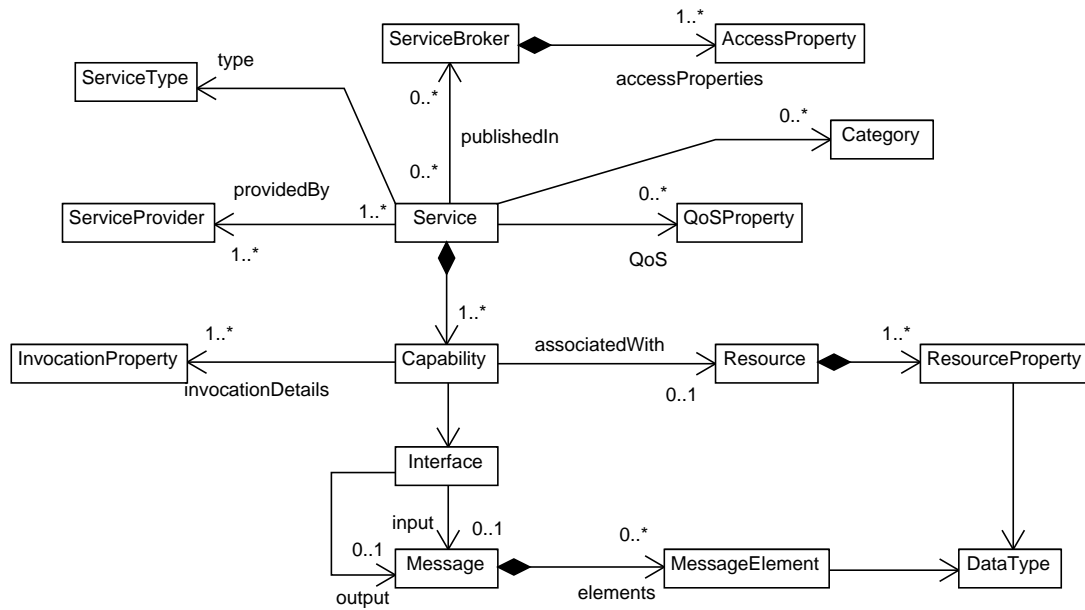
τεχνολογία εννοιών και χαρακτηριστικών. Επιπροσθέτως, στο μοντέλο ορίζεται μια σειρά ορθογώνιων επιπέδων επέκτασης, για την αντιμετώπιση μιας σειράς θεμάτων σχετικών με διάφορες πτυχές όλων των τύπων υπηρεσιών, όπως Ασφάλεια, Διαχείριση, Σημασιολογία, Ποιότητα, κτλ.

Στο βασικό επίπεδο του μοντέλου GeSMO, μια υπηρεσία περιγράφεται με τη χρήση ενός αριθμού συμπληρωματικών όψεων (viewpoints), ως ακολούθως:

- Η *αφαιρετική (abstract)* όψη αντιμετωπίζει την έννοια της υπηρεσίας σε σχέση με εννοιολογικά στοιχεία που συναντώνται, γενικότερα, στην περιοχή της ανάπτυξης λογισμικού.
- Η *βασική (basic)* όψη περιλαμβάνει το ελάχιστο σύνολο στοιχείων και εννοιών, που είναι απαραίτητα για τη μοντελοποίηση μιας υπηρεσίας. Τα στοιχεία αυτά μπορούν ακολούθως να αναλυθούν περαιτέρω.
- Η *δομική (structure)* όψη μοντελοποιεί τη δομή μιας υπηρεσίας, υπό την έννοια των δομικών στοιχείων που αυτή περιλαμβάνει.
- Στην όψη *σημασιολογίας και ποιοτικών χαρακτηριστικών (semantics & QoS)*, αναγνωρίζονται τα στοιχεία μιας υπηρεσίας, στα οποία μπορούν να επικολληθούν σημασιολογικές περιγραφές καθώς και ποιοτικά χαρακτηριστικά.
- Η όψη *δομής μηνυμάτων (message structure)* αναπαριστά τη δομή και τα στοιχεία των μηνυμάτων, που ανταλλάσσονται μεταξύ μιας υπηρεσίας και των χρηστών της.
- Η όψη *επικοινωνίας (communication)* αναγνωρίζει και περιγράφει στοιχεία σχετικά με τις λεπτομέρειες του δικτύου επικοινωνιών, στο οποίο λειτουργεί μια υπηρεσία, όπως π.χ. πρωτόκολλα επικοινωνίας, δικτυακές διευθύνσεις, κτλ.

Μέσω των πολλαπλών όψεων που ορίζει, το μοντέλο δίνει μια αρκετά ευρεία περιγραφή των κοινών χαρακτηριστικών που διέπουν κάθε τύπο υπηρεσίας. Έτσι, χάρη στη





Σχήμα 6: Ενιαίο εννοιολογικό μοντέλο υπηρεσιών

γενικότητά του και την ενιαία προσέγγιση που ακολουθεί, το μοντέλο GeSMO αποτέλεσε τη βάση για τη σχεδίαση της γλώσσας USQL, που αποτελεί μια από τις δύο βασικές συνεισφορές αυτής της διατριβής και παρουσιάζεται στο Κεφάλαιο 3. Το βασικό επίπεδο του GeSMO συνέβαλε στον ορισμό του εννοιολογικού μοντέλου της USQL και στον προσδιορισμό των κατηγοριών κριτηρίων αναζήτησης, έτσι ώστε η γλώσσα να είναι ανεξάρτητη από τις διάφορες τεχνολογίες υπηρεσιών. Επίσης, το σχήμα ορισμού της γλώσσας διατήρησε το χαρακτηριστικό επεκτασιμότητας του μοντέλου, επιτρέποντας την ενσωμάτωση επιπλέον στοιχείων για την υποστήριξη διαφόρων τύπων υπηρεσιών, κριτηρίων αναζήτησης, κτλ.

Το μοντέλο που απεικονίζεται στο Σχήμα 6 αποτελεί την εννοιολογική βάση της γλώσσας USQL. Στο μοντέλο αυτό, στοιχεία από τις διάφορες όψεις του μοντέλου GeSMO συνδυάζονται, επεκτείνονται ή/και τροποποιούνται κατάλληλα, προκειμένου να οριστεί μια νέα, ενιαία εννοιολογική όψη, με χαρακτηριστικά και ιδιότητες των υπηρεσιών ιστού, πλέγματος και ομότιμου-προς-ομότιμο, οι οποίες παρουσιάζουν ενδιαφέρον κατά την αναζήτησή τους. Η νέα αυτή όψη αποτελεί ένα συγκερασμό των εννοιολογικών μοντέλων των υπηρεσιών ιστού, πλέγματος και ομότιμου-προς-ομότιμο, ορίζοντας μια σειρά αφηρημένων εννοιών, για την περιγραφή ετερογενών χαρακτηριστικών τους κατά ενιαίο τρόπο. Συγκεκριμένα, συγκρίνοντας τα τρία επιμέρους μοντέλα των Σχημάτων 2, 3 και 4, με μοντέλο του Σχήματος 6, παρατηρούμε τα εξής:

- Η έννοια Service αποτελεί υπερχλάση των εννοιών WebService, P2PService και GridService
- Οι έννοιες Business, Organization, ServiceOwner, Peer και PeerGroup αντικαθι-

στώνται από την αφηρημένη κλάση ServiceProvider

- Οι έννοιες Registry, Repository, Catalog, και Peer αντικαθίστανται από την αφηρημένη κλάση ServiceBroker
- Η έννοια Capability αποκτά μια προαιρετική συσχέτιση με την έννοια Resource

Συνοψίζοντας, στην ενότητα αυτή αναδείξαμε την ετερογένεια που χαρακτηρίζει τους διαφόρους τύπους υπηρεσιών και παρουσιάσαμε συνοπτικά το μοντέλο GeSMO, ως ένα πρώτο βήμα επίλυσης της ετερογένειας αυτής σε εννοιολογικό επίπεδο. Στις επόμενες ενότητες, παρουσιάζουμε τις διάφορες τεχνολογίες που χρησιμοποιούνται για την περιγραφή και αναζήτηση υπηρεσιών. Πολλές από τις τεχνολογίες αυτές εφαρμόζονται και στους τρεις τύπους υπηρεσιών που μελετούμε, ενώ οι κάποιες επικαλύπτονται μεταξύ τους, αφού χρησιμοποιούνται για την ικανοποίηση κοινών αναγκών. Έτσι, οι τεχνολογίες αυτές προσθέτουν μια επιπλέον διάσταση στο πρόβλημα της ετερογένειας, που δεν αντιμετωπίζεται πλήρως από τις υπάρχουσες γλώσσες και μηχανές αναζήτησης υπηρεσιών.

## 2.3 Ετερογενείς Τεχνολογίες Περιγραφής Υπηρεσιών

Τα τελευταία χρόνια, έχουν κάνει την εμφάνισή τους διάφορες τεχνολογίες περιγραφής υπηρεσιών. Το καθιερωμένο πρότυπο για τη συντακτική περιγραφή υπηρεσιών, δηλαδή για την περιγραφή των διεπαφών που αυτές προσφέρουν, είναι η γλώσσα WSDL (Web Service Description Language) [26]. Σύμφωνα με την επίσημη προδιαγραφή της, η WSDL αποτελεί *“μια XML μορφοποίηση για την περιγραφή υπηρεσιών δικτύου ως ένα σύνολο δικτυακών ακροδεκτών (endpoints), οι οποίοι λειτουργούν ανταλλάσσοντας μηνύματα που περιλαμβάνουν πληροφορία προσανατολισμένη είτε σε έγγραφα (document-oriented) είτε σε διαδικασίες (procedure-oriented)”*. Σύμφωνα με τη δομή της γλώσσας, η WSDL χρησιμοποιείται για να περιγράψει τη διεπαφή μιας υπηρεσίας, δηλαδή τις διαθέσιμες λειτουργίες της και την υπογραφή αυτών (μηνύματα εισόδου/εξόδου), καθώς και τους συγκεκριμένους συνδέσμους (bindings) για την πρόσβαση στους ακροδέκτες της υπηρεσίας.

Αν και αρχικά η WSDL προοριζόταν για την περιγραφή υπηρεσιών ιστού, η χρήση της γρήγορα επεκτάθηκε σε άλλες υπηρεσιοστρεφείς τεχνολογίες, χάρη στην απλότητα και ευελιξία που διέπουν την προδιαγραφή της. Με την έλευση του πλαισίου WSRF (Web Service Resource Framework) [128], η WSDL υιοθετήθηκε για την περιγραφή υπηρεσιών πλέγματος, ενώ, μέσω ενός αριθμού επεκτάσεων, τα περιεχόμενά της εμπλουτίστηκαν με δομές για την περιγραφή των τύπων πόρων (resource types), στους οποίους οι υπηρεσίες πλέγματος επενεργούν. Από την άλλη, όπως δείξαμε σε προηγούμενη εργασία [118], η WSDL μπορεί επίσης να χρησιμοποιηθεί αποτελεσματικά για την περιγραφή των διεπαφών υπηρεσιών ομότιμου-προς-ομότιμο, ενώ πληροφορίες σχετικές με την κλήση τους

μπορούν να ενσωματωθούν με τη χρήση των καταλλήλων επεκτάσεων, για την εκάστοτε τεχνολογία ομότιμου-προς-ομότιμο.

Παρόλη την αποδοχή και διαδεδομένη χρήση της, η WSDL δεν μπορεί να περιγράψει επαρκώς τη σημασιολογία (semantics) μιας υπηρεσίας. Αυτό της το μειονέκτημα αποτέλεσε κίνητρο για τη διεξαγωγή εκτενούς έρευνας, με αποτέλεσμα την εμφάνιση ενός αριθμού επιπρόσθετων προτύπων περιγραφής υπηρεσιών, βασισμένων στη χρήση οντολογιών. Η OWL-S [77] ορίζει μια οντολογία υπηρεσιών με σκοπό την αυτοματοποίηση των διαδικασιών αναζήτησης, σύνθεσης και κλήσης τους. Εκτός από την υποστήριξη σημασιολογικών περιγραφών των λειτουργιών μιας υπηρεσίας, καθώς και των μηνυμάτων εισόδου/εξόδου τους, η OWL-S υποστηρίζει τον προσδιορισμό προϋποθέσεων (preconditions), για τη χρήση μιας υπηρεσίας, και επιδράσεων (effects), που προκύπτουν από την κλήση της. Μοιραζόμενη τους ίδιους στόχους με την OWL-S, η οντολογία WSMO (Web Service Modeling Ontology) [101] προτάθηκε ως ένα εννοιολογικό πλαίσιο συνοδευόμενο από μια επίσημη γλώσσα, την WSM (Web Service Modeling Language), το οποίο επιτρέπει τη σημασιολογική περιγραφή όλων των πτυχών μιας υπηρεσίας.

Τόσο η OWL-S όσο και η WSMO χαρακτηρίζονται από υψηλού βαθμού εκφραστικότητα κι έχουν χρησιμοποιηθεί επιτυχώς σε διάφορες ερευνητικές προσπάθειες, όπου η διαδικασία αναζήτησης υπηρεσιών βάσει σημασιολογικών κριτηρίων αναβαθμίστηκε σημαντικά. Ωστόσο, οι δύο αυτές προδιαγραφές είναι εξαιρετικά πολύπλοκες και αυτός είναι ο κύριος λόγος, για τον οποίο δεν έτυχαν ευρείας αποδοχής στον επιχειρηματικό κόσμο. Σε μια προσπάθεια αντιμετώπισης αυτού του προβλήματος, δημιουργήθηκε μια κοινοπραξία υπό την αιγίδα του οργανισμού προτυποποίησης του διαδικτύου, W3C (World Wide Web Consortium), οι εργασίες της οποίας είχαν ως αποτέλεσμα τη προδιαγραφή SAWSDL (Semantic Annotations for WSDL and XML Schema) [43]. Η SAWSDL αποτελεί ουσιαστικά μετεξέλιξη της γλώσσας WSDL-S [1]: συνδυάζει τα καλύτερα χαρακτηριστικά των OWL-S και WSMO, ενώ χαρακτηρίζεται από την απλότητά της. Πρόκειται για μια προσέγγιση, η οποία υποστηρίζει την προσθήκη σημασιολογικού σχολιασμού (semantic annotations) σε υπάρχουσες περιγραφές υπηρεσιών WSDL, καθώς και σε έγγραφα XML Schema, που περιγράφουν τους τύπους των μηνυμάτων εισόδου/εξόδου μιας υπηρεσίας. Πέραν της απλότητάς της, ένα ακόμα συγκριτικό πλεονέκτημα της SAWSDL έναντι των OWL-S και WSMO είναι η ανεξαρτησία της από τις οντολογίες, που χρησιμοποιούνται για την προσθήκη των σημασιολογικών σχολιασμών. Το γεγονός αυτό επιτρέπει το συνδυασμό και διαλειτουργικότητά της με οποιαδήποτε οντολογία πεδίου ή/και σημασιολογικό λεξικό.

Ένας αριθμός προτύπων έχουν επίσης προταθεί στη βιβλιογραφία για την περιγραφή των μη λειτουργικών ιδιοτήτων μιας υπηρεσίας. Οι Keller και Ludwig ανέπτυξαν την προδιαγραφή WSLA (Web Service Level Agreements) [63], η οποία ορίζει μια γλώσσα XML για τη σύνταξη των ισχυρισμών ενός παρόχου, σχετικά με την προσφορά μιας υπηρεσίας σε συγκεκριμένο πλαίσιο εγγυήσεων ποιότητας. Η προδιαγραφή WSLA ορίζει το στοιχείο

Service Object, το οποίο αποτελεί αφαίρεση του στοιχείου operation της γλώσσας WSDL. Έτσι, δημιουργείται μια υποννοούμενη σύνδεση μεταξύ μιας περιγραφής WSDL και μιας περιγραφής WSLA, που επιτρέπει στις δύο γλώσσες να συνδυαστούν για την περιγραφή των λειτουργικών και μη λειτουργικών ιδιοτήτων μιας υπηρεσίας και των λειτουργιών της.

Το πλαίσιο WS-QoS [116] περιγράφει τα ποιοτικά χαρακτηριστικά μιας υπηρεσίας στο επίπεδο εφαρμογής. Το πλαίσιο ορίζει ένα συντακτικό XML, που υποστηρίζει τη δήλωση των προσφορών QoS ενός παρόχου υπηρεσιών, καθώς και των αντίστοιχων απαιτήσεων QoS ενός αιτούντος. Οι προσφορές του παρόχου μπορούν να ενσωματωθούν στην περιγραφή WSDL της υπηρεσίας, ή εναλλακτικά καταγράφονται σε ανεξάρτητο έγγραφο, το οποίο στη συνέχεια συνδέεται με την περιγραφή WSDL. Μια προσφορά QoS μπορεί να αφορά στη διεπαφή μιας υπηρεσίας –και συνεπώς σε όλες τις περιεχόμενες από αυτή λειτουργίες, ή μπορεί να συνδέεται με κάποια μεμονωμένη λειτουργία της. Η βασική διαφορά του πλαισίου WS-QoS από την προδιαγραφή WSLA έγκειται στο ότι το πρώτο ορίζει ρητά ένα σύνολο παραμέτρων για την περιγραφή ποιοτικών χαρακτηριστικών και απαιτήσεων, όπως είναι η διαθεσιμότητα (availability), η αξιοπιστία (reliability), η τιμή της υπηρεσίας κτλ. Αντίθετα, η προδιαγραφή WSLA προσφέρει μια σειρά από γενικά στοιχεία-αποδέκτες τιμών, τα οποία μπορούν να χρησιμοποιηθούν για να εκφράσουν διάφορων ειδών ποιοτικά χαρακτηριστικά.

Καθώς η χρήση οντολογιών και η σύγκλιση τεχνολογιών του Σημασιολογικού Ιστού (Semantic Web) [15] με τις διάφορες υπηρεσιοστρεφείς τεχνολογίες άρχισε να γίνεται δημοφιλής, έκαναν την εμφάνισή τους ειδικές οντολογίες για την περιγραφή των ποιοτικών χαρακτηριστικών μιας υπηρεσίας. Η οντολογία DAML-QoS [134] ορίζει ένα προφίλ ποιοτικών χαρακτηριστικών (QoS Profile), σε αντιστοιχία με το προφίλ υπηρεσιών που ορίζεται από την οντολογία DAML-S [6]. Το προφίλ αυτό περιλαμβάνει σημασιολογικούς ορισμούς διάφορων ποιοτικών μετρικών και χαρακτηριστικών, όπως είναι η αξιοπιστία, ο χρόνος απόκρισης, το κόστος και η ρυθμοαπόδοση (throughput) της υπηρεσίας. Εν τούτοις, η οντολογία DAML-QoS είναι επεκτάσιμη και συνεπώς μπορεί να χρησιμοποιηθεί για να εκφράσει και άλλα είδη ποιοτικών χαρακτηριστικών, καθώς και τους αντίστοιχους περιορισμούς.

Η οντολογία OntQoS [35] παρέχει ένα γενικό μοντέλο περιγραφής ποιοτικών χαρακτηριστικών, το οποίο μπορεί να εφαρμοστεί σε διάφορα πεδία. Η OntQoS συσχετίζεται με την οντολογία περιγραφής υπηρεσιών OWL-S και ορίζει ένα σύνολο ποιοτικών χαρακτηριστικών και μετρικών, ενώ, όπως και η DAML-QoS, χρησιμοποιείται επίσης για την έκφραση απαιτήσεων ως προς τα ποιοτικά χαρακτηριστικά μιας υπηρεσίας. Επιπλέον, υποστηρίζει τη σύνδεση των προσδιορισμένων ποιοτικών χαρακτηριστικών με ολόκληρη την υπηρεσία, ή με μια συγκεκριμένη λειτουργία της.

## 2.4 Ετερογενείς Τεχνολογίες Αναζήτησης Υπηρεσιών

Η διάδοση του υπηρεσιοστρεφούς μοντέλου οδήγησε στην ανάπτυξη διαφόρων προτύπων μηχανισμών αναζήτησης υπηρεσιών, οι οποίοι υποστηρίζουν ετερογενή πρωτόκολλα και προσφέρουν ετερογενείς διεπαφές στο χρήστη.

Η προδιαγραφή UDDI (Universal Description, Discovery, and Integration) [28] ορίζει ένα σύνολο πρωτοκόλλων και διεπαφών για τη δημοσίευση και αναζήτηση πληροφορίας σχετικής με επιχειρήσεις και υπηρεσίες που αυτές προσφέρουν. Ο προκαθορισμένος μηχανισμός αναζήτησης των μητρώων UDDI υποστηρίζει επερωτήσεις βασισμένες σε κλειδιά-λέξεις (keywords), οι οποίες περιλαμβάνουν απλά κριτήρια αναζήτησης, όπως το όνομα της υπηρεσίας, το όνομα της επιχείρησης που την προσφέρει, η κατηγορία της υπηρεσίας ή/και της επιχείρησης κτλ. Η προδιαγραφή UDDI ορίζει ένα σημείο επέκτασης υπό τη μορφή της δομής tModel, που μπορεί να χρησιμοποιηθεί για τη συσχέτιση μιας εγγραφής επιχείρησης ή υπηρεσίας στο UDDI με επιπρόσθετη, εξωτερική πληροφορία. Σε συνδυασμό με τις διάφορες προσεγγίσεις σημασιολογικής περιγραφής υπηρεσιών, που παρουσιάστηκαν στην προηγούμενη ενότητα, η δομή tModel έχει χρησιμοποιηθεί αποτελεσματικά σε διάφορες ερευνητικές εργασίες, για τη δημοσίευση και αναζήτηση σημασιολογικά περιγεγραμμένων υπηρεσιών ιστού (βλ. π.χ. [109]).

Τα μητρώα ή/και αποθετήρια που είναι συμβατά με το πρότυπο ebXML (Electronic Business using eXtensible Markup Language) [38] προσφέρουν κατάλληλες δομές για την αποθήκευση πληροφοριών σχετικών με επιχειρηματικούς Οργανισμούς. Οι πληροφορίες αυτές χρησιμοποιούνται από τους μηχανισμούς του ebXML για την πραγματοποίηση διεργασιών B2B (Business-to-Business) καθώς και δοσοληψιών. Το περιεχόμενο καθεμιάς εγγραφής Οργανισμού σε ένα μητρώο/αποθετήριο τύπου ebXML μπορεί να είναι ποικιλόμορφο, περιλαμβάνοντας περιγραφές σε απλό κείμενο, διαγράμματα UML [85], έγγραφα XML, ή ακόμα και συστατικά λογισμικού. Στην περίπτωση δημοσίευσης των υπηρεσιών, που προσφέρονται από ένα συγκεκριμένο Οργανισμό, το περιεχόμενο των αντίστοιχων εγγραφών συνήθως αποτελείται από το έγγραφο περιγραφής τους WSDL. Όπως και στην περίπτωση των μητρώων UDDI, ο εγγενής μηχανισμός αναζήτησης του προτύπου ebXML υποστηρίζει επερωτήσεις με κριτήρια αναζήτησης βάσει λέξεων-κλειδιών. Ο περιορισμός αυτός αναγνωρίστηκε και τρόποι αντιμετώπισης προτάθηκαν από διάφορες ερευνητικές εργασίες, όπως αυτή των Dogac et al. [36], όπου σημασιολογικές γλώσσες σαν την OWL-S χρησιμοποιήθηκαν για το σημασιολογικό εμπλουτισμό των δημοσιευμένων πληροφοριών.

Τόσο η δημοσίευση όσο και η αναζήτηση υπηρεσιών ομότιμου-προς-ομότιμο πραγματοποιούνται εντός των ορίων ενός δικτύου ομότιμων κόμβων και η υλοποίησή τους εξαρτάται άμεσα από το είδος του δικτύου αυτού. Στις περισσότερες περιπτώσεις, οι υπηρεσίες δεν δημοσιεύονται σε κάποιο κεντρικό μητρώο, αλλά, αντίθετα, οι περιγραφές τους διαχέονται στο δίκτυο ομότιμων κόμβων, ή κατανέμονται βάσει κάποιου μηχανισμού κατακερμα-

τισμού. Στην περίπτωση της τεχνολογίας JXTA [53], αλλά και των διαφόρων παραλλαγών της (π.χ. Edutella [83]), οι υπηρεσίες που προσφέρονται από έναν ομότιμο κόμβο, ή από ένα σύνολο ομότιμων κόμβων, διαφημίζονται με τη χρήση των ειδικών εγγράφων MSA (Module Spec Advertisement), τα οποία διαχέονται στους κόμβους του δικτύου. Τα έγγραφα MSA ακολουθούν ένα απλό συντακτικό XML και περιλαμβάνουν στοιχεία όπως το όνομα της υπηρεσίας, η περιγραφή της, το όνομα του δημιουργού της, κτλ. Αντίστοιχα, ο εγγενής μηχανισμός αναζήτησης που προσφέρει η υλοποίηση της πλατφόρμας JXTA υποστηρίζει την εκτέλεση επερωτήσεων με κριτήρια αναζήτησης που αντιστοιχούν σε τέτοιου είδους πληροφορίες. Εξάλλου, ένα έγγραφο MSA μπορεί να επεκταθεί, προκειμένου να συμπεριλάβει αναφορές προς εξωτερικά έγγραφα και περιγραφές (π.χ. αναφορά σε ένα έγγραφο WSDL), μέσω του στοιχείου SpecURI.

Οι επερωτήσεις υπηρεσιών σε ένα δίκτυο JXTA αποστέλλονται από τον κόμβο-αιτούντα στους κόμβους rendezvous του δικτύου του, οι οποίοι προσφέρουν υπηρεσίες δρομολόγησης, μέσω της υπηρεσίας Discovery Service που προσφέρεται στο δίκτυο. Αυτοί οι ειδικοί τύποι κόμβων διατηρούν καταλόγους με πληροφορίες σχετικά με τους κόμβους του δικτύου και των πόρων/υπηρεσιών που αυτοί διαμοιράζονται. Σε περίπτωση που η διαφήμιση της ζητούμενης υπηρεσίας βρίσκεται αποθηκευμένη τοπικά σε κάποιον κόμβο rendezvous, αποστέλλεται απευθείας στον αιτούντα ως απόκριση. Διαφορετικά, ο κόμβος rendezvous συμβουλεύεται τον κατάλογό του και επιστρέφει τις διεθύνσεις των κόμβων εκείνων, οι οποίοι διαθέτουν την ζητούμενη υπηρεσία. Σε περίπτωση που ο κατάλογος δεν περιέχει αυτούς τους κόμβους, ο κόμβος rendezvous διαχέει την επερώτηση σε άλλους ομότιμους του κόμβους.

Γενικά, παρατηρούμε ότι η διαδικασία αναζήτησης υπηρεσιών ομότιμου-προς-ομότιμο είναι στενά συνυφασμένη με το εκάστοτε δίκτυο ομότιμων κόμβων και προϋποθέτει ότι ο αιτών αποτελεί μέλος του. Το γεγονός αυτό επιφέρει σημαντικές απαιτήσεις σε πόρους, σε περιπτώσεις που ο αιτών επιθυμεί να πραγματοποιήσει μια αναζήτηση σε περισσότερα από ένα δίκτυα ομότιμων κόμβων.

Σύμφωνα με τις προδιαγραφές της υποδομής Πλέγματος Globus [45], οι υπηρεσίες πλέγματος, που διατίθενται από τους συμμετέχοντες σε έναν Εικονικό Οργανισμό (Virtual Organization, VO), καθώς και οι ιδιότητες των διαμοιραζόμενων πόρων δημοσιεύονται και καθιστώνται διαθέσιμες προς αναζήτηση σε έναν ή περισσότερους καταλόγους. Στην τρέχουσα έκδοση της υποδομής Globus (GT4), οι κατάλογοι αυτοί ακολουθούν το πρωτόκολλο WS-ServiceGroup και αποτελούν μέρος του συστήματος MDS, που υποστηρίζει την παρακολούθηση και ανακάλυψη πόρων και υπηρεσιών πλέγματος.

Εξάλλου, στη σχετικά πρόσφατη υποδομή gLite [74], όλα τα βασικά συστατικά ενός πλεγματοειδούς περιβάλλοντος (grid environment) υλοποιούνται ως υπηρεσίες. Η πλατφόρμα gLite υποστηρίζει την εγγραφή υπηρεσιών τετριμμένης ή εξεζητημένης λειτουργικότητας στο πλαίσιο ενός Εικονικού Οργανισμού. Οι εγγραφές αυτού του είδους ακολουθούν τη

μορφοποίηση του σχήματος GLUE [5], το οποίο με τη σειρά του χρησιμοποιεί την προδιαγραφή καταλόγων LDAP [70]. Έτσι, η αναζήτηση υπηρεσιών πλέγματος στο πλαίσιο ενός VO πραγματοποιείται με τη χρήση μιας εξειδικευμένης υπηρεσίας αναζήτησης (Discovery Service), η οποία προσφέρεται από την πλατφόρμα gLite. Καί σε αυτήν την περίπτωση, τα κριτήρια αναζήτησης περιορίζονται στη χρήση λέξεων-κλειδιών και αναφέρονται σε απλές ιδιότητες των υπηρεσιών, όπως το όνομά τους, η περιγραφής τους, το όνομα του Εικονικού Οργανισμού, κτλ.

## 2.5 Γλώσσες Αναζήτησης Υπηρεσιών

Σε αυτήν την ενότητα, παρουσιάζουμε κάποιες από τις πιο γνωστές ερευνητικές εργασίες στην περιοχή της αναζήτησης υπηρεσιών, εστιάζοντας στις προτεινόμενες από αυτές λύσεις για τη σύνταξη των επερωτήσεων και εξετάζοντας τη δυνατότητα εφαρμογής τους στην περίπτωση της αναζήτησης ετερογενών υπηρεσιών.

Οι Chamberlin et al. παρουσίασαν την Quilt [22], μια βασισμένη στην XML γλώσσα, η οποία συγκεντρώνει χαρακτηριστικά και έννοιες πολλών δημοφιλών γλωσσών επερωτήσεων, όπως η SQL, η XPath [131] κ.ά. Η γλώσσα Quilt χαρακτηρίζεται από έναν υψηλό βαθμό εκφραστικότητας και το συντακτικό της περιλαμβάνει μια ευρεία γκάμα στοιχείων, εκφράσεων και συναρτήσεων. Αν και, πρωτογενώς, η Quilt έχει ως πεδίο εφαρμογής την αναζήτηση ενός μεγάλου φάσματος εγγράφων XML, θα μπορούσε να εφαρμοστεί και για την αναζήτηση υπηρεσιών, καθώς η πλειοψηφία των προτύπων περιγραφής τους είναι βασισμένα στην XML. Ωστόσο, αφού σχηματιστεί μια επερώτηση με τη χρήση της Quilt, μπορεί να εκτελεστεί μόνο σε συλλογές συγκεκριμένων εγγράφων, τα οποία ακολουθούν μια συγκεκριμένη σύνταξη/μορφοποίηση. Με άλλα λόγια, η μορφοποίηση της επερώτησης εξαρτάται σε μεγάλο βαθμό από αυτή των εγγράφων, στα οποία πραγματοποιείται η αναζήτηση. Το γεγονός αυτό καθιστά την Quilt μη εφαρμόσιμη στην περίπτωση αναζήτησης ετερογενών υπηρεσιών, των οποίων οι περιγραφές αποτελούνται από έγγραφα διαφορετικών γλωσσών και γραμματικών.

Το πλαίσιο WSDA (Web Service Discovery Architecture), που προτάθηκε από τον Hoschek [55], ορίζει μια απλοποιημένη εκδοχή της γλώσσας WSDL, την οποία ονομάζει SWSDL, για την περιγραφή των υπηρεσιών. Ακολουθώντας, αξιοποιεί την εκφραστικότητα της γλώσσας XQuery [132], χρησιμοποιώντας τη για τη σύνταξη επερωτήσεων. Το πλαίσιο αυτό χρησιμοποιείται για την αναζήτηση υπηρεσιών σε περιβάλλοντα πλέγματος. Όπως και στην περίπτωση της γλώσσας Quilt, που εξετάσαμε προηγουμένως, οι επερωτήσεις είναι στενά συνδεδεμένες με τη γλώσσα περιγραφής των υπηρεσιών, με αποτέλεσμα η ίδια επερώτηση να μη μπορεί να αξιολογηθεί έναντι διαφορετικών ειδών εγγράφων περιγραφής υπηρεσιών.

Η ανάγκη για υποστήριξη λειτουργικών και μη λειτουργικών απαιτήσεων στην ανα-

ζήτηση υπηρεσιών εντοπίζεται σχετικά νωρίς στη βιβλιογραφία, όπου παρουσιάστηκαν διάφορες προσεγγίσεις-επεκτάσεις του προτύπου UDDI. Το μοντέλο που προτάθηκε από τον Ran [96] προσθέτει έναν ακόμα ρόλο στο κλασικό υπηρεσιοστρεφές μοντέλο, αυτόν του εγγυητή ποιότητας υπηρεσίας (QoS Certifier). Επιπροσθέτως, το μοντέλο επαυξάνει την πληροφορία που αποθηκεύεται σε μητρώα τύπου UDDI, χρησιμοποιώντας τις δομές επέκτασης tModels των προδιαγραφών του UDDI, προκειμένου να εκφράσει τα ποιοτικά χαρακτηριστικά μιας υπηρεσίας. Η νέα αυτή δομή είναι αρκετά γενική και μπορεί να χρησιμοποιηθεί για τον προσδιορισμό πολλαπλών ειδών ποιοτικών χαρακτηριστικών, όπως η διαθεσιμότητα της υπηρεσίας, η αξιοπιστία της, οι δυνατότητες κλιμάκωσής της, κτλ. Στη συνέχεια, τόσο τα αιτήματα αναζήτησης υπηρεσιών όσο και οι αποκρίσεις με τα αποτελέσματα αναζήτησης εκφράζονται σύμφωνα με τις προδιαγραφές και το σχήμα του UDDI. Προφανώς, μια τέτοια προσέγγιση δεν μπορεί να εφαρμοστεί στην περίπτωση ετερογενών στόχων αναζήτησης, αποτελούμενων από μητρώα, αποθετήρια, καταλόγους και δίκτυα διαφόρων τεχνολογιών. Η μέθοδος αναζήτησης WSQBE, που προτάθηκε από τους Crasso et al. [33], προτείνει μια λύση που διευκολύνει τόσο το σχηματισμό επερωτήσεων όσο και τη διαδικασία αντιστοίχισης κατά την αναζήτηση υπηρεσιών. Σύμφωνα με την προσέγγιση αυτή, η αντιστοίχιση των υπηρεσιών λαμβάνει χώρα σε δύο στάδια, όπου, αρχικά, ο χώρος αναζήτησης περιορίζεται με αυτοματοποιημένο τρόπο σε κάποιο υποσύνολο. Στη συνέχεια, αξιολογούνται ως προς τα περιεχόμενα των επερωτήσεων μόνο οι περιγραφές υπηρεσιών που υπάγονται στο υποσύνολο εκείνο του συνολικού χώρου αναζήτησης. Οι αλγόριθμοι στους οποίους βασίζεται η μέθοδος αναζήτησης WSQBE περιλαμβάνουν τεχνικές από το επιστημονικό πεδίο εξόρυξης δεδομένων και από τη θεωρία διανυσματικών χώρων, τις οποίες εφαρμόζουν για την εξαγωγή λέξεων-κλειδιά από τις περιγραφές WSDL των υπηρεσιών. Η μέθοδος WSQBE δεν προϋποθέτει επιπρόσθετη προσπάθεια από την πλευρά του παρόχου μιας υπηρεσίας, σε ό,τι αφορά την περιγραφή της, ενώ είναι ανεξάρτητη από τις υπάρχουσες τεχνολογίες περιγραφής και αναζήτησης υπηρεσιών. Παρόλα αυτά, όπως και άλλες ερευνητικές προσπάθειες (βλ. π.χ. [112]), η μέθοδος αυτή λαμβάνει υπόψη μόνο τις λειτουργικές και δομικές ιδιότητες των υπηρεσιών, όπως αυτές εκφράζονται από τη γλώσσα WSDL. Ως αποτέλεσμα, δεν υποστηρίζει την αναζήτηση υπηρεσιών βάσει κριτηρίων ως προς μη λειτουργικές ιδιότητες, ενώ επιπλέον δεν είναι συμβατή με τις διάφορες τεχνολογίες για σημασιολογική περιγραφή υπηρεσιών, καθώς αγνοεί ενδεχόμενες επεκτάσεις της WSDL (π.χ. SAWSDL, OWL-S, κτλ.). Τέλος, η μέθοδος WSQBE έχει εφαρμοστεί αποκλειστικά για την αναζήτηση υπηρεσιών ιστού, ενώ δεν υπάρχει ένδειξη σχετικά με την καταλληλότητά της για την αναζήτηση άλλων, ετερογενών τύπων υπηρεσιών.

Λόγω ελλείψεως κατάλληλων προδιαγραφών και γλωσσών επερωτήσεων, κάποιες ερευνητικές προσεγγίσεις στην αναζήτηση υπηρεσιών υποθέτουν τη χρήση του ίδιου προτύπου για την περιγραφή των υπηρεσιών και τη σύνταξη των επερωτήσεων, όπως η



γλώσσα WSDL (βλ. [112]), η οντολογία DAML-S [6] κ.ά. Αυτού του είδους οι υποθέσεις θέτουν περιορισμούς στην εκφραστικότητα των επερωτήσεων, ενώ επιπλέον επιβάλλουν τη χρήση μιας συγκεκριμένης τεχνολογίας, υποχρεώνοντας το χρήστη να ασχοληθεί με τεχνικές λεπτομέρειες χαμηλού επιπέδου.

Το Πλαίσιο Ενιαίας Αναζήτησης Υπηρεσιών (Unified Service Discovery Framework), που προτάθηκε από τους Jin et al. [58], παρέχει κατάλληλα μοντέλα για την περιγραφή και αναζήτηση υπηρεσιών, ενώ είναι εφαρμόσιμο σε ένα ευρύ φάσμα περιβαλλόντων, όπου απαιτείται η αναζήτηση υπηρεσιών, συμπεριλαμβανομένων περιβαλλόντων πλέγματος. Τα προτεινόμενα μοντέλα υποστηρίζουν την έκφραση πληροφοριών σχετικών με την ποιότητα και το πλαίσιο (context) μιας υπηρεσίας, τόσο στη διαδικασία περιγραφής όσο και αναζήτησής της. Σε ό,τι αφορά τις επερωτήσεις, η προτεινόμενη προσέγγιση χρησιμοποιεί τη γλώσσα XQuery για τη σύνταξή τους. Ωστόσο, όπως έχει ήδη αναφερθεί, μια τέτοια επιλογή καθιστά τις επερωτήσεις άμεσα εξαρτώμενες από το εκάστοτε πρότυπο περιγραφής υπηρεσιών και τη συνεπαγόμενη δομή των περιγραφών.

Οι Li et al. πρότειναν ένα δίκτυο επικάλυψης ομότιμων κόμβων [75], πάνω από το οποίο η αναζήτηση υπηρεσιών μπορεί να εφαρμοστεί με αξιοσημείωτες δυνατότητες κλιμάκωσης και ευρωστείας. Στο προτεινόμενο πλαίσιο, οι περιγραφές υπηρεσιών, δοσμένες ως έγγραφα WSDL, αποκτούν επιπλέον μετα-δεδομένα προερχόμενα από τον πάροχο της εκάστοτε υπηρεσίας και δημοσιεύονται σε συγκεκριμένο κόμβο του δικτύου, που προσδιορίζεται βάσει ενός μηχανισμού κλειδιών κατακερματισμού. Ακολουθώς, οι επερωτήσεις, εκφρασμένες σε XML, αντιστοιχίζονται με κάποιο κλειδί κατακερματισμού και προωθούνται στον κόμβο εκείνο, ο οποίος φιλοξενεί την περιγραφή της ζητούμενης υπηρεσίας. Μια σειρά τελεστών μπορούν να χρησιμοποιηθούν στη σύνταξη της επερωτήσεως, η οποία όμως υποστηρίζει την έκφραση συντακτικών απαιτήσεων μόνο, ως προς τη ζητούμενη υπηρεσία. Έτσι, αν και η προτεινόμενη υποδομή παρέχει ένα ισχυρό περιβάλλον για την αναζήτηση υπηρεσιών, η αποτελεσματικότητά της προσέγγισης μειώνεται εξαιτίας της έλλειψης μιας πλούσιας σε περιεχόμενο και δυνατότητες γλώσσας επερωτήσεων.

Στο πλαίσιο ASD, ένα εργαλείο αναζήτησης υπηρεσιών βάσει αρχιτεκτονικής, που προτάθηκε από τους Kozlenkon et al. [71], οι επερωτήσεις σχετικά με μια υπηρεσία, ή μια σύνθεση υπηρεσιών, εκφράζονται με τη χρήση διαγραμμάτων κλάσεων και ακολουθίας της UML. Συγκεκριμένα, ορίζονται διάφορα εξειδικευμένα προφίλ (UML profiles), τα οποία επιτρέπουν τον προσδιορισμό κριτηρίων αναζήτησης ως προς διάφορες ιδιότητες μιας υπηρεσίας, όπως η διεπαφή της, η συμπεριφορά της (σε περίπτωση σύνθετης υπηρεσίας), καθώς και την έκφραση επιπλέον περιορισμών προς το πλαίσιο (context) της ζητούμενης υπηρεσίας. Παρατηρούμε ότι, αν και δεν υποδηλώνεται άμεσα, η γλώσσα επερωτήσεων του εργαλείου ASD θα μπορούσε μελλοντικά να χρησιμοποιηθεί για την αναζήτηση ετερογενών τύπων υπηρεσιών, με τις κατάλληλες προσθήκες. Σε κάθε περίπτωση, το βασικό μειονέκτημα/πρόβλημα χρήσης της UML για τη σύνταξη των επερωτήσεων έγκειται στην

πολυπλοκότητα της χαμηλού επιπέδου γλώσσας XMI (XML Metadata Interchange) [84], που χρησιμοποιείται για την αναπαράσταση των διαγραμμάτων UML σε XML, και η οποία είναι απαραίτητη για την επεξεργασία των επερωτήσεων από κάποιο εργαλείο λογισμικού.

## 2.6 Μηχανές Αναζήτησης Υπηρεσιών

Παράλληλα με τα ερευνητικά αποτελέσματα σχετικά με γλώσσες αναζήτησης υπηρεσιών, ένας μεγάλος αριθμός μηχανών αναζήτησης έχουν προταθεί τα τελευταία χρόνια στη βιβλιογραφία, επεκτείνοντας τις υπάρχουσες τεχνολογίες περιγραφής και αναζήτησης υπηρεσιών, με στόχο τη βελτίωσή τους. Βασικά σημεία εστίασης των περισσότερων ερευνητικών εργασιών αποτελούν η αρχιτεκτονική των διαφόρων συστημάτων αναζήτησης υπηρεσιών, οι μέθοδοι για την επίλυση σημασιολογικής αμφισημίας μεταξύ περιγραφών υπηρεσιών και των αντιστοίχων επερωτήσεων των χρηστών, καθώς και οι υποστηριζόμενοι αλγόριθμοι ταυτοποίησης. Στην ενότητα αυτή, εξετάζουμε κάποιες αντιπροσωπευτικές εργασίες για την αναζήτηση υπηρεσιών ιστού, πλέγματος και ομότιμου-προς-ομότιμο.

Η χρήση τεχνολογιών του Σημασιολογικού Ιστού (Semantic Web) για τη βελτίωση των αποτελεσμάτων της διαδικασίας αναζήτησης υπηρεσιών αποτελεί κοινό σημείο αρκετών μηχανών αναζήτησης. Το σύστημα METEOR-S [92] προσφέρει τους απαραίτητους μηχανισμούς για τον εμπλουτισμό των περιγραφών υπηρεσιών ιστού με σημασιολογικό περιεχόμενο, κάνοντας χρήση κατάλληλων οντολογιών πεδίου, εκφρασμένων σε DAML-S. Σύμφωνα με την ακολουθούμενη προσέγγιση, ιδιαίτερη έμφαση δίνεται στη σημασιολογική περιγραφή των λειτουργιών που προσφέρονται από μια υπηρεσία ιστού, ως προς τους τύπους δεδομένων των μηνυμάτων εισόδου/εξόδου τους. Στόχος της χρήσης οντολογιών είναι η σημασιολογική κατηγοριοποίηση των υπηρεσιών και κατ' επέκταση ο ευκολότερος εντοπισμός τους κατά τη φάση αναζήτησης. Επιπλέον, προσφέρονται μια σειρά αλγορίθμων για την ταυτοποίηση των σημασιολογικά εμπλουτισμένων περιγραφών υπηρεσιών, με τις επερωτήσεις των χρηστών.

Το πλαίσιο MWSDI [86] αποτελεί εξέλιξη του METEOR-S και προσφέρει μια αρχιτεκτονική βασισμένη σε συνομοσπονδίες μητρώων υπηρεσιών (registry federations). Το MWSDI ακολουθεί μια κατανεμημένη αρχιτεκτονική ομότιμων κόμβων και ορίζει την οντολογία XTRO για τη σημασιολογική ομαδοποίηση των μητρώων. Με τη χρήση της οντολογίας αυτής, οι υπηρεσίες ιστού δημοσιεύονται στα κατάλληλα μητρώα, ανάλογα με το σημασιολογικό τους περιεχόμενο. Αντίστοιχα, οι επερωτήσεις αναζήτησης υπόκεινται σε μια διαδικασία επεξεργασίας, κατά την οποία προσδιορίζονται τα μητρώα στα οποία θα εκτελεστούν. Το πλαίσιο MWSDI αντιμετωπίζει μερικώς το πρόβλημα της ετερογένειας που χαρακτηρίζει τις διάφορες τεχνολογίες μητρώων υπηρεσιών, υποστηρίζοντας μητρώα τύπου UDDI και ebXML.

Το περιβάλλον ανάπτυξης OWL-S IDE [110] υποστηρίζει την αναζήτηση σημασιολο-

γικά εμπλουτισμένων υπηρεσιών ιστού επεκτείνοντας κατάλληλα το πρότυπο UDDI, έτσι ώστε να καταστεί δυνατή η δημοσίευση και αναζήτηση υπηρεσιών βάσει των περιγραφών τους σε OWL-S. Συγκεκριμένα, προτείνεται η απεικόνιση του τμήματος OWL-S Profile της περιγραφής μιας υπηρεσίας, με τη χρήση ειδικά ορισμένων στοιχείων επέκτασης tModels του UDDI. Στη συνέχεια, επεκτείνονται οι εγγενείς μηχανισμοί αναζήτησης του UDDI προκειμένου να υποστηριχθεί ο αλγόριθμος σημασιολογικής ταυτοποίησης του συστήματος OWL-S IDE, η λειτουργία του οποίου συνίσταται στην κατάλληλη ποσοτικοποίηση των ιεραρχικών σχέσεων μεταξύ των σημασιολογικών εννοιών. Εξάλλου, η τεχνολογία της OWL-S χρησιμοποιήθηκε και στην περίπτωση αναζήτησης υπηρεσιών ομότιμου-προς-ομότιμο στο πλαίσιο του συστήματος Oden [42], όπου η οντολογία χρησιμοποιήθηκε για το σημασιολογικό εμπλουτισμό των διαφημίσεων υπηρεσιών σε δίκτυα ομότιμων κόμβων JXTA.

Το συστατικό ταυτοποίησης OWLS-MX [76] ακολουθεί μια υβριδική προσέγγιση στην ταυτοποίηση περιγραφών υπηρεσιών ιστού, συνδυάζοντας αλγορίθμους βασισμένους σε οντολογίες, με μια σειρά δημοφιλών τεχνικών ανάκτησης πληροφορίας. Όπως υποδηλώνει το όνομα του συστατικού, απαραίτητη προϋπόθεση για την εφαρμογή του αποτελεί η χρήση της οντολογίας υπηρεσιών OWL-S, τόσο για την περιγραφή των υπηρεσιών ιστού, όσο και για το σχηματισμό των επερωτήσεων αναζήτησής τους.

Ξεφεύγοντας από τη χρήση τεχνολογιών βασισμένων στην OWL-S, το πλαίσιο FLORA [65] εφαρμόζει μια σειρά αλγορίθμων *F*-Λογικής [66] για τη σημασιολογική αναζήτηση και ταυτοποίηση υπηρεσιών ιστού. Η προδιαγραφή και υλοποίηση του συστήματος είναι στενά συνδεδεμένη με το σημασιολογικό πλαίσιο WSMO και χρησιμοποιεί τους φορμαλισμούς που αυτό προσφέρει, για την περιγραφή των δυνατοτήτων των υπηρεσιών ιστού, καθώς και των στόχων των χρηστών. Αντιστοίχως, στην πρόσφατη εργασία των Kourtesis and Paraskakis [69], προτείνεται ένα πλαίσιο βασισμένο στο πρότυπο SAWSDL, για τη σημασιολογική αναζήτηση υπηρεσιών σε μητρώα UDDI.

Στο σύστημα UDDI-M<sup>T</sup> [78], η αναζήτηση υπηρεσιών πλέγματος πραγματοποιείται με την εκτέλεση επερωτήσεων RDQL [79] σε μετα-δεδομένα εκφρασμένα σε RDF. Η κεντρική ιδέα του συστήματος συνίσταται στην εκμετάλλευση των συντακτικών περιγραφών, που δημιουργούνται από τους χρήστες των υπηρεσιών, και στην εκτέλεση των επερωτήσεων σε συγκεκριμένες συνομοσπονδίες μητρώων UDDI, με απώτερο στόχο την εξατομίκευση της διαδικασίας αναζήτησης. Τα μετα-δεδομένα RDF που δημιουργούνται από τους χρήστες αποθηκεύονται σε μια υποδομή, που παρεμβάλλεται μεταξύ των μητρώων UDDI και του χρήστη. Έτσι, κατά την εκτέλεση μιας επερώτησης, ένα μέρος αυτής δρομολογείται στα μητρώα UDDI, ενώ ένα άλλο μέρος, που αφορά στα εξατομικευμένα μετα-δεδομένα, εκτελείται τοπικά από το σύστημα UDDI-M<sup>T</sup>.

Η επέκταση και χρήση των μητρώων UDDI στην αναζήτηση υπηρεσιών πλέγματος βρήκε επίσης εφαρμογή στη μηχανή αναζήτησης DUDE [12], βασικός στόχος της οποίας

ήταν η αντιμετώπιση των περιορισμών κλιμάκωσης. Χρησιμοποιώντας μηχανισμούς κατανεμημένων πινάκων κατακερματισμού (Distributed Hash Tables, DHT), η μηχανή αναζήτησης υπηρεσιών πλέγματος DUDE επιτρέπει τη διενέργεια επερωτήσεων ταυτόχρονα σε έναν μεγάλο αριθμό μητρώων UDDI, χωρίς ωστόσο να προτείνει βελτιώσεις στον τρόπο με τον οποίο δημιουργούνται και αξιολογούνται οι επερωτήσεις. Σε μια άλλη προσπάθεια βελτίωσης των προτύπων αναζήτησης υπηρεσιών πλέγματος, το πλαίσιο G-QoSM [2] επεκτείνει το μοντέλο της αρχιτεκτονικής OGSA έτσι ώστε οι αναζητήσεις να λαμβάνουν υπόψη τα ποιοτικά χαρακτηριστικά των υπηρεσιών. Οι εννοιολογικές επεκτάσεις υλοποιούνται μέσω κατάλληλων προσθηκών στην προδιαγραφή του UDDI, αλλά και στη γλώσσα περιγραφής υπηρεσιών WSDL. Με τον τρόπο αυτό, καθίσταται δυνατή η χρήση των μηχανισμών αναζήτησης του UDDI για την επιλογή υπηρεσιών πλέγματος, βάσει κριτηρίων προς ποιοτικά χαρακτηριστικά, τα οποία είναι στενά συνυφασμένα με το πεδίο εφαρμογής της υπο αναζήτησης υπηρεσίας.

Σε μια προσπάθεια αντιμετώπισης των προβλημάτων κλιμάκωσης της διαδικασίας αναζήτησης υπηρεσιών ιστού στο διαδίκτυο, οι Lausen and Haselwanger [54] προτείνουν μια μηχανή αναζήτησης, της οποίας η αρχιτεκτονική είναι παρόμοια με αυτή των παραδοσιακών μηχανών αναζήτησης ιστού. Συγκεκριμένα, η προτεινόμενη λύση χρησιμοποιεί ένα συστατικό για τον εντοπισμό και την ανάκτηση περιγραφών WSDL στο διαδίκτυο. Ακολουθώντας, τα ανακτηθέντα έγγραφα WSDL υπόκεινται σε κατάλληλη επεξεργασία και ανάλυση, προκειμένου να εξαχθεί χρήσιμη σημασιολογική πληροφορία, ενώ τα αποτελέσματα της διαδικασίας αυτής αξιολογούνται λαμβάνοντας υπόψη τα δεδομένα ανάδρασης, από την αλληλεπίδραση των χρηστών με τη μηχανή αναζήτησης.

## 2.7 Συζήτηση

Στο κεφάλαιο αυτό, παρουσιάσαμε το υπηρεσιοστρεφές μοντέλο και περιγράψαμε τρεις δημοφιλείς υλοποιήσεις του, τις τεχνολογίες των υπηρεσιών ιστού, υπηρεσιών πλέγματος και υπηρεσιών ομότιμου-προς-ομότιμο. Οι τρεις αυτοί τύποι υπηρεσιών παρουσιάζουν αρκετές ομοιότητες, αλλά και διαφορές. Παρόλα αυτά, όπως δείξαμε στην παράγραφο 2.2.4, η ενιαία εννοιολογική προσέγγισή τους είναι δυνατή μέσω του μοντέλου GeSMO, ενός κατάλληλου, ευέλικτου πλαισίου για την πολυδιάστατη περιγραφή ετερογενών υπηρεσιών. Το μοντέλο αυτό αποτελεί τη βάση της προτεινόμενης από την παρούσα διατριβή λύσης στην αναζήτηση ετερογενών υπηρεσιών.

Η περιγραφή των υπαρχουσών τεχνολογιών περιγραφής και αναζήτησης υπηρεσιών κατέδειξε το πρόβλημα της ετερογένειας: τα λειτουργικά και μη λειτουργικά χαρακτηριστικά των υπηρεσιών ιστού, πλέγματος και ομότιμου-προς-ομότιμο περιγράφονται με τη χρήση διαφόρων γλωσσών και σχημάτων, ενώ, επιπλέον, οι περιγραφές αυτές δημοσιεύονται σε μια ευρεία γκάμα μητρώων, αποθετηρίων, καταλόγων, δικτύων κτλ. Η κατάσταση αυτή

δυσχεραίνει την πραγματοποίηση αναζητήσεων υπηρεσιών, καθώς, όπως διαφάνηκε από την παρουσίαση των διαθέσιμων ερευνητικών αποτελεσμάτων, οι επερωτήσεις αναζήτησης είναι στενά συνδεδεμένες με το εκάστοτε σχήμα περιγραφής των υπηρεσιών, καθώς και με την υποδομή, στην οποία αυτές έχουν δημοσιευτεί. Από την άλλη, οι υπάρχουσες μηχανές αναζήτησης, αν και προτείνουν λύσεις σε σημαντικά προβλήματα όπως η κλιμάκωση, η ακρίβεια των αποτελεσμάτων αναζήτησης, η απόδοση κτλ., δεν είναι σε θέση να προσφέρουν υποστήριξη στην ενιαία αναζήτηση ετερογενών υπηρεσιών. Πράγματι, όπως φάνηκε στην ενότητα 2.6, όλες οι προτεινόμενες λύσεις βασίζονται στη χρήση συγκεκριμένων γλωσσών περιγραφής, στοχεύουν σε έναν μόνο τύπο υπηρεσιών και υποστηρίζουν αναζητήσεις σε συγκεκριμένα είδη μητρώων.

Σε μια προσπάθεια να καλύψουμε το τεχνολογικό αυτό κενό, προκειμένου να προσφέρουμε μια ενιαία προσέγγιση στην αναζήτηση ετερογενών υπηρεσιών, ανεξάρτητα από τη γλώσσα περιγραφής τους και την υποδομή στην οποία έχουν δημοσιευτεί, παρουσιάζουμε στα επόμενα κεφάλαια ένα ευέλικτο, επεκτάσιμο πλαίσιο ενιαίας αναζήτησης ετερογενών υπηρεσιών. Το προτεινόμενο πλαίσιο περιλαμβάνει μια ενιαία γλώσσα αναζήτησης υπηρεσιών, την USQL, καθώς και μια επεκτάσιμη μηχανή αναζήτησης με την ονομασία Proteus. Οι δύο αυτές συνεισφορές παρουσιάζονται λεπτομερώς στα επόμενα κεφάλαια.

Ανάπτυξη μιας γλώσσας και του απαραίτητου λογισμικού εκτέλεσής της, για την αναζήτηση ετερογενών υπηρεσιών με ενιαίο τρόπο

## ΚΕΦΑΛΑΙΟ 3

# Η ΓΛΩΣΣΑ ΑΝΑΖΗΤΗΣΗΣ ΕΤΕΡΟΓΕΝΩΝ ΥΠΗΡΕΣΙΩΝ

## USQL

Το κεφάλαιο αυτό παρουσιάζει τη γλώσσα αναζήτησης ετερογενών υπηρεσιών, USQL (Unified Service Query Language). Η γλώσσα USQL αποτελεί μια εκ των δύο βασικών συνεισφορών της διατριβής και προσφέρει μια υψηλού επιπέδου αφαίρεσης λύση, για το σχηματισμό εγγράφων επερωτήσεων και αποκρίσεων, που χρησιμοποιούνται για την ενιαία αναζήτηση ετερογενών υπηρεσιών. Η παρουσίαση της γλώσσας ξεκινά με μια συνοπτική περιγραφή της (Ενότητα 3.2), ενώ στη συνέχεια παρουσιάζεται λεπτομερώς το μετα-μοντέλο της (Ενότητα 3.3). Το συντακτικό της USQL παρουσιάζεται μέσω μιας σειράς παραδειγμάτων (Ενότητα 3.4). Το μαθηματικό μοντέλο της γλώσσας, το οποίο προσδιορίζει τον τρόπο αντιστοίχισης των επερωτήσεων με τα περιεχόμενα των περιγραφών υπηρεσιών, παρουσιάζεται στην Ενότητα 3.5. Τέλος, παρουσιάζουμε στην Ενότητα 3.6 στοιχεία σχετικά με τη συμβατότητα της USQL με τα διάφορα υπάρχοντα πρότυπα περιγραφής και αναζήτησης υπηρεσιών, καθώς και τις δυνατότητες επέκτασής της.

### 3.1 Κίνητρο

Όπως έδειξε η ανάλυση των διαφόρων γλωσσών αναζήτησης υπηρεσιών στο προηγούμενο κεφάλαιο, οι περισσότερες από τις υπάρχουσες προσεγγίσεις παρουσιάζουν περιορισμούς, ως προς τα υποστηριζόμενα είδη κριτηρίων αναζήτησης, τη συμβατότητα με ετερογενή πρότυπα περιγραφής και αναζήτησης υπηρεσιών, αλλά και τη δυνατότητα χρήσης τους για την αναζήτηση επιπροσθέτων τύπων υπηρεσιών, πέραν των υπηρεσιών ιστού. Επιπλέον, οι γραμματικές των προτεινόμενων γλωσσών είτε χαρακτηρίζονται από υπερβολική απλότητα, ή είναι αρκετά πολύπλοκες και συνεπώς δύσχρηστες, τουλάχιστον από χρήστες με σχετικά περιορισμένο τεχνολογικό υπόβαθρο. Από την άλλη, όπως υποδηλώνεται από τα δύο σενάρια που παρουσιάσαμε στις παραγράφους 1.1.1 και 1.1.2, η ανάγκη αναζήτησης, σύνθεσης και χρήσης ετερογενών τύπων υπηρεσιών είναι υπαρκτή, ενώ το Γενικό Μοντέλο Υπηρεσιών, στο οποίο αναφερθήκαμε στην παράγραφο 2.2.4, αποδεικνύει ότι μια ενοποιημένη προσέγγιση για την αντιμετώπιση αυτής της ανάγκης είναι δυνατή και μπορεί να αποτελέσει λύση στο πρόβλημα της ετερογένειας των διαφόρων υπηρεσιοστρεφών τεχνολογιών. Για όλους αυτούς τους λόγους, εισάγουμε την USQL (Unified Service Query Language), μια νέα γλώσσα για την αναζήτηση ετερογενών υπηρεσιών κατά ενιαίο τρόπο, και αφιερώνουμε το παρόν κεφάλαιο στη λεπτομερή παρουσίαση και περιγραφή των χαρακτηριστικών της.

Καθώς κάποιος θα μπορούσε να διερωτηθεί σχετικά με την ανάγκη εισαγωγής μιας

νέας γλώσσας στην ήδη πλούσια σε προδιαγραφές, πρότυπα και γλώσσες στοίβα υπηρεσιοστρεφών τεχνολογιών, λαμβάνοντας επιπλέον υπόψη την ύπαρξη διαφόρων προτύπων γλωσσών επερωτήσεων, όπως για παράδειγμα η XQuery [132], η SPARQL [108], η SQL [7] κ.ά., θα θέλαμε να τονίσουμε τα διαφορετικά πεδία εφαρμογής τους.

Η USQL χρησιμοποιείται για την έκφραση των απαιτήσεων ενός χρήστη σχετικά με τις λειτουργικές και μη λειτουργικές ιδιότητες μιας υπηρεσίας, ανεξάρτητα από τον τύπο της, τον τρόπο με τον οποίο έχει περιγραφεί, καθώς και την υποδομή στην οποία η περιγραφή της έχει δημοσιευτεί. Επίσης, χρησιμοποιείται για τη σύνταξη αποκρίσεων αναζήτησης υπηρεσιών, οι οποίες περιλαμβάνουν όλες τις απαραίτητες πληροφορίες για την κλήση καθεμιάς από τις επιστρεφόμενες υπηρεσίες. Έτσι, κατά κάποιον τρόπο, η USQL μπορεί να χαρακτηριστεί ως μια γλώσσα περιγραφής απαιτήσεων για την αναζήτηση ετερογενών υπηρεσιών. Από την άλλη, η XQuery είναι μια γλώσσα εκφράσεων (expression language), που σχεδιάστηκε για την επεξεργασία και μορφοποίηση δεδομένων σε XML. Η λειτουργικότητα που προσφέρει είναι συμπληρωματική ως προς αυτήν της USQL και ως εκ τούτου θα μπορούσε να χρησιμοποιηθεί συνδυαστικά από κάποια υλοποίηση, προκειμένου να υποστηρίξει την επεξεργασία περιγραφών υπηρεσιών σε XML, βάσει των απαιτήσεων που έχουν προσδιοριστεί σε ένα έγγραφο USQL. Παρομοίως, η SPARQL είναι μια γλώσσα που χρησιμοποιείται για τη σύνταξη επερωτήσεων έναντι πόρων σημασιολογικά περιγεγραμμένων με τη χρήση της RDF [98]. Συνεπώς, και η SPARQL θα μπορούσε να συνδυαστεί με την USQL και να χρησιμοποιηθεί από κάποια υλοποίηση για την ανάκτηση πληροφορίας από περιγραφές υπηρεσιών βασισμένες στο πλαίσιο RDF.

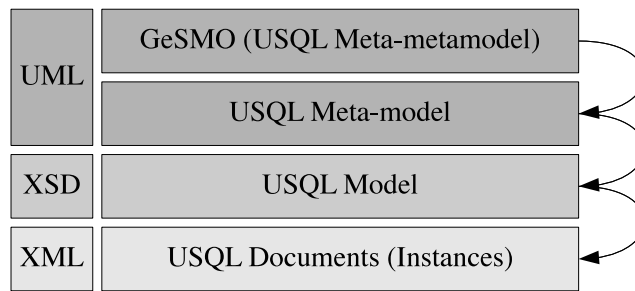
Τέλος, όπως έχει ήδη επισημανθεί στη βιβλιογραφία [64], η πρότυπη γλώσσα SQL [7] δεν μπορεί να υποστηρίξει την έκφραση διαφορετικών προτιμήσεων, ή προτεραιοτήτων, του χρήστη, σε σχέση με τα διάφορα κριτήρια αναζήτησης σε μια επερώτηση. Επιπλέον, η αξιολόγηση μιας επερώτησης SQL από συμβατές μηχανές αναζήτησης επιστρέφει μόνο αποτελέσματα, που ικανοποιούν πλήρως τα κριτήρια αναζήτησης. Ωστόσο, στην αναζήτηση υπηρεσιών, απαιτείται τόσο η υποστήριξη διαφορετικών προτεραιοτήτων κατά το σχηματισμό μιας επερώτησης, όσο και η δυνατότητα επιστροφής αποτελεσμάτων, τα οποία ικανοποιούν μερικώς τα κριτήρια αναζήτησης. Όπως θα δούμε στη συνέχεια, η USQL ικανοποιεί τις προϋποθέσεις αυτές.

### 3.2 Επισκόπηση της USQL

Η γλώσσα USQL (Unified Service Query Language) υποστηρίζει την αναζήτηση ετερογενών τύπων υπηρεσιών σε ετερογενή μητρώα (registries), αποθετήρια (repositories) και δίκτυα υπηρεσιών, καθιστώντας δυνατή τη σύνταξη επερωτήσεων κατά τρόπο *ενιαίο*, ως εξής:

- Οι επερωτήσεις σχηματίζονται έτσι ώστε να είναι ανεξάρτητες από τους διαφόρους





Σχήμα 7: Διαδικασία ορισμού της USQL

τύπους υπηρεσιών και τις αντίστοιχες με αυτούς τεχνολογίες περιγραφής και αναζήτησης

- Οι επερωτήσεις μπορούν να εκφράσουν απαιτήσεις του χρήστη προς πολλαπλές λειτουργικές ή/και ποιοτικές ιδιότητες της υπό αναζήτηση υπηρεσίας

Η διαδικασία ορισμού της γλώσσας USQL απεικονίζεται στο Σχήμα 7. Ο ορισμός της γλώσσας USQL βασίστηκε πρωτογενώς στο Γενικό Μοντέλο Υπηρεσιών GeSMO, για το οποίο ήδη μιλήσαμε στην παράγραφο 2.2.4. Μέσω του μοντέλου αυτού, έγινε εφικτή η αναγνώριση των ομοιοτήτων και διαφορών των υπηρεσιών ιστού, πλέγματος και ομότιμου-προς-ομότιμο, η οποία οδήγησε στον κατάλληλο σχεδιασμό της USQL, προκειμένου να υποστηρίξει την αναζήτησή τους κατά ενιαίο τρόπο. Χρησιμοποιώντας το GeSMO ως μετα-μεταμοντέλο, το μετα-μοντέλο της USQL ορίστηκε με τη χρήση της γλώσσας UML [85] σε ένα σχετικά υψηλό επίπεδο αφάιρεσης και έτσι τοποθετείται ορθογώνια στις διάφορες τεχνολογίες περιγραφής και αναζήτησης των τριών αυτών τύπων υπηρεσιών. Έτσι, χάριν αυτής της αφαιρετικότητας στο επίπεδο του μετα-μοντέλου, η γλώσσα USQL είναι κατάλληλη και προσιτή όχι μόνο σε έμπειρους, αλλά και σχετικά άπειρους τεχνολογικά χρήστες. Επιπλέον, όπως θα δούμε στη συνέχεια, χάρη στη δυνατότητα σημασιολογικού εμπλουτισμού των επερωτήσεων USQL, καθίσταται δυνατή η χρήση της γλώσσας από πράκτορες λογισμικού (software agents), οι οποίοι είναι υπεύθυνοι για την αυτόματη αναζήτηση υπηρεσιών κατά την εκτέλεση μιας εφαρμογής.

Ένα άλλο ουσιώδες χαρακτηριστικό της γλώσσας αποτελεί η επεκτασιμότητά της, μέσω της οποίας καθίσταται δυνατή η ενσωμάτωση επιπλέον χαρακτηριστικών. Κατά αυτόν τον τρόπο, η USQL μπορεί εύκολα να προσαρμοστεί στις διάφορες μελλοντικές εξελίξεις στο χώρο των υπηρεσιοστρεφών τεχνολογιών.

Εκλεπτόνοντας κατάλληλα το μετα-μοντέλο της, το μοντέλο της USQL, ορισμένο με τη χρήση της τεχνολογίας XML Schema [127], προσφέρει ένα συντακτικό σε μορφή XML για το σχηματισμό και συντακτικό έλεγχο των αιτήσεων (requests) του χρήστη και των αντίστοιχων αποκρίσεων (responses) της μηχανής αναζήτησης, κατά την αναζήτηση υπηρεσιών διάφορων ετερογενών τύπων. Η χρήση της XML για την υλοποίηση της γλώσσας προσφέρει μια σειρά πλεονεκτημάτων, όπως ευελιξία και επεκτασιμότητα, ενώ διευκολύνει την

υλοποίηση κατάλληλων εργαλείων χειρισμού των εγγράφων USQL. Σε κάθε περίπτωση, χάρη στην ανεξαρτησία του μετα-μοντέλου της USQL από συγκεκριμένες τεχνολογίες υλοποίησης, είναι δυνατός ο ορισμός εναλλακτικών μοντέλων της γλώσσας, επιπροσθέτως του προτύπου XML μοντέλου.

Τέλος, το μετα-μοντέλο της USQL συνοδεύεται από μια άλγεβρα, βάσει της οποίας πραγματοποιείται η αξιολόγηση μιας υπηρεσίας, ως προς τα κριτήρια αναζήτησης σε μια επερώτηση του χρήστη, και ποσοτικοποιείται η καταλληλότητά της. Βασικά χαρακτηριστικά του μαθηματικού αυτού μοντέλου αποτελούν:

- Η ανεξαρτησία του από τους αλγορίθμους αντιστοίχισης των διαφόρων ειδών τιμών, που χρησιμοποιούνται κατά τη σύνταξη των κριτηρίων αναζήτησης (π.χ. κείμενο, οντολογικές έννοιες, αριθμητικές τιμές, κτλ.).
- Η σαφής υποστήριξη των διαφορετικών προτεραιοτήτων, που ενδεχομένως έχουν τα διάφορα κριτήρια αναζήτησης για το χρήστη.
- Η ευθυγράμμιση με τις απαιτήσεις ρεαλιστικών σεναρίων αναζήτησης υπηρεσιών, όπου είναι επιθυμητή η συμπερίληψη αποτελεσμάτων που ικανοποιούν μερικώς τα κριτήρια αναζήτησης, καθώς και η ποσοτικοποίηση του βαθμού ικανοποίησης αυτών.

Στις επόμενες ενότητες του κεφαλαίου περιγράφονται διεξοδικά το μετα-μοντέλο, το συντακτικό και το μαθηματικό μοντέλο της γλώσσας USQL.

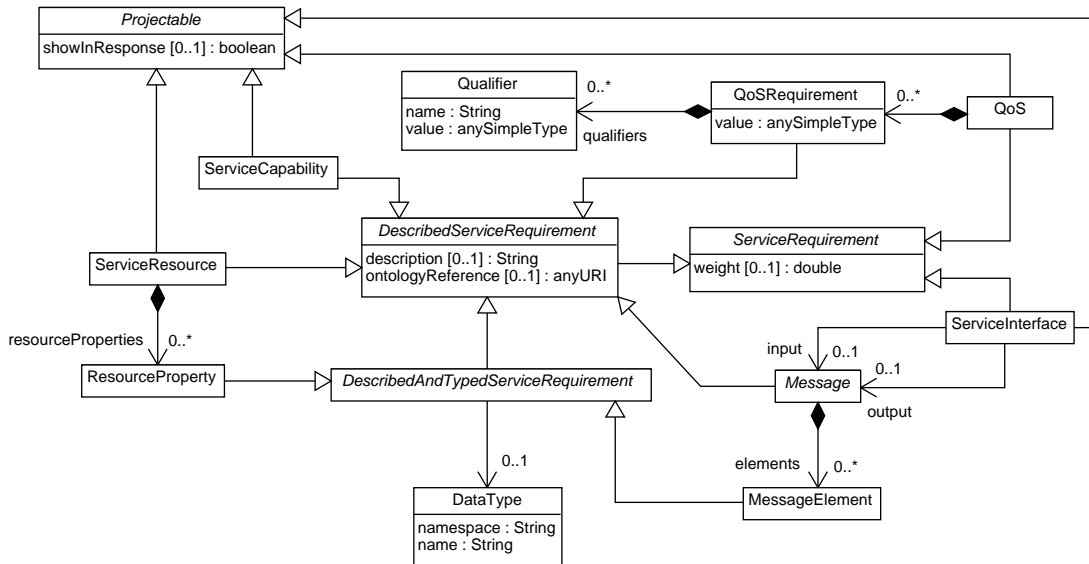
### 3.3 Μετα-Μοντέλο της USQL

Το μετα-μοντέλο της γλώσσας USQL προσδιορίζει, εννοιολογικά και συντακτικά, τη μορφή και το περιεχόμενο των υποστηριζόμενων εγγράφων αιτήσεως και αποκρίσεως. Βασικό του χαρακτηριστικό αποτελεί το υψηλό επίπεδο αφαίρεσης, το οποίο εξασφαλίζει στην USQL ανεξαρτησία από τις διάφορες υπηρεσιοστρεφείς τεχνολογίες.

#### 3.3.1 Στοιχεία Απαιτήσεων

Στον πυρήνα του μετα-μοντέλου βρίσκεται μια ιεραρχία κλάσεων, οι οποίες χρησιμοποιούνται για την έκφραση των διαφόρων ειδών απαιτήσεων (requirements) προς μια υπηρεσία. Η ιεραρχία αυτή απεικονίζεται στο Σχήμα 8.

Στο υψηλότερο επίπεδο της ιεραρχίας, ορίζεται η αφαιρετική κλάση *ServiceRequirement*, που μοντελοποιεί την έννοια της απαιτήσεως του χρήστη, ως προς κάποια ιδιότητα της αναζητούμενης υπηρεσίας. Σε μια επερώτηση, η κάθε μια επιμέρους απαίτηση προσλαμβάνει ένα αριθμητικό βάρος (*weight*), που υποδηλώνει τη σημαντικότητά της σε σχέση με



Σχήμα 8: Ιεραρχία στοιχείων απαιτήσεων της USQL

τις υπόλοιπες απαιτήσεις. Με τον τρόπο αυτό, η USQL μπορεί να εκφράσει τις, διαισθητικά ή ρητά προσδιορισμένες, διαφορετικές προτεραιότητες των κριτηρίων αναζήτησης του χρήστη.

Η, επίσης αφαιρετική, κλάση *DescribedServiceRequirement* επεκτείνει την προαναφερθείσα κλάση *ServiceRequirement* και αποτελεί τη βάση για τον προσδιορισμό απαιτήσεων προς ιδιότητες της αναζητούμενης υπηρεσίας, οι οποίες μπορούν να περιγραφούν (α) με τη χρήση απλού κειμένου, ή/και (β) με τη χρήση σημασιολογικών εννοιών προερχόμενων από κάποια οντολογία. Χάρη στη διπλή αυτή δυνατότητα, επιτυγχάνεται μια ισορροπία μεταξύ της εκφραστικότητας της γλώσσας και της ευκολίας στη χρήση της. Πράγματι, η USQL μπορεί να χρησιμοποιηθεί από χρήστες με σχετικά χαμηλό τεχνολογικό επίπεδο, συνηθισμένους στην έκφραση κριτηρίων αναζήτησης με απλό κείμενο και λέξεις-κλειδιά, καθώς κι από πιο έμπειρους χρήστες, με εξειδικευμένη γνώση του πεδίου εφαρμογής της αναζητούμενης υπηρεσίας, οι οποίοι μπορούν να χειριστούν εξεζητημένα οντολογικά σχήματα για τον προσδιορισμό των απαιτήσεών τους.

Η κλάση *DescribedAndTypedServiceRequirement* αποτελεί ειδική περίπτωση της κλάσης *DescribedServiceRequirement* και χρησιμεύει στην έκφραση απαιτήσεων, οι οποίες αντιστοιχούν σε ιδιότητες της αναζητούμενης υπηρεσίας, που έχουν τύπο δεδομένων. Χαρακτηριστικό παράδειγμα αυτής της κατηγορίας ιδιοτήτων αποτελούν τα στοιχεία των μηνυμάτων εισόδου/εξόδου μιας υπηρεσίας. Ο επιθυμητός τύπος δεδομένων προσδιορίζεται μέσω του χώρου ονομάτων, στον οποίο έχει οριστεί, και του ονόματός του, σύμφωνα με την κλάση *DataType* του μετα-μοντέλου.

Στη βάση των τριών προαναφερθεισών αφαιρετικών κλάσεων, το μετα-μοντέλο της USQL ορίζει έναν αριθμό στοιχείων απαιτήσεων προς συγκεκριμένες ιδιότητες μιας υπηρεσίας. Τα στοιχεία αυτά καταγράφονται συγκεντρωτικά στον Πίνακα 1.

Πίνακας 1: Στοιχεία απαιτήσεων της USQL

Στοιχείο Απαιτήσης	Βασική κλάση	Περιγραφή
ServiceCapability	DescribedServiceRequirement	Λειτουργικότητα υπηρεσίας
ServiceInterface	ServiceRequirement	Διεπαφή υπηρεσίας
Message	DescribedServiceRequirement	Μήνυμα εισόδου/εξόδου
MessageElement	DescribedAndTypedRequirement	Στοιχείο μηνύματος εισόδου/εξόδου
ServiceResource	DescribedServiceRequirement	Πόρος υπηρεσίας
ResourceProperty	DescribedAndTypedRequirement	Ιδιότητα πόρου υπηρεσίας
QoS	ServiceRequirement	Σύνολο ποιοτικών χαρακτηριστικών
QoSRequirement	DescribedServiceRequirement	Ποιοτικό χαρακτηριστικό υπηρεσίας

Σε μια επερώτηση USQL, η επιθυμητή λειτουργικότητα, για την οποία εκτελείται η αναζήτηση, μπορεί να καταγραφεί με τη χρήση του στοιχείου ServiceCapability. Η καταγραφή της επιθυμητής διεπαφής, που η υπηρεσία προσφέρει για την πρόσβαση στη ζητούμενη λειτουργικότητα, πραγματοποιείται μέσω του στοιχείου ServiceInterface. Το στοιχείο αυτό περιλαμβάνει τα μηνύματα εισόδου/εξόδου, που ανταλλάσσονται μεταξύ της υπηρεσίας και του χρήστη, κατά την εκτέλεση της συγκεκριμένης λειτουργικότητας. Οι απαιτήσεις προς τα μηνύματα αυτά συντάσσονται με τη χρήση του στοιχείου Message, το οποίο μπορεί να περιλαμβάνει απαιτήσεις προς μηδέν ή περισσότερα στοιχεία MessageElement.

Εξάλλου, ενδεχόμενες απαιτήσεις του χρήστη ως προς τον πόρο, στον οποίο επενεργεί η ζητούμενη υπηρεσία, εκφράζονται μέσω του στοιχείου ServiceResource. Παρόμοια με την περίπτωση των μηνυμάτων εισόδου/εξόδου, το στοιχείο αυτό μπορεί να περιλαμβάνει απαιτήσεις προς μηδέν ή περισσότερες ιδιότητες του πόρου, οι οποίες συντάσσονται βάσει του στοιχείου ResourceProperty.

Όλα τα προαναφερθέντα στοιχεία απαιτήσεων απευθύνονται στις διάφορες λειτουργικές ιδιότητες της αναζητούμενης υπηρεσίας. Για την υποστήριξη απαιτήσεων ως προς τα ποιοτικές ιδιότητες μιας υπηρεσίας, το μετα-μοντέλο της USQL ορίζει την κλάση QoS, η οποία περιλαμβάνει μηδέν ή περισσότερους περιορισμούς προς ποιοτικά χαρακτηριστικά επιπέδου εφαρμογής. Συγκεκριμένα, οι απαιτήσεις αυτού του είδους εκφράζονται με τη χρήση της αφαιρετικής δομής QoSRequirement. Η δομή αυτή επιτρέπει τον προσδιορισμό απαιτήσεων προς ένα ανοιχτό σύνολο ποιοτικών χαρακτηριστικών (π.χ. διαθεσιμότητα, αξιοπιστία, κόστος, χρόνος απόκρισης, κτλ.), ανεξάρτητα από τον τρόπο, που αυτά έχουν περιγραφεί από τον πάροχο της υπηρεσίας. Καθώς η δομή QoSRequirement αποτελεί εξειδίκευση της αφαιρετικής κλάσης DescribedServiceRequirement, το είδος του ποιοτικού χαρακτηριστικού, στο οποίο αφορά η απαίτηση, προσδιορίζεται με τη χρήση κειμένου, ή οντολογικών εννοιών. Η τιμή του μπορεί να έχει οποιονδήποτε απλό τύπο δεδομένων (π.χ. αριθμητική τιμή), ενώ ένας αριθμός στοιχείων Qualifier μπορούν να χρησιμοποιηθούν, προκειμένου να προσδώσουν περαιτέρω σημασιολογική πληροφορία. Για παράδειγμα, στην περίπτωση μιας απαίτησης ως προς το κόστος της υπηρεσίας, το στοιχείο Qualifier μπορεί να χρησιμοποιηθεί για να προσδιορίσει τη νομισματική μονάδα, στην οποία έχει

εκφραστεί η τιμή της απαίτησης, καθώς και το είδος της χρέωσης (π.χ. ανά ώρα, ημέρα, κτλ.).

Όπως φαίνεται στο Σχήμα 8, όλα τα στοιχεία απαιτήσεων πρώτου επιπέδου του μετα-μοντέλου (ServiceCapability, ServiceInterface, ServiceResource και QoS) κληρονομούν το ιδιοχαρακτηριστικό showInResponse της αφαιρετικής κλάσης Projectable. Το ιδιοχαρακτηριστικό αυτό επιτρέπει στο χρήστη να προσδιορίσει ρητά, αν επιθυμεί τη συμπερίληψη πληροφορίας σχετικής με τις αντίστοιχες ιδιότητες της υπηρεσίας, στις εγγραφές των αποτελεσμάτων αναζήτησης. Έτσι, τα περιεχόμενα του εγγράφου απόκρισης διαμορφώνονται με τρόπο δυναμικό, αντικατοπτρίζοντας τις προτιμήσεις του εκάστοτε αιτούντος.

### 3.3.2 Το Έγγραφο Επερωτήσεων USQLRequest

Έχοντας παρουσιάσει τα στοιχεία που προσφέρει το μετα-μοντέλο της γλώσσας για την καταγραφή των απαιτήσεων του χρήστη κατά την αναζήτηση ετερογενών υπηρεσιών, συνεχίζουμε με την παρουσίαση και περιγραφή της δομής του εγγράφου επερωτήσεων USQLRequest.

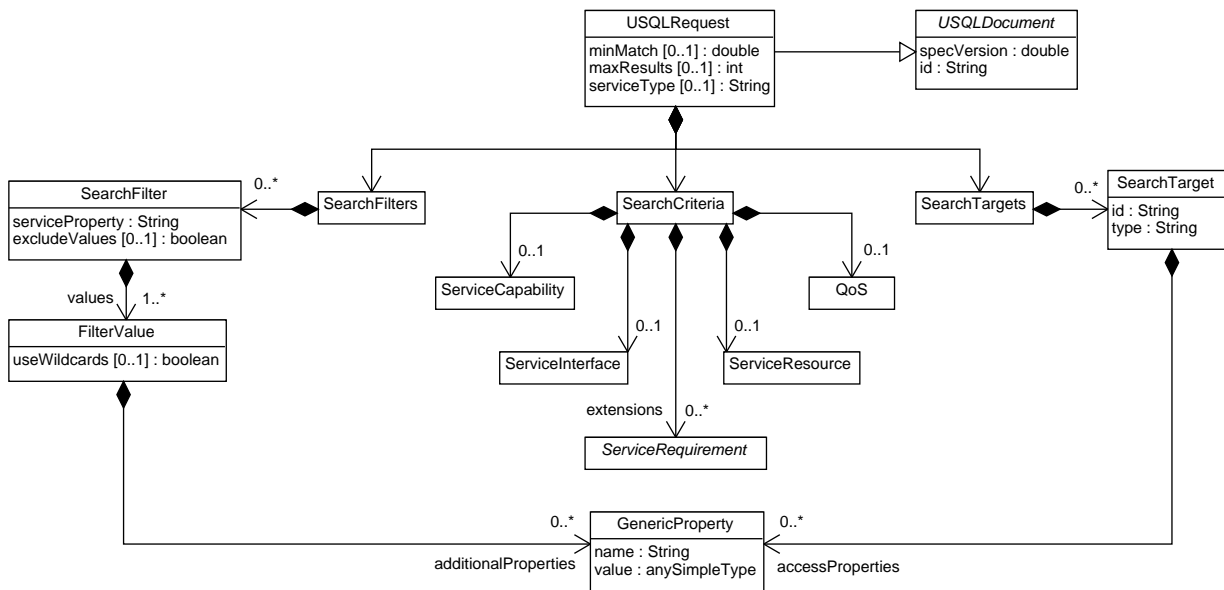
Ο τύπος εγγράφου USQLRequest χρησιμοποιείται για το σχηματισμό ενιαίων επερωτήσεων αναζήτησης ετερογενών υπηρεσιών, οι οποίες (επερωτήσεις) μπορεί να περιλαμβάνουν ποικίλα κριτήρια αναζήτησης (search criteria) προς πολλαπλές λειτουργικές και ποιοτικές ιδιότητες της υπό αναζήτηση υπηρεσίας. Επιπλέον, σε ένα έγγραφο USQLRequest, μπορεί κανείς να προσδιορίσει ρητά τους στόχους της αναζήτησης (search targets), δηλαδή τα διάφορα μητρώα, αποθετήρια και δίκτυα, όπου έχουν δημοσιευτεί οι υπό αναζήτηση υπηρεσίες.

Καθώς επεκτείνει την αφαιρετική κλάση εγγράφων USQLDocument (βλ. Σχήμα 9), κάθε έγγραφο τύπου USQLRequest χαρακτηρίζεται μοναδικά από ένα αναγνωριστικό (id). Επιπροσθέτως, προσφέρει μια σειρά ιδιοχαρακτηριστικών, μέσω των οποίων μπορούν να προσδιοριστούν προαιρετικά:

- Ο μέγιστος επιθυμητός αριθμός αποτελεσμάτων (ιδιοχαρακτηριστικό maxResults)
- Ο ελάχιστος αποδεκτός βαθμός ομοιότητας/καταλληλότητας των επιστρεφόμενων αποτελεσμάτων, ως προς τα διάφορα κριτήρια αναζήτησης (ιδιοχαρακτηριστικό minMatch)
- Ο τύπος της αναζητούμενης υπηρεσίας (ιδιοχαρακτηριστικό serviceType)

Όπως απεικονίζεται στο διάγραμμα του Σχήματος 9, τα περιεχόμενα του εγγράφου USQLRequest χωρίζονται σε τρεις βασικές κατηγορίες, οι οποίες περιγράφονται παρακάτω.

**Φίλτρα Αναζήτησης.** Το στοιχείο SearchFilters περιλαμβάνει τα διάφορα *φίλτρα αναζήτησης*, που ορίζονται με τη χρήση της αφαιρετικής δομής SearchFilter. Ως φίλτρο αναζή-



Σχήμα 9: Δομή του εγγράφου USQLRequest

τησης θεωρείται οποιοσδήποτε περιορισμός, του οποίου η αξιολόγηση έχει ως αποτέλεσμα μια τιμή αλήθειας. Για παράδειγμα, η αξιολόγηση ενός περιορισμού ως προς τον επιθυμητό (ή ανεπιθύμητο) πάροχο της αναζητούμενης υπηρεσίας επιφέρει την τιμή `true`, αν η υπηρεσία προσφέρεται (ή δεν προσφέρεται) από τον προσδιορισμένο πάροχο, ή `false`, σε οποιαδήποτε άλλη περίπτωση. Άλλα είδη περιορισμών-φίλτρων, που απαντώνται στους περισσότερους υπάρχοντες μηχανισμούς αναζήτησης υπηρεσιών, είναι η κατηγορία (classification) της αναζητούμενης υπηρεσίας, το όνομά της κτλ.

Κάθε φίλτρο ορίζει την ιδιότητα της υπηρεσίας, στην οποία εφαρμόζεται, μέσω του ιδιοχαρακτηριστικού `serviceProperty`, ενώ περιλαμβάνει μια ή περισσότερες εναλλακτικές τιμές (FilterValue) για την ιδιότητα αυτή. Οι τιμές αυτές μπορεί να είναι απόλυτες, ή ελαστικές, αναλόγως της τιμής του ιδιοχαρακτηριστικού `wildcards`. Επιπλέον, για κάθε μια τιμή, είναι δυνατός ο προσδιορισμός επιπρόσθετων ιδιοτήτων. Για παράδειγμα, οι τιμές ενός φίλτρου σχετικού με την επιθυμητή κατηγορία της υπό αναζήτηση υπηρεσίας χρειάζεται να ορίσουν το σχήμα κατηγοριοποίησης (classification scheme) (π.χ. NAICS [82], UNSPSC [123], κ.ά.), μέσω μιας επιπρόσθετης ιδιότητας. Τέλος, στην περίπτωση που το ιδιοχαρακτηριστικό `excludeValues` λάβει την τιμή `true`, οι τιμές του φίλτρου εκλαμβάνονται αρνητικά.

**Κριτήρια Αναζήτησης.** Το στοιχείο `SearchCriteria` ομαδοποιεί τα κριτήρια αναζήτησης της επερώτησης, τα οποία λαμβάνουν τη μορφή απαιτήσεων, όπως αυτές ορίζονται από το μετα-μοντέλο της USQL. Έτσι, σε ένα έγγραφο `USQLRequest`, ο χρήστης μπορεί να περιγράψει την επιθυμητή λειτουργικότητα της αναζητούμενης υπηρεσίας, να προδιαγράψει με λεπτομέρειες την επιθυμητή διεπαφή της, τον πόρο στον οποίο η υπηρεσία επενεργεί, καθώς και να καταγράψει τις απαιτήσεις του σχετικά με τα ποιοτικά της χαρακτηριστικά. Το στοιχείο `SearchCriteria` ορίζει επίσης ένα σημείο επέκτασης, μέσω της αφαιρετικής κλά-

σης `ServiceRequirement`. Έτσι, μπορεί κανείς να εισάγει στο μετα-μοντέλο του εγγράφου `USQLRequest` νέες κλάσεις, που αντιστοιχούν σε επιπρόσθετα κριτήρια αναζήτησης, αρκεί να κληρονομούν την κλάση `ServiceRequirement`, ή κάποια από τις υπόλοιπες υποκλάσεις της.

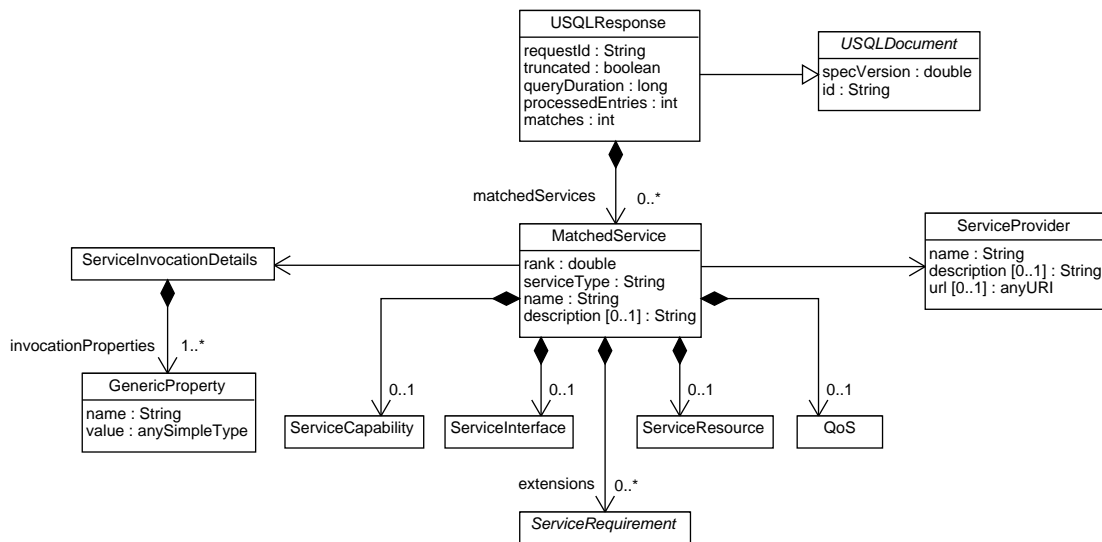
**Στόχοι Αναζήτησης.** Σε πολλές περιπτώσεις, οι αιτούντες μια υπηρεσία επιθυμούν να στοχεύσουν τις επερωτήσεις τους προς συγκεκριμένα, ιδιωτικά μητρώα, αποθετήρια, δίκτυα κτλ., αντί να αναζητήσουν την υπηρεσία στο ανοικτό διαδίκτυο. Η ανάγκη αυτή μπορεί να σχετίζεται για παράδειγμα με περιορισμούς ασφαλείας, που μια εταιρία επιβάλλει στους εργαζομένους της, ή να οφείλεται στην εκ των προτέρων γνώση του αιτούντος, σχετικά με το πού έχει δημοσιευτεί η υπηρεσία. Η δεύτερη περίπτωση απαντάται συχνά στον επιχειρηματικό κόσμο, όπου οι εταιρίες συνάπτουν ειδικές συμφωνίες συνεργασίας σε επίπεδο υπηρεσιών (`Service Level Agreements`, `SLA`), με αποτέλεσμα η αναζήτηση υπηρεσιών να πραγματοποιείται αποκλειστικά στα μητρώα αυτών και των συνεργατών τους.

Για όλους τους παραπάνω λόγους, η `USQL` εισάγει τις κατάλληλες δομές υποστηρίζοντας τον ρητό προσδιορισμό των στόχων αναζήτησης ενός αιτήματος `USQLRequest`. Συγκεκριμένα, το μετα-μοντέλο της γλώσσας ορίζει το στοιχείο `SearchTarget`, η γενική δομή του οποίου επιτρέπει τον προσδιορισμό διαφόρων τύπων μητρώων, αποθετηρίων, καταλόγων, δικτύων κτλ., ως στόχους αναζήτησης. Για κάθε στοιχείο `SearchTarget`, ορίζεται απαραίτητως ο τύπος του (ιδιοχαρακτηριστικό `type`) καθώς και το αναγνωριστικό του (ιδιοχαρακτηριστικό `id`). Επιπλέον, ορίζονται μηδέν ή περισσότερες ιδιότητες πρόσβασης, που περιλαμβάνουν όλες τις απαραίτητες πληροφορίες για την πρόσβαση/σύνδεση με το μητρώο, αποθετήριο, ή δίκτυο, στο οποίο αντιστοιχεί το στοιχείο `SearchTarget`. Οι ιδιότητες αυτές συντάσσονται σύμφωνα με τον αφηρημένο τύπο `GenericProperty`, που ορίζεται από το μετα-μοντέλο της `USQL`. Έτσι, για κάθε προσδιορισμένη ιδιότητα απαιτείται το όνομα και η τιμή της.

### 3.3.3 Το Έγγραφο Αποκρίσεων `USQLResponse`

Ο τύπος εγγράφου `USQLResponse` παρέχει όλες τις απαιτούμενες δομές για τη συγκέντρωση των αποτελεσμάτων της αναζήτησης, ενώ παρέχει επιπλέον πληροφορίες σχετικά με τη διαδικασία εκτέλεσης της αντίστοιχης επερώτησης. Η δομή ενός εγγράφου `USQLResponse`, όπως ορίζεται από το μετα-μοντέλο της `USQL`, απεικονίζεται στο Σχήμα 10.

Όπως και στην περίπτωση των εγγράφων `USQLRequest`, κάθε έγγραφο `USQLResponse` διαθέτει ένα αναγνωριστικό (`id`), ενώ συσχετίζεται με το έγγραφο `USQLRequest`, του οποίου αποτελεί την απόκριση, κάνοντας αναφορά στο αναγνωριστικό του μέσω του ιδιοχαρακτηριστικού `requestId`. Μια σειρά επιπλέον ιδιοχαρακτηριστικών στο έγγραφο `USQLResponse`



Σχήμα 10: Δομή του εγγράφου USQLResponse

προσφέρουν πληροφορίες σχετικά με:

- το συνολικό χρόνο εκτέλεσης της επερώτησης (ιδιοχαρακτηριστικό queryDuration)
- τον αριθμό διαφημίσεων υπηρεσιών, οι οποίες υπέστησαν τη διαδικασία αντιστοίχισης με τα κριτήρια αναζήτησης του χρήστη (ιδιοχαρακτηριστικό processedEntries)
- τον αριθμό αποτελεσμάτων, που περιλαμβάνονται στο έγγραφο USQLResponse (ιδιοχαρακτηριστικό matches)
- το αν τα αποτελέσματα της αναζήτησης έχουν υποστεί αποκοπή (ιδιοχαρακτηριστικό truncated), κατόπιν αιτήσεως του χρήστη (βλ. ιδιοχαρακτηριστικό maxResults του τύπου εγγράφου USQLRequest)

Για κάθε μια λειτουργία υπηρεσίας (service operation) που ικανοποιεί τα κριτήρια αναζήτησης, το έγγραφο USQLResponse περιλαμβάνει το αντίστοιχο στοιχείο MatchedService, στο οποίο αποδίδεται ένας βαθμός αντιστοιχίας (ιδιοχαρακτηριστικό rank), ενώ επιπλέον προσδιορίζεται ο τύπος της αντίστοιχης υπηρεσίας (ιδιοχαρακτηριστικό serviceType). Πέραν τούτων, ένα στοιχείο MatchedService περιέχει πληροφορία που εξυπηρετεί δύο σκοπούς:

1. Πρώτον, προσφέρονται όλα τα απαραίτητα δεδομένα για την κλήση της συγκεκριμένης λειτουργίας υπηρεσίας, έτσι ώστε το έγγραφο USQLResponse να μπορεί να χρησιμοποιηθεί από το κατάλληλο λογισμικό κλήσης υπηρεσιών. Για το σκοπό αυτό, το μετα-μοντέλο της γλώσσας ορίζει τη δομή ServiceInvocationDetails, η οποία αποτελεί στην ουσία μια συλλογή ιδιοτήτων. Η γενικότητα της δομής αυτής επιτρέπει τη συμπερίληψη οποιασδήποτε πληροφορίας κρίνεται απαραίτητη για την κλήση της αντίστοιχης λειτουργίας υπηρεσίας, ανάλογα με τον τύπο της.



```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <USQLRequest specVersion="2.0" id="1" serviceType="WebService" xmlns="http://usql.di.uoa.gr">
3   <SearchFilters>
4     <Filter serviceProperty="ServiceProvider">
5       <Option value="Medisystem" useWildcards="true"></Option>
6     </Filter>
7   </SearchFilters>
8   <SearchCriteria>
9     <Capability ontologyReference="http://medisystem.ro/ehealth.owl#GetPatientMedication"/>
10    <Interface>
11      <Input ontologyReference="http://medisystem.ro/ehealth.owl#GetPatientMedicationRequest">
12        <Element ontologyReference="http://medisystem.ro/ehealth.owl#PatiendCode">
13          <Type namespace="http://www.w3.org/2001/XMLSchema">string</Type>
14        </Element>
15        <Element ontologyReference="http://medisystem.ro/ehealth.owl#EpisodeCode">
16          <Type namespace="http://www.w3.org/2001/XMLSchema">string</Type>
17        </Element>
18      </Input>
19      <Output ontologyReference="http://medisystem.ro/ehealth.owl#GetPatientMedicationResponse">
20        <Element ontologyReference="http://medisystem.ro/ehealth.owl#PatientMedication"/>
21      </Output>
22    </Interface>
23    <QoS>
24      <QoSProperty value="0.9999">
25        <Description>Availability</Description>
26        <Qualifier name="min" value="0"/>
27        <Qualifier name="max" value="1"/>
28      </QoSProperty>
29    </QoS>
30  </SearchCriteria>
31  <SearchTargets>
32    <Target type="UDDI" id="uddi@medisystem">
33      <AccessProperty value="http://medisystem.ro/uddi/inquire" name="inquiryURL"/>
34    </Target>
35  </SearchTargets>
36 </USQLRequest>
```

### Παράδειγμα 1: Επερώτηση για υπηρεσία ιστού

2. Δεύτερον, η πληροφορία περιλαμβάνει λεπτομέρειες όπως ο πάροχος της υπηρεσίας, το όνομα και η περιγραφή της κτλ., οι οποίες προορίζονται κυρίως για χρήστες-ανθρώπους, προκειμένου να διευκολύνουν τη διαδικασία επιλογής. Επιπλέον, ως αποτέλεσμα των εκφρασμένων προτιμήσεων του χρήστη στο αίτημα USQLRequest (βλ. ιδιοχαρακτηριστικό `showInResponse`), το στοιχείο `MatchedService` μπορεί να περιλαμβάνει λεπτομέρειες σχετικά με την προσφερόμενη λειτουργικότητα της υπηρεσίας, τη διεπαφή της, τον πόρο, στον οποίο επενεργεί η αντίστοιχη λειτουργία υπηρεσίας, καθώς και τα διάφορα ποιοτικά της χαρακτηριστικά. Οι λεπτομέρειες αυτές περιγράφονται με τα ίδια στοιχεία του μετα-μοντέλου, που χρησιμοποιούνται για την περιγραφή των αντίστοιχων απαιτήσεων.

## 3.4 Συντακτικό της USQL

Όπως έχει προαναφερθεί, η υλοποίηση της USQL έχει πραγματοποιηθεί με τη χρήση της γλώσσας XML, ενώ βασίζεται εννοιολογικά στο μετα-μοντέλο που παρουσιάστηκε στην προηγούμενη ενότητα. Στην ενότητα αυτή, παρουσιάζουμε το XML συντακτικό της γλώσσας USQL μέσα από μια σειρά παραδειγμάτων. Συγκεκριμένα, χρησιμοποιώντας

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <USQLResponse specVersion="2.0" id="2" requestId="1" truncated="false" queryDuration="4326"
3   processedEntries="129" matches="1" xmlns="http://usql.di.uoa.gr">
4   <Match rank="1.0" serviceType="WebService" name="GetMedication">
5     <Provider name="Medisystem" />
6     <InvocationDetails>
7       <InvocationProperty name="ServiceName" value="GetMedication" />
8       <InvocationProperty name="PortName" value="GetMedicationSOAP" />
9       <InvocationProperty name="OperationName" value="GetAll" />
10      <InvocationProperty name="WSDL" value="http://medisystem.ro/ws/GetMedication.aspx?wsdl" />
11    </InvocationDetails>
12  </Match>
13 </USQLRequest>
```

## Παράδειγμα 2: Απόκριση στην επερώτηση για υπηρεσία ιστού

το σενάριο της παραγράφου 1.1.1, θα σχηματίσουμε τρία αντιπροσωπευτικά έγγραφα USQLRequest, ένα για κάθε υποστηριζόμενο τύπο υπηρεσίας, ενώ θα δείξουμε πώς δομούνται τα αντίστοιχα έγγραφα USQLResponse, ως αποκρίσεις στις τρεις αυτές επερωτήσεις. Έτσι, εκτός από την επίδειξη του συντακτικού της γλώσσας, τα παραδείγματα αυτά θα δείξουν πώς, με τη χρήση μιας κοινής μορφοποίησης, μπορεί κανείς να σχηματίζει επερωτήσεις προς ετερογενείς τύπους υπηρεσιών, με ενιαίο τρόπο.

Για λόγους πληρότητας, ο τυπικός ορισμός του XML μοντέλου της USQL, με τη χρήση της μετα-γλώσσας XML Schema [127], δίνεται στο Παράρτημα Ι.

### 3.4.1 Επερώτηση Αναζήτησης Υπηρεσίας Ιστού

Ξεκινούμε την παρουσίαση του συντακτικού της γλώσσας με το σχηματισμό μιας επερώτησης USQL, για την αναζήτηση μιας υπηρεσίας ιστού. Συγκεκριμένα, η επερώτηση αφορά στην προδιαγεγραμμένη λειτουργία ανάκτησης της φαρμακευτικής αγωγής ενός ασθενούς, στο πλαίσιο της διεργασίας καθορισμού θεραπευτικής αγωγής (βλ. παράγραφο 1.1.1).

Η επερώτηση καταγράφεται στο Παράδειγμα 1. Καθώς η αναζήτηση στοχεύει αποκλειστικά σε υπηρεσίες ιστού, που προσφέρονται από την κλινική, ο τύπος της αναζητούμενης υπηρεσίας έχει προσδιοριστεί ως “WebService” (γραμμή 2), ενώ επιπλέον ο χρήστης έχει ορίσει ένα φίλτρο αναζήτησης ως προς τον πάροχο της υπηρεσίας, χρησιμοποιώντας το όνομα της κλινικής (γραμμές 3-7).

Τα κριτήρια αναζήτησης για τη συγκεκριμένη υπηρεσία αφορούν στη λειτουργικότητά της (γραμμή 9), την διεπαφή της (γραμμές 10-22), καθώς και τα ποιοτικά της χαρακτηριστικά (γραμμές 23-29). Καθώς οι υπηρεσίες ιστού της κλινικής έχουν περιγραφεί σημασιολογικά με τη χρήση μιας κοινής οντολογίας, έχει επιλεχθεί η χρήση του κοινού ιδιοχαρακτηριστικού ontologyReference, για την περιγραφή τόσο της επιθυμητής λειτουργικότητας, όσο και των μηνυμάτων εισόδου/εξόδου της αναζητούμενης υπηρεσίας (βλ. γραμμές 9, 11, 12, 15, 19 και 20). Επιπλέον, για τα δύο στοιχεία του μηνύματος εισόδου, έχει προσδιοριστεί ο επιθυμητός τύπος δεδομένων τους (γραμμές 13 και 16). Συμπληρωματικά προς τις

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <USQLRequest specVersion="2.0" id="3" serviceType="P2PService" xmlns="http://usql.di.uoa.gr">
3   <SearchFilters>
4     <Filter serviceProperty="ServiceProvider">
5       <Option value="urn:jxta:uuid-59616261646162614E5..." />
6     </Filter>
7   </SearchFilters>
8   <SearchCriteria>
9     <Capability showInResponse="true">
10      <Description>Messaging service for medical opinions exchange</Description>
11    </Capability>
12    <QoS>
13      <QoSProperty value="1.0">
14        <Description>Reliability</Description>
15        <Qualifier name="min" value="0" />
16        <Qualifier name="max" value="1" />
17      </QoSProperty>
18    </QoS>
19  </SearchCriteria>
20  <SearchTargets>
21    <Target type="JXTA" id="jxta-subnet@medisystem">
22      <AccessProperty name="peerGroupid" value="urn:jxta:uuid-4d6172676572696e2..." />
23      <AccessProperty name="identity" value="..." />
24      <AccessProperty name="identityPassword" value="..." />
25      <AccessProperty name="keystorePassword" value="..." />
26    </Target>
27  </SearchTargets>
28 </USQLRequest>

```

### Παράδειγμα 3: Επερώτηση για υπηρεσία ομότιμου-προς-ομότιμο

Λειτουργικές αυτές απαιτήσεις, η επερώτηση του παραδείγματος καταγράφει μια ποιοτική απαίτηση σχετικά με τη διαθεσιμότητα της υπηρεσίας (γραμμές 24-28), ορίζοντας την επιθυμητή ελάχιστη τιμή, 0.9999, σε ένα εύρος τιμών από 0 έως 1. Έτσι, σε περίπτωση που η κλινική έχει αναπτύξει τη συγκεκριμένη υπηρεσία σε περισσότερους από έναν διακομιστές, θα διευκολυνθεί η επιλογή μιας εξ αυτών από τα αποτελέσματα της αναζήτησης.

Τέλος, με τη χρήση του στοιχείου Target, η επερώτηση προσδιορίζει ως στόχο αναζήτησης το ιδιωτικό μητρώο UDDI της κλινικής (γραμμές 31-35).

Το έγγραφο USQLResponse, που αποτελεί απόκριση στην επερώτηση του παραδείγματος, απεικονίζεται στο Παράδειγμα 2. Χάριν συντομίας, τα αποτελέσματα αναζήτησης περιλαμβάνουν μία μόνο εγγραφή (στοιχείο Match), η οποία αντιστοιχεί σε μια συγκεκριμένη λειτουργία της υπηρεσίας ιστού GetMedication (γραμμές 4-12). Η εγγραφή αυτή προσδιορίζει τον τύπο και το όνομα της υπηρεσίας, καθώς και το βαθμό αντιστοίχισης με τα κριτήρια αναζήτησης της επερώτησης (γραμμή 4). Επιπλέον, περιλαμβάνει πληροφορίες σχετικά με τον πάροχο της υπηρεσίας (γραμμή 5), ενώ παρέχει τις απαραίτητες λεπτομέρειες για την κλήση της συγκεκριμένης λειτουργίας (γραμμές 6-11). Παρατηρούμε ότι, για την περίπτωση υπηρεσιών ιστού, οι λεπτομέρειες αυτές συνίστανται στο όνομα της υπηρεσίας, στο όνομα της θύρας (port), μέσω της οποίας είναι προσβάσιμη, στο όνομα της λειτουργίας, όπως και στη διεύθυνση URL του εγγράφου WSDL, που περιγράφει την υλοποίηση της υπηρεσίας. Τέλος, σε συμφωνία με το μετα-μοντέλο της γλώσσας, το έγγραφο USQLResponse δίνει επιπρόσθετες πληροφορίες σχετικά με την εκτέλεση της επερώτησης, όπως αυτές καταγράφονται στις γραμμές 2 και 3 του Παραδείγματος 2.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <USQLResponse specVersion="2.0" matches="1" id="4" requestId="3"
3   truncated="false" processedEntries="26" queryDuration="1784" xmlns="http://usql.di.uoa.gr">
4   <Match serviceType="P2PService" name="MedMessenger" rank="0.736">
5     <Provider name="..." />
6     <InvocationDetails>
7       <InvocationProperty name="NetworkType" value="JXTA" />
8       <InvocationProperty name="PipeType" value="JxtaUnicast" />
9       <InvocationProperty name="PipeId" value="urn:jxta:uuid-9CCCDF5AD8..." />
10    </InvocationDetails>
11    <Capability>
12      <Description>Messenger for exchange of medical diagnoses</Description>
13    </Capability>
14  </Match>
15 </USQLResponse>
```

Παράδειγμα 4: Απόκριση στην επερώτηση για υπηρεσία ομότιμου-προς-ομότιμο

### 3.4.2 Επερώτηση Αναζήτησης Υπηρεσίας Ομότιμου-προς-Ομότιμο

Συνεχίζουμε την παρουσίαση του συντακτικού της γλώσσας USQL με τη δημιουργία ενός εγγράφου USQLRequest, για την αναζήτηση μιας υπηρεσίας ομότιμου-προς-ομότιμο. Όπως και στην προηγούμενη περίπτωση αναζήτησης υπηρεσίας ιστού, το παράδειγμά μας βασίζεται στο σενάριο της παραγράφου 1.1.1 και συγκεκριμένα αφορά στην απαιτούμενη λειτουργικότητα επικοινωνίας μεταξύ των ιατρών της κλινικής, για την ανταλλαγή απόψεων σχετικά με κάποιο ιατρικό περιστατικό.

Η επερώτηση για μια τέτοια υπηρεσία ομότιμου-προς-ομότιμο, προσφερόμενη από κόμβους ενός δικτύου τύπου JXTA, καταγράφεται στο Παράδειγμα 3. Για την αποφυγή αποτελεσμάτων που δεν αντιστοιχούν σε υπηρεσίες ομότιμου-προς-ομότιμο, ο τύπος της αναζητούμενης υπηρεσίας έχει προσδιοριστεί ρητά ως "P2PService" (γραμμή 2). Όπως και στην επερώτηση του Παραδείγματος 1, το έγγραφο USQLRequest περιλαμβάνει ένα φίλτρο αναζήτησης σχετικά με τον πάροχο της υπηρεσίας (γραμμές 3-7). Τη φορά αυτή ωστόσο, ο επιθυμητός πάροχος αντιστοιχεί σε κάποιον κόμβο του δικτύου ομότιμων κόμβων και προσδιορίζεται με τη χρήση του αναγνωριστικού του στο δίκτυο αυτό.

Τα κριτήρια αναζήτησης της επερώτησης περιλαμβάνουν απαιτήσεις σχετικά με τη λειτουργικότητα και αξιοπιστία της υπηρεσίας (γραμμές 8-19). Η επιθυμητή λειτουργικότητα έχει εκφραστεί με τη χρήση απλού κειμένου, ενώ επιπλέον ο αιτών έχει ζητήσει τη συμπερίληψη της περιγραφής της αντίστοιχης ιδιότητας, για κάθε υπηρεσία στα αποτελέσματα αναζήτησης (γραμμές 9-11). Εξάλλου, ο επιθυμητός βαθμός αξιοπιστίας έχει περιγραφεί παρόμοια με το βαθμό διαθεσιμότητας για την υπό αναζήτηση υπηρεσία ιστού του Παραδείγματος 1.

Εκτός από τα διάφορα κριτήρια αναζήτησης, στην επερώτηση έχει προσδιοριστεί κι ένας στόχος αναζήτησης. Συγκεκριμένα, με τη χρήση της γενικής δομής Target, έχει περιγραφεί ένα υποδίκτυο ομότιμων κόμβων τύπου JXTA. Οι απαραίτητες πληροφορίες για την απόκτηση πρόσβασης σε αυτό αφορούν στο αναγνωριστικό του υποδικτύου, καθώς και σε μια σειρά στοιχείων που απαιτούνται για την ταυτοποίηση σε αυτό (γραμμές 21-26).

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <USQLRequest specVersion="2.0" id="5" minMatch="0.5" serviceType="GridService"
3   xmlns="http://usql.di.uoa.gr">
4   <SearchFilters/>
5   <SearchCriteria>
6     <Capability weight="3">
7       <Description>Get resource property</Description>
8     </Capability>
9     <Resource weight="2">
10      <Description>Drug</Description>
11      <Property>
12        <Description>Manufacturer</Description>
13        <Type namespace="http://www.w3.org/2001/XMLSchema">string</Type>
14      </Property>
15      <Property>
16        <Description>ATC Code</Description>
17        <Type namespace="http://www.w3.org/2001/XMLSchema">string</Type>
18      </Property>
19      <Property>
20        <Description>Statistics Information</Description>
21        <Type namespace="http://www.w3.org/2001/XMLSchema">string</Type>
22      </Property>
23    </Resource>
24    <QoS weight="1">
25      <QoSProperty value="5.0">
26        <Description>Cost</Description>
27        <Qualifier name="Currency" value="EUR"/>
28        <Qualifier name="Unit" value="perCall"/>
29      </QoSProperty>
30    </QoS>
31  </SearchCriteria>
32  <SearchTargets>
33    <Target type="Globus" id="caGrid">
34      <AccessProperty name="url" value="http://localhost:8080/wsrf/services/cagrid/DataService"/>
35    </Target>
36  </SearchTargets>
37 </USQLRequest>
```

### Παράδειγμα 5: Επερώτηση για υπηρεσία πλέγματος

Το έγγραφο USQLResponse, που αντιστοιχεί στην επερώτηση, δίνεται στο Παράδειγμα 4. Ως αποτέλεσμα της αναζήτησης εμφανίζεται μια εγγραφή με βαθμό 0.736 (γραμμή 3), η οποία αντιστοιχεί σε μια υπηρεσία ομότιμου-προς-ομότιμο, που προσφέρεται από ένα δίκτυο ομότιμων κόμβων τύπου JXTA. Οι λεπτομέρειες κλήσης της υπηρεσίας (γραμμές 6-10) αφορούν σε στοιχεία της τεχνολογίας JXTA και σχετίζονται με την εγκαθίδρυση του καναλιού επικοινωνίας με τον κόμβο-πάροχο της υπηρεσίας.

### 3.4.3 Επερώτηση Αναζήτησης Υπηρεσίας Πλέγματος

Ολοκληρώνουμε την παρουσίαση του συντακτικού της γλώσσας USQL με τη δημιουργία μιας επερώτησης για υπηρεσία πλέγματος. Η αναζητούμενη υπηρεσία σχετίζεται με τη λήψη στατιστικών πληροφοριών για ένα φάρμακο, λειτουργικότητα που απαιτείται από τη διεργασία του σεναρίου, που περιγράφηκε στην παράγραφο 1.1.1.

Η επερώτηση για τη συγκεκριμένη υπηρεσία πλέγματος δίνεται στο Παράδειγμα 5. Τα κριτήρια αναζήτησης της επερώτησης περιλαμβάνουν απαιτήσεις σχετικά με την επιθυμητή λειτουργικότητα, αλλά και τον πόρο, τον οποίο η υπηρεσία διαχειρίζεται. Όπως φαίνεται στον κώδικα του παραδείγματος (γραμμές 9-23), ο πόρος αφορά σε φαρμακευτικά

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <USQLResponse specVersion="2.0" id="6" requestId="5" truncated="false"
3   processedEntries="23" queryDuration="2784" matches="1" xmlns="http://usql.di.uoa.gr">
4   <Match serviceType="GridService" name="GetResourceProperty" rank="0.7853">
5     <Provider name="..." />
6     <InvocationDetails>
7       <InvocationProperty name="ServiceName" value="DrugInfoService" />
8       <InvocationProperty name="PortName" value="DrugInfoSoap" />
9       <InvocationProperty name="OperationName" value="GetResourceProperty" />
10      <InvocationProperty name="WSDL" value="..." />
11    </InvocationDetails>
12  </Match>
13 </USQLResponse>
```

### Παράδειγμα 6: Απόκριση στην επερώτηση για υπηρεσία πλέγματος

στατιστικά στοιχεία, ενώ έχουν προσδιοριστεί τρεις σχετικές ιδιότητες.

Όπως και στα προηγούμενα παραδείγματα, η επερώτηση υπαγορεύει το στόχο της αναζήτησης (γραμμές 32-36), ενώ, εκτός από τις ανωτέρω λειτουργικές απαιτήσεις, περιλαμβάνει και ένα κριτήριο αναζήτησης σχετικά με το μέγιστο επιθυμητό κόστος ανά κλήση της συγκεκριμένης υπηρεσίας (γραμμές 24-30). Τέλος, παρατηρούμε ότι, στα διάφορα κριτήρια αναζήτησης έχουν αποδοθεί διαφορετικά βάρη, σηματοδοτώντας τις προτεραιότητες του χρήστη. Έτσι, η ικανοποίηση της απαίτησης σχετικά με τη λειτουργικότητα της υπηρεσίας έχει ύψιστη προτεραιότητα, ενώ ακολουθεί η απαίτηση σχετικά με τον διαχειριζόμενο πόρο. Το κόστος έχει μικρότερη σημασία για τον αιτούντα, αν οι προηγούμενες δύο απαιτήσεις ικανοποιούνται από κάποια υπηρεσία.

Μια ενδεικτική απόκριση στην επερώτηση για υπηρεσία πλέγματος δίνεται στο Παράδειγμα 6. Καθώς η υπηρεσία που βρέθηκε ακολουθεί τις προδιαγραφές του πλαισίου WSRF, οι λεπτομέρειες κλήσης της (γραμμές 6-11) είναι ίδιες με αυτές που απαιτούνται για την κλήση μιας υπηρεσίας ιστού.

## 3.5 Μαθηματικό Μοντέλο της USQL

Στις προηγούμενες ενότητες, δείξαμε πώς η γλώσσα USQL υποστηρίζει τη σύνταξη επερωτήσεων, που περιλαμβάνουν κριτήρια σχετικά με διάφορες λειτουργικές ή/και ποιοτικές ιδιότητες της αναζητούμενης υπηρεσίας. Κατά την αναζήτηση υπηρεσιών που ανταποκρίνονται στις προσδιορισμένες αυτές απαιτήσεις, οι επιθυμητές τιμές για τις διάφορες ιδιότητες πρέπει να συγκριθούν με αυτές που ορίζονται στις αντίστοιχες περιγραφές των υπηρεσιών. Έτσι, η επεξεργασία ενός εγγράφου USQLRequest και η αξιολόγηση μιας υπηρεσίας ως προς τα διάφορα κριτήρια αναζήτησης απαιτούν την ύπαρξη μοντέλων για την αντιστοίχιση (matchmaking) ποικίλων ειδών τιμών, καθώς και για την εκτίμηση του συνολικού βαθμού αντιστοιχίας (degree of match) της υπηρεσίας. Με άλλα λόγια, για να καταστεί η γλώσσα USQL εφαρμόσιμη στο πεδίο της αναζήτησης υπηρεσιών, απαιτείται κατάλληλη αλγοριθμική υποστήριξη κατά την επεξεργασία των επερωτήσεών της.

Τα τελευταία χρόνια, έχει υπάρξει ένας μεγάλος αριθμός ερευνητικών προσπαθειών

επίλυσης του προβλήματος αντιστοίχισης και μέτρησης ομοιότητας (μετα)δεδομένων. Με την εμφάνιση των υπηρεσιοστρεφών τεχνολογιών και την ανάγκη για αναζήτηση και αποδοτική αντιστοίχιση υπηρεσιών βάσει συγκεκριμένων κριτηρίων, πολλές από τις υπάρχουσες προσεγγίσεις προσαρμόστηκαν και εφαρμόστηκαν στη συγκεκριμένη περιοχή, ενώ ιδιαίτερη προσοχή δόθηκε στην ανάπτυξη μεθόδων για την εκτίμηση της σημασιολογικής ομοιότητας υπηρεσιών, βάσει σημασιολογικών εννοιών και ιδιοτήτων ορισμένων σε οντολογίες (π.χ. [72], [17]). Οι Klein και Bernstein [67] πρότειναν μια προσέγγιση ανάκτησης υπηρεσιών, σύμφωνα με την οποία η σημασιολογία μιας υπηρεσίας καταγράφεται σε μοντέλα διεργασιών, στα οποία εφαρμόζεται ένας αλγόριθμος αναγνώρισης προτύπων για τον εντοπισμό των επιθυμητών υπηρεσιών. Μια ακόμα βασισμένη στη χρήση οντολογιών προσέγγιση προτάθηκε από τους Ribeiro et al. [99], η οποία λαμβάνει υπόψιν τις διαφορετικές προτεραιότητες των κριτηρίων αναζήτησης, όπως έχουν οριστεί από το χρήστη.

Ακολουθώντας μια διαφορετική κατεύθυνση, η μηχανή αναζήτησης Woogie [37] υλοποιεί ένα σύνολο αλγορίθμων, που επιτρέπουν την αναζήτηση λειτουργιών υπηρεσιών ιστού, οι οποίες είναι παρόμοιες (συμπληρωματικές) με μια δεδομένη λειτουργία. Η βασική ιδέα των αλγορίθμων αυτών έγκειται στην ομαδοποίηση των μηνυμάτων εισόδου/εξόδου των διαφόρων λειτουργιών υπηρεσιών ιστού με τη χρήση σημασιολογικών εννοιών. Κατ' αυτόν τον τρόπο, η συντακτική πληροφορία που συνήθως εμπεριέχεται στις περιγραφές υπηρεσιών αποκτά σημασιολογία, με αποτέλεσμα την καλύτερη εκμετάλλευσή της από τη μηχανή αναζήτησης. Διάφορες εργασίες έχουν επίσης εμφανιστεί τα τελευταία χρόνια, οι οποίες επικεντρώνονται στην αλγοριθμική υποστήριξη για την αντιστοίχιση τιμών ποιοτικών χαρακτηριστικών υπηρεσιών. Για παράδειγμα, οι Taher et al. [114] παρουσίασαν ένα πλαίσιο, το οποίο αντιστοιχίζει τις ποιοτικές απαιτήσεις των καταναλωτών υπηρεσιών στις δημοσιευμένες περιγραφές των ποιοτικών χαρακτηριστικών των υπηρεσιών, λαμβάνοντας επίσης υπόψιν ποιοτικούς περιορισμούς.

Οι αλγόριθμοι όλων των προαναφερθεισών προσεγγίσεων προσφέρουν υποστήριξη για αντιστοίχιση είτε λειτουργικών είτε μη λειτουργικών κριτηρίων με τις αντίστοιχες ιδιότητες μιας υπηρεσίας. Ακόμα και στην περίπτωση εργασιών που υποστηρίζουν αντιστοίχιση περισσότερων κατηγοριών κριτηρίων, όπως π.χ. το σύστημα LARKS [113], παρατηρούμε ότι οι προτεινόμενοι αλγόριθμοι είναι τις περισσότερες φορές στενά συνδεδεμένοι με συγκεκριμένα σχήματα και μορφές περιγραφών υπηρεσιών. Ως εκ τούτου, η προσαρμογή τους σε ένα νέο φορμαλισμό για την περιγραφή αιτημάτων αναζήτησης υπηρεσιών, όπως η USQL, καθίσταται ιδιαίτερα δύσκολη έως αδύνατη.

Για να καλύψουμε τις ανάγκες της δικής μας προσέγγισης, σε ό,τι αφορά την αλγοριθμική υποστήριξη αντιστοίχισης υπηρεσιών, παρουσιάζουμε στην ενότητα αυτή ένα μαθηματικό μοντέλο, το οποίο, έχοντας ως βάση το μετα-μοντέλο της γλώσσας USQL, καθορίζει τον τρόπο υπολογισμού του βαθμού αντιστοίχισης μιας υπηρεσίας, με τα κριτήρια αναζήτησης μιας επερώτησης USQLRequest. Όπως θα δούμε στη συνέχεια, το μοντέλο

χαρακτηρίζεται από ευελιξία, καθώς διατηρεί την ανεξαρτησία του από τους επιμέρους αλγόριθμους αντιστοίχισης διαφόρων ειδών τιμών, όπως κείμενο, οντολογικές έννοιες, κτλ.

### 3.5.1 Βασικοί Ορισμοί

Ξεκινούμε την παρουσίαση του μαθηματικού μοντέλου της USQL παραθέτοντας μια σειρά βασικών ορισμών. Αρχικά, προσδιορίζουμε μαθηματικά τα σύνολα απαιτήσεων και ιδιοτήτων υπηρεσιών. Ένα σύνολο απαιτήσεων περιέχει ένα ή περισσότερα στοιχεία απαιτήσεων, όπως αυτά ορίζονται από το μετα-μοντέλο της USQL. Ένα σύνολο ιδιοτήτων υπηρεσίας περιέχει τις ιδιότητες μιας υπηρεσίας, που αντιστοιχούν στα στοιχεία απαιτήσεων.

**Ορισμός 3.1 (Σύνολο Απαιτήσεων)** Έστω  $Q = \{r_i\}, 1 \leq i \leq n$ , ένα διατεταγμένο σύνολο  $n > 0$  στοιχείων απαιτήσεων προς μια υπηρεσία και  $W = \{w_i\}, 1 \leq i \leq k$ , τα  $k$  διαφορετικά βάρη που έχουν αποδοθεί στα στοιχεία αυτά, όπου  $1 \leq k \leq n$ ,  $w_i \in \mathbb{R}^+$  και  $w_i < w_j \forall i, j : 1 \leq i < j \leq k$ . Τότε,  $Q \equiv Q_1 \cup \dots \cup Q_k$ , όπου, για  $1 \leq i \leq k$ , το κάθε υποσύνολο  $Q_i \subseteq Q$  περιλαμβάνει τα  $n_i \leq n$  στοιχεία απαιτήσεων του  $Q$ , στα οποία έχει αποδοθεί βάρος  $w_i$ . Επιπλέον, ισχύει ότι  $Q_i \cap Q_j = \emptyset, \forall i, j : i \neq j, 1 \leq i < j \leq k$  και  $\sum_{i=1}^k n_i = n$ .

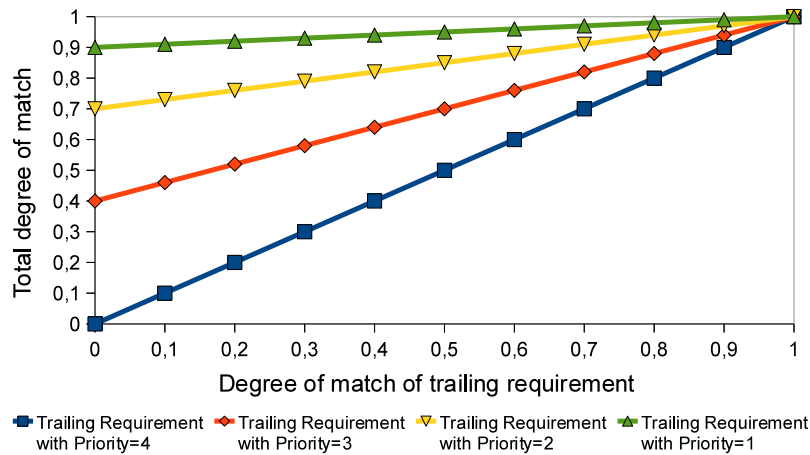
**Ορισμός 3.2 (Σύνολο Ιδιοτήτων)** Έστω  $A = \{p_i\}, 1 \leq i \leq n$ , ένα διατεταγμένο σύνολο  $n > 0$  στοιχείων ιδιοτήτων μιας υπηρεσίας, που αντιστοιχούν στα  $n > 0$  στοιχεία απαιτήσεων του συνόλου απαιτήσεων, όπως αυτό δίνεται στον Ορισμό 3.1. Τότε, με αντίστοιχο τρόπο, έχουμε ότι  $A \equiv A_1 \cup \dots \cup A_k$ , όπου, για  $1 \leq i \leq k$ , το κάθε υποσύνολο  $A_i \subseteq A$  περιλαμβάνει τα  $n_i \leq n$  στοιχεία ιδιοτήτων του  $A$ , τα οποία αντιστοιχούν στα  $n_i \leq n$  στοιχεία απαιτήσεων του υποσυνόλου  $Q_i \subseteq Q$ . Επιπλέον, ισχύει ότι  $A_i \cap A_j = \emptyset, \forall i, j : i \neq j, 1 \leq i < j \leq k$  και  $\sum_{i=1}^k n_i = n$ .

Έχοντας ορίσει τα σύνολα απαιτήσεων και ιδιοτήτων υπηρεσιών, χρειαζόμαστε τις κατάλληλες συναρτήσεις για την αντιστοίχιση των περιεχομένων τους. Ο ακόλουθος ορισμός προσδιορίζει το σύνολο των συναρτήσεων αυτών.

**Ορισμός 3.3 (Συναρτήσεις Αντιστοίχισης)** Έστω ένα σύνολο  $n > 0$  απαιτήσεων προς μια υπηρεσία,  $Q$ , κι ένα σύνολο με τις αντίστοιχες  $n > 0$  ιδιότητες υπηρεσίας,  $A$ . Για την αντιστοίχιση των περιεχομένων των δύο αυτών συνόλων, ορίζουμε την οικογένεια συναρτήσεων αντιστοίχισης,  $M = m : Q^n \times A^n \mapsto [0, 1]$ .

Οι συναρτήσεις αντιστοίχισης του Ορισμού 3.3 χρησιμοποιούνται συνδυαστικά για τον υπολογισμό του συνολικού βαθμού αντιστοιχίας μεταξύ των συνόλων  $Q$  και  $A$ , σύμφωνα με τον επόμενο ορισμό.





Σχήμα 11: Γραφική απεικόνιση της συμπεριφοράς του Υπολογιστή Αντιστοιχίας

**Ορισμός 3.4 (Υπολογιστής Αντιστοιχίας)** Έστω ένα σύνολο  $n > 0$  απαιτήσεων προς μια υπηρεσία,  $Q$ , κι ένα σύνολο με τις αντίστοιχες  $n > 0$  ιδιότητες υπηρεσίας,  $A$ , όπως αυτά δίνονται στους Ορισμούς 3.1 και 3.2, αντίστοιχα. Τότε, ο συνολικός βαθμός αντιστοιχίας τους,  $d$ , δίνεται από την παρακάτω εξίσωση:

$$d = \sum_{i=1}^k (w'_i \cdot \prod_{j=i}^k m_j(Q_j, A_j)) \quad (1)$$

όπου, για  $1 \leq i \leq j \leq k$ , έχουμε  $w'_i = \frac{w_i}{\sum_{i=1}^k w_i}$  και  $m_j \in M : m_j(Q_j, A_j) = \frac{\sum_{l=1}^{n_j} m_l(r_l, p_l)}{n_j}$ ,  $m_l \in M$ ,  $r_l \in Q_j$  και  $p_l \in A_j$ .

Όπως μπορεί εύκολα να αποδειχθεί, η εξίσωση 1 παράγει τιμές στο διάστημα  $[0,1]$ , ενώ βασίζεται στις ακόλουθες διαισθητικές υποθέσεις:

- Οι απαιτήσεις ενός χρήστη σχετικά τα επιθυμητά λειτουργικά ή/και ποιοτικά χαρακτηριστικά της αναζητούμενης υπηρεσίας συνήθως έχουν διαφορετικές προτεραιότητες, οι οποίες μπορεί να δηλώνονται ρητά ή να υπονοούνται από τη φύση των ίδιων των απαιτήσεων. Κατά συνέπεια, η ικανοποίηση των απαιτήσεων με τη μεγαλύτερη προτεραιότητα είναι πιο σημαντική από την ικανοποίηση απαιτήσεων με μικρότερη βαρύτητα. Για παράδειγμα, μια υπηρεσία, που προσφέρει την ίδια λειτουργικότητα με αυτή που αιτήθηκε ο χρήστης, αλλά δεν ικανοποιεί τις ποιοτικές απαιτήσεις του, είναι πιο κατάλληλη/σχετική από μια άλλη υπηρεσία, η οποία ικανοποιεί μόνο τα διάφορα ποιοτικά κριτήρια αναζήτησης του χρήστη.
- Όσο πιο σημαντική θεωρείται μια απαίτηση στην επερώτηση αναζήτησης, τόσο περισσότερο επηρεάζει ο βαθμός αντιστοιχίας της το συνολικό βαθμό καταλληλότητας μιας υποψήφιας προς επιλογή υπηρεσίας.

Η συμπεριφορά της εξίσωσης 1 βάσει των διαφορετικών προτεραιοτήτων απεικονίζεται στο διάγραμμα του Σχήματος 11. Το διάγραμμα καταγράφει τα αποτελέσματα εφαρμογής της εξίσωσης, στην περίπτωση που έχουμε τέσσερα στοιχεία απαιτήσεων με τέσσερα διαφορετικά βάρη. Καθεμία από τις τέσσερις γραφικές παραστάσεις αναπαριστά το συνολικό βαθμό αντιστοιχίας στην περίπτωση που, κρατώντας τους βαθμούς αντιστοιχίας των τριών από τα τέσσερα στοιχεία απαιτήσεων σταθερούς και ίσους με 1 (ή ισοδύναμα 100%), αναθέσουμε στο εναπομείναν στοιχείο απαίτησης, με βάρος  $w=1, 2, 3, 4$ , βαθμό αντιστοίχισης ίσο με  $0, 0.01, 0.02, \dots, 1$  (ή ισοδύναμα  $0, 1, 2, \dots, 100\%$ ).

Έτσι, παρατηρούμε ότι καθοριστικός παράγοντας στην διαμόρφωση της τιμής του συνολικού βαθμού αντιστοιχίας είναι ο επιμέρους βαθμός αντιστοιχίας του στοιχείου απαίτησης με τη μεγαλύτερη προτεραιότητα. Για παράδειγμα, βλέπουμε ότι ακόμα κι όταν το κριτήριο με τη χαμηλότερη προτεραιότητα (ίση με 1) λαμβάνει βαθμό αντιστοιχίας ίσο με 0, ο συνολικός βαθμός αντιστοιχίας επηρεάζεται ελάχιστα, λαμβάνοντας την τιμή 0.9. Ακόμα και όταν το κριτήριο με τη δεύτερη μεγαλύτερη προτεραιότητα λαμβάνει βαθμό αντιστοιχίας ίσο με 0, ο συνολικός βαθμός αντιστοιχίας δεν πέφτει κάτω από 0.4, δεδομένου ότι το κριτήριο με τη μεγαλύτερη προτεραιότητα έχει βαθμό αντιστοιχίας ίσο με 1. Με άλλα λόγια επιβεβαιώνεται ο διαισθητικός κανόνας, στον οποίο βασίστηκε ο Υπολογιστής Αντιστοιχίας, ότι, όσο μεγαλύτερη προτεραιότητα έχει ένα κριτήριο αναζήτησης, τόσο περισσότερο ο βαθμός αντιστοιχίας του επηρεάζει το συνολικό βαθμό αντιστοιχίας.

Ο Υπολογιστής Αντιστοιχίας είναι κατάλληλος για την αξιολόγηση οποιουδήποτε είδους κριτηρίου αναζήτησης αποτελείται από περισσότερα του ενός επιμέρους κριτήρια, στα οποία έχουν αποδοθεί διαφορετικά βάρη. Ως εκ τούτου, χρησιμοποιείται για τον υπολογισμό του συνολικού βαθμού αντιστοιχίας μιας υπηρεσίας με τα κριτήρια αναζήτησης ενός αιτήματος USQLRequest. Πράγματι, οι βαθμοί αντιστοιχίας των κριτηρίων αναζήτησης ServiceCapability, ServiceInterface, ServiceResource, QoS, καθώς και οποιουδήποτε άλλου στοιχείου απαίτησης τύπου ServiceRequirement συμπεριλαμβάνεται στα κριτήρια αναζήτησης της επερώτησης, συνδυάζονται λαμβάνοντας υπόψη τα βάρη τους από την εξίσωση 1, προκειμένου να υπολογιστεί ο συνολικός βαθμός αντιστοιχίας μιας υπηρεσίας.

Άλλωστε, όπως θα δούμε στη συνέχεια, ο Υπολογιστής Αντιστοιχίας χρησιμοποιείται επίσης, για τον υπολογισμό του βαθμού αντιστοιχίας των επιμέρους κριτηρίων αναζήτησης ServiceInterface και QoS.

### 3.5.2 Αντιστοίχιση Κριτηρίων Αναζήτησης

Έχοντας ως βάση τους Ορισμούς 3.1 - 3.4, το μαθηματικό μας μοντέλο ορίζει μια σειρά εξισώσεων για την αντιστοίχιση των επιμέρους κριτηρίων αναζήτησης, που υποστηρίζονται από το μετα-μοντέλο της USQL.

**Ορισμός 3.5 (Αντιστοίχιση περιγραφόμενων απαιτήσεων)** Έστω  $r = (v_o^r, v_d^r) \in Q$  ένα στοιχείο απαίτησης τύπου *DescribedServiceRequirement*, όπου  $v_o^r$  είναι η τιμή του ιδιο-χαρακτηριστικού *ontologyReference* και  $v_d^r$  είναι η τιμή του ιδιοχαρακτηριστικού *description*. Δεδομένου ενός αντίστοιχου στοιχείου ιδιότητας υπηρεσίας,  $p = (v_o^p, v_d^p) \in A$ , ο βαθμός αντιστοιχίας τους υπολογίζεται με τη χρήση της ακόλουθης συνάρτησης,  $\phi \in M$ :

$$\phi(r, p) = \max(\alpha(v_o^r, v_o^p), \beta(v_d^r, v_d^p)) \quad (2)$$

όπου  $\alpha$  και  $\beta$  είναι δύο συναρτήσεις αντιστοίχισης τιμών κειμένου και τιμών οντολογικών εννοιών, αντίστοιχα, ορισμένες εκτός του μαθηματικού μας μοντέλου και με σύνολο τιμών το διάστημα  $[0, 1]$ .

**Ορισμός 3.6 (Αντιστοίχιση περιγραφόμενων απαιτήσεων με τύπο)** Έστω  $r = (v_o^r, v_d^r, v_t^r) \in Q$  ένα στοιχείο απαίτησης τύπου *DescribedAndTypedServiceRequirement*, όπου  $v_o^r$  είναι η τιμή του ιδιοχαρακτηριστικού *ontologyReference*,  $v_d^r$  είναι η τιμή του ιδιοχαρακτηριστικού *description* και  $v_t^r$  η τιμή του τύπου δεδομένων του, που αντιστοιχεί στην κλάση *DataType*. Δεδομένου ενός αντίστοιχου στοιχείου ιδιότητας υπηρεσίας,  $p = (v_o^p, v_d^p, v_t^p) \in A$ , ο βαθμός αντιστοιχίας τους υπολογίζεται με τη χρήση της ακόλουθης συνάρτησης,  $\psi \in M$ :

$$\psi(r, p) = \frac{1}{2} \cdot (\phi(r, p) + \gamma(v_t^r, v_t^p)) \quad (3)$$

όπου  $\gamma$  είναι μια συνάρτηση ορισμένη εκτός του μαθηματικού μας μοντέλου με σύνολο τιμών το διάστημα  $[0, 1]$ , κατάλληλη για την αντιστοίχιση τιμών τύπων δεδομένων.

**Αντιστοίχιση στοιχείων *ServiceCapability*.** Σύμφωνα με το μετα-μοντέλο της USQL, το στοιχείο απαίτησης *Capability* έχει ως τύπο την αφαιρετική κλάση *DescribedServiceRequirement*. Συνεπώς, ο υπολογισμός του βαθμού αντιστοιχίας του είναι άμεσος και βασίζεται στην εφαρμογή της εξίσωσης 2.

**Ορισμός 3.7 (Αντιστοίχιση στοιχείων *Capability*)** Έστω  $r_{capability}$  ένα στοιχείο απαίτησης τύπου *Capability* και  $p_{capability}$  το στοιχείο που περιγράφει την αντίστοιχη ιδιότητα μιας υπηρεσίας. Τότε, ο βαθμός αντιστοιχίας τους,  $d_{capability}$  υπολογίζεται ως εξής:

$$d_{capability} = \phi(r_{capability}, p_{capability}) \quad (4)$$

**Αντιστοίχιση στοιχείων *Interface*.** Για τον υπολογισμό του βαθμού αντιστοιχίας μιας διαφημιζόμενης διεπαφής υπηρεσίας με την αιτούμενη διεπαφή στο έγγραφο USQLRequest, λαμβάνουμε υπόψιν τους ακόλουθους παράγοντες:

- Πρώτον, συγκρίνουμε τα μηνύματα εισόδου/εξόδου ως προς τη σημασιολογία τους και υπολογίζουμε το σημασιολογικό βαθμό αντιστοιχίας,  $d_{sem}$ , αναφορικά με το τί αντιπροσωπεύει η πληροφορία που η υπηρεσία ανταλλάσσει μέσω αυτών. Καθώς το στοιχείο Message επεκτείνει την αφαιρετική κλάση DescribedServiceRequirement, ο σημασιολογικός βαθμός αντιστοιχίας υπολογίζεται απευθείας με τη χρήση της εξίσωσης 2.
- Δεύτερον, υπολογίζουμε το βαθμό αντιστοιχίας υπογραφής,  $d_{sign}$ , μεταξύ της αιτούμενης και διαφημιζόμενης διεπαφής υπηρεσίας, συγκρίνοντας ενδελεχώς τα μηνύματα εισόδου/εξόδου, ως προς τα στοιχεία MessageElement, από τα οποία αποτελούνται. Όπως έχουμε δει στο μετα-μοντέλο της γλώσσας, τα στοιχεία αυτά έχουν ως τύπο την αφαιρετική κλάση DescribedAndTypedServiceRequirement. Ως εκ τούτου, ο υπολογισμός του βαθμού αντιστοιχίας δύο τέτοιων στοιχείων πραγματοποιείται με τη χρήση της εξίσωσης 3.

Πάντως, ο υπολογισμός του βαθμού αντιστοιχίας υπογραφής,  $d_{sign}$ , μεταξύ δύο μηνυμάτων μπορεί να οριστεί ως ειδική περίπτωση του Προβλήματος Ανάθεσης (Assignment Problem) [49]. Συγκεκριμένα, στην περίπτωσή μας, ζητείται να βρεθεί ο συνδυασμός μεταξύ των στοιχείων MessageElement του αιτούμενου μηνύματος και αυτών του διαφημιζόμενου μηνύματος, με το μέγιστο δυνατό βαθμό αντιστοιχίας. Καθώς τα στοιχεία MessageElement του αναζητούμενου μηνύματος έχουν ενδεχομένως διαφορετικά βάρη, ο βαθμός αντιστοιχίας για κάθε επιτρεπτό συνδυασμό υπολογίζεται με τη χρήση της εξίσωσης 1. Υπενθυμίζεται ότι, για κάθε ζεύγος αναζητούμενων και διαφημιζόμενων στοιχείων MessageElement,  $r_{element}$  και  $p_{element}$  αντίστοιχα, ο βαθμός αντιστοιχίας τους υπολογίζεται ως  $\psi(r_{element}, p_{element})$ .

Επιλύουμε το Πρόβλημα Ανάθεσης σε πολυωνυμικό χρόνο, με τη χρήση της γνωστής Ουγγρικής Μεθόδου (Hungarian Method) [73],  $h$ , την οποία προσαρμόζουμε κατάλληλα, βάσει των παραπάνω στοιχείων, αλλά και λαμβάνοντας υπόψη ενδεχόμενες διαφορές μεταξύ των δύο συνόλων στοιχείων MessageElement, ως προς τον αριθμό και σειρά καταγραφής τους.

**Ορισμός 3.8 (Αντιστοίχιση μηνυμάτων εισόδου/εξόδου)** Έστω  $r_{msg} = (v_o^r, v_d^r, \mathcal{E}^r)$  ένα στοιχείο απαίτησης τύπου Message και  $p_{msg} = (v_o^p, v_d^p, \mathcal{E}^p)$  το αντίστοιχο στοιχείο ιδιότητας μιας υπηρεσίας, όπου

- $v_o^r$  και  $v_o^p$  οι αντίστοιχες τιμές του ιδιοχαρακτηριστικού τους ontologyReference
- $v_d^r$  και  $v_d^p$  οι αντίστοιχες τιμές του ιδιοχαρακτηριστικού τους description
- $\mathcal{E}^r$  και  $\mathcal{E}^p$  τα αντίστοιχα σύνολα των στοιχείων τους MessageElement

Τότε, ο βαθμός αντιστοιχίας των  $r_{msg}$  και  $p_{msg}$ ,  $d_{msg}$ , υπολογίζεται από την ακόλουθη εξίσωση:

$$d_{msg} = \frac{1}{2} \cdot (d_{sem} + d_{sign}) = \frac{1}{2} \cdot (\phi(r_{msg}, p_{msg}) + h(\mathcal{E}^r, \mathcal{E}^p)) \quad (5)$$

Έτσι, ο βαθμός αντιστοιχίας μεταξύ δύο διεπαφών υπολογίζεται βάσει του βαθμού αντιστοιχίας των μηνυμάτων εισόδου/εξόδου τους, με τη χρήση της εξίσωσης 1, καθώς στα μηνύματα αυτά ενδέχεται να έχουν αποδοθεί διαφορετικά βάρη. Ο επόμενος ορισμός αναπτύσσει κατάλληλα την εν λόγω εξίσωση.

**Ορισμός 3.9 (Αντιστοίχιση στοιχείων ServiceInterface)** Έστω  $r_{interface} = (msg_{in}^r, msg_{out}^r)$  ένα στοιχείο απαίτησης τύπου ServiceInterface,  $w_{in}$  και  $w_{out}$  τα βάρη των μηνυμάτων εισόδου/εξόδου, αντίστοιχα, και  $p_{interface} = (msg_{in}^p, msg_{out}^p)$  το αντίστοιχο στοιχείο ιδιότητας μιας υπηρεσίας. Τότε, ο βαθμός αντιστοιχίας τους,  $d_{interface}$ , υπολογίζεται ως εξής:

$$d_{interface} = \begin{cases} d_{msg}^{in} \cdot (w'_{in} + w'_{out} \cdot d_{msg}^{out}) & , w_{in} > w_{out} \\ d_{msg}^{out} \cdot (w'_{out} + w'_{in} \cdot d_{msg}^{in}) & , w_{in} < w_{out} \\ \frac{1}{2} \cdot (d_{msg}^{in} \cdot w'_{in} + w'_{out} \cdot d_{msg}^{out}) & , w_{in} = w_{out} \end{cases} \quad (6)$$

όπου:

- $d_{msg}^{in}$  είναι ο βαθμός αντιστοιχίας των μηνυμάτων εισόδου,  $msg_{in}^r$  και  $msg_{in}^p$ , που προκύπτει από την εφαρμογή της εξίσωσης 5
- $d_{msg}^{out}$  είναι ο βαθμός αντιστοιχίας των μηνυμάτων εξόδου,  $msg_{out}^r$  και  $msg_{out}^p$ , που επίσης προκύπτει από την εφαρμογή της εξίσωσης 5
- $w'_{in} = \frac{w_{in}}{w_{in}+w_{out}}$  και  $w'_{out} = \frac{w_{out}}{w_{in}+w_{out}}$

**Αντιστοίχιση στοιχείων ServiceResource.** Σύμφωνα με το μετα-μοντέλο της γλώσσας USQL, τα στοιχεία απαιτήσεων Message και ServiceResource είναι δομικά και συντακτικά παρόμοια. Πράγματι, ένα στοιχείο Message περιέχει σημασιολογική πληροφορία, μέσω των ιδιοχαρακτηριστικών ontologyReference και description, ενώ περιλαμβάνει ένα διατεταγμένο σύνολο μερών (στοιχείων MessageElement), τα οποία έχουν επίσης σημασιολογική πληροφορία καθώς και συγκεκριμένο τύπο δεδομένων. Αντίστοιχα, ένα στοιχείο ServiceResource επίσης περιέχει σημασιολογική πληροφορία και περιλαμβάνει ένα διατεταγμένο σύνολο ιδιοτήτων (στοιχείων ResourceProperty), που είναι συντακτικά όμοια με τα στοιχεία MessageElement.

Με άλλα λόγια, το πρόβλημα αντιστοίχισης δύο στοιχείων τύπου ServiceResource αποτελεί επίσης ειδική περίπτωση του Προβλήματος Ανάθεσης και μπορεί να επιλυθεί με προσαρμογή της Ουγκρικής Μεθόδου, παρόμοια με την περίπτωση της αντιστοίχισης στοιχείων τύπου Message. Η μόνη διαφορά στην προσαρμογή έγκειται στο ότι, στην περίπτωση της αντιστοίχισης στοιχείων τύπου ServiceResource, δεν λαμβάνονται υπόψη ενδεχόμενες διαφορές στη σειρά καταγραφής των επιμέρους στοιχείων ResourceProperty.

**Ορισμός 3.10 (Αντιστοίχιση στοιχείων ServiceResource)** Έστω  $r_{res} = (v_o^r, v_d^r, \mathcal{P}^r)$  ένα στοιχείο απαίτησης τύπου *ServiceResource* και  $p_{res} = (v_o^p, v_d^p, \mathcal{P}^p)$  το αντίστοιχο στοιχείο ιδιότητας μιας υπηρεσίας, όπου

- $v_o^r$  και  $v_o^p$  οι αντίστοιχες τιμές του ιδιοχαρακτηριστικού τους *ontologyReference*
- $v_d^r$  και  $v_d^p$  οι αντίστοιχες τιμές του ιδιοχαρακτηριστικού τους *description*
- $\mathcal{P}^r$  και  $\mathcal{P}^p$  τα αντίστοιχα σύνολα των στοιχείων τους *ResourceProperty*

Τότε, ο βαθμός αντιστοιχίας των  $r_{res}$  και  $p_{res}$ ,  $d_{res}$ , υπολογίζεται από την ακόλουθη εξίσωση:

$$d_{msg} = \frac{1}{2} \cdot (d_{sem} + d_{sign}) = \frac{1}{2} \cdot (\phi(r_{res}, p_{res}) + h(\mathcal{P}^r, \mathcal{P}^p)) \quad (7)$$

**Αντιστοίχιση στοιχείων QoS.** Το μαθηματικό μοντέλο της USQL δεν προσδιορίζει τον τρόπο, με τον οποίο υπολογίζεται ο βαθμός αντιστοιχίας των διαφόρων στοιχείων απαίτησης τύπου *QoSRequirement*. Καθώς η κλάση αυτή μπορεί να χρησιμοποιηθεί για να εκφραστούν απαιτήσεις προς διάφορα ποιοτικά χαρακτηριστικά μιας υπηρεσίας, αναμένεται ότι, για κάθε υποστηριζόμενο είδος ποιοτικού χαρακτηριστικού, θα χρησιμοποιείται και η κατάλληλη για αυτό συνάρτηση αντιστοίχισης.

Παρόλα αυτά, σύμφωνα με το μετα-μοντέλο της γλώσσας, το στοιχείο *QoS* αποτελεί μια συλλογή στοιχείων απαιτήσεων *QoSRequirement*, στα οποία μπορεί να έχουν αποδοθεί διαφορετικά βάρη. Έτσι, ο συνολικός βαθμός αντιστοιχίας των ποιοτικών απαιτήσεων,  $d_{qos}$ , υπολογίζεται με τη χρήση της εξίσωσης 1.

### 3.6 Συμβατότητα και Επεκτασιμότητα

Η γενικότητα της γλώσσας USQL και το υψηλό επίπεδο αφαίρεσης, στο οποίο έχει οριστεί το μετα-μοντέλο της, εξασφαλίζουν διαλειτουργικότητα (*interoperability*) με διάφορους υπάρχοντες μηχανισμούς αναζήτησης υπηρεσιών ιστού, πλέγματος και ομότιμου-προσ-ομότιμο, καθώς και συμβατότητα με διάφορες υπάρχουσες γλώσσες περιγραφής υπηρεσιών. Η διαλειτουργικότητα και συμβατότητα επιτυγχάνονται μέσω συγκεκριμένων κανόνων μετασχηματισμού και αντιστοίχισης, οι οποίοι μπορούν να υλοποιηθούν είτε ως μικρές μονάδες λογισμικού, είτε υπό τη μορφή εκφράσεων XSLT [133] και XPath [131].

Ο Πίνακας 2 παρουσιάζει συνοπτικά τις αντιστοιχίες (*mappings*) μεταξύ των περιεχομένων ενός εγγράφου *USQLRequest* και των επερωτήσεων, που υποστηρίζονται από ένα μη εξαντλητικό σύνολο τεχνολογιών αναζήτησης υπηρεσιών.

Βάσει της εικόνας που δίνει ο Πίνακας 2, μπορούμε να κάνουμε τις εξής παρατηρήσεις:

Πίνακας 2: Διαλειτουργικότητα USQL - τεχνολογιών αναζήτησης υπηρεσιών

USQL	Τεχνολογίες Αναζήτησης Υπηρεσιών				
	UDDI	ebXML	JXTA	MDS	gLite
SearchFilters	X	X	X	X	X
ServiceCapability	X	X	X	X	X
ServiceInterface	X				
ServiceResource				X	X
QoS					

Πίνακας 3: Συμβατότητα USQL - τεχνολογιών περιγραφής υπηρεσιών

USQL	Τεχνολογίες Περιγραφής Υπηρεσιών						
	WSDL	WSRP	SAWSDL	OWL-S	MSA	WS-QoS	WSLA
SearchFilters			X	X	X		
ServiceCapability	X		X	X	X		
ServiceInterface	X		X	X			
ServiceResource		X					
QoS						X	X

- Η USQL μπορεί να εφαρμοστεί σε διάφορα ετερογενή υπηρεσιοστρεφή περιβάλλοντα, ως προς τους υποστηριζόμενους μηχανισμούς αναζήτησης υπηρεσιών.
- Με τη χρήση της USQL, ο αιτών μπορεί να εκφράσει τα κριτήρια αναζήτησης κατά τρόπο ενιαίο, ανεξάρτητα από τις διάφορες τεχνολογίες αναζήτησης υπηρεσιών. Τα κριτήρια αυτά μπορούν στη συνέχεια να αντιστοιχηθούν κατάλληλα με τις μορφοποιήσεις και διεπαφές επερωτήσεων, που υποστηρίζει η καθεμία επιμέρους τεχνολογία αναζήτησης υπηρεσιών.
- Σε σύγκριση με τις υπάρχουσες τεχνολογίες αναζήτησης υπηρεσιών ιστού, πλέγματος και ομότιμου-προς-ομότιμο, η USQL προσφέρει μια αρκετά πιο πλούσια διεπαφή για το σχηματισμό επερωτήσεων. Πράγματι, καθεμία από τις εξετασθείσες τεχνολογίες αναζήτησης υπηρεσιών υποστηρίζει ένα υποσύνολο των κριτηρίων αναζήτησης, που μπορούν να εκφραστούν σε ένα έγγραφο USQLRequest.
- Σε αντίθεση με την USQL, καμία από τις υπάρχουσες τεχνολογίες αναζήτησης υπηρεσιών δεν παρέχει εγγενή υποστήριξη για την περιγραφή απαιτήσεων ως προς τα ποιοτικά χαρακτηριστικά της αναζητούμενης υπηρεσίας, παρά το σημαντικό τους ρόλο στην τελική επιλογή μιας υπηρεσίας από το χρήστη.

Εξάλλου, στον Πίνακα 3, απεικονίζεται συνοπτικά η συμβατότητα της USQL με ένα μη εξαντλητικό σύνολο τεχνολογιών περιγραφής υπηρεσιών. Συγκεκριμένα, ο πίνακας δείχνει τις δυνατότητες αντιστοίχισης καθεμιάς τεχνολογίας με τα κριτήρια αναζήτησης, που υποστηρίζονται από την USQL. Σε αυτή την περίπτωση, παρατηρούμε ότι:

- Η USQL μπορεί να χρησιμοποιηθεί για το σχηματισμό επερωτήσεων για υπηρεσίες σε διάφορα ετερογενή περιβάλλοντα, ως προς τις υποστηριζόμενες από αυτά τεχνολογίες περιγραφής υπηρεσιών.
- Καθεμία από τις εξετασθείσες τεχνολογίες υποστηρίζει την περιγραφή ενός μέρους μόνο από τις ιδιότητες, για τις οποίες μπορεί κανείς να εκφράσει απαιτήσεις σε ένα έγγραφο USQLRequest. Το γεγονός αυτό μαρτυρεί τη συγκριτική ανωτερότητα της USQL ως γλώσσα επερωτήσεων, από άλλες προσεγγίσεις, στις οποίες οι επερωτήσεις σχηματίζονται βάσει του συντακτικού μιας συγκεκριμένης γλώσσας περιγραφής υπηρεσιών.
- Ο πάροχος μιας υπηρεσίας χρειάζεται περισσότερες από μια γλώσσες, προκειμένου να περιγράψει τις διάφορες λειτουργικές και ποιοτικές ιδιότητες της υπηρεσίας του. Αντιθέτως, ο αιτών μιας υπηρεσίας περιγράφει τις απαιτήσεις του μέσω ενός ενιαίου και υψηλού επιπέδου αφάιρησης συντακτικού, που προσφέρει η USQL.

Κλείνοντας αυτή την ενότητα, θα θέλαμε να υπενθυμίσουμε τη δυνατότητα επέκτασης της USQL, έτσι ώστε να υποστηρίξει την έκφραση περαιτέρω κριτηρίων αναζήτησης. Χάρη σε αυτή της την ευελιξία, αλλά και την αφαιρετικότητα των δομών του μετα-μοντέλου της, όπως το στοιχείο SearchFilter, καθίσταται δυνατή μελλοντικά η επίτευξη ακόμα μεγαλύτερης συμβατότητας με τις υπάρχουσες τεχνολογίες αναζήτησης και περιγραφής υπηρεσιών ιστού, πλέγματος και ομότιμου-προς-ομότιμο. Για παράδειγμα, μέσω κατάλληλων επεκτάσεων της αφαιρετικής κλάσης ServiceRequirement, ή κάποιας υποκλάσης της, θα μπορούσε να υποστηριχθεί η περιγραφή των απαιτήσεων του χρήστη, ως προς τις προϋποθέσεις (pre-conditions) ή/και επιδράσεις (effects) εκτέλεσης της αναζητούμενης υπηρεσίας, καθώς η περιγραφή αυτών των ιδιοτήτων υποστηρίζεται από κάποιες γλώσσες περιγραφής υπηρεσιών (π.χ. OWL-S, WSML).

### 3.7 Συζήτηση

Στο κεφάλαιο αυτό παρουσιάσαμε μια νέα γλώσσα αναζήτησης ετερογενών υπηρεσιών, την USQL (Unified Service Query Language). Η USQL επιτρέπει τη σύνταξη επερωτήσεων, το περιεχόμενο των οποίων αντικατοπτρίζει τις πραγματικές απαιτήσεις του χρήστη ως προς τις λειτουργικές και ποιοτικές ιδιότητες της αναζητούμενης υπηρεσίας. Η γλώσσα βασίζεται σε ένα αφαιρετικό μετα-μοντέλο και παρέχει ένα συντακτικό XML, τα οποία την καθιστούν ανεξάρτητη από τις τεχνολογίες που διέπουν τους διάφορους ετερογενείς τύπους υπηρεσιών και ειδικά εκείνες που αφορούν στην περιγραφή και αναζήτησή τους. Πράγματι, μέσω κατάλληλων κανόνων μετασχηματισμού και αντιστοίχισης, η USQL είναι συμβατή με την πλειοψηφία των υπάρχοντων προτύπων περιγραφής υπηρεσιών, όπως είναι οι γλώσσες WSDL, SAWSDL, OWL-S, WSLA, WS-QoS κ.ά. Με τον ίδιο τρόπο, η



USQL μπορεί να χρησιμοποιηθεί για την αναζήτηση υπηρεσιών σε πληθώρα ετερογενών μητρώων, αποθετηρίων, δικτύων ομότιμων κόμβων κτλ., όπως τα UDDI, ebXML, LDAP, JXTA, κτλ.

Έχοντας οριστεί σε ένα υψηλό επίπεδο αφαίρεσης, η USQL είναι ανεξάρτητη από το πεδίο εφαρμογής των αναζητούμενων υπηρεσιών και περιλαμβάνει μια επαρκή γκάμα δομών και στοιχείων για τον προσδιορισμό απαιτήσεων ως προς τη ζητούμενη λειτουργικότητα, διεπαφή, τον επιθυμητό πόρο και τα ποιοτικά χαρακτηριστικά της υπό αναζήτηση υπηρεσίας. Ταυτόχρονα, η προδιαγραφή της γλώσσας παρέχει κατάλληλα σημεία επέκτασης, τα οποία επιτρέπουν τη μελλοντική ενσωμάτωση επιπρόσθετων στοιχείων και δυνατοτήτων, χωρίς την ανάγκη επανορισμού του συντακτικού της. Οι δυνατότητες επέκτασης της γλώσσας χρησιμοποιήθηκαν ήδη για την ενσωμάτωση κριτήριων αναζήτησης σχετικά με τη συμπεριφορά, ή χορογραφία (choreography) σύνθετων υπηρεσιών, στο πλαίσιο μιας γενικότερης προσπάθειας ενσωμάτωσης της USQL στο πλαίσιο αναζήτησης υπηρεσιών ASD [71]. Τα πρώιμα αποτελέσματα της προσπάθειας αυτής είναι διαθέσιμα στη βιβλιογραφία (βλ. [90]). Εν κατακλείδι, η επεκτασιμότητα της USQL επιτρέπει τη βιωσιμότητά της σε βάθος χρόνου, καθιστώντας την ευέλικτη και ικανή να προσαρμόζεται, καθώς οι υπάρχουσες υπηρεσιοστρεφείς τεχνολογίες εξελίσσονται και νέες τεχνολογίες κάνουν την εμφάνισή τους.

Όλα τα χαρακτηριστικά της παρούσης έκδοσης της USQL έχουν υλοποιηθεί και υποστηρίζονται από το σύστημα Proteus, το οποίο παρουσιάζουμε λεπτομερώς στο Κεφάλαιο 4. Στο σημείο αυτό αξίζει να αναφερθεί, ότι προηγούμενες εκδόσεις της USQL έχουν ήδη χρησιμοποιηθεί ως επιλογή γλώσσας για τη σύνταξη επερωτήσεων από άλλα δύο συστήματα αναζήτησης υπηρεσιών. Στο πλαίσιο DIRE [13], το οποίο παρέχει ένα κοινό μοντέλο υπηρεσιών για ετερογενή μητρώα και υποστηρίζει την επικοινωνία τους μέσω ενός μηχανισμού δημοσίευσης/συνδρομής (publish/subscribe mechanism), η USQL χρησιμοποιείται για τη σύνταξη των προτιμήσεων του χρήστη ως προς κάποια ζητούμενη υπηρεσία. Οι προτιμήσεις αυτές λαμβάνουν συνεπώς τη μορφή ενός εγγράφου USQLRequest, το οποίο στη συνέχεια μεταφράζεται κατάλληλα στην υποστηριζόμενη από το σύστημα γλώσσα περιγραφής συνδρομών. Από την άλλη, το πλαίσιο PYRAMID-S [94] επεκτείνει και χρησιμοποιεί τη γλώσσα USQL για το σχηματισμό σημασιολογικά εμπλουτισμένων επερωτήσεων και την εκτέλεσή τους σε ετερογενή μητρώα δημοσίευσης σημασιολογικά περιγεγραμμένων υπηρεσιών ιστού. Κλείνοντας αυτό το κεφάλαιο, θα θέλαμε να σημειώσουμε, ότι το γεγονός της χρήσης της USQL από άλλα συστήματα αναζήτησης υπηρεσιών προσφέρει μια ένδειξη σχετικά με τη χρησιμότητά της και την ορθότητα της εννοιολογικής της σύλληψης.

Ανάπτυξη μιας γλώσσας και του απαραίτητου λογισμικού εκτέλεσής της, για την αναζήτηση ετερογενών υπηρεσιών με ενιαίο τρόπο

## ΚΕΦΑΛΑΙΟ 4

# Η ΜΗΧΑΝΗ ΑΝΑΖΗΤΗΣΗΣ ΕΤΕΡΟΓΕΝΩΝ ΥΠΗΡΕΣΙΩΝ

### Proteus

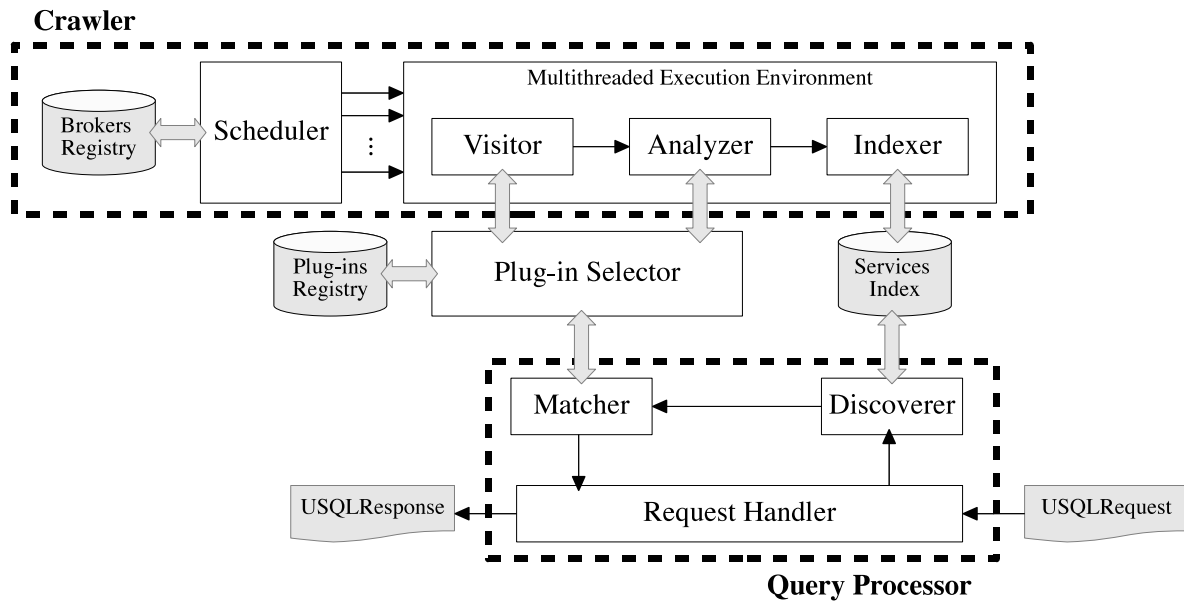
Σε αυτό το κεφάλαιο, παρουσιάζουμε τη μηχανή αναζήτησης Proteus, η οποία υποστηρίζει την αναζήτηση ετερογενών υπηρεσιών σε ετερογενή μητρώα, αποθετήρια, καταλόγους, δίκτυα ομότιμων κόμβων, κτλ., κάνοντας χρήση της γλώσσας USQL. Η μηχανή αναζήτησης Proteus αποτελείται από δύο υποσυστήματα, τα οποία περιγράφουμε στις Ενότητες 4.3 και 4.5. Ένα βασικό χαρακτηριστικό που διέπει τη συνολική αρχιτεκτονική του συστήματος είναι η επεκτασιμότητά του, υποστηριζόμενη από ένα ισχυρό μηχανισμό επεκτάσεων και μια σειρά σημείων επέκτασης, τα οποία επιτρέπουν τη σύνδεση επιπρόσθετων συστατικών στο κυρίως σώμα της μηχανής αναζήτησης. Τόσο ο μηχανισμός επεκτάσεων όσο και ο μηχανισμός κλιμάκωσης του συστήματος Proteus παρουσιάζονται στις Ενότητες 4.6 και 4.7, αντίστοιχα.

#### 4.1 Κίνητρο

Η ευρεία αποδοχή του υπηρεσιοστρεφούς μοντέλου για την ανάπτυξη κατανεμημένων εφαρμογών λογισμικού είχε ως αποτέλεσμα την εμφάνιση ενός μεγάλου αριθμού ετερογενών υπηρεσιών, οι οποίες είναι διαθέσιμες προς αναζήτηση και χρήση. Οι υπηρεσίες αυτές δημοσιεύονται ως επί το πλείστον σε διάφορα ετερογενή ιδιωτικά μητρώα, αποθετήρια κτλ., ή στα όρια ελεγχόμενων δικτύων ομότιμων κόμβων. Ο μεγάλος αριθμός των πιθανών στόχων αναζήτησης, η κατανομή τους, καθώς και ο συνεχώς αυξανόμενος αριθμός δημοσιευμένων υπηρεσιών γεννά τα εξής προβλήματα απόδοσης και κλιμάκωσης στην αναζήτηση υπηρεσιών:

- Η εκτέλεση μιας επερώτησης αναζήτησης υπηρεσιών μπορεί να έχει μεγάλη διάρκεια και ο χρόνος απόκρισης εξαρτάται από αστάθμητους παράγοντες, όπως η καθυστέρηση του δικτύου, ο φόρτος εργασίας των εξυπηρετητών που φιλοξενούν τα διάφορα μητρώα, αποθετήρια κτλ., το εύρος ζώνης της σύνδεσης του χρήστη κ.ά.
- Για την εκτέλεση μιας επερώτησης απαιτείται η συγκομιδή, επεξεργασία και αντιστοίχιση ενός μεγάλου εν δυνάμει αριθμού εγγράφων περιγραφών υπηρεσιών, από τη μηχανή αναζήτησης.

Η ανάγκη εφαρμογής της αναζήτησης υπηρεσιών σε ευρεία κλίμακα έχει ήδη αναγνωριστεί στη βιβλιογραφία, όπου παρουσιάστηκαν διάφορες προσεγγίσεις για την επίλυση των ανωτέρω προβλημάτων, κυρίως με τη χρήση μεθόδων καταλογοποίησης [60] [62].



Σχήμα 12: Αρχιτεκτονική της μηχανής αναζήτησης Proteus

Μια ακόμα διαδεδομένη τεχνική για τη βελτίωση των χρόνων απόκρισης των μηχανών αναζήτησης υπηρεσιών συνίσταται στο διαχωρισμό της διαδικασίας εκτέλεσης της επερώτησης ενός χρήστη από τη διαδικασία επίσκεψης στα διάφορα μητρώα, αποθετήρια, δίκτυα κτλ., παρόμοια με τον τρόπο που λειτουργούν οι σύγχρονες μηχανές αναζήτησης ιστού (web search engines), π.χ. η διάσημη μηχανή Google [18]. Τα τελευταία χρόνια, αναπτύχθηκαν διάφορες διαδικτυακές μηχανές αναζήτησης υπηρεσιών υιοθετώντας στην τεχνική αυτή, με πρόσφατο παράδειγμα τη μηχανή Seekda [105]. Παρόλα αυτά, οι υπάρχουσες προσεγγίσεις στο πρόβλημα της κλιμάκωσης δεν μπορούν να εφαρμοστούν στην περίπτωση αναζήτησης ετερογενών υπηρεσιών, καθώς δεν προτείνουν λύση στο πρόβλημα της ετερογένειας, που διέπει τους διάφορους τύπους υπηρεσιών, καθώς και τα πρότυπα περιγραφής και αναζήτησής τους.

Για τους λόγους αυτούς, αλλά και επειδή στην Πληροφορική μια γλώσσα δεν έχει καμία αξία χωρίς τη μηχανή υλοποίησής της, παρουσιάζουμε στο κεφάλαιο αυτό μια νέα, ευέλικτη και επεκτάσιμη μηχανή αναζήτησης ετερογενών υπηρεσιών, με το όνομα Proteus. Όπως θα δούμε, το προτεινόμενο σύστημα υποστηρίζει τη γλώσσα ενιαίας αναζήτησης ετερογενών υπηρεσιών USQL, ενώ, μέσω του σχεδιασμού του, δίνει λύσεις στα προαναφερθέντα ζητήματα ετερογένειας, απόδοσης και κλιμάκωσης.

Στις παραγράφους που ακολουθούν, παρουσιάζουμε την αρχιτεκτονική της μηχανής αναζήτησης Proteus και περιγράφουμε αναλυτικά τα βασικά της συστατικά καθώς και τις λειτουργίες που αυτά επιτελούν.

## 4.2 Αρχιτεκτονική Συστήματος

Η μηχανή αναζήτησης Proteus έχει σχεδιαστεί κατά τέτοιο τρόπο, ώστε να επιτυγχάνεται πλήρως ο προαναφερθείς διαχωρισμός της διαδικασίας εκτέλεσης των επερωτήσεων USQL, από τη διαδικασία επίσκεψης στις υποδομές των διαφόρων μεσαζόντων υπηρεσιών, για την ανάκτηση και επεξεργασία των δημοσιευμένων περιγραφών υπηρεσιών.

Όπως απεικονίζεται στο Σχήμα 12, η αρχιτεκτονική του συστήματος προδιαγράφει ένα τοπικό ευρετήριο υπηρεσιών (*services index*) και μια σειρά μητρώων με δεδομένα υποστήριξης (*brokers registry* και *plug-ins registry*), ενώ χωρίζει την προσφερόμενη λειτουργικότητά του σε δύο βασικά υποσυστήματα, ως εξής:

- Ο **Αναζητητής (Crawler)** περικλύει όλα τα απαραίτητα συστατικά, που υποστηρίζουν (α) την περιοδική επίσκεψη σε διαφόρους, εκ των προτέρων γνωστούς στόχους αναζήτησης, με σκοπό τη συγκομιδή των δημοσιευμένων σε αυτούς περιγραφών υπηρεσιών, (β) την επεξεργασία των περιγραφών, προκειμένου να μετατραπούν σε μια κοινή μορφή εγγράφων διαφημίσεων λειτουργιών, και (γ) την κατάλληλη αποθήκευση των διαφημίσεων αυτών, με τη δημιουργία πολλαπλών ευρετηρίων για γρήγορη ανάκτηση.
- Ο **Επεξεργαστής Επερωτήσεων (Query Processor)** προσφέρει μια διεπαφή στους χρήστες, μέσω της οποίας υποστηρίζεται η ενιαία αναζήτηση ετερογενών υπηρεσιών, με τη χρήση της γλώσσας USQL. Συγκεκριμένα, το υποσύστημα δέχεται επερωτήσεις υπό τη μορφή εγγράφων τύπου *USQLRequest* και, με βάση τα εκφρασμένα από αυτές φίλτρα και κριτήρια αναζήτησης, ανακτά τις κατάλληλες περιγραφές υπηρεσιών, των οποίων το περιεχόμενο αξιολογεί στη συνέχεια. Τα αποτελέσματα της διαδικασίας αυτής διαμορφώνονται σε ένα έγγραφο τύπου *USQLResponse*, το οποίο επιστρέφεται στο χρήστη, ως απόκριση.

Στις ακόλουθες παραγράφους, περιγράφουμε διεξοδικά τα δύο υποσυστήματα της μηχανής αναζήτησης Proteus, καθώς και τη λειτουργία των επιμέρους συστατικών τους.

## 4.3 Το Υποσύστημα Αναζήτησης

Ακολουθώντας το παράδειγμα παραδοσιακών μηχανών αναζήτησης στο διαδίκτυο, η μηχανή Proteus περιλαμβάνει έναν *Αναζητητή (Crawler)*, ο οποίος έχει σχεδιαστεί για την υποστήριξη ταυτόχρονων, περιοδικών επισκέψεων σε (πιθανότατα ετερογενείς) υποδομές πολλαπλών μεσαζόντων υπηρεσιών (*service brokers*). Η αρχιτεκτονική του Αναζητητή προσομοιάζει αυτή των παραδοσιακών συστατικών *Web Crawler* (βλ. [11], [8], [106], [10]), που συναντώνται στις περισσότερες μηχανές αναζήτησης Ιστού. Ας περιγράψουμε πιο λεπτομερώς τη λειτουργία του.

Το υποσύστημα αναζήτησης εκκινείται δεχόμενο μια λίστα μεσαζόντων υπηρεσιών, οι οποίοι αποτελούν τους στόχους αναζήτησης. Ένας μεσάζων καθίσταται γνωστός στο σύστημα μέσω της καταχώρησής του στο *Μητρώο Μεσαζόντων (Brokers Registry)*. Η διαδικασία αυτή πραγματοποιείται χειροκίνητα από το διαχειριστή του συστήματος Proteus, ενώ η αντίστοιχη για κάθε μεσάζοντα εγγραφή έχει την ακόλουθη μορφή:

$$rec_{broker} = (id, t_b, t_s, s_{spec}, T_0, T, T_{min}, T_{max}, d_v, r_{inc}, \mathcal{P}) \quad (8)$$

Η σημασιολογία των στοιχείων του ανωτέρω τύπου εγγραφής έχει ως εξής:

- *id*: το αναγνωριστικό του μεσάζοντα, βάσει του οποίου γίνεται η ταυτοποίησή του από το σύστημα Proteus
- *t<sub>b</sub>*: το είδος της υποδομής του μεσάζοντα (π.χ. UDDI, ebXML, κτλ.)
- *t<sub>s</sub>*: ο τύπος των υπηρεσιών, που δημοσιεύονται στο μεσάζοντα (π.χ. WebService, GridService, P2PService)
- *s<sub>spec</sub>*: το σχήμα περιγραφής των δημοσιευμένων υπηρεσιών (π.χ. WSDL, OWL-S, κτλ.)
- *T<sub>0</sub>*: η αρχική περίοδος επισκέψεων, σε χιλιοστά του δευτερολέπτου
- *T*: η τρέχουσα περίοδος επισκέψεων, σε χιλιοστά του δευτερολέπτου (αρχικά ισούται με την τιμή της αρχικής περιόδου επισκέψεων, *T<sub>0</sub>*)
- *T<sub>min</sub>*: η ελάχιστη επιτρεπτή τιμή για την περίοδο επισκέψεων, σε χιλιοστά του δευτερολέπτου
- *T<sub>max</sub>*: η μέγιστη επιτρεπτή τιμή για την περίοδο επισκέψεων, σε χιλιοστά του δευτερολέπτου
- *d<sub>v</sub>*: η ημερομηνία τελευταίας επίσκεψης του Αναζητητή στο μεσάζοντα, εκφρασμένη σε χιλιοστά του δευτερολέπτου
- *r<sub>inc</sub>*: ο σταθερός ρυθμός αύξησης της περιόδου επισκέψεων, με τιμές μεταξύ 0 και 1
- *℘*: το σύνολο ιδιοτήτων πρόσβασης του μεσάζοντα (σε αντιστοιχία με το μετα-μοντέλο της USQL, βλ. παράγραφο 3.3.2)

#### 4.3.1 Χρονοπρογραμματισμός Επισκέψεων

Κατά την αρχικοποίηση της μηχανής αναζήτησης Proteus, όλες οι καταχωρημένες εγγραφές τύπου *rec<sub>broker</sub>* του Μητρώου Μεσαζόντων ανακτώνται από τον Αναζητητή και δίνονται ως είσοδος στο *Χρονοπρογραμματιστή (Scheduler)*. Το συστατικό αυτό είναι υπεύθυνο

για τον κατάλληλο προγραμματισμό των επισκέψεων στους μεσάζοντες, που αντιστοιχούν στις εγγραφές.

Για κάθε μεσάζοντα, ο Χρονοπρογραμματιστής δεσμεύει ένα ξεχωριστό νήμα εκτέλεσης (execution thread), του οποίου ο χρόνος ενεργοποίησης καθορίζεται βάσει (1) του χρόνου τελευταίας επίσκεψης,  $d_v$ , (2) της τρέχουσας περιόδου επισκέψεων,  $T$ , του μεσάζοντα και (3) του τρέχοντος χρόνου,  $d_{now}$ , εκφρασμένου σε χιλιοστά του δευτερολέπτου. Συγκεκριμένα, ο χρόνος αναμονής του νήματος εκτέλεσης,  $t_{delay}$ , υπολογίζεται ως εξής:

$$t_{delay} = \max((d_v + T) - d_{now}, 0) \quad (9)$$

Αφού ολοκληρωθεί ο χρονοπρογραμματισμός των νημάτων εκτέλεσης για την πραγματοποίηση των επισκέψεων, ο Χρονοπρογραμματιστής αδρανοποιείται, έως ότου χρειαστεί να καθοριστεί εκ νέου η επόμενη επίσκεψη σε κάποιον από τους μεσάζοντες.

### 4.3.2 Πραγματοποίηση Επισκέψεων

Αμέσως μόλις ενεργοποιηθεί κάποιο από τα χρονοπρογραμματιζόμενα νήματα εκτέλεσης του Αναζητητή, για την επίσκεψη σε κάποιο μεσάζοντα, δημιουργείται από το σύστημα ένα νέο στιγμιότυπο του συστατικού *Επισκέπτης* (*Visitor*). Το συστατικό αυτό υλοποιεί τη διαδικασία επίσκεψης και λήψης των περιεχομένων ενός μεσάζοντα. Καθώς, όπως έχει ήδη περιγραφεί στην ενότητα 2.4, οι μηχανισμοί αναζήτησης των διαφόρων μεσαζόντων χαρακτηρίζονται από ετερογένεια, η διαδικασία αυτή διεκπεραιώνεται με τη χρήση ενός κατάλληλου, *συνδεδεμένου συστατικού* (*plug-in component*), το οποίο ο Επισκέπτης ανακτά δυναμικά από το *Μητρώο Συνδεδεμένων Συστατικών* (*Plug-ins Registry*) του συστήματος (για περισσότερες λεπτομέρειες, βλ. ενότητα 4.6). Η επιλογή πραγματοποιείται βάσει του είδους του μεσάζοντα,  $t_b$ , όπως αυτό δηλώνεται στην αντίστοιχη εγγραφή  $rec_{broker}$ .

Το αποτέλεσμα της επίσκεψης στο μεσάζοντα είναι μια λίστα εγγραφών, οι οποίες αντιστοιχούν στις υπηρεσίες, που βρέθηκαν δημοσιευμένες σε αυτόν. Οι εγγραφές αυτές έχουν την εξής μορφή:

$$rec_{pub} = (id_b, id_s, spec, \mathcal{P}) \quad (10)$$

Έτσι, για κάθε μια δημοσίευση, η αντίστοιχη εγγραφή  $rec_{pub}$  περιλαμβάνει πληροφορία σχετικά με:

- το αναγνωριστικό του μεσάζοντα ( $id_b$ ), από τον οποίο ανακτήθηκε,
- το αναγνωριστικό της υπηρεσίας ( $id_s$ ), όπως αυτό έχει καταχωρηθεί στο μεσάζοντα,
- το έγγραφο περιγραφής της υπηρεσίας ( $spec$ ) και

- μια σειρά ιδιοτήτων ( $\mathcal{P}$ ), όπως για παράδειγμα το όνομα της υπηρεσίας, το όνομα του παρόχου της, η κατηγορία στην οποία ανήκει κτλ., οι οποίες ανταποκρίνονται στα φίλτρα αναζήτησης των επερωτήσεων της USQL.

Στην περίπτωση που ο Αναζητητής έχει επισκεφθεί το συγκεκριμένο μεσάζοντα στο παρελθόν, ο Επισκέπτης αναλαμβάνει τον προσδιορισμό του αριθμού των κοινών δημοσιεύσεων ( $s_{com}$ ), τον αριθμό των νέων δημοσιεύσεων ( $s_{new}$ ), καθώς και τον αριθμό των διαγεγραμμένων δημοσιεύσεων ( $s_{del}$ ), συγκρίνοντας τις λίστες εγγραφών  $rec_{pub}$ , που ανακτήθηκαν κατά τρέχουσα και αμέσως προηγούμενη επίσκεψη στο μεσάζοντα. Ακολούθως, χρησιμοποιεί τα στοιχεία αυτά για να ποσοτικοποιήσει τη μεταβλητότητα (volatility),  $v$ , των περιεχομένων του μεσάζοντα, σύμφωνα με την ακόλουθη εξίσωση:

$$v = \frac{s_{new} + s_{del}}{s_{new} + s_{del} + s_{com}} \quad (11)$$

Βάσει της μεταβλητότητας  $v$ , η οποία λαμβάνει τιμές μεταξύ του 0 και του 1, υπολογίζεται η νέα περίοδος επισκέψεων του μεσάζοντα, ως εξής:

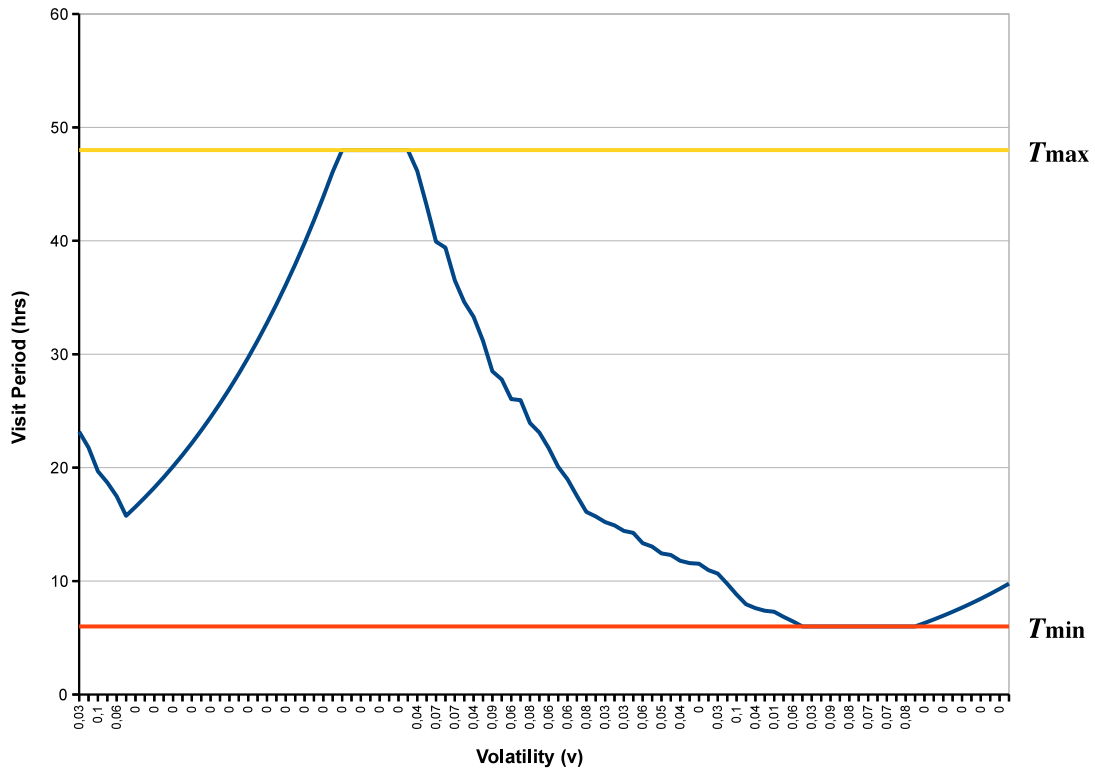
$$T = \begin{cases} \min(T \cdot (1 + r_{inc}), T_{max}) & , \quad v = 0 \\ T_0 & , \quad v = \text{undefined} \quad (s_{new} = s_{del} = s_{com} = 0) \\ \max(T \cdot (1 - v), T_{min}) & , \quad v > 0 \end{cases} \quad (12)$$

Όπως φαίνεται στην Εξίσωση (12), η περίοδος επισκέψεων σε ένα μεσάζοντα αυξάνει, καθώς παρατηρείται μηδενική ρευστότητα στα περιεχόμενά του, ενώ μειώνεται, στις περιπτώσεις που στα περιεχόμενά του παρατηρούνται αλλαγές. Κατ' εξαίρεση, η περίοδος επισκέψεων λαμβάνει την αρχική της τιμή,  $T_0$ , σε περιπτώσεις δύο διαδοχικών επισκέψεων, κατά τις οποίες ο μεσάζων βρέθηκε να έχει μηδενικά περιεχόμενα, με αποτέλεσμα να ισχύει  $s_{new} = s_{del} = s_{com} = 0$ .

Η νέα τιμή της περιόδου επισκέψεων χρησιμοποιείται στη συνέχεια από το Χρονοπρογραμματιστή, προκειμένου να προσδιορίσει το χρόνο της επόμενης επίσκεψης στο μεσάζοντα, σύμφωνα με την Εξίσωση (9).

Στη βιβλιογραφία (π.χ. [8]), έχουν οριστεί διάφορες πολιτικές επαναληπτικών επισκέψεων, για τους Αναζητητές των παραδοσιακών μηχανών αναζήτησης ιστού. Στην περίπτωση της δικής μας μηχανής αναζήτησης ετερογενών υπηρεσιών, όπως διαφαίνεται κι από τις Εξισώσεις (11) και (12), ακολουθείται μια αναλογική πολιτική (proportional policy) στον προγραμματισμό των επαναληπτικών επισκέψεων σε κάθε στόχο αναζήτησης, έτσι ώστε να εξασφαλίζεται ο συγχρονισμός των περιεχομένων του τοπικού καταλόγου υπηρεσιών, με αυτά των εξωτερικών μεσαζόντων υπηρεσιών, στο βαθμό που αυτό είναι δυνατό. Σημειώνεται, ότι η αναλογική αυτή πολιτική έχει προσαρμοστεί ειδικά στην περίπτωση αναζήτησης υπηρεσιών, όπου οι συχνότητες εμφάνισης νέων δημοσιεύσεων ή/και αλλαγών





Σχήμα 13: Αυξομείωση της περιόδου επισκέψεων

σε υπάρχουσες δημοσιεύσεις είναι σαφώς μικρότερες από αυτές των ιστοσελίδων ανά το διαδίκτυο.

Όπως άλλωστε υποδηλώνει η Εξίσωση 12, βασικό χαρακτηριστικό της ακολουθούμενης αναλογικής πολιτικής είναι, ότι η περίοδος επισκέψεων σε ένα μεσάζοντα τείνει να αυξάνεται με σταθερό, αλλά παραμετροποιήσιμο, ρυθμό, όσο τα περιεχόμενά του παραμένουν αμετάβλητα. Διαισθητικά, αυτό σημαίνει ότι η μηχανή αναζήτησης Proteus επισκέπτεται όλο και πιο σπάνια εκείνους τους μεσάζοντες υπηρεσιών, στους οποίους σπανίως δημοσιεύονται νέες υπηρεσίες, ή από τους οποίους σπανίως διαγράφονται υπάρχουσες διαφημίσεις υπηρεσιών. Παρόλα αυτά, για την αποφυγή υπερβολικά μεγάλων περιόδων επισκέψεων, η αύξηση ελέγχεται από μια προσδιορισμένη μέγιστη τιμή,  $T_{max}$ .

Αντίστοιχα, η περίοδος επισκέψεων τείνει να μειώνεται σύμφωνα με το ρυθμό αλλαγών στα περιεχόμενα του μεσάζοντα, αλλά δεν μπορεί να ξεπεράσει το προκαθορισμένο κατώτατο κατώφλι,  $T_{min}$ . Διαισθητικά, αυτό σημαίνει ότι η μηχανή Proteus θα επισκέπτεται ολοένα και πιο συχνά εκείνα τα μητρώα, αποθετήρια, δίκτυα κτλ., των οποίων τα περιεχόμενα εμφανίζουν μεγάλη μεταβλητότητα.

Η προσαρμοστική αυτή συμπεριφορά του συστήματός μας, σε ό,τι αφορά τις διαδοχικές επισκέψεις σε ένα μεσάζοντα υπηρεσιών, φαίνεται γραφικά στο διάγραμμα του Σχήματος 13.

Στο συγκεκριμένο παράδειγμα, αναπαρίσταται η αυξομείωση της περιόδου επισκέψεων σε ένα μεσάζοντα, για τον οποίο ισχύουν οι εξής τιμές:  $T_0 = 24hrs$ ,  $T_{min} = 6hrs$ ,

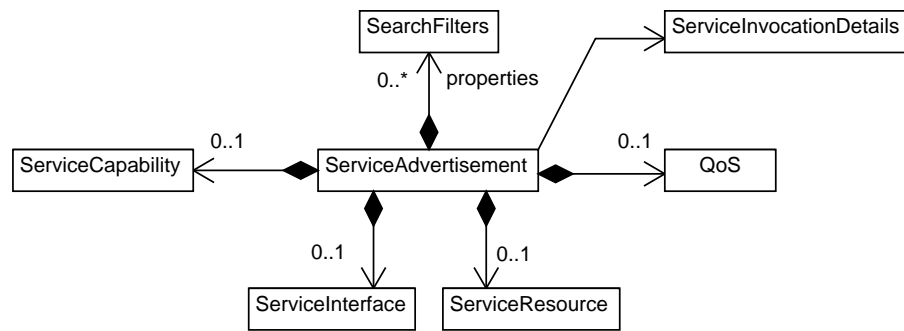
$T_{max} = 48hrs$ , και  $r_{inc} = 5\%$ . Παρατηρούμε ότι, καθώς τα περιεχόμενα του στόχου αναζήτησης εμφανίζουν αλλαγές ( $v > 0$ ), η περίοδος επισκέψεων  $T$  μειώνεται. Αντίθετα, η περίοδος επισκέψεων αρχίζει να αυξάνει, καθώς διαδοχικές επισκέψεις στο μεσάζοντα αναφέρουν μηδενικό αριθμό αλλαγών ( $v = 0$ ) στα, μη μηδενικά, περιεχόμενά του. Σε κάθε περίπτωση, όπως απεικονίζεται στο διάγραμμα, η περίοδος επισκέψεων αυξομειώνεται εντός των ορίων που καθορίζουν οι προκαθορισμένες τιμές  $T_{min}$  και  $T_{max}$ .

### 4.3.3 Ανάλυση Περιγραφών Υπηρεσιών

Με την ολοκλήρωση της λειτουργίας του Επισκέπτη, και καθώς ο Χρονοπρογραμματιστής προγραμματίζει την επόμενη επίσκεψη στο μεσάζοντα, ο Αναζητητής δημιουργεί ένα νέο στιγμιότυπο του συστατικού *Αναλυτής (Analyzer)*, για την επεξεργασία των  $s_{new}$  νέων δημοσιεύσεων. Για κάθε μια από αυτές τις δημοσιεύσεις, το πρώτο στάδιο επεξεργασίας συνίσταται στην ανάλυση του εγγράφου περιγραφής της αντίστοιχης υπηρεσίας (υπενθυμίζεται, ότι το έγγραφο αυτό έχει καταστεί προσβάσιμο μέσω του στοιχείου *spec* της αντίστοιχης εγγραφής *rec<sub>pub</sub>*). Για την αντιμετώπιση της ετερογένειας που χαρακτηρίζει τις διάφορες γλώσσες περιγραφής υπηρεσιών, η επεξεργασία των εγγράφων αυτών πραγματοποιείται με τη βοήθεια κατάλληλων συνδεδεμένων συστατικών (plug-ins) του συστήματος Proteus. Κάθε τέτοιο συστατικό συνδέεται δυναμικά με τον Αναλυτή και είναι υπεύθυνο για την απεικόνιση των περιεχομένων του υπό επεξεργασία εγγράφου περιγραφής της υπηρεσίας σε ένα ή περισσότερα έγγραφα διαφήμισης των επιμέρους λειτουργιών της, σύμφωνα με το μετα-μοντέλο της USQL. Η επιλογή του κατάλληλου συνδεδεμένου συστατικού για την επεξεργασία του εγγράφου περιγραφής μιας υπηρεσίας πραγματοποιείται βάσει του σχήματος της γλώσσας περιγραφής,  $s_{spec}$ , που περιλαμβάνεται στην εγγραφή του μεσάζοντα, από τον οποίο ανακτήθηκε. Έτσι, ο Αναλυτής καθίσταται ανθεκτικός στη συντακτική ή/και εννοιολογική ετερογένεια, που χαρακτηρίζει τα έγγραφα περιγραφής υπηρεσιών ιστού, πλέγματος και ομότιμου-προς-ομότιμο.

Κατά το δεύτερο στάδιο της επεξεργασίας, όλες οι τιμές κειμένου στα παραχθέντα έγγραφα διαφημίσεων υπόκεινται σε περαιτέρω ανάλυση, με την εκτέλεση των ακόλουθων βημάτων Επεξεργασίας Φυσικής Γλώσσας (Natural Language Processing, NLP):

1. **Διάσπαση (Tokenization)**. Κάθε τιμή κειμένου διασπάται σύμφωνα με την εμφάνιση μή αλφαριθμητικών χαρακτήρων και κεφαλαίων γραμμάτων, τα οποία χρησιμοποιούνται συνήθως για το διαχωρισμό λέξεων στα έγγραφα περιγραφής υπηρεσιών. Έτσι, για παράδειγμα, η διαδικασία διάσπασης της τιμής “GetWeatherReport” παράγει τρεις επιμέρους λέξεις, τις “get”, “weather” και “report”.
2. **Εξεύρεση Λεξικών Ριζών (Word Stemming)**. Κάθε μια λέξη από τη λίστα, που παράγεται από τη διαδικασία διάσπασης του προηγούμενου βήματος, αντικαθίσταται



Σχήμα 14: Δομή των εγγράφων διαφήμισης λειτουργιών υπηρεσιών

από τη γλωσσολογική της ρίζα, με τη βοήθεια του ηλεκτρονικού λεξικού WordNet [44]. Σε γενικές γραμμές, το βήμα αυτό οδηγεί συνήθως σε ελάτωση του μεγέθους της λίστας λέξεων, που αντιστοιχεί σε μια δεδομένη τιμή κειμένου.

- 3. Απομάκρυνση Τετριμμένων Λέξεων (Stopwords Removal).** Η λίστα με τις λέξεις-ρίζες ελατώνεται περαιτέρω, με την απομάκρυνση όλων των κοινών όρων, των οποίων η σημασιολογική συνεισφορά είναι ασήμαντη, όπως άρθρα, προθέσεις, κτλ. Για την επίτευξη αυτού του βήματος, ο Αναλυτής χρησιμοποιεί μια προκαθορισμένη λίστα από συνολικά 318 κοινές λέξεις της αγγλικής γλώσσας.
- 4. Λεξική Επέκταση (Word Expansion).** Κάθε μια από τις εναπομείνουσες λέξεις επεκτείνεται μέσω της αντικατάστασής της από τα αντίστοιχα σύνολα συνωνύμων, όπως αυτά ορίζονται στο λεξικό WordNet. Ο στόχος του τελευταίου αυτού βήματος στην επεξεργασία των τιμών κειμένου είναι ο εμπλουτισμός των, συχνά περιορισμένων σε περιεχόμενο, περιγραφών υπηρεσιών με επιπρόσθετες και σημασιολογικά σχετικές έννοιες.

Αξίζει να σημειωθεί ότι, από τα παραπάνω βήματα επεξεργασίας των τιμών κειμένου, σκόπιμα παραλείπουμε τη διαδικασία εννοιολογικής αποσαφήνισης των λέξεων (word sense disambiguation). Καθώς οι περιγραφές των υπηρεσιών ως επί το πλείστον είναι αρκετά σύντομες και περιορισμένες σε κείμενο, καθίσταται ιδιαίτερα δύσκολη η εξαγωγή κάποιου εννοιολογικού πλαισίου (context), το οποίο είναι απαραίτητο για την ομαλή λειτουργία της διαδικασίας αυτής. Το γεγονός αυτό, το οποίο έχει ήδη επισημανθεί στη βιβλιογραφία (βλ. [68]), καθώς και το κόστος εφαρμογής της εννοιολογικής αποσαφήνισης των λέξεων σε χρόνο και πόρους του συστήματος, συνετέλεσαν στην απόφασή μας να μην τη συμπεριλάβουμε στη συνολική διαδικασία επεξεργασίας φυσικής γλώσσας, που υποστηρίζεται από τον Αναλυτή του υποσυστήματος αναζήτησης.

Έτσι, με το πέρας της διαδικασίας ανάλυσης του εγγράφου περιγραφής μιας υπηρεσίας, ο Αναλυτής παράγει ένα ή περισσότερα έγγραφα διαφήμισης των προσφερόμενων από την υπηρεσία λειτουργιών (ένα έγγραφο για κάθε λειτουργία). Όπως απεικονίζεται

στο Σχήμα 14, τα έγγραφα αυτά ακολουθούν το μετα-μοντέλο της γλώσσας USQL, γεγονός που διευκολύνει τη διαδικασία αντιστοίχισης των περιεχομένων τους με τα κριτήρια αναζήτησης ενός αιτήματος USQLRequest. Ακόμα, όλες οι περιγραφές κειμένου στα έγγραφα διαφήμισης έχουν αντικατασταθεί από τα αντίστοιχα σύνολα συνωνύμων των περιεχομένων λέξεών τους. Στο σημείο αυτό, ο Αναλυτής ολοκληρώνει την εκτέλεσή του και ο έλεγχος περνάει στον Ευρετηριοποιητή (Indexer) του υποσυστήματος.

#### 4.3.4 Αποθήκευση Διαφημίσεων Υπηρεσιών

Ο Ευρετηριοποιητής (Indexer) του υποσυστήματος αναζήτησης είναι υπεύθυνος για τη συντήρηση και ενημέρωση του τοπικού καταλόγου υπηρεσιών της μηχανής αναζήτησης Proteus. Η βασική λειτουργία του συστατικού αυτού συνίσταται στην αποθήκευση των εγγράφων διαφημίσεων λειτουργιών υπηρεσιών, που παράγονται από τον Αναλυτή σύμφωνα με τις διαδικασίες που περιγράψαμε στην προηγούμενη παράγραφο, καθώς και στη δημιουργία κατάλληλων ευρετηρίων για τη γρήγορη ανάκτησή τους.

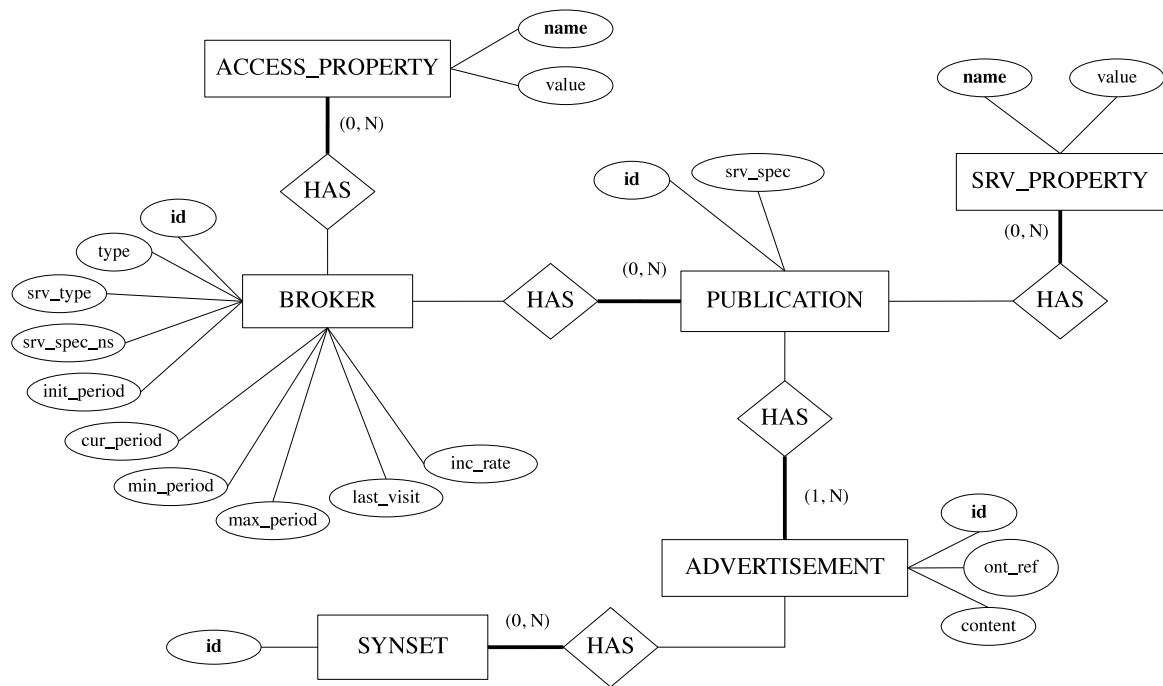
Για κάθε έγγραφο διαφήμισης λειτουργίας μιας υπηρεσίας, ο Ευρετηριοποιητής δημιουργεί μια εγγραφή  $rec_{adv} = (id_s, c, S, r_{ont})$ , όπου:

- $id_s$  είναι το αναγνωριστικό της υπηρεσίας, όπως έχει καταχωρηθεί στην αντίστοιχη εγγραφή  $rec_{pub}$
- $c$  είναι το περιεχόμενο του εγγράφου
- $S$  είναι το σύνολο των συνωνύμων, που αντιστοιχούν στην τιμή του ιδιοχαρακτηριστικού description του στοιχείου Capability του εγγράφου
- $r_{ont}$  είναι η τιμή του ιδιοχαρακτηριστικού ontologyReference του στοιχείου Capability του εγγράφου

Ακολούθως, η εγγραφή  $rec_{adv}$  αποθηκεύεται στον τοπικό κατάλογο υπηρεσιών (services index), ενώ ενημερώνονται κατάλληλα τα διάφορα ευρετήριά του. Τέλος, ο Ευρετηριοποιητής διαγράφει από τον κατάλογο όλες τις εγγραφές που αντιστοιχούν στις  $s_{del}$  διεγραμμένες δημοσιεύσεις υπηρεσιών του μεσάζοντα. Στο σημείο αυτό, το νήμα εκτέλεσης, το οποίο ο Αναζητητής είχε δημιουργήσει για την επίσκεψη στο συγκεκριμένο μεσάζοντα, την ανάκτηση και την επεξεργασία των περιεχομένων του, τερματίζει σηματοδοτώντας την ολοκλήρωση της διαδικασίας αναζήτησης.

### 4.4 Ο Τοπικός Κατάλογος Υπηρεσιών

Όπως φαίνεται στο Σχήμα 12, ο τοπικός κατάλογος υπηρεσιών του συστήματος χρησιμοποιείται από το υποσύστημα επερωτήσεων, για την ανάκτηση των αποθηκευμένων σε



Σχήμα 15: Σχήμα δεδομένων του τοπικού καταλόγου υπηρεσιών

αυτών διαφημίσεων λειτουργιών υπηρεσιών, κατά την εκτέλεση μιας επερώτησης. Συνεπώς, πριν προχωρήσουμε στην παρουσίαση του υποσυστήματος επερωτήσεων της μηχανής αναζήτησης Proteus, κρίνεται σκόπιμο να περιγράψουμε τη δομή του καταλόγου, στον οποίο αποθηκεύονται και ευρετηριοποιούνται από τον Αναζητητή του συστήματος οι διαφημίσεις λειτουργιών των διαφόρων υπηρεσιών ιστού, πλέγματος και ομότιμου-προς-ομότιμο.

Ο τοπικός κατάλογος υπηρεσιών υλοποιεί ένα απλό σχήμα δεδομένων, το οποίο απεικονίζεται στο διάγραμμα Οντοτήτων-Συσχετίσεων (Entity-Relationship Diagram, ERD) [23] του Σχήματος 15. Το σχήμα δεδομένων υποστηρίζει συνολικά πέντε (5) οντότητες, τις οποίες περιγράψουμε στη συνέχεια.

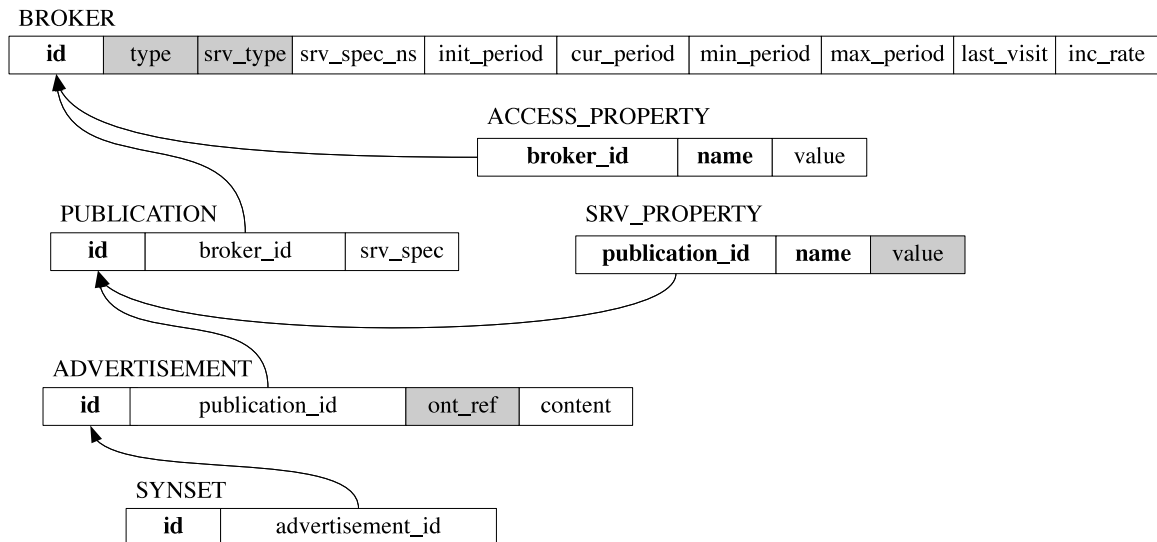
- **BROKER.** Η οντότητα αυτή χρησιμοποιείται για την αποθήκευση εγγραφών μεσαζόντων υπηρεσιών,  $rec_{broker}$ , των οποίων το περιεχόμενο προσδιορίστηκε προηγουμένως, από την εξίσωση 8.
- **ACCESS\_PROPERTY.** Κάθε εγγραφή αυτής της οντότητας αναπαριστά μια ιδιότητα πρόσβασης ενός συγκεκριμένου μεσάζοντα υπηρεσιών. Οι ιδιότητες αυτές είναι απαραίτητες για την σύνδεση της μηχανής αναζήτησης Proteus με το μεσάζοντα και την πρόσβαση στα περιεχόμενά του. Για μια συγκεκριμένη εγγραφή της οντότητας BROKER, κάθε μια ιδιότητα του αντίστοιχου στόχου αναζήτησης αναγνωρίζεται μοναδικά από το όνομά της (name) και έχει μια τιμή (value).
- **PUBLICATION.** Η οντότητα αυτή αντιπροσωπεύει τα περιεχόμενα των μεσαζόντων.

Για κάθε εγγραφή της οντότητας BROKER, μπορούν να υπάρχουν μηδέν ή περισσότερες εγγραφές τύπου  $rec_{pub}$ , που αντιστοιχούν στις υπηρεσίες που βρέθηκαν δημοσιευμένες στον αντίστοιχο μεσάζοντα. Για κάθε υπηρεσία, κρατούνται το αναγνωριστικό της (id) και το έγγραφο περιγραφής της (srv\_spec).

- **SRV\_PROPERTY.** Κάθε εγγραφή της οντότητας PUBLICATION μπορεί να έχει μηδέν ή περισσότερες εγγραφές της οντότητας SRV\_PROPERTY. Οι εγγραφές αυτές αντιστοιχούν στα περιεχόμενα του συνόλου  $\mathcal{P}$  των εγγραφών  $rec_{pub}$  και αφορούν σε ιδιότητες της υπηρεσίας, οι οποίες ταυτοποιούνται από το όνομά τους (name) και λαμβάνουν μια τιμή (value). Όπως έχει ήδη αναφερθεί, οι ιδιότητες αυτές χρησιμοποιούνται κατά τη διαδικασία της εκτέλεσης μιας επερώτησης USQL, για την αντιστοίχιση των εκφρασμένων σε αυτή φίλτρων αναζήτησης.
- **ADVERTISEMENT.** Κάθε εγγραφή της οντότητας PUBLICATION συσχετίζεται με μια ή περισσότερες εγγραφές της οντότητας ADVERTISEMENT. Ουσιαστικά, η οντότητα αυτή αναπαριστά τις διαφημίσεις των προσφερόμενων από μια υπηρεσία λειτουργιών, ενώ οι εγγραφές της έχουν τη μορφή  $rec_{adv}$ , που περιγράφηκε στην προηγούμενη ενότητα. Επιπλέον, για την επιτάχυνση ανάκτησης εγγραφών αυτής της οντότητας από το υποσύστημα επερωτήσεων, ο τοπικός κατάλογος διατηρεί ευρετήριο για το γνώρισμα ont\_ref.
- **SYNSET.** Η οντότητα αυτή αντιστοιχεί στο στοιχείο  $\mathcal{S}$  των εγγραφών τύπου  $rec_{adv}$  και χρησιμοποιείται για την αποθήκευση των συνόλων συνωνύμων μιας λειτουργίας υπηρεσίας, τα οποία παράγονται κατά την επεξεργασία του εγγράφου περιγραφής της, όπως είδαμε στην Παράγραφο 4.3.3. Ο τοπικός κατάλογος διατηρεί ευρετήριο για το αναγνωριστικό (id) καθενός συνόλου συνωνύμων, το οποίο επιτρέπει τη γρήγορη ανάκτηση των αντίστοιχων εγγραφών  $rec_{adv}$ , κατά την εκτέλεση μιας επερωτήσεως.

Ο τοπικός κατάλογος υπηρεσιών του συστήματός μας υλοποιείται ως μια σχεσιακή βάση δεδομένων (relational database), η οποία ακολουθεί το Σχεσιακό Μοντέλο Δεδομένων (Relational Data Model) [29]. Ανατρέχοντας στη λειτουργία του Ευρετηριοποιητή, αυτό σημαίνει ότι, διαγράφοντας μια εγγραφή  $rec_{pub}$ , που αντιστοιχεί σε μια από τις  $\mathcal{S}_{del}$  διαγραμμένες δημοσιεύσεις στο μεσάζοντα, απομακρύνονται ταυτόχρονα όλες οι εγγραφές  $rec_{adv}$ , που συσχετίζονται με αυτή.

Το σχεσιακό σχήμα δεδομένων του τοπικού καταλόγου υπηρεσιών, το οποίο απεικονίζεται στο Σχήμα 16, είναι κανονικοποιημένο σε μορφή BCNF (Boyce-Codd Normalization Form) [30]. Οι συσχετίσεις των οντοτήτων, που παρατηρούνται στο διάγραμμα του Σχήματος 15, υλοποιούνται ως αναφορές δευτερεύοντος κλειδιού (foreign key references), οι οποίες σημειώνονται στο διάγραμμα του σχεσιακού σχήματος με κατευθυνόμενα βέλη.



Σχήμα 16: Σχισιακό σχήμα δεδομένων του τοπικού καταλόγου υπηρεσιών

Εξάλλου, με πιο σκούρο φόντο σημειώνονται τα πεδία των πινάκων, για τα οποία δημιουργούνται ευρετήρια, επιπροσθέτως των ευρετηρίων που αντιστοιχούν στα πεδία-κλειδιά.

Στην επόμενη ενότητα, περιγράφεται ο τρόπος χρήσης του τοπικού καταλόγου κατά την εκτέλεση επερωτήσεων USQL, για την ανάκτηση των αποθηκευμένων σε αυτόν διαφημίσεων λειτουργιών υπηρεσιών.

## 4.5 Το Υποσύστημα Επερωτήσεων

Το Υποσύστημα Επερωτήσεων είναι υπεύθυνο για την επεξεργασία και εκτέλεση των επερωτήσεων USQL, που υποβάλλονται στη μηχανή αναζήτησης Proteus από τους χρήστες. Αποτελεί ουσιαστικά το σημείο εισόδου του συστήματος και προσφέρει τη λειτουργικότητά του ως υπηρεσία ιστού.

Όπως μπορούμε να δούμε στο Σχήμα 12, ο *Χειριστής Αιτήσεων (Request Handler)* αποτελεί το βασικό συστατικό του υποσυστήματος και είναι υπεύθυνος για το χειρισμό των εισερχόμενων επερωτήσεων USQL. Η εκτέλεση της επερωτήσης συντονίζεται από το συστατικό αυτό, ενώ η συνολική διαδικασία αναζήτησης πραγματοποιείται σε δύο στάδια:

1. Αρχικά, τα έγγραφα διαφημίσεων των υπηρεσιών ανακτώνται από τον τοπικό κατάλογο, αφού προηγουμένως εξεταστεί η καταλληλότητά τους βάσει των προσδιορισμένων από το χρήστη φίλτρων στο αίτημα USQLRequest. Το βήμα αυτό υλοποιείται από το *Συστατικό Ανακάλυψης (Discoverer)* του υποσυστήματος επερωτήσεων.
2. Στη συνέχεια, το σύνολο των εγγράφων διαφημίσεων υπηρεσιών που ανακτήθηκαν υπόκεινται σε ενδεδεγμένη αξιολόγηση προκειμένου να υπολογιστεί ο βαθμός αντιστοιχίας τους με τα κριτήρια αναζήτησης του χρήστη. Η επεξεργασία αυτή υποστηρίζεται

από το *Συστατικό Αντιστοίχισης (Matcher)*.

Καθώς ο αριθμός των διαθέσιμων υπηρεσιών αυξάνεται συνεχώς, μια προσέγγιση εξαντλητικής αντιστοίχισης κάθε μιας διαφήμισης υπηρεσίας που βρίσκεται στο τοπικό αποθετήριο, με τα κριτήρια αναζήτησης μιας επερώτησης USQL, θα ήταν μη ρεαλιστική, λόγω των συνεπαγόμενων υψηλών απαιτήσεων σε πόρους και των μεγάλων χρόνων απόκρισης. Η προσέγγιση διαχωρισμού της διαδικασίας αναζήτησης υπηρεσιών στα δύο παραπάνω επιμέρους βήματα είναι επομένως ουσιώδης για τη βελτίωση της απόδοσης της μηχανής αναζήτησης. Πέραν τούτου, η δυνατότητα επιλογής ενός υποσυνόλου από το σύνολο των διαθέσιμων περιγραφών υπηρεσιών για αντιστοίχιση αποτελεί κοινή πρακτική και αναφέρεται από αρκετές ερευνητικές εργασίες, όπου προτείνονται διάφορες τεχνικές υλοποίησης (π.χ. σημασιολογική ομαδοποίηση μητρώων [94] ή υπηρεσιών [92], κατακερματισμός δικτύου και περιγραφών υπηρεσιών [102], κτλ.).

Στη συνέχεια, περιγράφουμε αναλυτικά τη διαδικασία αναζήτησης υπηρεσιών, όπως υποστηρίζεται και υλοποιείται από το υποσύστημα επερωτήσεων της μηχανής αναζήτησης Proteus.

#### **4.5.1 Αρχικοποίηση και Επέκταση Επερωτήσεων USQL**

Με την άφιξη μιας επερώτησης μέσω της διεπαφής, που προσφέρει το υποσύστημα επερωτήσεων, αρχικοποιείται μια νέα εργασία συστήματος (system job) σε ένα ξεχωριστό νήμα εκτέλεσης. Με τη σειρά της, η εργασία αυτή αρχικοποιεί το Χειριστή Αιτήσεων, για το χειρισμό της επερώτησης. Κατά την αρχικοποίηση, διεξάγεται ένας συντακτικός έλεγχος του αιτήματος USQLRequest, βάσει του συντακτικού XML που ορίζει το σχήμα της γλώσσας USQL. Σε περίπτωση που το εισερχόμενο αίτημα δεν είναι συντακτικά έγκυρο, η εκτέλεση της εργασίας αναζήτησης διακόπτεται και Χειριστής Αιτήσεων τερματίζει τη λειτουργία του.

Αφού περάσει επιτυχώς το συντακτικό έλεγχο, το αίτημα USQLRequest υφίσταται από το Χειριστή Αιτήσεων μια διαδικασία επέκτασης, προκειμένου να εμπλουτιστούν τα περιεχόμενά του. Συγκεκριμένα, με τη βοήθεια του ηλεκτρονικού λεξικού WordNet, όλα τα κριτήρια αναζήτησης που περιλαμβάνουν τιμές κειμένου αναλύονται λεκτικά, έτσι ώστε οι τιμές κειμένου να αντικατασταθούν από τα αντίστοιχα σύνολα συνωνύμων. Όπως και στην περίπτωση ανάλυσης των περιγραφών υπηρεσιών, που ανακτώνται από τους διαφόρους μεσάζοντες υπηρεσιών, όπου έχουν δημοσιευτεί, η λεκτική ανάλυση ακολουθεί τα βήματα επεξεργασίας φυσικής γλώσσας που περιγράφονται στην παράγραφο 4.3.3.

#### **4.5.2 Ανάκτηση Διαφημίσεων Υπηρεσιών**

Αφού η επερώτηση USQL επεκταθεί και οι τιμές κειμένου των διαφόρων κριτηρίων αναζήτησης αντικατασταθούν με τα αντίστοιχα σύνολα συνωνύμων τους, ακολουθεί η εξαγωγή



```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <USQLRequest specVersion="2.0" id="1" serviceType="WebService" xmlns="http://usql.di.uoa.gr">
3   <SearchFilters>
4     <Filter serviceProperty="ServiceProvider" excludeValues="true">
5       <Option value="Xignite" useWildcards="true"/>
6       <Option value="ServiceObjects" useWildcards="true"/>
7       <Option value="IventaStage" useWildcards="true"/>
8     </Filter>
9   </SearchFilters>
10  <SearchCriteria>
11    <Capability>
12      <Description>Get weather report</Description>
13    </Capability>
14  </SearchCriteria>
15  <SearchTargets />
16 </USQLRequest>
```

### Παράδειγμα 7: Μια απλή επερώτηση υπηρεσίας ιστού

των φίλτρων αναζήτησης και η χρήση τους για την ανάκτηση των εγγράφων διαφημίσεων υπηρεσιών από τον τοπικό κατάλογο του συστήματος. Στα φίλτρα αναζήτησης, που εξάγονται από το Χειριστή Αιτήσεων, εκτός από τα περιεχόμενα του στοιχείου SearchFilters, συμπεριλαμβάνονται επίσης:

1. ο τύπος της επιθυμητής υπηρεσίας, όπως προσδιορίζεται από το ιδιοχαρακτηριστικό serviceType του αιτήματος,
2. η σημασιολογική περιγραφή της επιθυμητής υπηρεσίας, εφόσον έχει προσδιοριστεί με τη χρήση του ιδιοχαρακτηριστικού ontologyReference του κριτηρίου αναζήτησης Capability, καθώς και
3. τα σύνολα συνωνύμων, που αντιστοιχούν στην περιγραφή της λειτουργικότητάς της, εφόσον αυτή έχει δοθεί από τον αιτούντα, όπως αυτά περιέχονται, πλέον, μετά την επέκταση της επερώτησης, στο στοιχείο Description του κριτηρίου αναζήτησης Capability.

Θεωρώντας τη σχεσιακή υλοποίηση του σχήματος δεδομένων του τοπικού καταλόγου υπηρεσιών, οι τιμές των φίλτρων αναζήτησης χρησιμοποιούνται στη συνέχεια από το Συστατικό Ανακάλυψης (Discoverer) του υποσυστήματος επερωτήσεων, για τη δημιουργία και εκτέλεση της κατάλληλης επερώτησης SQL [21] [7], προκειμένου να ανακτηθούν από τον τοπικό κατάλογο τα έγγραφα διαφημίσεων λειτουργιών υπηρεσιών.

Για την καλύτερη κατανόηση της όλης διαδικασίας, ας θεωρήσουμε ως παράδειγμα το αίτημα USQLRequest του Παραδείγματος 7. Όπως μπορεί κανείς να δει, ο αιτών έχει προσδιορίσει ως φίλτρο αναζήτησης ένα σύνολο μη επιθυμητών παρόχων υπηρεσιών, ενώ επίσης έχει δώσει μια σύντομη περιγραφή της επιθυμητής λειτουργικότητας, για την υπηρεσία ιστού που αναζητά. Ο μετασχηματισμός του αιτήματος αυτού από το Συστατικό Ανακάλυψης (Discoverer) δίνει την επερώτηση SQL, που καταγράφεται στο Παράδειγμα 8.

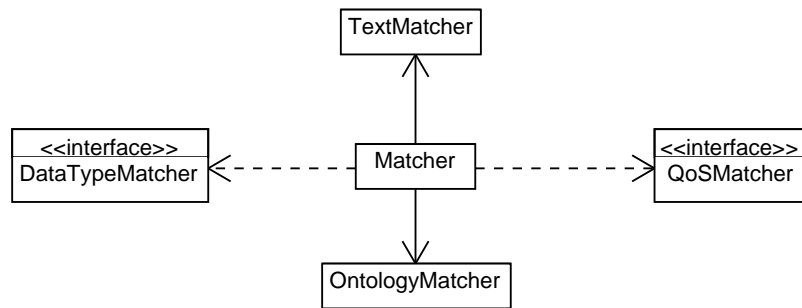
```
1 SELECT DISTINCT advertisement.content
2 FROM broker, publication, srv_property, advertisement, synset
3 WHERE publication.broker_id = broker.id
4 AND advertisement.publication_id = publication.id
5 AND synset.advertisement_id = advertisement.id
6 AND srv_property.publication_id = publication_id
7 AND broker.srv_type = 'WebService'
8 AND (srv_property.name = 'ServiceProvider'
9 AND (srv_property.value NOT LIKE '%Xignite%')
10 AND (srv_property.value != 'ServiceObjects')
11 AND (srv_property.value != 'IventaStage'))
12 AND synset.id IN (
13 'SID-06207199-N',
14 'SID-02707251-V',
15 'SID-06409752-N',
16 'SID-00965390-V',
17 'SID-00965035-V',
18 'SID-02038278-V',
19 'SID-01928283-V',
20 'SID-07218470-N',
21 'SID-06681551-N',
22 'SID-11524662-N',
23 'SID-07217924-N',
24 'SID-00965687-V',
25 'SID-00965542-V',
26 'SID-01400464-A',
27 'SID-07391516-N',
28 'SID-07220300-N',
29 'SID-00967098-V',
30 'SID-00966809-V',
31 'SID-00275088-V');
```

Παράδειγμα 8: Επερώτηση SQL για την ανάκτηση διαφημίσεων υπηρεσιών

### 4.5.3 Μηχανισμοί Αντιστοίχισης

Αν και οι διαφημίσεις λειτουργιών υπηρεσιών, που ανακτώνται από τον τοπικό κατάλογο υπηρεσιών του συστήματος, ικανοποιούν εν πρώτοις τα φίλτρα αναζήτησης του χρήστη, χρειάζεται να ελεγχθούν λεπτομερώς, προκειμένου να υπολογιστεί ο ακριβής βαθμός αντιστοιχίας τους με το σύνολο των κριτηρίων αναζήτησης, που περιλαμβάνει το αίτημα USQLRequest. Η αντιστοίχιση των περιεχομένων του εγγράφου διαφήμισης μιας λειτουργίας υπηρεσίας με τα κριτήρια αναζήτησης της επερώτησης USQLRequest πραγματοποιείται από το συστατικό αντιστοίχισης (Matcher) του υποσυστήματος επερωτήσεων. Το συστατικό αυτό εφαρμόζει το μαθηματικό μοντέλο της USQL, που παρουσιάστηκε στην Ενότητα 3.5, προκειμένου να υπολογίσει το συνολικό βαθμό αντιστοιχίας μιας διαφημιζόμενης λειτουργίας υπηρεσίας.

Τα αποτελέσματα της διαδικασίας αντιστοίχισης χρησιμοποιούνται στη συνέχεια από το Χειριστή Αιτήσεων, για την κατασκευή του εγγράφου USQLResponse, ως απόκριση στην επερώτηση. Για κάθε έγγραφο διαφήμισης μιας λειτουργίας υπηρεσίας, του οποίου ο συνολικός βαθμός αντιστοιχίας είναι μεγαλύτερος του μηδενός, ή ισούται/υπερβαίνει το κατώφλι minMatch (σε περίπτωση που αυτό έχει προσδιοριστεί στην επερώτηση USQLRequest), δημιουργείται και προστίθεται στο έγγραφο USQLResponse το αντίστοιχο στοιχείο Match. Έτσι, το υποσύστημα επερωτήσεων ολοκληρώνει τη λειτουργία του επιστρέφοντας στο χρήστη το τελικό έγγραφο USQLResponse, με τα αποτελέσματα της αναζήτησης.



Σχήμα 17: Αρχιτεκτονική του συστατικού αντιστοίχισης

Από αρχιτεκτονική άποψη, το συστατικό αντιστοίχισης είναι αρθρωτό και χαρακτηρίζεται από μεγάλη ευελιξία, η οποία υποστηρίζεται από τον εγγενή μηχανισμό επεκτάσεων του συστήματος. Όπως φαίνεται στο διάγραμμα του Σχήματος 17, οι διαδικασίες αντιστοίχισης τιμών κειμένου και οντολογικών εννοιών υλοποιούνται από τα συστατικά TextMatcher και OntologyMatcher, αντίστοιχα. Επίσης, οι διαδικασίες αντιστοίχισης τύπων δεδομένων, καθώς και των διαφόρων ποιοτικών χαρακτηριστικών της υπηρεσίας με τα αντίστοιχα κριτήρια αναζήτησης, πραγματοποιούνται από κατάλληλα συνδεδεμένα συστατικά (plug-ins), τα οποία χρησιμοποιούνται μέσω των αντίστοιχων διεπαφών DataTypeMatcher και QoSMatcher. Κατά αυτό τον τρόπο, το συστατικό αντιστοίχισης μπορεί εύκολα να ενσωματώσει υπάρχοντες αλγορίθμους αντιστοίχισης, που αφορούν στις προαναφερθείσες κατηγορίες τιμών. Επιπλέον, μπορεί να επεκταθεί για την υποστήριξη, στο επίπεδο αντιστοίχισης, ενός ανοικτού συνόλου ποιοτικών χαρακτηριστικών, καθώς και διαφόρων σχημάτων τύπων δεδομένων.

Σύμφωνα με την Εξίσωση 2, το μαθηματικό μοντέλο της USQL είναι ανεξάρτητο από τους αλγορίθμους αντιστοίχισης τιμών κειμένου και οντολογικών εννοιών, καθώς οι αντίστοιχες συναρτήσεις  $\alpha$  και  $\beta$  εμφωλεύουν τις λεπτομέρειές τους. Ακόμα, σύμφωνα με την Εξίσωση 3, παρόμοια ανεξαρτησία ισχύει αναφορικά με τους αλγορίθμους αντιστοίχισης τύπων δεδομένων. Αναμένεται, ότι, κάθε μια υλοποίηση της διεπαφής DataTypeMatcher παρέχει τη δική της εκδοχή της συνάρτησης  $\gamma$ . Τέλος, οι υλοποιήσεις της διεπαφής QoSMatcher υποστηρίζουν τους κατάλληλους αλγορίθμους για την αντιστοίχιση τιμών των διαφόρων ειδών ποιοτικών χαρακτηριστικών μιας υπηρεσίας, όπως π.χ. το κόστος, η διαθεσιμότητα, η αξιοπιστία, κτλ.

Στη συνέχεια, περιγράφουμε τις υλοποιήσεις των συναρτήσεων  $\alpha$  και  $\beta$ , από τα συστατικά TextMatcher και OntologyMatcher, καθώς αποτελούν τμήμα του συστήματος Proteus.

**Αντιστοίχιση Τιμών Κειμένου.** Για την αντιστοίχιση τιμών κειμένου, υιοθετήσαμε το Μοντέλο Διανυσματικού Χώρου (Vector Space Model) [103]. Σύμφωνα με το μοντέλο αυτό, όλες οι τιμές κειμένου, τόσο στις διαφημίσεις λειτουργιών υπηρεσιών όσο και στα κριτήρια αναζήτησης της επερώτησης, αναπαριστώνται ως *έγγραφα* συνόλων συνωνύμων (synset

documents). Χάρη στην προπαρασκευαστική διαδικασία ανάλυσης των περιγραφών υπηρεσιών, που παρουσιάστηκε στην Παράγραφο 4.3.3, αλλά και στη διαδικασία επέκτασης επερωτήσεων, που περιγράφηκε στην Παράγραφο 4.5.1, τα έγγραφα αυτά έχουν ήδη δημιουργηθεί και είναι διαθέσιμα κατά τη φάση της αντιστοίχισης.

Για κάθε έγγραφο συνόλων συνωνύμων, δημιουργείται ένα διάνυσμα  $\vec{w}$ , το οποίο περιλαμβάνει τα βάρη των συνόλων συνωνύμων. Τα βάρη αυτά υπολογίζονται με τη χρήση της μετρικής Term Frequency - Inverse Document Frequency (TF-IDF) [100]. Σύμφωνα με τη μετρική TF-IDF, τα σύνολα συνωνύμων που εμφανίζονται σε λίγα έγγραφα είναι σημαντικότερα (δηλαδή έχουν μεγαλύτερο βάρος) από εκείνα που εμφανίζονται σε πολλά έγγραφα (δηλαδή είναι κοινά, και άρα ασήμαντου σημασιολογικού περιεχομένου).

Επιπροσθέτως, προκειμένου να ελαχιστοποιηθεί η διάσταση του διανυσματικού χώρου, τα έγγραφα συνόλων συνωνύμων χωρίζονται σε πέντε, ανεξάρτητα μεταξύ τους, σύνολα. Έτσι, δημιουργείται ένα σύνολο εγγράφων, αντίστοιχα, για τα έγγραφα που αντιστοιχούν στις τιμές κειμένου των στοιχείων Capability, Message, MessageElement, Resource και ResourceProperty, που περιλαμβάνονται στις διαφημίσεις λειτουργιών υπηρεσιών, αλλά και στις επερωτήσεις. Κατ' αυτό τον τρόπο, ο διανυσματικός χώρος καθεμίας συλλογής εγγράφων καθορίζεται από τον αριθμό των μοναδικών συνόλων συνωνύμων, που εμφανίζονται στα έγγραφά της.

Βάσει των παραπάνω, ο βαθμός αντιστοιχίας μεταξύ δύο δεδομένων εγγράφων συνόλων συνωνύμων, που ανήκουν στην ίδια συλλογή, υπολογίζεται με τη χρήση του συντελεστή συνημιτόνου (cosine coefficient), ως εξής:

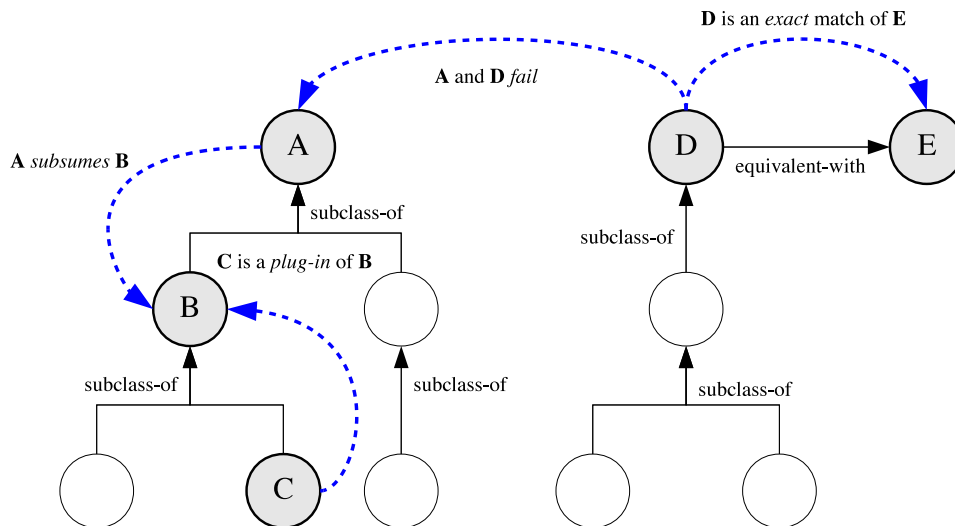
$$\cos(\vec{w}_r, \vec{w}_a) = \frac{\vec{w}_r^T \cdot \vec{w}_a}{\sqrt{\vec{w}_r^T \cdot \vec{w}_r} \cdot \sqrt{\vec{w}_a^T \cdot \vec{w}_a}} \quad (13)$$

Στην παραπάνω εξίσωση,  $\vec{w}_r$  και  $\vec{w}_a$  είναι τα διανύσματα βαρών TF-IDF, που αντιστοιχούν στα δύο δεδομένα έγγραφα συνόλων συνωνύμων. Έτσι, η συνάρτηση  $\alpha$  της εξίσωσης 2 υλοποιείται από το σύστημα Proteus ως εξής:

$$\alpha(v_d^r, v_d^a) = \cos(f_{TFIDF}(v_d^r), f_{TFIDF}(v_d^a)) \quad (14)$$

Σημειώνεται, ότι η συνάρτηση  $f_{TFIDF}(v_d) = \vec{w}$  της εξίσωσης 14 αναλαμβάνει την απεικόνιση μιας τιμής κειμένου,  $v_d$ , στο αντίστοιχο διάνυσμα συνόλων συνωνύμων,  $\vec{w}$ .

**Αντιστοίχιση Οντολογικών Εννοιών.** Για τον υπολογισμό του βαθμού αντιστοιχίας μιας οντολογικής έννοιας, που έχει προσδιοριστεί σε ένα έγγραφο USQLRequest με τη χρήση του ιδιοχαρακτηριστικού ontologyReference, με αυτές που έχουν προσδιοριστεί στις περιγραφές λειτουργιών των διαφόρων υπηρεσιών, λαμβάνουμε υπόψιν την ιεραρχική τους σχέση, όπως αυτή διαμορφώνεται στον γράφο εννοιών της οντολογίας. Συγκεκριμένα, ποσοτικοποιούμε τις σημασιολογικές σχέσεις *exact*, *plug-in*, *subsume* και *fail*, οι οποίες



Σχήμα 18: Ιεραρχικές σχέσεις στο γράφο μιας οντολογίας

εμφανίζονται εκτενώς στη βιβλιογραφία (π.χ. [109], [14]) και απεικονίζονται γραφικά στο Σχήμα 18, για να υπολογίσουμε το βαθμό αντιστοιχίας δύο σημασιολογικών εννοιών. Για τις σημασιολογικές αυτές σχέσεις, ισχύει ότι  $fail \prec plug-in \prec subsume \prec exact$ .

Έτσι, η συνάρτηση  $\beta$  της εξίσωσης 2 υλοποιείται από το συστατικό *OntologyMatcher* του συστήματος *Proteus* αντιστοιχίζοντας αριθμητικές τιμές στις διακριτές αυτές σημασιολογικές σχέσεις, ως εξής:

$$\beta(v_o^r, v_o^a) = \begin{cases} 1 & , \text{ if } v_o^a \text{ is an exact match of } v_o^r \\ \sigma & , \text{ if } v_o^a \text{ subsumes } v_o^r \\ \rho & , \text{ if } v_o^a \text{ is plug-in of } v_o^r \\ 0 & , \text{ else} \end{cases} \quad (15)$$

Στην εξίσωση 15, οι  $\rho$  και  $\sigma$  είναι αριθμητικές τιμές, τέτοιες ώστε  $0 < \rho < \sigma < 1$ . Σημειώνεται, ότι, όπως έχει αναφερθεί στη βιβλιογραφία [121], ο προσδιορισμός των τιμών αυτών είναι μια επίπονη διαδικασία, που εξαρτάται από το είδος της οντολογίας και τη θεματική της περιοχή. Καθώς η υλοποίηση αυτής της διαδικασίας ξεφεύγει από το ερευνητικό αντικείμενο της διατριβής, το συστατικό *OntologyMatcher* του υποσυστήματος επερωτήσεων λειτουργεί με την προϋπόθεση ότι, για κάθε υποστηριζόμενη οντολογία, δίνονται οι τιμές των παραμέτρων  $\rho$  και  $\sigma$ .

## 4.6 Μηχανισμός Επεκτάσεων

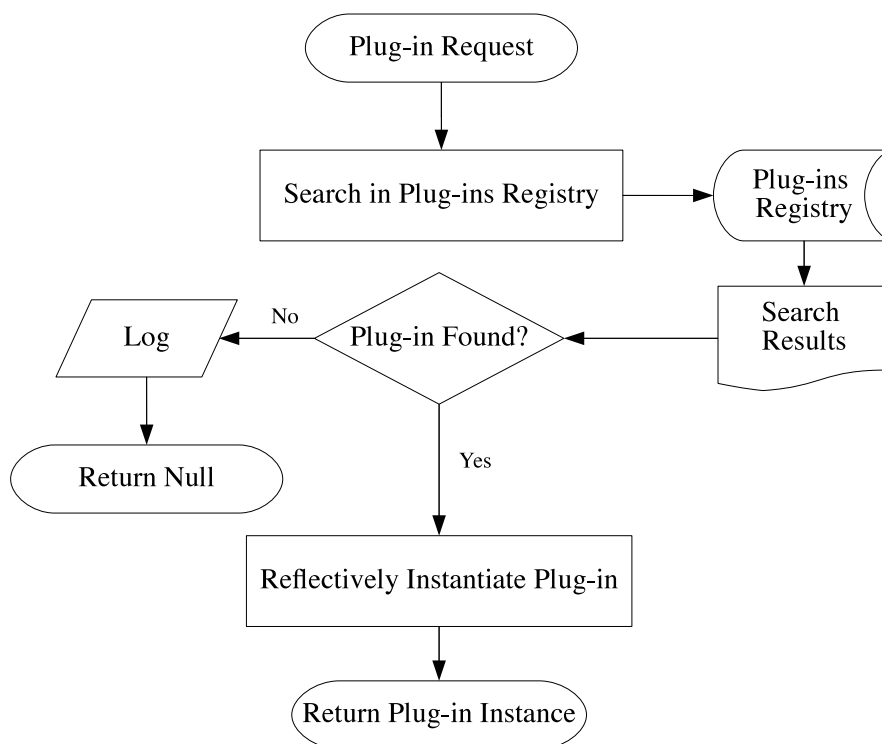
Ένα από τα σημαντικότερα χαρακτηριστικά της μηχανής αναζήτησης *Proteus* είναι η επεκτασιμότητά της, χάρη στην οποία μπορεί να αντιμετωπίσει αποτελεσματικά το πρόβλημα της ετερογένειας των υπαρχουσών υπηρεσιών, των περιγραφών τους, καθώς και των

μηχανισμών αναζήτησής τους. Πράγματι, όπως ήδη περιγράφηκε στις προηγούμενες παραγράφους, τόσο το υποσύστημα αναζήτησης όσο και το υποσύστημα επερωτήσεων της μηχανής αναζήτησης Proteus βασίζονται στη χρήση διαφόρων *συνδεδεμένων συστατικών (plug-in components)*, για τη διεκπεραίωση μερών της λειτουργικότητάς τους. Η χρήση των συστατικών αυτών υποστηρίζεται από ένα μηχανισμό επεκτάσεων, ο οποίος επιτρέπει την απρόσκοπτη ενσωμάτωση επιπρόσθετων συστατικών στο κυρίως σώμα του συστήματος Proteus. Συγκεκριμένα, ο μηχανισμός αυτός αποτελείται από ένα Μητρώο Συνδεδεμένων Συστατικών (Plug-ins Registry), στο οποίο καταχωρούνται τα στοιχεία των συνδεδεμένων συστατικών, επιτρέποντας τη δυναμική ανάκτηση, αρχικοποίηση και κλήση τους, οπότε-δήποτε η χρήση τους απαιτείται από το σύστημα.

Έτσι, η δυνατότητα επεκτάσεων προσδίδει στο σύστημα Proteus τα εξής πλεονεκτήματα:

- Συμβατότητα με τα διάφορα πρότυπα περιγραφής υπηρεσιών και τους υπάρχοντες μηχανισμούς αναζήτησης υπηρεσιών.
- Αναζήτηση ετερογενών υπηρεσιών μέσω μιας ενιαίας και ανεξάρτητης από τις διάφορες υπηρεσιοστρεφείς τεχνολογίες διεπαφής.
- Παραμετροποίηση, προσαρμογή και χρήση σε ετερογενή περιβάλλοντα εργασίας.
- Δυνατότητα ενσωμάτωσης νέων χαρακτηριστικών σε συνάρτηση με την εμφάνιση νέων υπηρεσιοστρεφών τεχνολογιών.

Η διεργασία που υλοποιεί ο μηχανισμός επεκτάσεων του συστήματος Proteus απεικονίζεται στο διάγραμμα ροής (flow chart) του Σχήματος 19. Ο μηχανισμός ενεργοποιείται οπότεδήποτε κάποιο από τα δύο υποσυστήματα χρειαστεί ένα συγκεκριμένο συστατικό plug-in, στο πλαίσιο της εκτέλεσης κάποιας λειτουργίας. Ακολουθεί μια αναζήτηση στο Μητρώο Συνδεδεμένων Συστατικών του συστήματος, βάσει κριτηρίων που εξαρτώνται από τον τύπο του συστατικού που αναζητείται. Σε περίπτωση που το ζητούμενο συστατικό δεν έχει εγγραφεί στο μητρώο, το γεγονός καταγράφεται στο ημερολόγιο του συστήματος και η διεργασία τερματίζει επιστρέφοντας μια τιμή NULL, προκειμένου να υποδηλώσει ότι το συστατικό δε βρέθηκε. Διαφορετικά, από την ανακτηθείσα εγγραφή του συστατικού, εξάγεται το όνομα της κλάσης υλοποίησής του και δημιουργείται ένα νέο στιγμιότυπό της σύμφωνα με το ανακλαστικό πρότυπο σχεδίασης (reflective design pattern) [48], [20]. Η διεργασία τερματίζει επιστρέφοντας στο σύστημα το νέο αυτό στιγμιότυπο, έτοιμο προς χρήση. Στο σημείο αυτό, θα πρέπει να αναφερθεί ότι το στιγμιότυπο διατηρείται στη μνήμη του συστήματος μόνο για το χρονικό διάστημα, κατά το οποίο χρησιμοποιείται. Συνεπώς, η διεργασία του μηχανισμού επεκτάσεων επιστρέφει σε κάθε περίπτωση ένα νέο στιγμιότυπο του ζητούμενου συνδεδεμένου συστατικού, εφόσον αυτό έχει δηλωθεί στο μητρώο.



Σχήμα 19: Διεργασία του μηχανισμού επεκτάσεων

Στη συνέχεια, περιγράφουμε τα είδη των συνδεδεμένων συστατικών (plug-ins), που μπορούν να ενσωματωθούν στο σύστημα Proteus.

**Συνδεδεμένα συστατικά τύπου Harvester:** Χρησιμοποιούνται από τον Επισκέπτη (Visitor) του υποσυστήματος αναζήτησης, για την ανάκτηση των δημοσιεύσεων υπηρεσιών σε διάφορους ετερογενείς μεσάζοντες υπηρεσιών. Η επιλογή τους πραγματοποιείται με βάση τον τύπο του μεσάζοντα, γεγονός που συνεπάγεται ότι, για την υποστήριξη αναζήτησης υπηρεσιών σε μεσάζοντες μιας συγκεκριμένης τεχνολογίας (π.χ. UDDI, ebXML, κτλ.), απαιτείται από το σύστημα Proteus η ύπαρξη του αντίστοιχου συνδεδεμένου συστατικού Harvester. Κάθε τέτοιο συστατικό δέχεται ως είσοδο μια εγγραφή  $rec_{broker}$ , που προσδιορίζει το στόχο της αναζήτησης, ενώ επιστρέφει ως έξοδο μια ή περισσότερες εγγραφές  $rec_{pub}$ , που αντιστοιχούν στις δημοσιευμένες σε αυτόν υπηρεσίες.

**Συνδεδεμένα συστατικά τύπου Parser:** Χρησιμοποιούνται από τον Αναλυτή (Analyzer) του υποσυστήματος αναζήτησης, για την επεξεργασία των περιεχομένων των διαφόρων ετερογενών εγγράφων περιγραφών υπηρεσιών ιστού, πλέγματος και ομότιμου-προς-ομότιμο. Για κάθε τέτοιο έγγραφο, ο Αναλυτής επιλέγει το κατάλληλο συστατικό Parser βάσει του χώρου ονομάτων (namespace) του σχήματος της γλώσσας περιγραφής υπηρεσιών, με την οποία έχει μορφοποιηθεί (π.χ. WSDL, OWL-S, κτλ.). Κάθε συνδεδεμένο συστατικό Parser δέχεται ως είσοδο μια εγγραφή  $rec_{pub}$ , που αντιστοιχεί στην υπηρεσία, της οποίας η πε-

ριγραφή πρόκειται να αναλυθεί, και επιστρέφει ένα ή περισσότερα έγγραφα διαφήμισης των λειτουργιών της συγκεκριμένης υπηρεσίας. Έτσι, με τη χρήση των συστατικών Parser, επιτυγχάνεται πλήρης ανεξαρτησία του συστήματος Proteus από τις διάφορες τεχνολογίες περιγραφής υπηρεσιών, αφού, αφενός, όλες οι τεχνικές λεπτομέρειες της επεξεργασίας του εγγράφου περιγραφής μιας υπηρεσίας εμφωλεύονται στην υλοποίηση του κατάλληλου συστατικού Parser, ενώ, αφετέρου, το αποτέλεσμα της επεξεργασίας είναι μια σειρά εγγράφων με ομοιογενή και συμβατή με τη γλώσσα USQL μορφοποίηση.

**Συνδεδόμενα συστατικά τύπου QoS Matcher:** Ενσωματώνονται δυναμικά στο συστατικό αντιστοίχισης (Matcher) του υποσυστήματος επερωτήσεων, προκειμένου να υποστηριχθεί η αξιολόγηση απαιτήσεων προς τα ποιοτικά χαρακτηριστικά της αναζητούμενης υπηρεσίας. Η αναγνώριση του κατάλληλου συστατικού QoS Matcher από το συστατικό αντιστοίχισης βασίζεται στην τιμή του ιδιοχαρακτηριστικού description του εκάστοτε στοιχείου απαίτησης QoSRequirement. Η κοινή διεπαφή των συνδεδόμενων συστατικών QoS Matcher ορίζει ως είσοδο δύο αντικείμενα, που αναπαριστούν το κριτήριο αναζήτησης QoSRequirement και την αντίστοιχη ιδιότητα της υπηρεσίας, σύμφωνα με το μετα-μοντέλο της USQL. Ως έξοδος, ορίζεται μια αριθμητική τιμή μεταξύ 0 και 1, που υποδηλώνει το βαθμό αντιστοιχίας των δύο αντικειμένων εισόδου. Τα συνδεδόμενα συστατικά QoS Matcher προσδίδουν στο σύστημα Proteus τη δυνατότητα υποστήριξης, κατά τη διαδικασία αντιστοίχισης, ενός ανοικτού συνόλου ποιοτικών χαρακτηριστικών υπηρεσιών. Ταυτόχρονα καθιστούν το υποσύστημα επερωτήσεων ανεξάρτητο από τους επιμέρους αλγόριθμους αντιστοίχισης, που χρησιμοποιούνται για την αξιολόγηση καθενός είδους κριτηρίου αναζήτησης προς τα ποιοτικά αυτά χαρακτηριστικά.

**Συνδεδόμενα συστατικά τύπου Datatype Matcher:** Χρησιμεύουν για την αντιστοίχιση τύπων δεδομένων, όπως αυτοί εκφράζονται με την κλάση DataType του μετα-μοντέλου της USQL. Η ενσωμάτωση ενός συστατικού αυτού του είδους στο υποσύστημα επερωτήσεων πραγματοποιείται βάσει της τιμής του ιδιοχαρακτηριστικού namespace του στοιχείου DataType, στο υπό αξιολόγηση στοιχείο απαίτησης με τύπο. Έτσι, το συστατικό αντιστοίχισης του Proteus μπορεί να υποστηρίξει διάφορες μετα-γλώσσες, όπως η XML Schema [127] ή η JSON [59], που χρησιμοποιούνται για την περιγραφή των τύπων δεδομένων στα μέρη ενός μηνύματος εισόδου/εξόδου, ή στις ιδιότητες ενός πόρου. Η κοινή διεπαφή, που ορίζεται από τις προδιαγραφές του συστήματος Proteus για όλα τα συνδεδόμενα συστατικά τύπου Datatype Matcher, προσδιορίζει ως είσοδο δύο αντικείμενα τύπου DataType, που αντιστοιχούν στον αιτούμενο και διαφημιζόμενο τύπο δεδομένων. Παρόμοια με τη διεπαφή των συστατικών QoS Matcher, ως έξοδος ορίζεται μια αριθμητική τιμή μεταξύ 0 και 1, που αντιπροσωπεύει το βαθμό αντιστοιχίας των δύο αντικειμένων εισόδου.



## 4.7 Μηχανισμός Κλιμάκωσης

Σε μια προσπάθεια ενίσχυσης των δυνατοτήτων κλιμάκωσης του συστήματος Proteus ως προς (1) τον αριθμό των στόχων αναζήτησης, (2) τον αριθμό των περιγραφών υπηρεσιών, αλλά και (3) τον αριθμό των εισερχόμενων επερωτήσεων, ο σχεδιασμός του περιλαμβάνει ένα μηχανισμό αντιγράφων (replication mechanism), που βασίζεται στη χρήση κατανεμημένων συστατικών. Συγκεκριμένα, τα δύο του υποσυστήματα, ο Αναζητητής και ο Επεξεργαστής Επερωτήσεων, μπορούν να προσφέρουν εναλλακτικά τη λειτουργικότητά τους ως υπηρεσίες ιστού. Έτσι, είναι δυνατή η δημιουργία πολλαπλών αντιγράφων και η επιλεγμένη χρήση τους, μέσω απομακρυσμένων κλήσεων. Με τον τρόπο αυτό, επιτυγχάνεται διαμοιρασμός του φόρτου εργασίας κατά τις διαδικασίες επίσκεψης σε μεσάζοντες υπηρεσιών, ανάκτησης των περιεχομένων τους, ανάγνωσης, μετατροπής και ευρετηριοποίησης των ανακτούμενων περιγραφών υπηρεσιών, καθώς και εκτέλεσης των επερωτήσεων, με απώτερο σκοπό την αποσυμφόρηση του συστήματος και την ελλάτωση των απαιτήσεων σε πόρους της υποδομής υλικού, που το φιλοξενεί.

Όλα τα αντίγραφα-υπηρεσίες ιστού των δύο υποσυστημάτων καθιστώνται διαθέσιμα σε ένα στιγμιότυπο της μηχανής αναζήτησης Proteus με την καταχώρηση από το διαχειριστή του συστήματος των αντίστοιχων εγγραφών τύπου  $rec_{rep} = (t_{rep}, addr_{rep}, d_{use}, n_{req}, req_{max})$ , όπου:

- $t_{rep}$  είναι ο τύπος του αντιγράφου (“crawler” ή “queryProcessor”),
- $addr_{rep}$  είναι η διεύθυνση ακροδέκτη (endpoint address) της αντίστοιχης υπηρεσίας ιστού,
- $d_{use}$  είναι η χρονική στιγμή τελευταίας χρήσης του συγκεκριμένου αντιγράφου,
- $n_{req}$  είναι ο τρέχων αριθμός αιτημάτων, που το αντίγραφο εξυπηρετεί και
- $req_{max}$  είναι ο μέγιστος αριθμός αιτημάτων, που το αντίγραφο μπορεί να εξυπηρετήσει ταυτόχρονα.

Κάθε φορά που το σύστημα δημιουργεί ένα νέο νήμα εκτέλεσης για τη διεξαγωγή αναζήτησης σε ένα μεσάζοντα, ή για την επεξεργασία μιας εισερχόμενης επερώτησης, διενεργείται ένας έλεγχος, προκειμένου να διαπιστωθεί αν το τοπικό στιγμιότυπο του αντίστοιχου υποσυστήματος έχει φτάσει στα όρια της απόδοσής του, σχετικά με τον αριθμό ταυτόχρονων αιτημάτων που εξυπηρετεί. Η τιμή του ορίου αυτού είναι παραμετροποιήσιμη από το διαχειριστή του συστήματος. Σε περίπτωση που ο αριθμός ταυτόχρονων αιτημάτων ισούται με το μέγιστο επιτρεπτό όριο, ξεκινά μια διαδικασία επιλογής και χρήσης του κατάλληλου αντιγράφου από το εκάστοτε υποσύστημα, η οποία έχει ως εξής:

Πίνακας 4: Παράδειγμα αντιγράφων του υποσυστήματος αναζήτησης

Crawler	Last Used ( $d_{use}$ )	Concurrent Jobs ( $n_{req}$ )	Max Concurrent Jobs ( $req_{max}$ )
c1	Aug 12, 2009 2:01:14 AM	5	5
c2	Aug 12, 2009 8:56:23 PM	4	5
c3	Aug 13, 2009 3:43:52 AM	2	6
c4	Aug 13, 2009 3:45:41 AM	2	2
c5	Aug 21, 2009 7:05:13 AM	4	6

1. Καταρχάς, από το σύνολο των εγγραφών  $rec_{rep}$ , απομακρύνονται εκείνες για τις οποίες ισχύει  $n_{req} = req_{max}$ . Με άλλα λόγια, αποφεύγεται η χρήση αντιγράφων, τα οποία βρίσκονται στο μέγιστο των δυνατοτήτων τους, από την άποψη της ταυτόχρονης εξυπηρέτησης αιτημάτων.
2. Στη συνέχεια, δοθείσης της τρέχουσας χρονικής στιγμής  $d_{now}$ , κατά την οποία πραγματοποιείται η διαδικασία επιλογής, επιλέγεται από τις εναπομείναντες εγγραφές εκείνη, για την οποία το χρονικό διάστημα  $d_{now} - d_{use}$  μεγιστοποιείται. Με άλλα λόγια, από ένα σύνολο αντιγράφων, το σύστημα προτιμά κάθε φορά να χρησιμοποιεί το λιγότερο ενεργό, δηλαδή εκείνο που χρησιμοποιήθηκε λιγότερο πρόσφατα.

Τέλος, σε περίπτωση που από την ανωτέρω διαδικασία δε βρεθεί κάποιο αντίγραφο, η διαδικασία της αναζήτησης στο μεσάζοντα, ή η επεξεργασία της εισερχόμενης ερώτησης, εκτελείται από το τοπικό στιγμιότυπο του υποσυστήματος αναζήτησης, ή επερωτήσεων, αντίστοιχα, αφού πρώτα ο αριθμός των ταυτόχρονα εξυπηρετούμενων αιτημάτων καταστεί μικρότερος από το μέγιστο επιτρεπτό όριο.

Για την καλύτερη κατανόηση της ανωτέρω διαδικασίας, ας δώσουμε ένα παράδειγμα εφαρμογής της. Ο Πίνακας 4 περιέχει μια σειρά αντιγράφων του υποσυστήματος αναζήτησης, δίνοντας για το καθένα πληροφορίες σχετικά με τη χρονική στιγμή τελευταίας χρήσης τους, τον τρέχοντα αριθμό εξυπηρετούμενων αιτημάτων, καθώς και το μέγιστο επιτρεπτό όριο ταυτόχρονα εξυπηρετούμενων αιτημάτων. Υποθέτουμε, ότι το τοπικό στιγμιότυπο του υποσυστήματος έχει φτάσει στα όριά του, τη στιγμή που δημιουργείται ένα νέο νήμα εκτέλεσης, για την επίσκεψη σε κάποιο μεσάζοντα. Έτσι, ξεκινά η διαδικασία επιλογής από τη λίστα διαθέσιμων αντιγράφων, για την εξυπηρέτηση της εκκρεμώσεως διεργασίας και συνεπώς το διαμοιρασμό του φόρτου εργασίας. Το πρώτο βήμα της διαδικασίας αποκλείει την επιλογή των αντιγράφων c1 και c4, καθώς, όπως φαίνεται στον Πίνακα 4, ισχύει  $n_{req} = req_{max}$ . Τελικά, από τα εναπομείναντα αντίγραφα c2, c3 και c5, το δεύτερο βήμα της διαδικασίας αποκλείει τα c3 και c5, οδηγώντας έτσι στην επιλογή προς χρήση του αντιγράφου c2.

## 4.8 Συζήτηση

Παρουσιάσαμε το σύστημα Proteus, που υποστηρίζει την ενιαία αναζήτηση ετερογενών υπηρεσιών. Βασικά χαρακτηριστικά του συστήματος είναι η ευελιξία και επεκτασιμότητά του, χάρη στις οποίες το πρόβλημα της ετερογένειας των περιγραφών υπηρεσιών και των διαφόρων μηχανισμών αναζήτησής τους αντιμετωπίζεται επιτυχώς. Η αρχιτεκτονική του συστήματος επιλύει επίσης προβλήματα απόδοσης στην αναζήτηση υπηρεσιών, τα οποία οφείλονται στην ευρεία διασπορά των σημείων δημοσίευσης των περιγραφών τους στο διαδίκτυο. Στην περίπτωση αναζήτησης ετερογενών υπηρεσιών, τα προβλήματα απόδοσης παρατηρούνται ακόμα πιο έντονα, καθώς οι περιγραφές τους δε δημοσιεύονται αποκλειστικά σε μητρώα, αλλά μπορούν να αναζητηθούν ακόμα σε καταλόγους εντός οικονομικών οργανισμών πλέγματος, ή ακόμα σε δίκτυα ομότιμων κόμβων. Στην τελευταία περίπτωση, ο ασύγχρονος τρόπος λειτουργίας των υποστηριζόμενων μηχανισμών αναζήτησης καθιστά την εύρεση υπηρεσιών ιδιαίτερα χρονοβόρα και αναποτελεσματική, σε ό,τι αφορά την ανάκληση των αποτελεσμάτων.

Για την επίλυση όλων αυτών των προβλημάτων απόδοσης, η μηχανή αναζήτησης Proteus σχεδιάστηκε λαμβάνοντας υπόψη την αρχιτεκτονική των σύγχρονων μηχανών αναζήτησης ιστού. Έτσι, το σύστημα αποτελείται από δύο ανεξάρτητες μεταξύ τους λειτουργικές μονάδες, με τη βοήθεια των οποίων επιτυγχάνεται πλήρης διαχωρισμός της διαδικασίας αναζήτησης, ανάκτησης και ανάλυσης των περιγραφών υπηρεσιών από τα διάφορα σημεία στα οποία έχουν δημοσιευθεί, από τη διαδικασία επεξεργασίας και εκτέλεσης μιας επερώτησης.

Υλοποιώντας την προδιαγραφή της γλώσσας USQL και βασιζόμενο στο μαθηματικό της μοντέλο για την αξιολόγηση υπηρεσιών ως προς τα κριτήρια αναζήτησης του χρήστη, το σύστημα Proteus μπορεί να λειτουργήσει αποδοτικά σε περιβάλλοντα, όπου οι περιγραφές των υπηρεσιών είναι σημασιολογικά εμπλουτισμένες με τη χρήση οντολογιών, αλλά και σε περιβάλλοντα, όπου οι περιγραφές υπηρεσιών βασίζονται κυρίως στη χρήση απλού κειμένου. Στην πρώτη περίπτωση, η μηχανή αναζήτησης του συστήματος βασίζεται σε μια απλή οντολογία ανωτέρου επιπέδου, προκειμένου να υποστηρίξει την αντιστοίχιση υπηρεσιών σε σημασιολογικό επίπεδο, γεγονός που επιφέρει έναν περιορισμό στη λειτουργία της. Εν τούτοις, χάρη στην αρθρωτή και επεκτάσιμη αρχιτεκτονική του, το σύστημα μπορεί εύκολα να επεκταθεί προκειμένου να υποστηρίξει πιο εξεζητημένους μηχανισμούς σημασιολογικής αντιστοίχισης, οι οποίοι περιλαμβάνουν προχωρημένες τεχνικές αυτόματης αντιστοίχισης οντολογιών [34], [61], [40]. Ωστόσο, κάτι τέτοιο ξεφεύγει από τα όρια του γνωστικού αντικείμενου και της συνεισφοράς της παρούσης διατριβής. Ως εκ τούτου, η ενσωμάτωση προχωρημένων τεχνικών οντολογικής αντιστοίχισης στο σύστημα Proteus θα μπορούσε να θεωρηθεί ως πεδίο μελλοντικής έρευνας.

Κλείνοντας αυτό το κεφάλαιο, θα θέλαμε να κάνουμε μια σύντομη αναφορά στην ιστο-

ρία και την έως τώρα χρήση του συστήματος που παρουσιάσαμε. Ουσιαστικά, το σύστημα Proteus αποτελεί εξέλιξη της μηχανής αναζήτησης USQL Engine, η οποία αρχικά αναπτύχθηκε μαζί με τη γλώσσα USQL στο πλαίσιο του ερευνητικού έργου SODIUM [119]. Πρώιμες εκδόσεις της μηχανής αυτής έχουν παρουσιαστεί σε διάφορες ερευνητικές εργασίες [120], [89], [88]. Χάρη στην ευελιξία και την ενιαία διεπαφή της μέσω της γλώσσας USQL, η μηχανή USQL Engine χρησιμοποιήθηκε στη συνέχεια από τα εργαλεία αναζήτησης υπηρεσιών του ερευνητικού έργου SeCSE [104], για την αναζήτηση υπηρεσιών ιστού σε εξωτερικά μητρώα και αποθετήρια, όπως το UDDI και ebXML, καθώς επίσης για την αναζήτηση υπηρεσιών ομότιμου-προς-ομότιμου σε δίκτυα ομότιμων κόμβων τύπου JXTA. Το σύστημα Proteus διατηρεί τα βασικά γνωρίσματα και πλεονεκτήματα της μηχανής USQL Engine, ενώ παρουσιάζει σημαντικές βελτιώσεις σε ό,τι έχει να κάνει με την απόδοση και επεκτασιμότητά της, προσφέροντας έτσι μια ολοκληρωμένη, αξιόπιστη και ευέλικτη λύση στο πρόβλημα της αναζήτησης ετερογενών υπηρεσιών.

## ΚΕΦΑΛΑΙΟ 5

### ΑΞΙΟΛΟΓΗΣΗ

Στο κεφάλαιο αυτό αξιολογούμε τα αποτελέσματα της παρούσης διατριβής. Μέσω της παρουσίασης μιας λεπτομερούς μελέτης των υπαρχουσών τεχνολογιών περιγραφής και αναζήτησης υπηρεσιών ιστού, πλέγματος και ομότιμου-προς-ομότιμο (Ενότητα 5.1), αποτιμάται η γενικότητα και καταλληλότητα της γλώσσας USQL, ως ενιαία διεπαφή αναζήτησης ετερογενών υπηρεσιών. Σε ό,τι αφορά τη μηχανή αναζήτησης Proteus, παρουσιάζονται και συζητούνται τα αποτελέσματα μετρήσεων, που ελήφθησαν στο πλαίσιο εκτέλεσης μιας σειράς πειραμάτων. Στόχος ήταν η αξιολόγηση παράμετρων του συστήματος, όπως η ακρίβεια και ανάκληση των αποτελεσμάτων αναζήτησης (Ενότητα 5.2), η απόδοση και οι δυνατότητες κλιμάκωσής του (Ενότητα 5.3).

Η αξιολόγηση της μηχανής αναζήτησης Proteus πραγματοποιήθηκε με τη βοήθεια μιας πρωτότυπης υλοποίησης στη γλώσσα προγραμματισμού Java (έκδοση 1.5.0.16) [56]. Η ανάπτυξη του πρωτότυπου βασίστηκε στην εξαιρετικά δημοφιλή πλατφόρμα Eclipse (έκδοση 3.4) [39] και αξιοποιεί την εξεζητημένη διεπαφή προγραμματισμού εφαρμογών (Application Programming Interface, API), που αυτή προσφέρει, για την υλοποίηση του μηχανισμού επεκτάσεων, που περιγράφηκε στην Ενότητα 4.6. Το σχεσιακό σχήμα δεδομένων του τοπικού καταλόγου υπηρεσιών υλοποιήθηκε με τη χρήση της τεχνολογίας MySQL [81] και αναπτύχθηκε σε έναν εξυπηρετητή MySQL έκδοσης 5.1.34. Η επικοινωνία μεταξύ των διαφόρων συστατικών της μηχανής αναζήτησης Proteus και του τοπικού καταλόγου υπηρεσιών υλοποιήθηκε με τη χρήση της τεχνολογίας JDBC (έκδοση 3.0) [57].

#### 5.1 Αξιολόγηση της Γλώσσας USQL

Στην ενότητα αυτή, προχωρούμε σε μια αξιολόγηση της γλώσσας USQL, παρουσιάζοντας μια λεπτομερή μελέτη της σχέσης της (i) με τους υπάρχοντες μηχανισμούς αναζήτησης υπηρεσιών ιστού, υπηρεσιών πλέγματος και υπηρεσιών ομότιμου-προς-ομότιμο, καθώς και (ii) με τις υπάρχουσες γλώσσες περιγραφής των λειτουργικών και ποιοτικών χαρακτηριστικών αυτών των ειδών υπηρεσιών. Συγκεκριμένα, προκειμένου να αποτιμήσουμε τη συνεισφορά της USQL, την ορθότητα της εννοιολογικής της σύλληψης, αλλά και τη βιωσιμότητά της στην αναζήτηση ετερογενών υπηρεσιών, εξετάσαμε:

1. Την εκφραστικότητά της, σε σχέση με τις διεπαφές των μηχανισμών αναζήτησης, που προσφέρουν οι πλέον διαδεδομένες και/ή προτυποποιημένες τεχνολογίες μεσαζόντων υπηρεσιών ιστού, πλέγματος και ομότιμου-προς-ομότιμο.
2. Τη δυνατότητα κάλυψης, υπό τη μορφή κριτηρίων αναζήτησης σε ένα έγγραφο επερώτησης USQLRequest, όλων των χαρακτηριστικών μιας υπηρεσίας, η περιγραφή

των οποίων υποστηρίζεται από τις πιο δημοφιλείς τεχνολογίες περιγραφής υπηρεσιών.

Στόχος της αξιολόγησης αυτής ήταν να δείξουμε, ότι, αν και η USQL προσφέρει μια ενοποιητική προσέγγιση στην αναζήτηση ετερογενών υπηρεσιών ιστού, πλέγματος και ομότιμου-προς-ομότιμο, η απώλεια σε εκφραστικότητα ή κάλυψη των διαφόρων κριτηρίων αναζήτησης είναι μηδενική, ή τουλάχιστον εξαιρετικά περιορισμένη.

### 5.1.1 Μελέτη Διεπαφών Αναζήτησης Υπηρεσιών

Σε όλες τις υπάρχουσες τεχνολογίες μεσαζόντων υπηρεσιών, οι υποστηριζόμενες διεπαφές για την αναζήτηση υπηρεσιών βασίζονται στη χρήση απλών φίλτρων, που λαμβάνουν κυρίως τη μορφή λέξεων-κλειδιών. Όπως θα δούμε στη συνέχεια αυτής της παραγράφου, η USQL παρέχει τις κατάλληλες δομές για την έκφραση τέτοιου είδους περιορισμών, εξασφαλίζοντας μηδενική απώλεια εκφραστικότητας. Ακολουθούν λεπτομερείς περιγραφές των πλέον διαδεδομένων διεπαφών αναζήτησης υπηρεσιών ιστού, υπηρεσιών πλέγματος και υπηρεσιών ομότιμου-προς-ομότιμο.

**Διεπαφή Αναζήτησης UDDI.** Οι προδιαγραφές του προτύπου UDDI προσφέρουν μια διεπαφή αποτελούμενη από μεθόδους για την αναζήτηση υπηρεσιών ιστού σε μητρώα αυτού του είδους. Η μέθοδος *find\_business* επιτρέπει την αναζήτηση παρόχων υπηρεσιών, βάσει του ονόματος ή της κατηγοριοποίησής τους, με τη χρήση κάποιου σχήματος ταξινόμησης, όπως η UNSPSC [123] ή η NAICS [82]. Τα αποτελέσματα αυτού του είδους αναζήτησης υπόκεινται σε περαιτέρω επεξεργασία, προκειμένου να ανακτηθούν από το μητρώο οι εγγραφές υπηρεσιών ιστού, που παρέχονται από τον κάθε πάροχο. Εξάλλου, η μέθοδος *find\_service* επιτρέπει την απευθείας αναζήτηση και ανάκτηση εγγραφών υπηρεσιών ιστού, υποστηρίζοντας κριτήρια αναζήτησης όπως το όνομα της υπηρεσίας, το όνομα του παρόχου της, καθώς και η κατηγορία στην οποία η υπηρεσία υπάγεται. Επιπλέον, κάνοντας χρήση της δομής *tModel* του UDDI, είναι δυνατή η αναζήτηση υπηρεσιών βάσει προκαθορισμένων διεπαφών, περιγεγραμμένων με τη χρήση της γλώσσας WSDL.

Με τη χρήση της USQL, μπορεί κανείς να εκφράσει όλα τα προαναφερθέντα κριτήρια αναζήτησης, που υποστηρίζονται από τη διεπαφή του UDDI για την αναζήτηση υπηρεσιών ιστού. Τόσο το όνομα και η κατηγορία της αναζητούμενης υπηρεσίας, όσο και λεπτομέρειες σχετικές με τον πάροχό της μπορούν εύκολα να εκφραστούν με τη χρήση της γενικής δομής *SearchFilter*, που ορίζεται από το μετα-μοντέλο της γλώσσας. Παρόμοια, απαιτήσεις σχετικά με τη διεπαφή της αναζητούμενης υπηρεσίας μπορούν να εκφραστούν με τη χρήση του κριτηρίου αναζήτησης *ServiceInterface*. Συνεπώς, συγκρίνοντας τη γλώσσα USQL με τη διεπαφή αναζήτησης υπηρεσιών του UDDI, συμπεραίνουμε ότι δεν υπάρχει απώλεια εκφραστικότητας.

**Διεπαφή Αναζήτησης ebXML.** Οι επερωτήσεις προς ένα μητρώο ebXML παίρνουν τη μορφή μηνυμάτων τύπου *AdhocQueryRequest*, σύμφωνα με τις προδιαγραφές του προτύπου αυτού. Σε ένα μήνυμα αυτού του τύπου, το στοιχείο *ServiceQuery* χρησιμοποιείται για τη σύνταξη φίλτρων αναζήτησης (στοιχεία *ServiceFilter*) σχετικών με το όνομα, την περιγραφή, την κατηγοριοποίηση, αλλά και τον πάροχο της αναζητούμενης υπηρεσίας. Εναλλακτικά, παρόμοια με τις προδιαγραφές του UDDI, μπορεί κανείς να πραγματοποιήσει αναζήτηση μιας υπηρεσίας σε δύο στάδια: Καταρχάς, με τη χρήση του στοιχείου φίλτρου *OrganizationFilter*, μπορούν να τεθούν περιορισμοί σχετικά με τον/τους επιθυμητούς παρόχους υπηρεσιών. Σε δεύτερο στάδιο, οι επιστρεφόμενες εγγραφές παρόχων υπηρεσιών υπόκεινται σε περαιτέρω επεξεργασία, έτσι ώστε να ανακτηθούν οι αντίστοιχες εγγραφές υπηρεσιών.

Όπως και στην περίπτωση του UDDI, η USQL παρουσιάζει συγκριτικά μηδενικές απώλειες σε ό,τι αφορά την εκφραστικότητα των επερωτήσεων αναζήτησης υπηρεσιών, καθώς το στοιχείο *SearchFilter* μπορεί να χρησιμοποιηθεί για τη σύνταξη κριτηρίων-φίλτρων αναζήτησης, όπως το όνομα, η κατηγοριοποίηση και ο πάροχος της υπηρεσίας, ενώ, με τη χρήση του κριτηρίου αναζήτησης *Capability*, μπορεί κανείς να περιγράψει την επιθυμητή λειτουργικότητα της υπηρεσίας με κείμενο.

**Διεπαφή Αναζήτησης JXTA.** Η αναζήτηση υπηρεσιών σε δίκτυα ομότιμων κόμβων JXTA πραγματοποιείται με την εφαρμογή του πρωτοκόλλου PDP (Peer Discovery Protocol). Σύμφωνα με το πρωτόκολλο αυτό, κάθε κόμβος, που ενδιαφέρεται για κάποια υπηρεσία, σχηματίζει ένα μήνυμα τύπου *DiscoveryQuery*, το οποίο ακολούθως διαχέει στην ομάδα κόμβων, της οποίας αποτελεί μέλος. Το συντακτικό του μηνύματος *DiscoveryQuery* υποστηρίζει τον προσδιορισμό φίλτρων σχετικά με το όνομα και την περιγραφή της υπό αναζήτηση υπηρεσίας, καθώς και με τον πάροχό της. Αντίστοιχα, η δομή *SearchFilter* της USQL μπορεί να χρησιμοποιηθεί για τον προσδιορισμό φίλτρων σχετικών με το όνομα ή/και τον πάροχο της αναζητούμενης υπηρεσίας, ενώ, όπως είδαμε στο Κεφάλαιο 3, η περιγραφή της μπορεί να προσδιοριστεί με τη χρήση του κριτηρίου αναζήτησης *Capability*.

**Διεπαφή Αναζήτησης gLite.** Η υποδομή πλέγματος gLite περιλαμβάνει μια σειρά συστατικών λογισμικού, τα οποία υποστηρίζουν το διαμοιρασμό πόρων και υπηρεσιών μεταξύ Ιδεατών Οργανισμών (Virtual Organizations). Συγκεκριμένα, ο μηχανισμός *Service Discovery (SD)* παρέχει μια διεπαφή, μέσω της οποίας μπορεί κανείς να αποκτήσει πρόσβαση σε στοιχεία των υπηρεσιών πλέγματος, που δημοσιεύονται σε καταλόγους σύμφωνα με το σχήμα μετα-δεδομένων GLUE. Έτσι, η διεπαφή του SD υποστηρίζει κριτήρια αναζήτησης, τα οποία αντιστοιχούν σε ιδιότητες των οντοτήτων *Site*, *Service* και *ServiceData*, οι οποίες ορίζονται από τις προδιαγραφές του GLUE.

Πιο αναλυτικά, ο μηχανισμός SD επιτρέπει την αναζήτηση μιας υπηρεσίας πλέγματος

βάσει του ονόματός της, που έχει καταχωρηθεί κατά τη δημοσίευσή της (ιδιότητα Name της οντότητας Service) στον κατάλογο υπηρεσιών, του είδους/κατηγορίας της (ιδιότητα Type της οντότητας Service), του παρόχου της (ιδιότητα Name της οντότητας Site), καθώς και των δεδομένων (ζεύγη ιδιοτήτων Key/Value της οντότητας ServiceData), τα οποία χειρίζεται. Όλα αυτά τα απλά είδη κριτηρίων αναζήτησης μπορούν να εκφραστούν σε μια επερώτηση USQLRequest ως φίλτρα αναζήτησης, με τη χρήση της δομής SearchFilter.

### 5.1.2 Μελέτη Γλωσσών Περιγραφής Υπηρεσιών

Η περιγραφή των διαφόρων χαρακτηριστικών μιας υπηρεσίας παραγματοποιείται με τη χρήση διαφόρων γλωσσών, οι οποίες έχουν εμφανιστεί τα τελευταία χρόνια. Ιδιαίτερο ενδιαφέρον παρουσιάζει το γεγονός, ότι καμία από τις υπάρχουσες τεχνολογίες περιγραφής υπηρεσιών δεν μπορεί να υποστηρίξει όλα τα χαρακτηριστικά μιας υπηρεσίας. Έτσι, για παράδειγμα, η περιγραφή των ποιοτικών χαρακτηριστικών μιας υπηρεσίας βασίζεται στη χρήση διαφορετικών γλωσσών, από αυτές που χρησιμοποιούνται για την περιγραφή της λειτουργικότητας και της διεπαφής της.

Από την άλλη, η USQL επιτρέπει τη δημιουργία επερωτήσεων, υπό τη μορφή εγγράφων USQLRequest, στις οποίες συνδυάζονται απαιτήσεις προς πολλαπλές διαφορετικές ιδιότητες της αναζητούμενης υπηρεσίας. Στις επόμενες παραγράφους, εξετάζουμε το βαθμό, στον οποίο τα περιεχόμενα μιας περιγραφής υπηρεσίας μπορούν να αξιοποιηθούν κατά την αναζήτησή της, μέσω αντιστοιχιών με τα διάφορα φίλτρα και κριτήρια αναζήτησης, που υποστηρίζονται από το συντακτικό της USQL. Η μελέτη αυτή εστιάζει σε έναν αντιπροσωπευτικό αριθμό γλωσσών, οι οποίες έχουν καταστεί κοινώς αποδεκτές και χρησιμοποιούνται εκτενώς για την περιγραφή υπηρεσιών ιστού, υπηρεσιών πλέγματος και υπηρεσιών ομότιμου-προς-ομότιμο.

**Η Γλώσσα WSDL.** Στη βασική του μορφή, ένα έγγραφο WSDL παρέχει συντακτική πληροφορία σχετικά με τις λειτουργίες, που προσφέρονται από μια υπηρεσία, τη μορφοποίηση των ανταλλασσόμενων μηνυμάτων, καθώς και των τύπων δεδομένων των μερών, από τα οποία τα μηνύματα αυτά αποτελούνται. Αυτού του είδους οι πληροφορίες συνιστούν το *αφηρημένο (abstract)* μέρος της γλώσσας WSDL και χρησιμοποιούνται κατά τη διαδικασία αναζήτησης, προκειμένου να διαπιστωθεί η καταλληλότητα μιας υπηρεσίας σε σχέση με τις λειτουργικές απαιτήσεις του αιτούντος.

Εξάλλου, με τη χρήση επεκτάσεων, επιπρόσθετες πληροφορίες μπορούν να συμπεριληφθούν σε ένα έγγραφο WSDL. Για παράδειγμα, το σχήμα SAWSDL ορίζει μια σειρά επεκτάσεων, οι οποίες επιτρέπουν την προσθήκη σημασιολογικής πληροφορίας στο αφηρημένο μέρος της WSDL, υπό τη μορφή οντολογικών εννοιών. Παρόμοια, σε περίπτωση που η WSDL χρησιμοποιείται για την περιγραφή μιας υπηρεσίας πλέγματος, κατάλληλες



Πίνακας 5: Αντιστοιχία περιγραφών WSDL και επερωτήσεων USQL

WSDL 1.1, SAWSDL, and WSRP Elements	USQL Filters and Criteria
wsdl:service@name	Capability/Description
wsdl:service/documentation	Capability/Description
wsdl:portType/sawSDL:attrExtensions	Filter@serviceProperty="ServiceCategory"
wsdl:portType@wsrp:ResourceProperties	Resource
wsdl:operation@name	Capability/Description
wsdl:operation/documentation	Capability/Description
wsdl:operation/wsdl:input	Interface/Input
wsdl:operation/wsdl:output	Interface/Output
wsdl:message/wsdl:part@name	Message/Element/Description
wsdl:message/wsdl:part/documentation	Message/Element/Description
wsdl:message/wsdl:part@type	Message/Element/Type
wsdl:message/wsdl:part@sawSDL:modelReference	Message/Element@ontologyReference

Πίνακας 6: Αντιστοιχία περιγραφών OWL-S και επερωτήσεων USQL

OWL-S Elements	USQL Filters and Criteria
Profile@about	Capability@ontologyReference
Profile/serviceName	Filter@serviceProperty="ServiceName"
Profile/textDescription	Capability/Description
Profile/contactInformation	Filter@serviceProperty="ServiceProvider"
Profile/serviceCategory	Filter@serviceProperty="ServiceCategory"
Profile/hasInput@resource	Interface/Input/Element@ontologyReference
Profile/hasOutput@resource	Interface/Output/Element@ontologyReference
Profile/hasPrecondition	-
Profile/hasResult	-

επεκτάσεις, σύμφωνα με τις προδιαγραφές του πλαισίου WS-RF, καθιστούν δυνατή την περιγραφή του πόρου, στον οποίο επενεργούν οι προσφερόμενες λειτουργίες.

Ο Πίνακας 5 καταγράφει τις αντιστοιχίες μεταξύ των στοιχείων της γλώσσας WSDL, αλλά και των επεκτάσεών της μέσω των σχημάτων SAWSDL και WSRP, και των στοιχείων κριτηρίων αναζήτησης της USQL. Παρατηρούμε, ότι, με τη χρήση της USQL, μπορούν να αξιοποιηθούν πλήρως τα περιεχόμενα του αφηρημένου μέρους των εγγράφων περιγραφών WSDL, καθώς η γλώσσα υποστηρίζει τον προσδιορισμό κριτηρίων αναζήτησης ως προς αυτά.

**Η Γλώσσα OWL-S.** Η προδιαγραφή της σημασιολογικής γλώσσας περιγραφής υπηρεσιών OWL-S ορίζει την κλάση *ServiceProfile*, για τη διαφήμιση και αναζήτησή τους. Εννοιολογικά, ένα στιγμιότυπο της κλάσης αυτής αντιστοιχεί σε μια λειτουργία υπηρεσίας και παρέχει μια υψηλού επιπέδου περιγραφή των δυνατοτήτων της. Συγκεκριμένα, σε ένα στιγμιότυπο *ServiceProfile*, μπορεί κανείς να βρει σημασιολογικές περιγραφές σχετικά με τη λειτουργικότητα της υπηρεσίας, τα μηνύματα εισόδου/εξόδου της, τις προαπαιτούμενες συνθήκες (preconditions) και τα αποτελέσματα εκτέλεσής της (results). Εξάλλου, ένα

Πίνακας 7: Αντιστοιχία στοιχείων WS-QoS και επερωτήσεων USQL

WS-QoS Elements	USQL Filters and Criteria
QoSDefinition/price QoSDefinition/price@currency	QoSProperty/Description="Price" QoSProperty/Qualifier@name="Currency"
serverQoSMetrics/processingTime	QoSProperty/Description="ProcessingTime"
serverQoSMetrics/requestsPerSecond	QoSProperty/Description="RequestsPerSecond"
serverQoSMetrics/reliability	QoSProperty/Description="Reliability" QoSProperty/Qualifier@name="min"@value=0 QoSProperty/Qualifier@name="max"@value=1
serverQoSMetrics/availability	QoSProperty/Description="Availability" QoSProperty/Qualifier@name="min"@value=0 QoSProperty/Qualifier@name="max"@value=1
serverQoSMetrics/customMetric@name serverQoSMetrics/customMetric@ontology	QoSProperty/Description QoSProperty@ontologyReference
transportQoSPriorities/delay	QoSProperty/Description="TransportQoS" QoSProperty/Qualifier@name="min"@value=0 QoSProperty/Qualifier@name="max"@value=10
transportQoSPriorities/jitter	QoSProperty/Description="Jitter" QoSProperty/Qualifier@name="min"@value=0 QoSProperty/Qualifier@name="max"@value=10
transportQoSPriorities/throughput	QoSProperty/Description="Throughput" QoSProperty/Qualifier@name="min"@value=0 QoSProperty/Qualifier@name="max"@value=10
transportQoSPriorities/packetLoss	QoSProperty/Description="PacketLoss" QoSProperty/Qualifier@name="min"@value=0 QoSProperty/Qualifier@name="max"@value=10
transportQoSPriorities/customPriority@name transportQoSPriorities/customPriority@ontology	QoSProperty/Description QoSProperty@ontologyReference

στιγμιότυπο ServiceProfile ενδέχεται να περιλαμβάνει επιπλέον πληροφορία για την υπηρεσία, όπως το όνομά της, η περιγραφή της, η κατηγορία και ο πάροχός της.

Στον Πίνακα 6 καταγράφονται οι αντιστοιχίσεις μεταξύ των διαφόρων στοιχείων της κλάσης ServiceProfile της OWL-S και των περιεχομένων ενός αιτήματος USQLRequest. Με εξαίρεση τις περιγραφές των προαπαιτούμενων συνθηκών και των αποτελεσμάτων εκτέλεσης της υπηρεσίας, βλέπουμε, ότι η USQL μπορεί να αξιοποιήσει πλήρως τα υπόλοιπα περιεχόμενα ενός εγγράφου περιγραφής ServiceProfile της OWL-S, επιτρέποντας τη σύνταξη φίλτρων και κριτηρίων αναζήτησης ως προς τις αντίστοιχες ιδιότητες της υπηρεσίας.

**Η Γλώσσα WS-QoS.** Η γλώσσα WS-QoS παρέχει ένα σχήμα για την περιγραφή προσφορών, αλλά και απαιτήσεων, σχετικά με τα ποιοτικά χαρακτηριστικά μιας υπηρεσίας. Το συντακτικό της γλώσσας υποστηρίζει την περιγραφή πολλαπλών διαστάσεων QoS στο επίπεδο εφαρμογής, όπως η διαθεσιμότητα της υπηρεσίας, η αξιοπιστία της, ο μέσος χρόνος επεξεργασίας, κτλ. Σύμφωνα με την προδιαγραφή της WS-QoS, μια προσφορά συσχετίζεται με ένα συγκεκριμένο στοιχείο Port του εγγράφου WSDL της υπηρεσίας και ως εκούτου αφορά στις λειτουργίες που προσφέρονται μέσω της αντίστοιχης θύρας. Εναλλα-

Πίνακας 8: Αντιστοιχία στοιχείων WSLA και USQL

WSLA Elements	USQL Filters and Criteria
SLAParameter@name	QoSProperty/Description
SLAParameter@unit	QoSProperty/Qualifier@name="Unit"
SLAParameter@type	QoSProperty/Qualifier@name="Type"
SLAParameter/Metric	QoSProperty/Qualifier@name="Metric"

κτικά, είναι δυνατή η αντιστοίχιση μιας προσφοράς με ένα συγκεκριμένο στοιχείο Operation του εγγράφου WSDL, σε περίπτωση που κάποια λειτουργία χαρακτηρίζεται από διαφορετικές τιμές στα ποιοτικά της χαρακτηριστικά.

Οι αντιστοιχίες μεταξύ των στοιχείων περιγραφής ποιοτικών χαρακτηριστικών της WS-QoS και του κριτηρίου αναζήτησης QoSProperty της USQL δίνονται στον Πίνακα 7. Αμέσως γίνεται σαφές, ότι, χάρη στο υψηλό επίπεδο αφαίρεσης, στο οποίο έχει οριστεί η δομή αυτή, η USQL είναι σε θέση να καλύψει πλήρως το χρήστη στην έκφραση απαιτήσεων ως προς τα ποιοτικά χαρακτηριστικά της αναζητούμενης υπηρεσίας, σε περιπτώσεις όπου αυτά έχουν περιγραφεί με τη χρήση του σχήματος WS-QoS.

**Η Γλώσσα WSLA.** Οι προδιαγραφές του πλαισίου WSLA ορίζουν μια γλώσσα βασισμένη στην XML, με το συντακτικό της οποίας οι πάροχοι υπηρεσιών μπορούν να εκφράσουν εγγυήσεις σχετικά με τα ποιοτικά χαρακτηριστικά των λειτουργιών, που προσφέρουν μέσω των υπηρεσιών τους. Η γλώσσα WSLA επιτρέπει τον προσδιορισμό μέτρων, που λαμβάνονται από τον πάροχο σε περίπτωση απόκλισης ή μη τήρησης των περιγεγραμμένων ποιοτικών χαρακτηριστικών, καθώς και τον προσδιορισμό μηχανισμών, με τους οποίους πραγματοποιούνται οι μετρήσεις. Επιπροσθέτως, σε ένα έγγραφο WSLA, μπορεί να συμπεριληφθεί πληροφορία σχετικά με οντότητες, οι οποίες επιτηρούν ή/και μετρούν τα ποιοτικά χαρακτηριστικά της υπηρεσίας και είναι υπεύθυνες για τη διαχείριση των διαφόρων αποκλίσεων.

Εννοιολογικά, ένα έγγραφο WSLA αποτελείται από τρία τμήματα:

1. Στο τμήμα *Parties* προσδιορίζονται όλα τα συμβαλλόμενα μέρη, ενώ, για καθένα από αυτά, δίνονται πληροφορίες ταυτοποίησης και τεχνικές λεπτομέρειες.
2. Το τμήμα *Service Description* προσδιορίζει τα ποιοτικά χαρακτηριστικά της υπηρεσίας, μαζί με τις παρατηρούμενες παραμέτρους τους. Οι πληροφορίες που περιέχονται σε αυτό το τμήμα του εγγράφου WSLA παρουσιάζουν ενδιαφέρον κατά την αναζήτηση υπηρεσιών. Συνεπώς, μια γλώσσα επερωτήσεων υπηρεσιών, όπως η USQL, πρέπει να μπορεί να εκφράσει τα αντίστοιχα κριτήρια αναζήτησης.
3. Το τελευταίο τμήμα, *Obligations*, ορίζει διάφορες εγγυήσεις και περιορισμούς και αξιοποιείται κατά τη διαδικασία παρακολούθησης των τιμών των ποιοτικών χαρακτηριστικών της υπηρεσίας, προκειμένου να διαπιστωθούν ενδεχόμενες παραβάσεις.

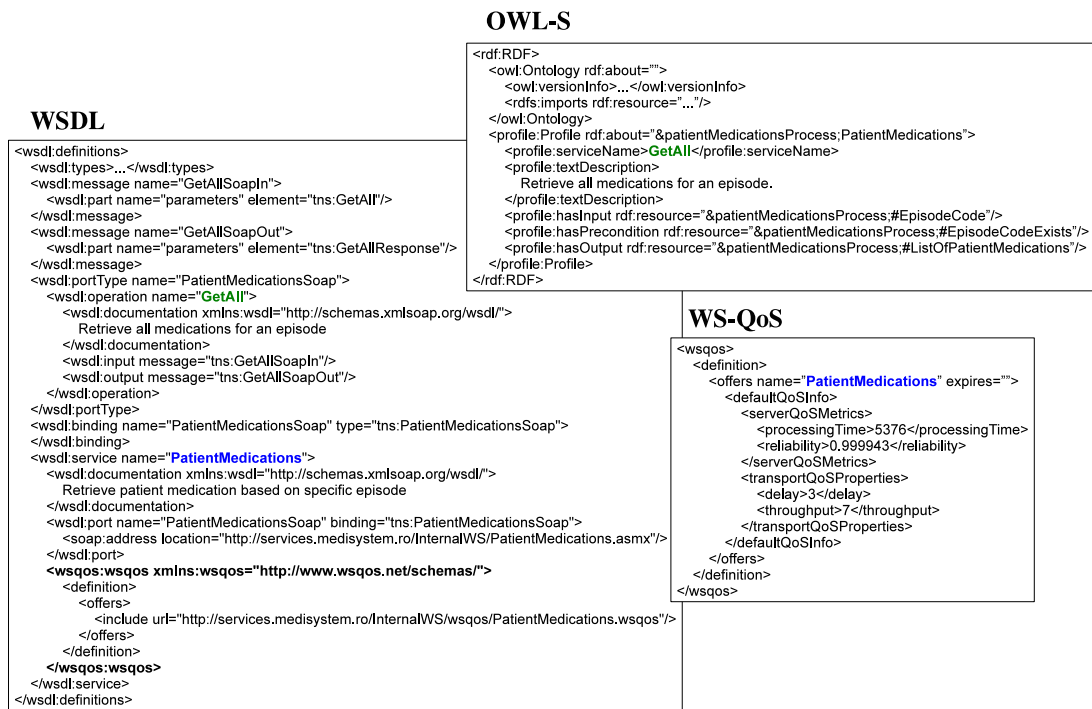
Εστιάζοντας στο τμήμα Service Description, τα ποιοτικά χαρακτηριστικά μιας υπηρεσίας περιγράφονται με τη χρήση της δομής *SLAParameter*. Κάθε στιγμιότυπο της δομής αυτής αντιστοιχεί σε ένα ποιοτικό χαρακτηριστικό, το οποίο έχει ένα όνομα (ιδιοχαρακτηριστικό *name*), έναν τύπο (ιδιοχαρακτηριστικό *type*), καθώς και μια μονάδα μέτρησης (ιδιοχαρακτηριστικό *unit*). Επιπλέον, κάθε στιγμιότυπο *SLAParameter* συμπεριλαμβάνει ένα στοιχείο *Metric*, το οποίο ορίζει τον τρόπο μέτρησης του αντίστοιχου ποιοτικού χαρακτηριστικού. Χάρη στη γενικότητα των δομών *SLAParameter*, της *WSLA*, και *QoSProperty*, της *USQL*, η αντιστοίχισή τους είναι άμεση, όπως φαίνεται στον Πίνακα 8. Συνεπώς, διενεργώντας αναζήτηση υπηρεσιών με τη χρήση της *USQL*, είναι δυνατή η πλήρης αξιοποίηση των περιεχομένων του τμήματος Service Description ενός εγγράφου *WSLA*.

### 5.1.3 Συζήτηση

Αξιολογήσαμε τη γλώσσα *USQL* ως προς την καταλληλότητά της στο σχηματισμό επερωτήσεων κατά την αναζήτηση υπηρεσιών ιστού, υπηρεσιών πλέγματος και υπηρεσιών ομότιμου-προς-ομότιμο. Η συγκριτική μελέτη των πλέον διαδεδομένων μηχανισμών αναζήτησης υπηρεσιών αυτών των τύπων έδειξε, ότι η *USQL* μπορεί να εκφράσει όλα τα φίλτρα και κριτήρια αναζήτησης, που υποστηρίζονται από τις διεπαφές αναζήτησης των εξετασθέντων μηχανισμών, *UDDI*, *ebXML*, *JXTA* και *gLite*. Αξίζει επίσης να σημειωθεί, ότι, συγκρίνοντας τις διεπαφές αυτές με την *USQL*, διαπιστώνουμε την ανωτερότητά της στην έκφραση επιπρόσθετων κριτηρίων αναζήτησης, που δεν υποστηρίζονται προς το παρόν από τους μηχανισμούς αναζήτησης των προαναφερθέντων τεχνολογιών μεσαζόντων υπηρεσιών. Συμπερασματικά, η *USQL* διευκολύνει τη διεξαγωγή ενιαίας αναζήτησης υπηρεσιών σε ετερογενείς μεσάζοντες (μητρώα, αποθετήρια, καταλόγους, δίκτυα κτλ.), χωρίς τη συνεπαγωγή απωλειών, ως προς την εκφραστικότητα των επερωτήσεων.

Μελετώντας τις επικρατέστερες γλώσσες περιγραφής υπηρεσιών διαπιστώσαμε τον υψηλό βαθμό συμβατότητας της *USQL* με τα περιεχόμενα των αντίστοιχων εγγράφων. Η *USQL* προσφέρει ένα πλούσιο σύνολο στοιχείων και δομών, που επιτρέπουν στο χρήστη να εκφράσει τις απαιτήσεις του προς τις διάφορες λειτουργικές και μη λειτουργικές ιδιότητες μιας υπηρεσίας, οι οποίες καταγράφονται στην περιγραφή της. Έτσι, με εξαίρεση τις περιγραφές των προαπαιτούμενων συνθηκών και των αποτελεσμάτων εκτέλεσης της υπηρεσίας, που μπορεί να δίνονται σε έγγραφα *OWL-S*, τα περιεχόμενα των εγγράφων περιγραφής υπηρεσιών, που συντάσσονται με τις γλώσσες *WSDL*, *SAWSDL*, *WSRP*, *OWL-S*, *WS-QoS* και *WSLA* αξιοποιούνται πλήρως κατά την αναζήτηση υπηρεσιών με τη χρήση της *USQL*. Σε κάθε περίπτωση, η επεκτασιμότητα που χαρακτηρίζει τη γλώσσα καθιστά δυνατή τη μελλοντική ενσωμάτωση επιπρόσθετων στοιχείων, έτσι ώστε να υποστηριχθεί η έκφραση κριτηρίων αναζήτησης ως προς διάφορες ιδιότητες υπηρεσιών.

Δευτερευόντως, η μελέτη κατέστησε σαφές το πλεονέκτημα του σχηματισμού επερω-



Σχήμα 20: Συνδυασμός πολλαπλών γλωσσών για την περιγραφή μιας υπηρεσίας ιστού.

τήσεων με ενιαίο τρόπο, χάρη στο σχήμα της USQL, έναντι της χρήσεως πολλαπλών διαφορετικών συντακτικών για την περιγραφή των αντίστοιχων ιδιοτήτων μιας υπηρεσίας από τον πάροχό της. Ως παράδειγμα, το Σχήμα 20 απεικονίζει την περιγραφή των συντακτικών, σημασιολογικών και ποιοτικών χαρακτηριστικών μιας υπηρεσίας ιστού με τη χρήση τριών διαφορετικών γλωσσών (WSDL, OWL-S και WS-QoS), ενώ το Σχήμα 21 απεικονίζει την αντίστοιχη, ενιαία επερώτηση με τη χρήση της USQL, στην οποία εκφράζονται κριτήρια αναζήτησης προς όλες τις υποστηριζόμενες ιδιότητες της υπηρεσίας.

Κλείνοντας την ενότητα αυτή πρέπει να σημειωθεί, ότι τα αποτελέσματα της αξιολόγησης επιτρέπουν μια ακόμα διαπίστωση, σχετικά με την καταλληλότητα, τη γενικότητα και το εύρος χρήσης της USQL στην αναζήτηση υπηρεσιών ιστού, υπηρεσιών πλέγματος και υπηρεσιών ομότιμου-προς-ομότιμο. Δεδομένου του εξαιρετικά υψηλού βαθμού συμβατότητας της USQL με τις τεχνολογίες αναζήτησης και περιγραφής υπηρεσιών, που εξετάστηκαν, καθίσταται δυνατή η χρήση της σε οποιοδήποτε υπηρεσιοστρεφές περιβάλλον βασίζεται σε αυτές. Με άλλα λόγια, η USQL μπορεί να χρησιμοποιηθεί για την αναζήτηση υπηρεσιών στα περιεχόμενα οποιουδήποτε μεσάζοντα υπηρεσιών, που διαθέτει υποδομή UDDI, ebXML, JXTA, ή gLite. Επιπλέον, τα κριτήρια αναζήτησης της επερώτησης USQLRequest μπορούν να αντιστοιχιστούν με τα περιεχόμενα οποιουδήποτε εγγράφου περιγραφής υπηρεσίας, που συντάσσεται με τη χρήση των γλωσσών WSDL, SAWSDL, WSRP, OWL-S, WS-QoS, ή/και WSLA.

## USQL

```
<USQLRequest id="1" specVersion="1.2" serviceType="WebService">
<SearchFilters/>
<SearchCriteria>
<Capability ontologyReference="">
<Description>Retrieve all medications for an episode</Description>
</Capability>
<Interface>
<Input>
<Element ontologyReference="patientMedicationsProcess.#EpisodeCode">
<Description>Episode code</Description>
<Type namespace="sodium.medisystem.ro">GetAll</Type>
</Element>
</Input>
<Output>
<Element ontologyReference="patientMedicationsProcess.#ListOfMedications">
<Description>List of medications</Description>
<Type namespace="sodium.medisystem.ro">GetAllResponse</Type>
</Element>
</Output>
</Interface>
<QoS>
<QoSProperty value="5376">
<Description>ServerProcessingTime</Description>
<Qualifier name="unit" value="milliseconds"/>
</QoSProperty>
<QoSProperty value="0.999943">
<Description>ServerReliability</Description>
<Qualifier name="min" value="0"/>
<Qualifier name="max" value="1"/>
</QoSProperty>
<QoSProperty value="3">
<Description>TransportDelay</Description>
<Qualifier name="min" value="0"/>
<Qualifier name="max" value="10"/>
</QoSProperty>
<QoSProperty value="7">
<Description>TransportThroughput</Description>
<Qualifier name="min" value="0"/>
<Qualifier name="max" value="10"/>
</QoSProperty>
</QoS>
</SearchCriteria>
<SearchTargets/>
</USQLRequest>
```

Σχήμα 21: Ενιαία επερώτηση USQL για μια υπηρεσία ιστού.

## 5.2 Απόδοση των Μηχανισμών Αντιστοίχισης

Η αξιολόγηση της απόδοσης των μηχανισμών αντιστοίχισης του συστήματος Proteus, που, όπως είδαμε, υλοποιούν το μαθηματικό μοντέλο της γλώσσας USQL, πραγματοποιήθηκε με τη διεξαγωγή ενός πειράματος, βάσει του οποίου καταγράφηκε η μετρική *F-score* [124]. Η μετρική αυτή αποτελεί δημοφιλή επιλογή για την αξιολόγηση μηχανών αναζήτησης, καθώς λαμβάνει υπόψη τόσο την ακρίβεια (*precision*) όσο και την ανάκληση (*recall*) των αποτελεσμάτων. Ο μαθηματικός της ορισμός δίνεται ως εξής:

$$F = 2 \cdot \frac{precision \cdot recall}{precision + recall} \quad (16)$$

Η ανωτέρω εξίσωση ερμηνεύεται κοινώς ως ο μέσος όρος της ακρίβειας και ανάκλησης των αποτελεσμάτων αναζήτησης, ενώ η *F* λαμβάνει τιμές μεταξύ 0 (χειρίστη απόδοση) και 1 (βέλτιστη απόδοση). Σημειώνεται επίσης, ότι η ακρίβεια και η ανάκληση υπολογίζονται βάσει των ακόλουθων εξισώσεων:

$$precision = \frac{RetRel}{Ret}, recall = \frac{RetRel}{Rel} \quad (17)$$

Στις εξισώσεις αυτές, αναπαριστούμε με *RetRel* τον αριθμό των σχετικών με την επερώτηση εγγραφών στα αποτελέσματα αναζήτησης, με *Rel* > 0 το συνολικό αριθμό των σχετικών με την επερώτηση εγγραφών, ενώ με *Ret* > 0 τον αριθμό των επιστρεφόμενων εγγραφών στα αποτελέσματα αναζήτησης, σχετικών και μη.

Για τις ανάγκες του πειράματος, απομονώσαμε και συλλέξαμε έναν αριθμό περιγραφών υπηρεσιών από διάφορες πηγές. Συγκεκριμένα, χρησιμοποιήσαμε τις περιγραφές των υπηρεσιών ιστού, πλέγματος και ομότιμου-προς-ομότιμο, που προέρχονται από τις εφαρμογές των ερευνητικών έργων SODIUM [119] και SeCSE [104], ενώ εμπλουτίσαμε τη συλλογή μας με περιγραφές υπηρεσιών ιστού από τη διαδικτυακή μηχανή αναζήτησης Seekda [105].

Συνολικά, συγκεντρώσαμε μια συλλογή περιγραφών αποτελούμενη από 258 λειτουργίες υπηρεσιών ιστού, 27 λειτουργίες υπηρεσιών ομότιμου-προς-ομότιμο και 18 λειτουργίες υπηρεσιών πλέγματος. Επομένως, οι μετρήσεις του πειράματος πραγματοποιήθηκαν με αναζητήσεις σε 303 λειτουργίες υπηρεσιών. Ορισμένες από τις περιγραφές των υπηρεσιών που συλλέξαμε περιελάμβαναν σημασιολογική πληροφορία υπό τη μορφή σημασιολογικών επισημειώσεων σε OWL-S και SAWSDL. Έτσι, κατά το σχηματισμό των επερωτήσεων μας για αυτές τις υπηρεσίες, χρησιμοποιήσαμε έννοιες από την ίδια οντολογία που χρησιμοποιήθηκε για τις περιγραφές τους. Επίσης, κάποιες από αυτές τις περιγραφές περιείχαν στοιχεία σχετικά με τα ποιοτικά χαρακτηριστικά των υπηρεσιών, εκφρασμένα με τη χρήση του σχήματος WS-QoS. Κάποιες άλλες περιγραφές υπηρεσιών, αν και δεν περιείχαν σημασιολογικές επισημειώσεις, ούτε στοιχεία ποιοτικών χαρακτηριστικών, ήταν πλούσιες σε ελεύθερο κείμενο. Σε κάθε περίπτωση, προκειμένου να εξασφαλίσουμε ότι η συλλογή περιγραφών θα μπορούσε να χρησιμοποιηθεί αποτελεσματικά για την αξιολόγηση του αλγόριθμου αντιστοίχισης τιμών ποιοτικών χαρακτηριστικών, δημιουργήσαμε με ημι-αυτοματοποιημένο τρόπο τις κατάλληλες περιγραφές ποιοτικών χαρακτηριστικών, όπου αυτές έλειπαν.

Για την εκτέλεση του πειράματος, χρησιμοποιήσαμε 9 επερωτήσεις υπό τη μορφή εγγράφων USQLRequest, που αφορούσαν σε υπηρεσίες διαφορετικών τύπων και πεδίων εφαρμογής, όπως φαίνεται στον Πίνακα 9. Τα κριτήρια αναζήτησης των επερωτήσεων προσδιορίστηκαν βάσει πραγματικών απαιτήσεων, οι οποίες καταγράφηκαν σε μορφή κειμένου (παραδοτέα έγγραφα) από τους χρήστες των προαναφερθέντων έργων SODIUM και SeCSE, αλλά και από το τμήμα μηχανογράφησης γνωστής ελληνικής τράπεζας.

Καθένα από τα έγγραφα USQLRequest σχηματίστηκε και εκτελέστηκε σταδιακά, ως εξής:

1. Αρχικά, προσδιορίσαμε και εκτελέσαμε αιτήματα USQLRequest που περιελάμβαναν κριτήρια αναζήτησης σχετικά με την επιθυμητή λειτουργικότητα, ή/και τον επιθυμητό πόρο (στις περιπτώσεις αναζήτησης υπηρεσιών πλέγματος).
2. Ακολούθως, εμπλουτίσαμε τα αιτήματα αυτά με κριτήρια αναζήτησης ως προς την επιθυμητή διεπαφή, δηλαδή ως προς τα μηνύματα εισόδου/εξόδου της ζητούμενης υπηρεσίας, και τα επανεκτελέσαμε.
3. Τέλος, συμπληρώσαμε τα αιτήματα με κριτήρια αναζήτησης σχετικά με τα επιθυμητά

Πίνακας 9: Επερωτήσεις για τη μέτρηση της απόδοσης των αλγορίθμων αντιστοίχισης

Query	Service Type	Domain
Q1	P2P	Healthcare
Q2	Web	Healthcare
Q3	P2P	Crisis Management
Q4	Grid	Crisis Management
Q5	Web	Automotive
Q6	Grid	Automotive
Q7	Grid	Weather
Q8	Web	Finance
Q9	Web	Finance

ποιοτικά χαρακτηριστικά των υπηρεσιών, πριν ξεκινήσουμε έναν τελευταίο γύρο εκτέλεσής τους.

Ο στόχος αυτής της στρατηγικής εκτέλεσης των επερωτήσεων ήταν να εκτιμήσουμε την αποτελεσματικότητα των διαφόρων ειδών κριτηρίων αναζήτησης της USQL και να μετρήσουμε την απόδοση των αντίστοιχων αλγορίθμων αντιστοίχισης, καθώς και τη συμβολή τους στον υπολογισμό του συνολικού βαθμού αντιστοιχίας για κάθε μια λειτουργία υπηρεσίας. Έτσι, με τη βοήθεια της πρωτότυπης υλοποίησης του συστήματος Proteus, εκτελέσαμε συνολικά 27 επερωτήσεις USQL, αναζητώντας υπηρεσίες ιστού, υπηρεσίες πλέγματος και υπηρεσίες ομότιμου-προς-ομότιμο προερχόμενες από πέντε διαφορετικά πεδία εφαρμογών. Όλα τα αιτήματα USQLRequest του πειράματος παραμετροποιήθηκαν ως εξής:

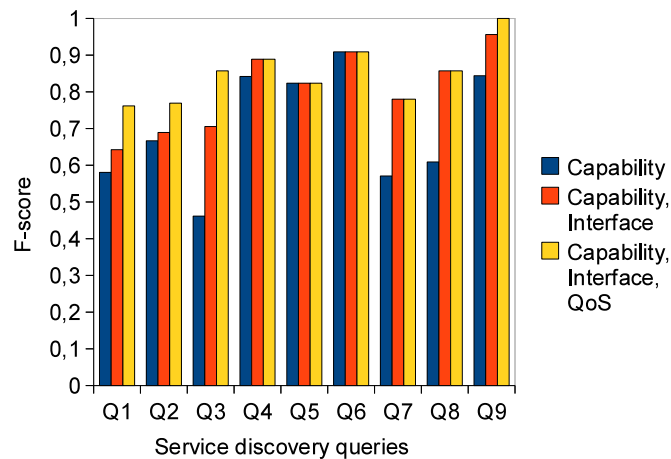
- Ο μέγιστος αριθμός αποτελεσμάτων περιορίστηκε σε 20, μέσω του ιδιοχαρακτηριστικού maxResults του στοιχείου USQLRequest.
- Ο ελάχιστος αποδεκτός βαθμός αντιστοιχίας για κάθε μια λειτουργία στα αποτελέσματα αναζήτησης προσδιορίστηκε σε 20%, μέσω του ιδιοχαρακτηριστικού minMatch του στοιχείου USQLRequest, προκειμένου να αποκλειστούν από τα αποτελέσματα λειτουργίες με μη μηδενικό, ωστόσο εξαιρετικά χαμηλό βαθμό αντιστοιχίας.

### 5.2.1 Αποτελέσματα Μετρήσεων

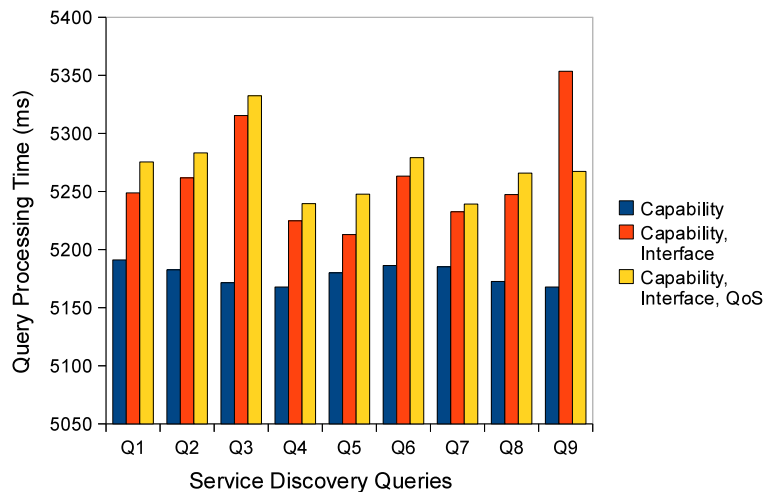
Το γράφημα του Σχήματος 22 αναπαριστά τα πειραματικά αποτελέσματα σχετικά με την ανάκληση και ακρίβεια των αποτελεσμάτων αναζήτησης, μετά την εκτέλεση των προαναφερθέντων αιτημάτων USQLRequest.

Σύμφωνα με τα αποτελέσματα, μπορεί κανείς να συμπεράνει ότι, υπό τις κατάλληλες προϋποθέσεις, οι μηχανισμοί αντιστοίχισης του συστήματος Proteus, εφαρμόζοντας





Σχήμα 22: Απόδοση των αλγορίθμων αντιστοίχισης



Σχήμα 23: Μέσος χρόνος επεξεργασίας των αλγορίθμων αντιστοίχισης

το μαθηματικό μοντέλο της γλώσσας USQL, χαρακτηρίζονται από αξιοσημείωτες επιδόσεις. Πράγματι, οι περιορισμοί της διαδικασίας αντιστοίχισης τιμών κειμένου, προερχόμενοι από το ελλιπές περιεχόμενο των περισσότερων περιγραφών υπηρεσιών, αντιμετωπίστηκαν επιτυχώς μέσω των προαναφερθεισών οδηγιών `minMatch` και `maxResults` στα έγγραφα `USQLRequest`. Έτσι, επιτεύχθηκε μια εξισορρόπηση της συμβολής των σχετικά χαμηλών αυτών βαθμών αντιστοίχισης στην ακρίβεια και ανάκληση των αποτελεσμάτων αναζήτησης.

Εξίσου ενδιαφέρουσα είναι η παρατήρηση ότι, στις περισσότερες περιπτώσεις, η προσθήκη κριτηρίων αναζήτησης σχετικά με τη διεπαφή και τα ποιοτικά χαρακτηριστικά της αναζητούμενης υπηρεσίας επέφερε βελτίωση στην ακρίβεια των αποτελεσμάτων. Το γεγονός αυτό επιβεβαιώνει τη σημαντικότητα του να μπορεί κανείς να εκφράζει και να συνδυάζει διάφορα είδη απαιτήσεων υπό τη μορφή κριτηρίων αναζήτησης, κατά την αναζήτηση υπηρεσιών.

Η αξιολόγηση των μηχανισμών αντιστοίχισης του συστήματος Proteus ολοκληρώνεται με την παρουσίαση των αποτελεσμάτων σχετικά με το μέσο χρόνο επεξεργασίας της επερώτησης, για κάθε μία από τις αναζητούμενες υπηρεσίες του πειράματος. Σκοπός των μετρήσεων αυτών ήταν να διαπιστώσουμε τον πρόσθετο χρόνο που επιβάλλουν οι αλγόριθμοι αντιστοίχισης των διαφόρων ειδών κριτηρίων αναζήτησης που υποστηρίζει η USQL.

Όπως δείχνει το σχετικό γράφημα στο Σχήμα 23, ο μέσος χρόνος επεξεργασίας των αλγορίθμων αντιστοίχισης, μετρημένος σε χιλιοστά του δευτερολέπτου, παρουσίασε μικρή διακύμανση για τα διάφορα πεδία εφαρμογής των υπηρεσιών που χρησιμοποιήσαμε στο πείραμα. Σε όλες τις περιπτώσεις, ο μέσος χρόνος επεξεργασίας επιβαρύνθηκε κατά λιγότερο από ένα δευτερόλεπτο, με την προσθήκη κριτηρίων αναζήτησης ως προς τη διεπαφή και τα ποιοτικά χαρακτηριστικά των ζητούμενων υπηρεσιών. Ειδικά στην περίπτωση του αλγορίθμου αντιστοίχισης ποιοτικών χαρακτηριστικών, ο μέσος επιπρόσθετος φόρτος ήταν αμελητέος (μικρότερος από 500 χιλιοστά του δευτερολέπτου) και σχετικά σταθερός για όλες τις υπηρεσίες.

Αξίζει να σημειωθεί, ότι, σε μεγάλο ποσοστό, ο χρόνος επεξεργασίας επηρεάζεται από το χρόνο που απαιτείται για τη δημιουργία των διανυσμάτων TF-IDF, τα οποία χρησιμοποιούνται από τον αλγόριθμο αντιστοίχισης τιμών κειμένου. Μια πιθανή λύση για την εξάλειψη του φόρτου αυτού θα μπορούσε να είναι η προσθήκη επιπλέον λειτουργικότητας στον Αναλυτή (Analyzer) του υποσυστήματος αναζήτησης, προκειμένου η δημιουργία των διανυσμάτων TF-IDF να λαμβάνει χώρα ανεξάρτητα από την εκτέλεση των επερωτήσεων.

## 5.2.2 Συζήτηση

Στην ενότητα αυτή, αξιολογήσαμε το μαθηματικό μοντέλο της γλώσσας USQL, που παρουσιάσαμε στο Κεφάλαιο 3, και το οποίο υποστηρίζει τον υπολογισμό του βαθμού αντιστοίχιας μιας λειτουργίας υπηρεσίας με τα κριτήρια αναζήτησης ενός εγγράφου USQLRequest. Η αξιολόγηση πραγματοποιήθηκε στο πλαίσιο ενός πειράματος, όπου μετρήσαμε την απόδοση των αλγορίθμων του μοντέλου, βάσει των ποσοστών ανάκλησης και ακριβείας των αποτελεσμάτων αναζήτησης. Στόχος των μετρήσεων ήταν η εκτίμηση τόσο του συνολικού αλγορίθμου αντιστοίχισης, όσο και της προσφοράς των επιμέρους αλγορίθμων, που χρησιμοποιούνται από τον Επεξεργαστή Επερωτήσεων του συστήματος Proteus, για την αντιστοίχιση απλών και σύνθετων κριτηρίων αναζήτησης.

Οι μετρήσεις έδειξαν ότι η αλγοριθμική μας προσέγγιση χαρακτηρίζεται από ικανοποιητική απόδοση και αυτό την καθιστά βιώσιμη και εφαρμόσιμη ως λύση στο πρόβλημα αντιστοίχισης ετερογενών υπηρεσιών. Επιπλέον, κατέστη σαφές ότι η χρήση κριτηρίων αναζήτησης προς πολλαπλές λειτουργικές και ποιοτικές ιδιότητες της ζητούμενης υπηρεσίας όχι μόνο είναι δυνατή, αλλά βελτιώνει σημαντικά την ποιότητα των επιστρεφόμενων αποτελεσμάτων.

Πίνακας 10: Στοιχεία πειράματος μέτρησης της κλιμάκωσης του Proteus

Αριθμός Μεσαζόντων	Αριθμός Περιγραφών Υπηρεσιών	Αριθμός Διαφημίσεων Υπηρεσιών
1	127	2.832
2	254	5.664
3	381	8.496
4	508	11.328
5	635	14.160
6	762	16.992

### 5.3 Απόδοση της Μηχανής Αναζήτησης Proteus

Σε αυτήν την ενότητα, παρουσιάζουμε τις πειραματικές μετρήσεις που λάβαμε σχετικά με την απόδοση της μηχανής αναζήτησης Proteus. Συγκεκριμένα, μετρήσαμε το χρόνο επεξεργασίας του Αναζητητή (Crawler) και του Επεξεργαστή Επερωτήσεων (Query Processor) της μηχανής, ως προς τον αριθμό των περιγραφών υπηρεσιών, αλλά και το μέγεθος των περιεχομένων τους. Για τις ανάγκες της αξιολόγησης αυτής, αυξήσαμε τον υπάρχοντα αριθμό των περιγραφών υπηρεσιών εισάγοντας τελικά στο σύστημά μας ένα σύνολο 4.077 εγγράφων διαφημίσεων λειτουργιών υπηρεσιών ιστού, υπηρεσιών ομότιμου-προς-ομότιμο και υπηρεσιών πλέγματος. Όλα αυτά τα έγγραφα περιελάμβαναν περιγραφές τόσο των λειτουργικών όσο και των ποιοτικών χαρακτηριστικών των αντίστοιχων υπηρεσιών. Οι υπηρεσίες των οποίων τις περιγραφές χρησιμοποιήσαμε στο πείραμα προέρχονται από διάφορα πεδία εφαρμογών, γεγονός ωστόσο που δεν επηρεάζει τις μετρήσεις σχετικά με την απόδοση της μηχανής αναζήτησης.

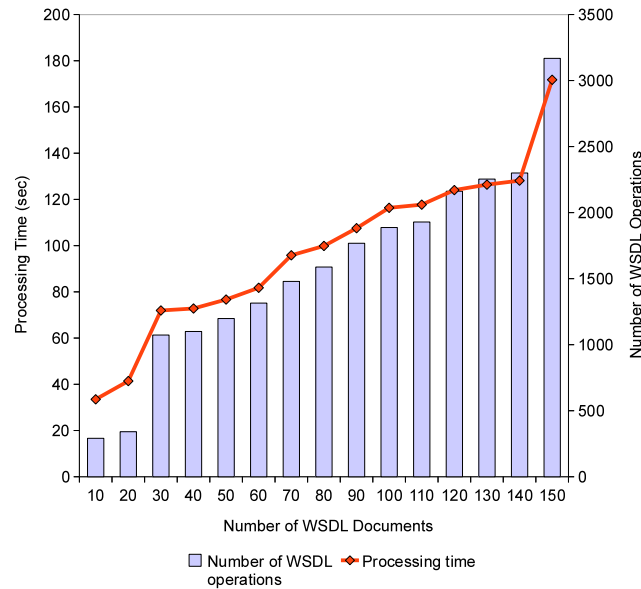
Εκτός από τις ανωτέρω μετρήσεις, πραγματοποιήσαμε μια αξιολόγηση των δυνατοτήτων κλιμάκωσης των δύο υποσυστημάτων της μηχανής αναζήτησης, εκτελώντας δύο αντίστοιχα πειράματα περιορισμένης κλίμακας. Συγκεκριμένα, για την περίπτωση του Αναζητητή, χρησιμοποιήσαμε έξι μεσάζοντες ως στόχους αναζήτησης, εφοδιασμένους με 127 περιγραφές υπηρεσιών WSDL ο καθένας. Από την ανάλυση των περιγραφών αυτών προέκυψαν 2.832 διαφημίσεις υπηρεσιών ανά στόχο αναζήτησης. Ο Πίνακας 10 συνοψίζει αυτές τις αντιστοιχίες.

Επιπλέον, για τις ανάγκες του πειράματος, αναπτύξαμε δύο ομάδες αντιγράφων υπηρεσιών ιστού, για καθένα από τα δύο υποσυστήματα, σε δύο υπολογιστές, τα χαρακτηριστικά των οποίων, σε υλικό και λογισμικό, καταγράφονται στον Πίνακα 11. Οι υπολογιστές αυτοί, καθώς και ο υπολογιστής, στον οποίο εγκαταστήσαμε το υπόλοιπο σύστημα Proteus, συνδέονταν μέσω ενός τοπικού δικτύου LAN, με ταχύτητα 100 Mbps.

Όπως περιγράψαμε στην Ενότητα 4.3 του προηγούμενου κεφαλαίου, ο Αναζητητής της μηχανής αναζήτησης Proteus ανακτά τις περιγραφές ετερογενών υπηρεσιών επισκεπτόμενος τα διάφορα ετερογενή μητρώα, αποθετήρια, δίκτυα κτλ., στα οποία αυτές δημοσιεύονται. Έτσι, στην πράξη, ο συνολικός χρόνος εκτέλεσης εξαρτάται από εξωγενείς παράγοντες, όπως η απόδοση του δικτύου, ο φόρτος εργασίας του εκάστοτε στόχου ανα-

Πίνακας 11: Ανάπτυξη της πειραματικής διάταξης

OS	CPU	RAM	Application Server
MS Windows XP Prof.	Intel Core 2 @ 2 GHz	2 GB	Jetty 6.1 & Axis 1.4
Ubuntu 8.04	Intel Core 2 Duo @ 2.66 GHz	3.2 GB	Jetty 6.1 & Axis 1.4



Σχήμα 24: Συνολικός χρόνος επεξεργασίας του Αναζητητή

ζήτησης, κτλ., που είναι ανεξάρτητοι από το σύστημά μας. Για το λόγο αυτό και έχοντας ως στόχο την αξιολόγηση της απόδοσης του Αναζητητή και μόνο, οι μετρήσεις που λάβαμε αφορούν αποκλειστικά στο μέρος της διεργασίας που ξεκινά αμέσως μετά τη μεταφόρτωση των ανακτηθεισών εγγράφων περιγραφών υπηρεσιών WSDL. Έτσι, οι χρόνοι επεξεργασίας του πειράματος που αφορούν στον Αναζητητή, αναφέρονται αποκλειστικά στην απόδοση των συστατικών, που αναλαμβάνουν την ανάλυση (Analyzer) και αποθήκευση (Indexer) των περιγραφών υπηρεσιών, στον τοπικό κατάλογο υπηρεσιών του συστήματος.

Η αξιολόγηση της απόδοσης του Επεξεργαστή Επερωτήσεων, πραγματοποιήθηκε με τη λήψη μετρήσεων του χρόνου εκτέλεσης των επερωτήσεων, ως προς

- τον αριθμό των εγγράφων διαφημίσεων υπηρεσιών, που ανακτώνται από τον τοπικό κατάλογο υπηρεσιών, και
- τον αριθμό των ταυτόχρονων επερωτήσεων.

Οι μετρήσεις του χρόνου εκτέλεσης των επερωτήσεων ως προς τον πρώτο παράγοντα, πραγματοποιήθηκαν με τη, σταδιακά αυξανόμενη, επεξεργασία έως και 3.000 εγγράφων διαφημίσεων υπηρεσιών, από τον Επεξεργαστή Επερωτήσεων, για την ίδια επερωτήση. Σημειώνεται, ότι, στην επερωτήση, αξιοποιήθηκε πλήρως το εύρος κριτηρίων αναζήτησης

που προσφέρει η USQL, καθώς συμπεριελήφθησαν κριτήρια σχετικά με τη λειτουργικότητα, τη διεπαφή, αλλά και τα ποιοτικά χαρακτηριστικά της αναζητούμενης υπηρεσίας.

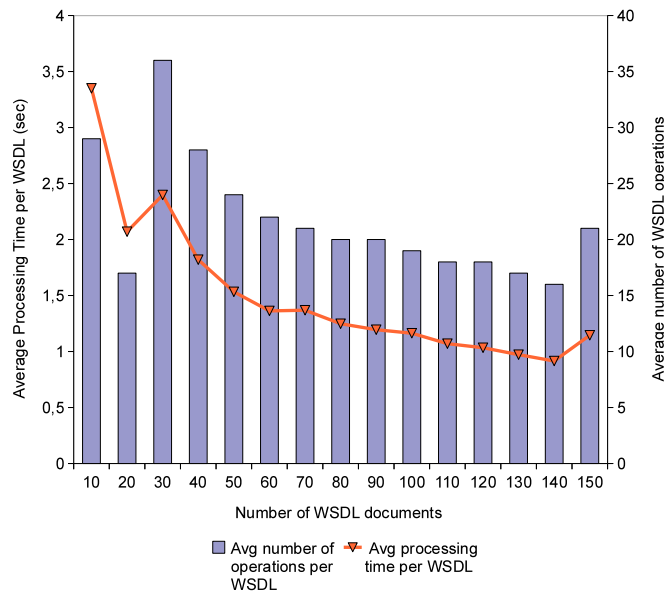
Για τη λήψη των μετρήσεων της απόδοσης του Επεξεργαστή Επερωτήσεων ως προς τον δεύτερο παράγοντα, δημιουργήσαμε ένα σύνολο από 20 επερωτήσεις USQL, των οποίων ο χρόνος εκτέλεσης διαπιστώθηκε διεξοδικά ότι κυμαίνεται μεταξύ 8 και 8,5 δευτερολέπτων. Στη συνέχεια, με τη χρήση ενός απλού μηχανισμού που αναπτύξαμε, εκτελέσαμε τις επερωτήσεις αυτές ταυτόχρονα, ξεκινώντας από 1 επερώτηση τη φορά και φτάνοντας μέχρι 20 επερωτήσεις τη φορά. Η διαδικασία αυτή επαναλήφθηκε συνολικά 3 φορές: αρχικά, το σύστημα δεν διέθετε κάποιο αντίγραφο του Επεξεργαστή Επερωτήσεων, ενώ τις επόμενες δύο φορές αυξήσαμε τον αριθμό των αντιγράφων κατά ένα, χρησιμοποιώντας την ίδια διάταξη υπολογιστών, των οποίων τα χαρακτηριστικά καταγράφει ο Πίνακας 11. Με τον τρόπο αυτό, παρατηρήσαμε την απόδοση του συστήματος Proteus, όταν αυτό είναι εφοδιασμένο με αντίγραφα του Επεξεργαστή Επερωτήσεων.

### 5.3.1 Αποτελέσματα Μετρήσεων

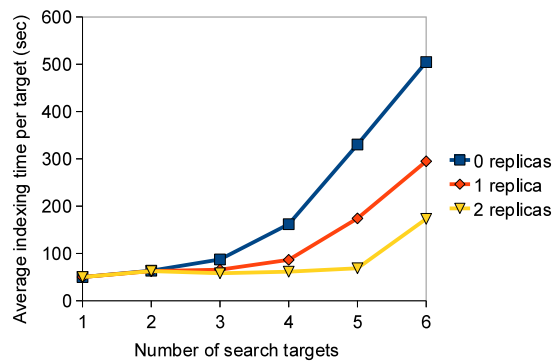
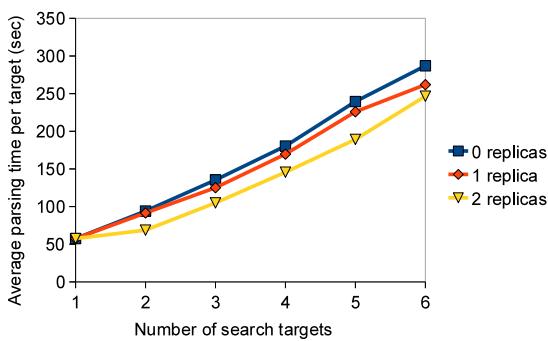
Στα διαγράμματα των Σχημάτων 24 και 25 καταγράφεται η απόδοση του Αναζητητή (Crawler), σε απόλυτους και μέσους χρόνους αντίστοιχα. Συγκεκριμένα, στο Σχήμα 24 φαίνεται ο χρόνος επεξεργασίας του Αναζητητή, συναρτήσει του αριθμού των περιγραφών WSDL, αλλά και του συνολικού αριθμού λειτουργιών, που αυτές περιέχουν. Παρατηρούμε, ότι, σε κάθε περίπτωση, ο χρόνος επεξεργασίας είναι άμεσα εξαρτώμενος από τον αριθμό των λειτουργιών. Η συμπεριφορά αυτή εξηγείται από το γεγονός, ότι τα έγγραφα διαφημίσεων υπηρεσιών που ευρετηριοποιούνται και αποθηκεύονται στον τοπικό κατάλογο υπηρεσιών του συστήματος Proteus αντιστοιχούν στις λειτουργίες των υπηρεσιών και όχι στις υπηρεσίες. Έτσι, από τη διαδικασία ευρετηριοποίησης, δημιουργείται ένα έγγραφο διαφήμισης υπηρεσίας για κάθε μια λειτουργία, που συναντάται στα έγγραφα WSDL.

Η προαναφερθείσα συμπεριφορά του Αναζητητή φαίνεται καλύτερα στο Σχήμα 25, όπου καταγράφεται ο μέσος χρόνος επεξεργασίας ανά έγγραφο WSDL. Πράγματι, η μέση απόδοση του υποσυστήματος εξαρτάται λιγότερο από το συνολικό αριθμό εγγράφων WSDL και είναι πάντως ανάλογη του μέσου αριθμού λειτουργιών ανά έγγραφο WSDL. Αξίζει επίσης να σημειωθεί, ότι, σε όλες τις περιπτώσεις, ο μέσος χρόνος επεξεργασίας ανά λειτουργία διατηρείται σχετικά σταθερός και δεν υπερβαίνει τα 0.2 δευτερόλεπτα.

Τα αποτελέσματα του πειράματος αξιολόγησης των δυνατοτήτων κλιμάκωσης του Αναζητητή απεικονίζονται στα διαγράμματα του Σχήματος 26. Από τα διαγράμματα φαίνεται, ότι η απόδοση του Αναζητητή σε περιπτώσεις ταυτόχρονης επίσκεψης σε περισσότερους από έναν στόχους αναζήτησης βελτιώνεται, καθώς αυξάνεται ο αριθμός αντιγράφων του, για την υποστήριξη των λειτουργιών ανάλυσης και ευρετηριοποίησης. Ιδιαίτερα αισθητή εμφανίζεται η βελτίωση στην περίπτωση ευρετηριοποίησης, όπου, ειδικά στην περίπτωση



Σχήμα 25: Μέσος χρόνος επεξεργασίας του Αναζητητή



(i) Μέσος χρόνος ανάλυσης περιγραφών υπηρεσιών

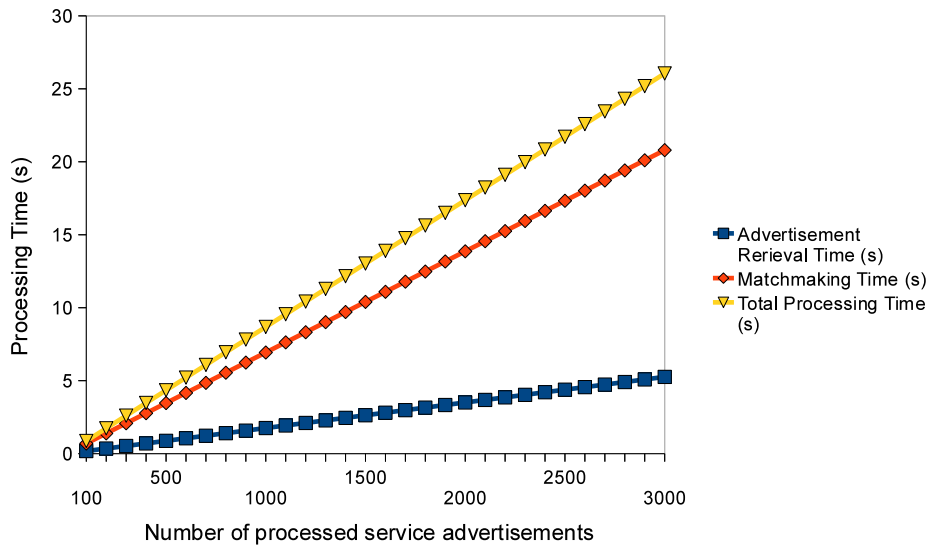
(ii) Μέσος χρόνος ευρετηριοποίησης διαφημίσεων υπηρεσιών

Σχήμα 26: Κλιμάκωση του Αναζητητή

ταυτόχρονης επίσκεψης σε έξι στόχους αναζήτησης, η μείωση του μέσου χρόνου επεξεργασίας ανά στόχο αναζήτησης με τη χρήση μόλις δύο αντιγράφων υπηρεσιών ιστού του Αναζητητή είναι εντυπωσιακή, σχεδόν υπο-πενταπλασιάζοντας τον αντίστοιχο απαιτούμενο χρόνο με τη χρήση μόνο του τοπικού συστατικού.

Προχωρώντας στην αξιολόγηση της απόδοσης του Επεξεργαστή Επερωτήσεων (Query Processor), το γράφημα του Σχήματος 27 απεικονίζει τα αποτελέσματα των μετρήσεων. Στο γράφημα, η απόδοση του υποσυστήματος επερωτήσεων της μηχανής αναζήτησης Proteus εκφράζεται από το μέσο χρόνο επεξεργασίας των επερωτήσεων, συναρτήσει του αριθμού των ανακτούμενων διαφημίσεων λειτουργιών υπηρεσιών.

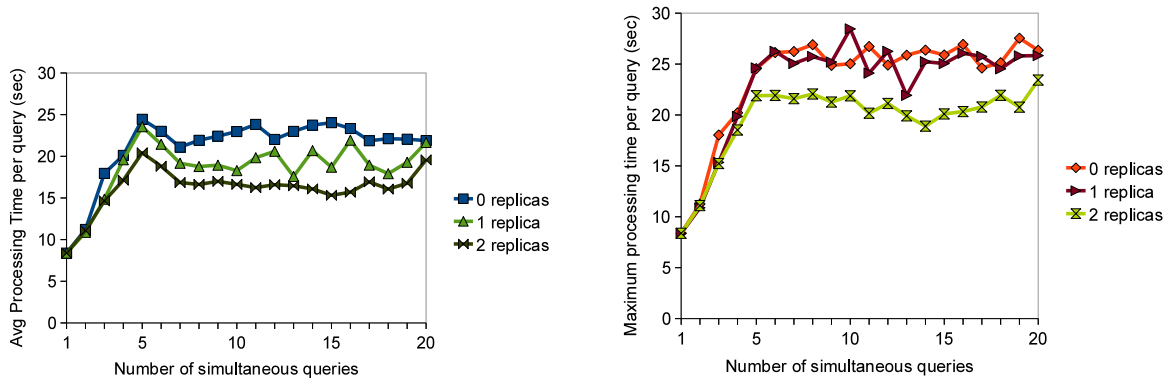
Παρατηρούμε, ότι ο χρόνος επεξεργασίας μιας επερώτησης αυξάνεται γραμμικά με τον αριθμό των διαφημίσεων υπηρεσιών που ανακτώνται από τον τοπικό κατάλογο υπηρεσιών. Βλέπουμε επίσης, ότι ο μέσος χρόνος ανάκτησης δεν ξεπερνά τα πέντε δευτερό-



Σχήμα 27: Απόδοση του Επεξεργαστή Επερωτήσεων

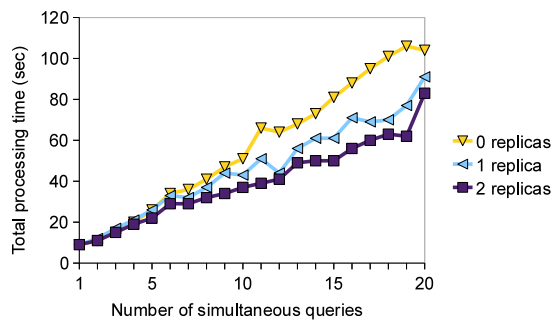
λεπτα, για αριθμούς διαφημίσεων υπηρεσιών που κειμούνται ακόμα και μεταξύ 2.500 και 3.000 τη φορά. Έτσι, στο μεγαλύτερο ποσοστό του, ο χρόνος επεξεργασίας μιας επερωτήσης προσδιορίζεται από το χρόνο εκτέλεσης των αλγορίθμων αντιστοίχισης από το συστατικό αντιστοίχισης (Matcher) του υποσυστήματος επερωτήσεων. Πάντως, ο συνολικός χρόνος εκτέλεσης μιας επερωτήσης, για περιπτώσεις επεξεργασίας έως και 500 εγγράφων διαφημίσεων, δεν ξεπερνά τα 5 δευτερόλεπτα. Καθώς, με την κατάλληλη εφαρμογή των φίλτρων αναζήτησης, που υποστηρίζονται από τη γλώσσα USQL, αλλά και από το σύστημα Proteus, ο αριθμός των ανακτούμενων εγγράφων διαφημίσεων από τον τοπικό κατάλογο συνήθως περιορίζεται σε αριθμούς της τάξης μερικών εκατοντάδων, και δεδομένης της απαιτητικής σε πόρους συστήματος φύσεως της διαδικασίας αντιστοίχισης, κρίνουμε ότι η απόδοση της μηχανής αναζήτησης είναι σε μεγάλο βαθμό επαρκής.

Τέλος, στα διαγράμματα του Σχήματος 28 παραθέτουμε τις μετρήσεις που λάβαμε σχετικά με τις δυνατότητες κλιμάκωσης του Επεξεργαστή Επερωτήσεων. Συγκεκριμένα, τα διαγράμματα 28(i) και 28(ii) αναπαριστούν το μέσο και μέγιστο αντίστοιχα χρόνο επεξεργασίας ανά επερωτήση, για περιπτώσεις ταυτόχρονης επεξεργασίας έως και 20 ισοδύναμων επερωτήσεων. Η αύξηση τόσο στους μέσους όσο και στους μέγιστους χρόνους επεξεργασίας ήταν απότομη για περιπτώσεις ταυτόχρονης εκτέλεσης μέχρι και πέντε επερωτήσεων, ενώ η χρήση αντιγράφων του Επεξεργαστή Επερωτήσεων επέφερε μικρή βελτίωση. Το γεγονός αυτό οφείλεται σε κάποια προβλήματα συγχρονισμού, που παρατηρήθηκαν στην πρότυπη υλοποίηση του συστήματος Proteus, αλλά και στην κεντροποιημένη υλοποίηση της βάσης δεδομένων, που περιέχει τον τοπικό κατάλογο υπηρεσιών. Παρόλα αυτά, το πλεονέκτημα της διαθεσιμότητας αντιγράφων για την κατανομή του φόρτου εργασίας του υποσυστήματος επερωτήσεων κατέστη σαφές, καθώς ο αριθμός των ταυτόχρονων επερωτήσεων αυξανόταν. Αξιοσημείωτη επίσης είναι η παρατήρηση ότι, είτε με ή



(i) Μέσος χρόνος επεξεργασίας επερωτήσης

(ii) Μέγιστος χρόνος επεξεργασίας επερωτήσης



(iii) Συνολικός χρόνος επεξεργασίας επερωτήσης

Σχήμα 28: Κλιμάκωση του Επεξεργαστή Επερωτήσεων

χωρίς τη χρήση αντιγράφων, οι χρόνοι επεξεργασίας ανά επερωτήση παρέμειναν σχετικά σταθεροί, για όλες τις επακόλουθες περιπτώσεις ταυτόχρονης υποβολής περισσότερων των πέντε επερωτήσεων.

Ο συνολικός χρόνος επεξεργασίας, συναρτήσει του αριθμού των ταυτόχρονων επερωτήσεων και του αριθμού αντιγράφων του Επεξεργαστή Επερωτήσεων, απεικονίζεται στο διάγραμμα 28(iii). Συνολικά, οι μετρήσεις έδειξαν, ότι το σύστημά μας κλιμακώνεται περίπου γραμμικά, καθώς ο αριθμός των ταυτόχρονων επερωτήσεων αυξάνει. Επίσης φάνηκε ότι, η χρήση καταμεμημένων αντιγράφων του Επεξεργαστή Επερωτήσεων βοηθά στη μείωση των χρόνων επεξεργασίας, για όλες τις περιπτώσεις ταυτόχρονης υποβολής στη μηχανή αναζήτησης περισσότερων των πέντε επερωτήσεων.

### 5.3.2 Συζήτηση

Στην ενότητα αυτή, παρουσιάσαμε μια σειρά μετρήσεων σχετικών με την απόδοση του συστήματος Proteus. Συγκεκριμένα, αξιολογήσαμε την απόδοση του Αναζητητή (Crawler) της μηχανής αναζήτησης, βάσει του αριθμού και περιεχομένου των περιγραφών υπηρεσιών, που επεξεργάζεται και αποθηκεύει στον τοπικό κατάλογο υπηρεσιών. Επίσης, μετρήσαμε το χρόνο επεξεργασίας των επερωτήσεων USQL από τον Επεξεργαστή Επερωτήσεων, ως προς τον αριθμό των ανακτούμενων διαφημίσεων λειτουργιών υπηρεσιών από τον



τοπικό κατάλογο, οι οποίες και υπόκεινται στη διαδικασία αντιστοίχισης.

Από τις μετρήσεις της προηγούμενης παραγράφου φάνηκε, ότι οι χρόνοι επεξεργασίας και για τα δύο υποσυστήματα της μηχανής αναζήτησης Proteus είναι ανάλογοι του αριθμού των αντικειμένων, που καλούνται να επεξεργαστούν. Ακόμα και σε ακραίες περιπτώσεις επεξεργασίας μεγάλου αριθμού αντικειμένων, το σύστημά μας ανταποκρίνεται εμφανίζοντας αρκετά ικανοποιητικές αποδόσεις. Επιπλέον, αντιμετωπίζοντας διάφορα ζητήματα υλοποίησης με πιο αποδοτικές μεθόδους (π.χ. κατάτμηση ή/και δημιουργία αντιγράφων του τοπικού καταλόγου υπηρεσιών), οι οποίες ωστόσο ξεφεύγουν από το ερευνητικό πεδίο της παρούσης διατριβής, υπάρχουν περιθώρια περαιτέρω βελτίωσης της απόδοσης του συστήματος. Αξίζει τέλος να τονιστεί ότι, παρατηρώντας συνδυαστικά τα αποτελέσματα των μετρήσεων, μπορεί κανείς να διαπιστώσει το κέρδος στην απόδοση, που αποκομίζεται από το διαχωρισμό των λειτουργιών των δύο υποσυστημάτων.

Ανάπτυξη μιας γλώσσας και του απαραίτητου λογισμικού εκτέλεσής της, για την αναζήτηση ετερογενών υπηρεσιών με ενιαίο τρόπο

## ΚΕΦΑΛΑΙΟ 6

### ΣΥΜΠΕΡΑΣΜΑΤΑ

Το κεφάλαιο αυτό περιλαμβάνει μια σύνοψη της έρευνας, που πραγματοποιήθηκε στο πλαίσιο της παρούσης διατριβής. Επισημαίνονται τα σημεία συνεισφοράς, ενώ δίνεται μια περιγραφή του προβλεπόμενου περιβάλλοντος εφαρμογής του προτεινόμενου πλαισίου. Τέλος, προδιαγράφονται μια σειρά κατευθύνσεων για μελλοντική έρευνα ή/και εξέλιξη των αποτελεσμάτων.

#### 6.1 Σύνοψη

Η επιτυχία του υπηρεσιοστρεφούς μοντέλου ανάπτυξης εφαρμογών λογισμικού έχει οδηγήσει στην εμφάνιση διαφόρων τεχνολογιών, οι οποίες χρησιμοποιούνται στις μέρες μας για την υλοποίηση συστατικών υπηρεσιών. Χαρακτηριστικοί αντιπρόσωποι των τεχνολογιών αυτών αποτελούν οι υπηρεσίες ιστού, οι υπηρεσίες πλέγματος, καθώς και οι υπηρεσίες ομότιμου-προς-ομότιμο. Παρόλο που, πρωτογενώς, βασίζονται στις ίδιες αρχές, όπως αυτές υπαγορεύονται από το υπηρεσιοστρεφές μοντέλο, οι τρεις αυτές κατηγορίες υπηρεσιών χαρακτηρίζονται από αξιοσημείωτη ετερογένεια, σε ό,τι αφορά τις γλώσσες περιγραφής και τους μηχανισμούς αναζήτησης που υποστηρίζουν. Ως εκ τούτου, και παρά το γεγονός ότι η συνδυασμένη χρήση τους είναι επιθυμητή σε πολλά πεδία εφαρμογών, οι υπηρεσίες των τύπων αυτών δεν είναι μεταξύ τους διαλειτουργικές.

Με την παρούσα εργασία, προτείναμε μια ενιαία προσέγγιση στην αναζήτηση υπηρεσιών ιστού, υπηρεσιών πλέγματος και υπηρεσιών ομότιμου-προς-ομότιμο, ως ένα πρώτο βήμα προς την επίλυση των προβλημάτων, που δημιουργούνται από την προαναφερθείσα ετερογένεια, και την επίτευξη της επιθυμητής διαλειτουργικότητας. Η προσέγγισή μας συνίσταται στη βασική ιδέα, ότι οι χαμηλού επιπέδου λεπτομέρειες των τεχνολογιών αυτών πρέπει να αποκρύπτονται από το χρήστη. Μέσω μιας υψηλού επιπέδου αφαίρεσης, ενιαίας διεπαφής ο χρήστης δύναται να εκφράζει τις λειτουργικές και ποιοτικές απαιτήσεις του και να εκτελεί τις παραγόμενες επερωτήσεις, με τρόπο ανεξάρτητο από τους τύπους των αναζητούμενων υπηρεσιών, από τις περιγραφές τους, αλλά κι από τους μηχανισμούς αναζήτησης, που προσφέρονται από τις υποδομές, στις οποίες οι περιγραφές έχουν δημοσιευτεί.

#### 6.2 Συνεισφορά της Διατριβής

Στη βάση της ενιαίας προσέγγισης για την αναζήτηση ετερογενών υπηρεσιών, η διατριβή αυτή προτείνει ένα πλήρες πλαίσιο, αποτελούμενο από τα εξής:

- *Μια ενιαία γλώσσα επερωτήσεων*, την USQL, η οποία προσφέρει στο χρήστη μια υψηλού επιπέδου διεπαφή για τη διεξαγωγή αναζητήσεων υπηρεσιών ιστού, υπηρεσιών πλέγματος και υπηρεσιών ομότιμου-προς-ομότιμο. Η USQL υποστηρίζει το σχηματισμό επερωτήσεων, οι οποίες μπορούν να εκφράσουν ένα ευρύ φάσμα λειτουργικών και ποιοτικών απαιτήσεων. Οι επερωτήσεις μορφοποιούνται ανεξάρτητα από τον τύπο της εκάστοτε αναζητούμενης υπηρεσίας, ενώ, χάρη στην επεκτασιμότητα και αφαιρετικότητα της, η USQL μπορεί εύκολα να ενσωματώσει επιπρόσθετη πληροφορία, νέα είδη κριτηρίων αναζήτησης, καθώς και να υποστηρίξει την αναζήτηση άλλων τύπων υπηρεσιών.
- *Μια μηχανή αναζήτησης ετερογενών υπηρεσιών*, με την ονομασία Proteus, που υποστηρίζει την εκτέλεση επερωτήσεων USQL, παρέχοντας πρόσβαση προς τα περιεχόμενα διαφόρων ετερογενών μητρώων, αποθετηρίων, καταλόγων και δικτύων, στα οποία έχουν δημοσιευτεί οι περιγραφές των αναζητούμενων υπηρεσιών. Η μηχανή Proteus ακολουθεί παρόμοια αρχιτεκτονική με τις παραδοσιακές μηχανές αναζήτησης Ιστού, σύμφωνα με την οποία η διαδικασία ανάκτησης, επεξεργασίας και κατηγοριοποίησης των περιγραφών υπηρεσιών είναι πλήρως διαχωρισμένη από τη διαδικασία εκτέλεσης των επερωτήσεων. Χάρη σε αυτή τη σχεδιαστική επιλογή, η μηχανή Proteus καθίσταται κατάλληλη για χρήση σε περιπτώσεις, όπου είναι επιθυμητή, ή αναγκαία, η ταυτόχρονη αναζήτηση υπηρεσιών σε μεγάλο αριθμό μητρώων, αποθετηρίων, δικτύων, κτλ. Επιπλέον, υλοποιώντας και συνδυάζοντας με κατάλληλο τρόπο τα αποτελέσματα υπάρχοντων αλγορίθμων ταυτοποίησης, η μηχανή προσδίδει στα αποτελέσματα αναζήτησης, που παράγονται από την εκτέλεση των επερωτήσεων USQL, αξιοσημείωτη ακρίβεια και ανάκληση, ενώ οι χρόνοι απόκρισης κρίνονται ικανοποιητικοί, ακόμα και για σχετικά μεγάλους αριθμούς υπηρεσιών.

Ένα βασικό χαρακτηριστικό του προτεινόμενου από τη διατριβή πλαισίου αναζήτησης ετερογενών υπηρεσιών, είναι η επεκτασιμότητά του. Το μετα-μοντέλο της γλώσσας USQL ορίζει κατάλληλα σημεία επέκτασης, τα οποία επιτρέπουν τη μελλοντική ενσωμάτωση στη γλώσσα στοιχείων και δομών, εφόσον αυτό κριθεί αναγκαίο. Έτσι, καθίσταται δυνατός ο εμπλουτισμός (i) των επερωτήσεων με επιπλέον κριτήρια αναζήτησης, τα οποία δεν υποστηρίζονται πρωτογενώς από τη γλώσσα και (ii) των αποτελεσμάτων αναζήτησης, με οποιαδήποτε πληροφορία κρίνεται απαραίτητη. Αξίζει στο σημείο αυτό να τονιστεί το γεγονός ότι, ήδη, όπως περιγράφεται σε σχετική μας εργασία [90], η δυνατότητα επέκτασης της USQL αξιοποιήθηκε στο πλαίσιο μιας προσπάθειας για την υποστήριξη αναζήτησης σύνθετων υπηρεσιών, βάσει χαρακτηριστικών συμπεριφοράς τους.

Σε αλγοριθμικό επίπεδο, το μαθηματικό μοντέλο της USQL, που εφαρμόζεται για τον υπολογισμό του συνολικού βαθμού αντιστοιχίας μιας υπηρεσίας, διατηρεί την ανεξαρτησία του από τους επιμέρους αλγορίθμους, που χρησιμοποιούνται για την αντιστοίχιση των

Πίνακας 12: Υλοποιημένα συνδεδεμένα συστατικά

Είδος Επέκτασης	Plug-in	Λειτουργικότητα
harvester	UDDIHarvester	Αναζήτηση σε μητρώα UDDI v2
harvester	EBXMLHarvester	Αναζήτηση σε μητρώα ebXML v2
harvester	JXTAHarvester	Αναζήτηση σε δίκτυα ομότιμων κόμβων JXTA 2.5
harvester	GlobusHarvester	Αναζήτηση σε καταλόγους Globus (GT4)
parser	WSDLParser	Επεξεργασία εγγράφων WSDL 1.1 με υποστήριξη επεκτάσεων SAWSDL, OWL-S, WSRP και WS-QoS
parser	MSAParser	Επεξεργασία εγγράφων MSA JXTA 2.5
datatypeMatcher	XSDMatcher	Αντιστοίχιση τύπων δεδομένων του XML Schema 1.1
qosMatcher	AvailabilityMatcher	Αντιστοίχιση τιμών διαθεσιμότητας υπηρεσίας
qosMatcher	ReliabilityMatcher	Αντιστοίχιση τιμών αξιοπιστίας υπηρεσίας
qosMatcher	ProcessTimeMatcher	Αντιστοίχιση τιμών χρόνου επεξεργασίας υπηρεσίας

διαφόρων ειδών τιμών. Η ευελιξία αυτή στον ορισμό του, επιτρέπει τη μελλοντική αντικατάσταση των υποστηριζόμενων αλγορίθμων αντιστοίχισης τιμών κειμένου, σημασιολογικών εννοιών και αριθμητικών πεδίων, εφόσον αυτό κριθεί απαραίτητο, ή το συνδυασμό τους με άλλους, εξειδικευμένους αλγορίθμους στις επιμέρους περιοχές.

Τέλος, η μηχανή αναζήτησης ετερογενών υπηρεσιών Proteus προσφέρει μια σειρά σημείων επέκτασης, υποστηριζόμενα από έναν ισχυρό μηχανισμό διαχείρισης επεκτάσεων, που επιτρέπουν την ενσωμάτωση επιπρόσθετης λειτουργικότητας και την υποστήριξη διαφόρων ετερογενών προτύπων περιγραφής και αναζήτησης υπηρεσιών, με τη μορφή συνδεδεμένων συστατικών (plug-ins). Η επεκτασιμότητα του συστήματος ελέγχθηκε κατά την εκπόνηση της παρούσης διατριβής, μέσω της ανάπτυξης διαφόρων συνδεδεμένων συστατικών (Πίνακας 12), τα οποία τεκμηριώνουν τη βιωσιμότητα της προσέγγισής μας.

### 6.3 Περιβάλλον Εφαρμογής

Ήδη από το εισαγωγικό κεφάλαιο της διατριβής, περιγράψαμε την ανάγκη χρήσεως ετερογενών τύπων υπηρεσιών, για την ανάπτυξη υπηρεσιοστρεφών συστημάτων σε διάφορα εφαρμοσμένα πεδία. Όπως φάνηκε από την παρουσίαση των δύο περιπτώσεων χρήσης (βλ. παραγράφους 1.1.1 και 1.1.2), η ανάγκη ενιαίας αναζήτησης ετερογενών υπηρεσιών εμφανίζεται έντονα στα τμήματα μηχανογράφησης οργανισμών, οι οποίοι αναπτύσσουν εφαρμογές αποτελούμενες από ετερογενείς υπηρεσίες, όπως υπηρεσίες ιστού, πλέγματος και ομότιμου-προς-ομότιμο.

Το προτεινόμενο πλαίσιο ενιαίας αναζήτησης ετερογενών υπηρεσιών προσφέρει μια ιδανική λύση σε περιβάλλοντα, όπου η αναζήτηση υπηρεσιών πραγματοποιείται κατά ελεγχόμενο τρόπο, τουλάχιστον ως προς τους στόχους αναζήτησης. Πράγματι, όπως περιγράψαμε στο Κεφάλαιο 4, ο κατάλογος υπηρεσιών του συστήματος Proteus τροφοδοτείται με περιγραφές υπηρεσιών, οι οποίες προέρχονται από μητρώα, αποθετήρια, δίκτυα ομότιμων κόμβων, ή άλλους στόχους αναζήτησης, των οποίων τα στοιχεία πρόσβασης

έχουν εκ των προτέρων καταγραφεί στο σύστημα. Έτσι, θεωρούμε οργανισμούς μεσαίου έως μεγάλου μεγέθους, με τμήματα, καθένα εκ των οποίων ενδεχομένως χρησιμοποιεί το δικό του μητρώο για τη δημοσίευση των υπηρεσιών που αναπτύσσει, ως το πλέον κατάλληλο περιβάλλον χρήσης του πλαισίου μας. Επίσης, το σύστημά μας μπορεί να καλύψει τις ανάγκες αναζήτησης σε εξωτερικούς ως προς έναν οργανισμό στόχους αναζήτησης, που δημιουργούνται σε περιπτώσεις (α) σύναψης μιας ή περισσότερων συμφωνιών σε επίπεδο υπηρεσιών, μεταξύ του οργανισμού και άλλων εταιρών, ή/και (β) συμμετοχής του οργανισμού σε δίκτυα διαμοιρασμού υπηρεσιών (π.χ. εξειδικευμένα, υπηρεσιοστρεφή δίκτυα ομότιμων κόμβων, ιδεατοί οργανισμοί πλέγματος, κ.ά.).

## **6.4 Κατευθύνσεις Μελλοντικής Έρευνας**

Το πλαίσιο ενιαίας αναζήτησης ετερογενών υπηρεσιών, που παρουσιάστηκε στην παρούσα διατριβή, ενσωματώνει ιδέες και τεχνικές προερχόμενες από ένα ευρύ ερευνητικό και τεχνολογικό φάσμα. Λαμβάνοντας υπόψη τις τρέχουσες εξελίξεις στην περιοχή των υπηρεσιοστρεφών τεχνολογιών, καθώς και τις γενικότερες τάσεις στο Διαδίκτυο, προδιαγράφουμε στην ενότητα αυτή μια σειρά θεμάτων, που συνδέονται με την παρούσα εργασία, χρήζουν ιδιαίτερου ενδιαφέροντος και θεωρούμε πως μπορούν να αποτελέσουν τους κατευθυντήριους άξονες για μελλοντική έρευνα, έχοντας ως στόχο τη βελτίωση και επέκταση των δυνατοτήτων του προτεινόμενου πλαισίου αναζήτησης ετερογενών υπηρεσιών. Αξίζει να σημειωθεί, ότι η επεκτασιμότητα, που χαρακτηρίζει το προτεινόμενο πλαίσιο, καθιστά εύκολη, και σε κάθε περίπτωση δυνατή, την ενσωμάτωση των αποτελεσμάτων καθεμιάς από τις ακόλουθες κατευθύνσεις μελλοντικής έρευνας.

### **6.4.1 Υποστήριξη Επιπρόσθετων Τύπων Υπηρεσιών**

Καθώς οι υπηρεσιοστρεφείς τεχνολογίες εξελίσσονται και η υπηρεσιοστρεφής θεώρηση ενσωματώνεται στο πλαίσιο διαφόρων αρχιτεκτονικών, νέοι τύποι υπηρεσιών κάνουν την εμφάνισή τους, χαρακτηριζόμενοι από ετερογένεια. Πρόσφατα παραδείγματα αποτελούν οι υπηρεσίες αρχιτεκτονικής REST (Representational State Transfer) [125], οι υπηρεσίες OSGi (Open Services Gateway initiative) [95], κ.ά., η υποστήριξη των οποίων από το πλαίσιο μας συνιστά μια πρόκληση. Πράγματι, η ενσωμάτωση κατάλληλων συστατικών για την αναζήτηση επιπλέον τύπων υπηρεσιών, πέραν των υπηρεσιών ιστού, πλέγματος και ομότιμου-προς-ομότιμο, ενισχύει τη χρησιμότητα της προσέγγισής μας, διευρύνοντας το πεδίο εφαρμογής της, ενώ την καθιστά βιώσιμη σε βάθος χρόνου.

## 6.4.2 Αναζήτηση Υπηρεσιών Βάσει Πληροφορίας Περιβάλλοντος Πλαισίου

Στην πλειοψηφία τους, οι διάφορες προσεγγίσεις στην αναζήτηση υπηρεσιών δε λαμβάνουν υπόψη πληροφορία, η οποία προέρχεται από το περιβάλλον πλαίσιο (context) του χρήστη, ή του παρόχου της αναζητούμενης υπηρεσίας. Τέτοιου είδους πληροφορία μπορεί να περιλαμβάνει τη γεωγραφική τοποθεσία, τόσο του χρήστη όσο και της υπηρεσίας, στοιχεία σχετικά με τις προσωπικές προτιμήσεις του χρήστη (προφίλ), την τρέχουσα κατάσταση της εφαρμογής, στην οποία πρόκειται να ενσωματωθεί η υπό αναζήτηση υπηρεσία κτλ. Σε πολλές περιπτώσεις, αυτού του είδους τα δεδομένα παίζουν καθοριστικό ρόλο στην επιλογή μιας υπηρεσίας έναντι μιας άλλης, συνεπώς θα έπρεπε να λαμβάνονται υπόψη κατά τη διαδικασία της αναζήτησης υπηρεσιών. Το γεγονός αυτό προδιαγράφει μια ενδιαφέρουσα κατεύθυνση για μελλοντική έρευνα, με στόχο την υποστήριξη πληροφορίας περιβάλλοντος πλαισίου, τόσο στο επίπεδο έκφρασης των σχετικών κριτηρίων αναζήτησης σε επερωτήσεις USQL, όσο και στο επίπεδο της αλγοριθμικής υποστήριξης αυτών από τη μηχανή αναζήτησης Proteus. Ήδη, στη βιβλιογραφία έχουν καταγραφεί κάποιες ερευνητικές προσπάθειες προς αυτή την κατεύθυνση (βλ. π.χ. [19], [107]), οι οποίες μπορούν να χρησιμοποιηθούν ως βάση για τη σχεδίαση και υλοποίηση των κατάλληλων επεκτάσεων στο σύστημά μας. Παρόλα αυτά, για την επιτυχή χρήση τέτοιων ειδών κριτηρίων αναζήτησης, ανάλογες προσπάθειες απαιτούνται να καταβληθούν στην περιοχή των τεχνολογιών περιγραφής υπηρεσιών, για την υποστήριξη αντίστοιχων πληροφοριών.

## 6.4.3 Αναζήτηση Σύνθετων Υπηρεσιών Βάσει Κριτηρίων Συμπεριφοράς

Στην παρούσα του μορφή, το προτεινόμενο πλαίσιο υποστηρίζει την ενιαία αναζήτηση απλών (ατομικών) ετερογενών υπηρεσιών. Παρόλα αυτά, καθώς το αποτέλεσμα της σύνθεσης υπηρεσιών προσφέρεται συνήθως ως μια νέα, σύνθετη υπηρεσία, η περιγραφή της συμπεριφοράς, που προσδιορίζεται από τις ανταλλαγές μηνυμάτων μεταξύ των συνιστωσών υπηρεσιών, αποκτά ιδιαίτερη σημασία. Στη βιβλιογραφία αναφέρονται μια σειρά ερευνητικών προσπαθειών, οι οποίες λαμβάνουν υπόψη τη συμπεριφορά σύνθετων υπηρεσιών ιστού, κατά την αναζήτησή τους [91] [71]. Το πλαίσιο μας θα μπορούσε μελλοντικά να επεκταθεί, ενσωματώνοντας ή/και επεκτείνοντας τις υπάρχουσες προσεγγίσεις, προκειμένου να υποστηρίξει την αναζήτηση σύνθετων υπηρεσιών, αποτελούμενων από ετερογενείς ατομικές υπηρεσίες, βάσει κριτηρίων σχετικών με την επιθυμητή συμπεριφορά. Ήδη, στο επίπεδο της γλώσσας USQL, έχουμε παρουσιάσει κάποια αρχικά αποτελέσματα προς την κατεύθυνση αυτή [90]. Η εργασία αυτή θα μπορούσε μελλοντικά να συμπερίλαβει την υλοποίηση των αντίστοιχων αλγορίθμων και την ενσωμάτωσή τους στη μηχανή αναζήτησης Proteus.

#### 6.4.4 Δυναμική Σύνθεση Υπηρεσιών

Όπως προειπώθηκε, το πλαίσιό μας υποστηρίζει την αναζήτηση απλών (ατομικών) ετερογενών υπηρεσιών, βάσει πολλαπλών κριτηρίων σχετικών με διάφορες λειτουργικές και ποιοτικές ιδιότητές τους. Ωστόσο, υπάρχουν περιπτώσεις όπου, αν και οι απαιτήσεις του χρήστη δεν ικανοποιούνται στο σύνολό τους από μία υπηρεσία, μπορούν εν τούτοις να ικανοποιηθούν από το γραμμικό συνδυασμό δύο ή περισσότερων υπηρεσιών (service chaining). Η μελέτη υπαρχουσών τεχνικών δυναμικής σύνθεσης υπηρεσιών (βλ. [97]) και η εξέταση της δυνατότητας ενσωμάτωσής τους στη διαδικασία αναζήτησης αποτελούν μια πρόκληση, η οποία θεωρούμε πως μελλοντικά θα χρειαστεί να αντιμετωπιστεί από τις διάφορες λύσεις αναζήτησης υπηρεσιών. Συγκεκριμένα, ο δυναμικός εντοπισμός υπηρεσιών, οι οποίες συνδυαστικά αντιστοιχούν στα κριτήρια της επερώτησης USQL, η αλγοριθμική υποστήριξη στον έλεγχο της καταλληλότητας των πιθανών συνδυασμών τους, καθώς και η κατάλληλη περιγραφή της σύνθεσης αυτών στην απόκριση USQL, προσφέρουν κατάλληλο θεματολογικό έδαφος για μελλοντική έρευνα, στο δρόμο προς τη βελτίωση και επέκταση των δυνατοτήτων του προτεινόμενου πλαισίου αναζήτησης ετερογενών υπηρεσιών.

#### 6.4.5 Υποστήριξη Συστατικών Σημασιολογικών Μεσολαβητών

Στην Παράγραφο 4.5.3, περιγράψαμε το συστατικό αντιστοίχισης του συστήματος Proteus, όπου αναφερθήκαμε στον, σχετικά απλό, αλγόριθμο αντιστοίχισης οντολογικών εννοιών, που υιοθετείται, μέσω του συστατικού OntologyMatcher. Στην παρούσα του υλοποίηση, το συστατικό αυτό δεν έχει τη δυνατότητα αντιστοίχισης οντολογικών εννοιών, οι οποίες προέρχονται από διαφορετικές οντολογίες ή/και ακολουθούν διαφορετικά σχήματα. Δεδομένης της έντονης ερευνητικής δραστηριότητας στην περιοχή των τεχνολογιών Σημασιολογικού Ιστού, ο προαναφερθής περιορισμός του συστήματός μας θα μπορούσε να αρθεί, μέσω της χρήσης υπαρχουσών λύσεων στο πρόβλημα της ετερογένειας μεταξύ των διάφορων οντολογιών, οι οποίες έχουν προταθεί και είναι διαθέσιμες στη βιβλιογραφία [61] [25]. Μια πιθανή προσέγγιση θα ήταν η χρήση συστατικών σημασιολογικών μεσολαβητών (semantic mediation components), τα οποία υποστηρίζουν την (ημι)αυτόματη απεικόνιση εννοιών, από μια οντολογία σε μια άλλη. Σε κάθε περίπτωση, οποιαδήποτε βελτίωση στην αντιστοίχιση οντολογικών εννοιών επηρεάζει μόνο την υλοποίηση του συστατικού OntologyMatcher, αφήνοντας το υπόλοιπο σύστημα ανέπαφο.

#### 6.4.6 Αξιοποίηση Τεχνικών Κοινωνικής Δικτύωσης

Τα τελευταία χρόνια, η ανάπτυξη μιας νέας γενιάς τεχνολογιών Ιστού (Web 2.0), έδωσε τη δυνατότητα στους χρήστες να συμμετέχουν ενεργά στο πλαίσιο διάφορων εφαρμογών, διαμορφώνοντας τα περιεχόμενά τους μέσω δραστηριοτήτων κοινωνικής δικτύωσης



(social networking). Όπως δείξαμε σε πρόσφατη εργασία μας [87], η αξιοποίηση τεχνικών κοινωνικής δικτύωσης, όπως ο σχολιασμός με τη χρήση λέξεων-κλειδιών από τους χρήστες, και η ενσωμάτωση αυτών στο υπηρεσιοστρεφές μοντέλο, μπορούν να βελτιώσουν τη διαδικασία αναζήτησης υπηρεσιών, μέσω της συλλογικής ευφυΐας (collective intelligence) που αναπτύσσεται αναφορικά με τη σημασιολογία των διαφημιζόμενων υπηρεσιών. Στο μέλλον, σχεδιάζουμε να εστιάσουμε τις ερευνητικές μας προσπάθειες προς αυτή την κατεύθυνση, προκειμένου να ολοκληρώσουμε την, ήδη εν εξελίξει, πλατφόρμα και να την ενσωματώσουμε στο σύστημα Proteus.

## 6.5 Επίλογος

Η χρήση υπηρεσιοστρεφών τεχνολογιών για την ανάπτυξη και διάθεση εξεζητημένων, διαλειτουργικών λύσεων λογισμικού αποτελεί μια από τις πλέον δημοφιλείς επιλογές στις μέρες μας. Το γεγονός αυτό έχει επιφέρει μια συνεχή αύξηση στον αριθμό των υπηρεσιών, που δημοσιεύονται στο διαδίκτυο και καθιστώνται από τους παρόχους τους διαθέσιμες προς αναζήτηση και χρήση. Η ιστοσελίδα της διαδικτυακής μηχανής δημοσίευσης υπηρεσιών ιστού Seekda [105] αναφέρει πάνω από 28.000 δημοσιευμένες υπηρεσίες ιστού. Παρόμοιες αυξητικές τάσεις, αν και ίσως σε μικρότερη κλίμακα, παρατηρούνται στον αριθμό των διαθέσιμων υπηρεσιών πλέγματος και υπηρεσιών ομότιμου-προς-ομότιμο.

Σε ένα τέτοιο περιβάλλον προσφοράς, η διαδικασία της αναζήτησης υπηρεσιών λαμβάνει εξέχουσα σημασία στον κύκλο ανάπτυξης υπηρεσιοστρεφών εφαρμογών, καθώς υποστηρίζει τους χρήστες στην εύρεση και αναγνώριση, από ένα δυνητικά μεγάλο σύνολο, κατάλληλων υπηρεσιών, που ανταποκρίνονται στις διάφορες απαιτήσεις τους. Ωστόσο, η ετερογένεια που διέπει τις υπάρχουσες υπηρεσιοστρεφείς τεχνολογίες και αφορά στα υποστηριζόμενα μοντέλα, πρότυπα περιγραφής και μηχανισμούς αναζήτησης, καθιστά την αναζήτησή τους ιδιαίτερα επίπονη, χρονοβόρα και απαιτητική σε πόρους. Έτσι, αν και η διαλειτουργικότητα αποτελεί έναν από τους πρωτεύοντες στόχους των υπηρεσιοστρεφών τεχνολογιών, η επίτευξή της στο επίπεδο της συνεργασίας διαφορετικών τύπων υπηρεσιών συναντά πρωτογενή εμπόδια.

Το προτεινόμενο από αυτή τη διατριβή πλαίσιο προσφέρει μια λύση στο πολυδιάστατο πρόβλημα της ετερογένειας, και αποτελεί ένα πρώτο βήμα προς την κατεύθυνση επίτευξης της επιθυμητής διαλειτουργικότητας. Συγκεκριμένα, η διατριβή εστιάζει στην άρση των παρακάτω βασικών περιορισμών των υφιστάμενων προσεγγίσεων: (α) υποστήριξη μιας συγκεκριμένης υπηρεσιοστρεφούς τεχνολογίας και συνεπώς αναζήτηση σε ένα υποσύνολο των διαθέσιμων υπηρεσιών, (β) εξάρτηση από συγκεκριμένα πρωτόκολλα και πρότυπα περιγραφής υπηρεσιών, (γ) αναζήτηση σε συγκεκριμένα είδη μητρώων δημοσίευσης υπηρεσιών και (δ) αδυναμία υποστήριξης πολυδιάστατων κριτηρίων αναζήτησης, τόσο στη σύνταξη των επερωτήσεων, όσο και στους αλγορίθμους ταυτοποίησης.

Σε αντιδιαστολή με τις υπάρχουσες προσεγγίσεις, το πλαίσιο που προτείνεται με την παρούσα διατριβή αντιμετωπίζει τα προαναφερθέντα προβλήματα, ενώ υποστηρίζει την ενιαία αναζήτηση τριών δημοφιλών, πλην όμως ετερογενών τύπων υπηρεσιών: ιστού, πλέγματος, και ομότιμου-προς-ομότιμο. Ταυτόχρονα, χάρη στις εγγενείς δυνατότητες επέκτασης, δίνεται η δυνατότητα μελλοντικής υποστήριξης άλλων εμφανιζόμενων υπηρεσιοστρεφών τεχνολογιών.

Ο συγκερασμός πρακτικών από πολλές ερευνητικές περιοχές για την ανάπτυξη της προτεινόμενης λύσης, σε συνδυασμό με τις εξελίξεις στην περιοχή των υπηρεσιοστρεφών τεχνολογιών, δημιουργούν πρόσφορο έδαφος δίνοντας πολλαπλές ενδιαφέρουσες ευκαιρίες για μελλοντική έρευνα στη θεματική περιοχή της αναζήτησης ετερογενών υπηρεσιών.

## ΠΑΡΑΡΤΗΜΑ

### ΣΧΗΜΑ XML ΤΗΣ ΓΛΩΣΣΑΣ USQL

Παρακάτω δίνεται ο ορισμός της γλώσσας USQL με τη χρήση της μετα-γλώσσας XML Schema.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns="http://usql.di.uoa.gr"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://usql.di.uoa.gr"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:complexType name="DataType">
    <xs:annotation>
      <xs:documentation>Type of data types</xs:documentation>
    </xs:annotation>
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute name="namespace" type="xs:string" use="required"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="ServiceRequirement" abstract="true">
    <xs:annotation>
      <xs:documentation>Abstract type for service requirements</xs:documentation>
    </xs:annotation>
    <xs:attribute name="weight" type="xs:double" use="optional"/>
  </xs:complexType>
  <xs:complexType name="DescribedServiceRequirement" abstract="true">
    <xs:annotation>
      <xs:documentation>
        Abstract type for service requirements that can be described
      </xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="ServiceRequirement">
        <xs:sequence>
          <xs:element name="Description" type="xs:string" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="ontologyReference" type="xs:anyURI" use="optional"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="DescribedAndTypedServiceRequirement" abstract="true">
    <xs:annotation>
      <xs:documentation>
        Abstract type for service requirements that can be described and also have a data type
      </xs:documentation>
    </xs:annotation>
    <xs:complexContent>
      <xs:extension base="DescribedServiceRequirement">
        <xs:sequence>
          <xs:element name="Type" type="DataType" minOccurs="0"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:schema>
```

```

        </xs:sequence>
    </xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:attributeGroup name="ProjectableAttributes">
    <xs:annotation>
        <xs:documentation>Attributes of projectable service requirements</xs:documentation>
    </xs:annotation>
    <xs:attribute name="showInResponse" type="xs:boolean" use="optional"/>
</xs:attributeGroup>
<xs:complexType name="ServiceCapability">
    <xs:annotation>
        <xs:documentation>Type of requirement towards the service capability</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
        <xs:extension base="DescribedServiceRequirement">
            <xs:attributeGroup ref="ProjectableAttributes"/>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="MessageElement">
    <xs:annotation>
        <xs:documentation>
            Type of requirement towards the service's input/output message elements
        </xs:documentation>
    </xs:annotation>
    <xs:complexContent>
        <xs:extension base="DescribedAndTypedServiceRequirement"/>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="Message">
    <xs:annotation>
        <xs:documentation>
            Type of requirement towards the service's input/output messages
        </xs:documentation>
    </xs:annotation>
    <xs:complexContent>
        <xs:extension base="DescribedServiceRequirement">
            <xs:sequence>
                <xs:element name="Element" type="MessageElement" minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="ServiceInterface">
    <xs:annotation>
        <xs:documentation>Type of requirement towards the service interface</xs:documentation>
    </xs:annotation>
    <xs:complexContent>
        <xs:extension base="ServiceRequirement">
            <xs:sequence>
                <xs:element name="Input" type="Message" minOccurs="0"/>
                <xs:element name="Output" type="Message" minOccurs="0"/>
            </xs:sequence>
            <xs:attributeGroup ref="ProjectableAttributes"/>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>

```

```

<xs:complexType name="ResourceProperty">
  <xs:annotation>
    <xs:documentation>
      Type of requirement towards the service's resource properties
    </xs:documentation>
  </xs:annotation>
  <xs:complexContent>
    <xs:extension base="DescribedAndTypedServiceRequirement"/>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="ServiceResource">
  <xs:annotation>
    <xs:documentation>Type of requirement towards the service's resource</xs:documentation>
  </xs:annotation>
  <xs:complexContent>
    <xs:extension base="DescribedServiceRequirement">
      <xs:sequence>
        <xs:element name="Property" type="ResourceProperty" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attributeGroup ref="ProjectableAttributes"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="GenericProperty">
  <xs:annotation>
    <xs:documentation>Generic type for name-value properties</xs:documentation>
  </xs:annotation>
  <xs:attribute name="name" type="xs:string" use="required"/>
  <xs:attribute name="value" type="xs:anySimpleType" use="required"/>
</xs:complexType>
<xs:complexType name="QoSRequirement">
  <xs:annotation>
    <xs:documentation>Type of requirement towards various QoS properties</xs:documentation>
  </xs:annotation>
  <xs:complexContent>
    <xs:extension base="DescribedServiceRequirement">
      <xs:sequence>
        <xs:element name="Qualifier" type="GenericProperty" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="value" type="xs:anySimpleType" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="QoS">
  <xs:annotation>
    <xs:documentation>Type of requirement towards the service's QoS</xs:documentation>
  </xs:annotation>
  <xs:complexContent>
    <xs:extension base="ServiceRequirement">
      <xs:sequence>
        <xs:element name="QoSProperty" type="QoSRequirement" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attributeGroup ref="ProjectableAttributes"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="SearchFilterValue">
  <xs:annotation>

```

```

    <xs:documentation>Type of search filter values</xs:documentation>
</xs:annotation>
<xs:sequence>
    <xs:element name="AdditionalProperty" type="GenericProperty" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
<xs:attribute name="useWildcards" type="xs:boolean" use="optional"/>
<xs:attribute name="value" type="xs:anySimpleType" use="required"/>
</xs:complexType>
<xs:complexType name="SearchFilter">
    <xs:annotation>
        <xs:documentation>Type of search filters</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="Option" type="SearchFilterValue" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="serviceProperty" type="xs:string" use="required"/>
    <xs:attribute name="excludeValues" type="xs:boolean" use="optional"/>
</xs:complexType>
<xs:complexType name="SearchFilters">
    <xs:annotation>
        <xs:documentation>Set of search filters</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="Filter" type="SearchFilter" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="SearchTarget">
    <xs:annotation>
        <xs:documentation>Type of search targets</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="AccessProperty" type="GenericProperty" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="type" type="xs:string" use="required"/>
</xs:complexType>
<xs:complexType name="SearchTargets">
    <xs:annotation>
        <xs:documentation>Set of search targets</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="Target" type="SearchTarget" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="SearchCriteria">
    <xs:annotation>
        <xs:documentation>Type of search criteria container</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="Capability" type="ServiceCapability" minOccurs="0"/>
        <xs:element name="Interface" type="ServiceInterface" minOccurs="0"/>
        <xs:element name="Resource" type="ServiceResource" minOccurs="0"/>
        <xs:element name="QoS" type="QoS" minOccurs="0"/>
        <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="USQLDocument" abstract="true">
    <xs:annotation>

```

```

    <xs:documentation>Abstract type of USQL documents</xs:documentation>
</xs:annotation>
<xs:attribute name="specVersion" type="xs:double" use="required"/>
<xs:attribute name="id" type="xs:string" use="required"/>
</xs:complexType>
<xs:complexType name="USQLRequest">
  <xs:annotation>
    <xs:documentation>Type of USQL requests</xs:documentation>
  </xs:annotation>
  <xs:complexContent>
    <xs:extension base="USQLDocument">
      <xs:sequence>
        <xs:element name="SearchFilters" type="SearchFilters"/>
        <xs:element name="SearchCriteria" type="SearchCriteria"/>
        <xs:element name="SearchTargets" type="SearchTargets"/>
      </xs:sequence>
      <xs:attribute name="minMatch" type="xs:double" use="optional"/>
      <xs:attribute name="maxResults" type="xs:int" use="optional"/>
      <xs:attribute name="serviceType" type="xs:string" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:element name="USQLRequest" type="USQLRequest">
  <xs:annotation>
    <xs:documentation>The USQL request element</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:complexType name="ServiceProvider">
  <xs:annotation>
    <xs:documentation>Type of service providers</xs:documentation>
  </xs:annotation>
  <xs:attribute name="name" type="xs:string" use="required"/>
  <xs:attribute name="description" type="xs:string" use="optional"/>
  <xs:attribute name="url" type="xs:anyURI" use="optional"/>
</xs:complexType>
<xs:complexType name="ServiceInvocationDetails">
  <xs:annotation>
    <xs:documentation>Type of invocation properties container</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="InvocationProperty" type="GenericProperty" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="MatchedService">
  <xs:annotation>
    <xs:documentation>Type of matched service entries</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="Provider" type="ServiceProvider"/>
    <xs:element name="InvocationDetails" type="ServiceInvocationDetails"/>
    <xs:element name="Capability" type="ServiceCapability" minOccurs="0"/>
    <xs:element name="Interface" type="ServiceInterface" minOccurs="0"/>
    <xs:element name="Resource" type="ServiceResource" minOccurs="0"/>
    <xs:element name="QoS" type="QoS" minOccurs="0"/>
    <xs:element ref="Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="rank" type="xs:double" use="required"/>
  <xs:attribute name="serviceType" type="xs:string" use="required"/>

```

```
<xs:attribute name="name" type="xs:string" use="required"/>
<xs:attribute name="description" type="xs:string" use="optional"/>
</xs:complexType>
<xs:complexType name="USQLResponse">
  <xs:annotation>
    <xs:documentation>Type of USQL responses</xs:documentation>
  </xs:annotation>
  <xs:complexContent>
    <xs:extension base="USQLDocument">
      <xs:sequence>
        <xs:element name="Match" type="MatchedService" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="requestId" type="xs:string" use="required"/>
      <xs:attribute name="truncated" type="xs:boolean" use="required"/>
      <xs:attribute name="queryDuration" type="xs:long" use="required"/>
      <xs:attribute name="processedEntries" type="xs:int" use="required"/>
      <xs:attribute name="matches" type="xs:int" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:element name="USQLResponse" type="USQLResponse">
  <xs:annotation>
    <xs:documentation>The USQL response element</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="Extension" type="ServiceRequirement" abstract="true">
  <xs:annotation>
    <xs:documentation>Extension point element</xs:documentation>
  </xs:annotation>
</xs:element>
</xs:schema>
```



## ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΩΝ

Ελληνικός Όρος	Αγγλικός Όρος
Ακροδέκτης Υπηρεσίας	Service Endpoint
Αναζήτηση Υπηρεσιών	Service Discovery
Ανακλαστικό Πρότυπο Σχεδίασης	Reflective Design Pattern
Ανάκτηση Πληροφορίας	Information Retrieval
Αναφορά Ξένου Κλειδιού	Foreign Key Reference
Αντιστοίχιση	Matchmaking
Αντιστοίχιση Οντολογιών	Ontology Mapping
Αξιοπιστία	Reliability
Αποθετήριο	Repository
Βαθμός Αντιστοιχίας	Degree of Match
Γλώσσα Εκφράσεων	Expression Language
Διάγραμμα Οντοτήτων-Συσχετίσεων	Entity-Relationship Diagram
Διάγραμμα Ροής	Flow Chart
Διαθεσιμότητα	Availability
Διαλειτουργικότητα	Interoperability
Διαφήμιση Υπηρεσίας	Service Advertisement
Διαχείριση Κύκλου Ζωής	Lifecycle Management
Εικονικοποίηση	Virtualization
Εικονικός Οργανισμός	Virtual Organization
Εννοιολογική Αποσαφήνιση Λέξεως	Word Sense Disambiguation
Επεξεργασία Φυσικής Γλώσσας	Natural Language Processing
Επιδράσεις	Effects
Εργασία Συστήματος	System Job
Ισχυρός Περιορισμός	Hard Constraint
Καταναλωτής Υπηρεσίας	Service Consumer
Κοινωνική Δικτύωση	Social Networking
Λέξη-Κλειδί	Keyword
Με Διατήρηση της Κατάστασης	Stateful
Μεσάζων Υπηρεσίας	Service Broker
Μεταβατικός	Transient
Μητρώο	Registry
Μηχανή Αναζήτησης Ιστού	Web Search Engine
Μηχανισμός Δημοσίευσης/Συνδρομής	Publish/Subscribe Mechanism
Μοντέλο Διανυσματικού Χώρου	Vector Space Model

<b>Ελληνικός Όρος</b>	<b>Αγγλικός Όρος</b>
Νήμα Εκτέλεσης	Execution Thread
Ολοκλήρωση	Integration
Οντολογία Ανωτέρου Επιπέδου	Upper Ontology
Οντολογία Πεδίου	Domain Ontology
Όψη	Viewpoint
Πάροχος Υπηρεσίας	Service Provider
Πλέγμα	Grid
Πλεγματοειδές Περιβάλλον	Grid Environment
Πράκτορας Λογισμικού	Software Agent
Πρόβλημα Ανάθεσης	Assignment Problem
Προσδιοριστής	Qualifier
Πρότυπο Υποννοούμενου Πόρου	Implied Resource Pattern
Προϋποθέσεις	Preconditions
Ρυθμοαπόδοση	Throughput
Σημασιολογία	Semantics
Σημασιολογικός Ιστός	Semantic Web
Σημασιολογικός Σχολιασμός	Semantic Annotation
Συλλογική Ευφυΐα	Collective Intelligence
Συμφωνία σε Επίπεδο Υπηρεσιών	Service Level Agreement
Συνδεόμενο Συστατικό	Plug-in Component
Σύνδεσμος Υπηρεσίας	Service Binding
Συνομοσπονδία Μητρώων	Registry Federation
Συντελεστής Συνημιτόνου	Cosine Coefficient
Σχεσιακή Βάση Δεδομένων	Relational Database
Σχεσιακό Μοντέλο Δεδομένων	Relational Data Model
Τύπος Πόρου	Resource Type
Υπηρεσία Ιστού	Web Service
Υπηρεσία Ομότιμου-προς-Ομότιμο	Peer-to-Peer Service
Υπηρεσία Πλέγματος	Grid Service
Υπηρεσιοστρέφεια	Service-Orientation
Υπηρεσιοστρεφής Αρχιτεκτονική	Service-Oriented Architecture
Υπολογιστής Ομοιότητας	Similarity Calculator
Υπολογιστική Πλέγματος	Grid Computing
Φορητότητα	Portability
Χρηστικότητα	Usability
Χώρος Ονομάτων	Namespace

## ΑΚΡΩΝΥΜΙΑ – ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ

<b>Ακρωνύμιο</b>	<b>Ανάπτυξη-Επεξήγηση</b>
API	Application Programming Interface
B2B	Business-to-Business
BCNF	Boyce-Codd Normalization Form
COM	Component Object Model
CORBA	Common Object Request Broker Architecture
EAI	Enterprise Application Integration
EBNF	Extended Backus Naur Form
ebXML	Electronic Business using eXtensible Markup Language
EJB	Enterprise Java Beans
ERD	Entity-Relationship Diagram
GeSMO	Generic Service Model
IEEE	Institute of Electrical and Electronics Engineers
JDBC	Java DataBase Connectivity
JSON	JavaScript Object Notation
MSA	Module Spec Advertisement
NAICS	North American Industry Classification System
OGSA	Open Grid Services Architecture
OGSI	Open Grid Services Infrastructure
OSGi	Open Services Gateway initiative
P2P	Peer-to-Peer
PDA	Personal Data Assistant
PDP	Peer Discovery Protocol
QoS	Quality of Service
RDF	Resource Description Framework
REST	REpresentational State Transfer
SAWSDL	Semantic Annotations for WSDL and XML Schema
SeCSE	Service-Centric Systems Engineering
SLA	Service Level Agreement
SOA	Service-Oriented Architecture
SOAP	Simple Object Access Protocol
SOC	Service-Oriented Computing
SODIUM	Service-Oriented Development In a Unified fraMework
SQL	Structured Query Language
SUMO	Suggested Upper Merged Ontology

<b>Ακρωνύμιο</b>	<b>Ανάπτυξη-Επεξήγηση</b>
TF-IDF	Term Frequency – Inverse Term Frequency
UDDI	Universal Description, Discovery, and Integration
UNSPSC	United Nations Standard Products and Services Code
URI	Uniform Resource Identifier
USQL	Unified Service Query Language
VO	Virtual Organization
W3C	World Wide Web Consortium
WS-RF	Web Services Resource Framework
WSDL	Web Services Description Language
WSLA	Web Service Level Agreements
WSML	Web Service Modeling Language
WSMO	Web Service Modeling Ontology
XMI	XML Metadata Interchange
XML	eXtensible Markup Language

## ΑΝΑΦΟΡΕΣ

- [1] R. Akkiraju, J. Farrell, J. Miller, M. Nagarajan, M.-T. Schmidt, A. Sheth, and K. Verma. Web Service Semantics - WSDL-S. <http://www.w3.org/Submission/WSDL-S/>, November 2005.
- [2] R. Al-Ali, O. Rana, D. Walker, S. Jha, and S. Sohail. G-QoS: Grid service discovery using QoS properties. *Computing and Informatics Journal, Special Issue on Grid Computing*, 21(4):363–382, 2002.
- [3] G. Alonso, F. Casati, H. Kuno, and V. Machiraju. *Web Services: Concepts, Architectures and Applications*. Springer-Verlag, 2003.
- [4] M. Amoretti, F. Zanichelli, and G. Conte. SP2A: A Service-Oriented Framework for P2P-based Grids. In *Proceedings of the 3rd International Workshop on Middleware for Grid Computing, MGC'05*, pages 1–6, 2005.
- [5] S. Andreatto, M. Sgaravatto, and C. Vistoli. Sharing a conceptual model of grid resources and services. In *Proceedings of the 2003 Conference on Computing in High Energy and Nuclear Physics*, 2003.
- [6] A. Ankolekar, M. Burstein, J. R. Hobbs, O. Lassila, D. L. Martin, D. McDermott, S. A. McIlraith, S. Narayanan, M. Paolucci, T. R. Payne, and K. Sycara. DAML-S: Web service description for the Semantic Web. In *Proceedings of the First International Semantic Web Conference (ISWC)*, pages 348–363, 2002.
- [7] ANSI. The Database Language SQL. American National Standards Institute, Document ANSI X3.135, 1986.
- [8] A. Arasu, J. Cho, H. Garcia-Molina, A. Paepcke, and S. Raghavan. Searching the Web. *ACM Transactions on Internet Technology*, 1(1):2–43, August 2001.
- [9] G. Athanasopoulos, A. Tsalgatidou, and M. Pantazoglou. Interoperability among heterogeneous services. In *Proceedings of the 2006 IEEE International Conference on Services Computing, SCC 2006*, pages 174–181. IEEE Computer Society, 2006.
- [10] R. A. Baeza-Yates and C. Castillo. Balancing volume, quality, and freshness in web crawling. In *HIS 2002*, volume 87, pages 565–572. IOS Press, 2002.
- [11] R. A. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley, Reading, MA, 1999.
- [12] S. Banerjee, S. Basu, S. Garg, S. Garg, S.-L. Lee, P. Mullan, and P. Sharma. Scalable grid service discovery based on UDDI. In *Proceedings of the Third International*

- Workshop on Middleware for Grid Computing, MGC 2005*, pages 1–6. ACM Press, 2005.
- [13] L. Baresi, M. Miraz, and P. Plebani. A flexible and semantic-aware publication infrastructure for web services. In *Proceedings of the 20th International Conference on Advanced Information Systems Engineering (CAiSE 2008)*, volume 5074 of *Lecture Notes in Computer Science*, pages 435–449. Springer, 2008.
- [14] U. Bellur and R. Kulkarni. Improved matchmaking algorithm for semantic web services based on bipartite graph matching. In *IEEE International Conference on Web Services, ICWS'07*, pages 86–93. IEEE Computer Society, 2007.
- [15] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, 284(5):34–43, 2001.
- [16] D. Booth, H. Haas, F. McCabe, E. Newcomer, M. Champion, C. Ferris, and D. Orchard. Web services architecture. <http://www.w3.org/TR/ws-arch/>.
- [17] A. Bramantoro, S. Krishnaswamy, and M. Indrawan. A semantic distance measure for matching web services. In *Proceedings of the Web Information Systems Engineering – WISE 2005 Workshops: WISE 2005 International Workshops*, pages 217–226, 2005.
- [18] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *Proceedings of the Seventh World Wide Web Conference, WWW7*, pages 107–117, 1998.
- [19] T. Broens, S. Pokraev, M. van Sinderen, J. Koolwaaij, and P. D. Costa. Context-aware, Ontology-based Service Discovery. In *Proceedings of the Second European Symposium on Ambient Intelligence, EUSAI 2004*, volume 3295 of *Lecture Notes in Computer Science*, pages 72–83. Springer, 2004.
- [20] W. Cazzola, A. Ghoneim, and G. Saake. Reflective analysis and design for adapting object run-time behavior. In *Proceedings of the 8th International Conference on Object-Oriented Information Systems, OOIS 2002*, volume 2425 of *Lecture Notes in Computer Science*, pages 242–254. Springer, 2002.
- [21] D. Chamberlin and R. Boyce. EQUOL: A Structured English Query Language. In *ACM SIGMOD Workshop*, volume 1, pages 249–264, 1974.
- [22] D. Chamberlin, J. Robie, and D. Florescu. Quilt: An XML query language for heterogeneous data sources. In *WebDB (Selected Papers)*, volume 1997 of *Lecture Notes in Computer Science*. Springer, 2000.

- [23] P. Chen. The Entity-Relationship Model: Toward a Unified View of Data. *ACM Transactions on Database Systems*, 1(1):9–36, March 1976.
- [24] X. Chen, W. Cai, S. J. Turner, and Y. Wang. SOAr-DSGrid: Service-Oriented Architecture for Distributed Simulation on the Grid. In *Proceedings of the 20th ACM/IEEE/SCS Workshop on Principles of Advanced and Distributed Simulation, PADS'06*, pages 65–73. IEEE Computer Society, May 2006.
- [25] N. Choi, I. Y. Song, and H. Han. A survey on ontology mapping. *SIGMOD Record*, 35(3):34–41, September 2006.
- [26] E. Christensen, F. Curbera, G. Meredith, and S. Weerawarana. Web Service Description Language (WSDL) 1.1. <http://www.w3.org/TR/wsdl>, March 2001. W3C Note, World Wide Web Consortium (W3C).
- [27] I. Clarke, O. Sandberg, B. Wiley, and T. W. Hong. Freenet: A distributed anonymous information storage and retrieval system. In *Proceedings of the International Workshop on Design Issues in Anonymity and Unobservability*, volume 2009 of *Lecture Notes in Computer Science*, pages 46–66. Springer-Verlag, 2001.
- [28] L. Clement, A. Hatley, C. von Riegen, and T. Rogers. UDDI version 3.0.2. [http://uddi.org/pubs/uddi\\_v3.htm](http://uddi.org/pubs/uddi_v3.htm), October 2004. UDDI Spec Technical Committee.
- [29] E. F. Codd. A relational model for large shared data banks. *Communications of the ACM*, 13(6):377–387, June 1970.
- [30] E. F. Codd. Recent investigations in relational database systems. In *Proceedings of IFIP Congress 74*, pages 1017–1021, 1974.
- [31] COM, Microsoft Component Object Model technologies. <http://www.microsoft.com/com/>. (Accessed July 2009).
- [32] CORBA. <http://www.corba.org/>. (Accessed July 2009).
- [33] M. Crasso, A. Zunino, and M. Campo. Easy web service discovery: A query-by-example approach. *Science of Computer Programming*, 71(2):144–164, 2008.
- [34] Y. Ding and S. Foo. Ontology Research and Development. Part 2 - A Review of Ontology Mapping and Evolving. *Journal of Information Science*, 28(5):375–388, 2002.
- [35] G. Dobson, R. Lock, and I. Sommerville. Qosont: A qos ontology for service-centric systems. In *31st EUROMICRO Conference on Software Engineering and Advanced Applications*, pages 80–87. IEEE Computer Society, 2005.

- [36] A. Dogac, Y. Kabak, and G. B. Laleci. Enriching ebXML registries with OWL ontologies for efficient service discovery. In *Proceedings of the 14th International Workshop on Research Issues on Data Engineering: Web Services for e-Commerce and e-Government Applications*, pages 69–76, 2004.
- [37] X. Dong, A. Halevy, J. Madhavan, E. Nemes, and J. Zhang. Similarity search for web services. In *Proceedings of the Thirtieth international Conference on Very Large Data Bases*, volume 30, pages 372–383, 2004.
- [38] ebXML, Electronic Business using eXtensible Markup Language. <http://www.ebxml.org>. (Accessed July 2009).
- [39] Eclipse IDE, <http://www.eclipse.org/>.
- [40] M. Ehrig and Y. Sure. Ontology Mapping - An Integrated Approach. In *Proceedings of the First European Semantic Web Symposium*, volume 3053 of *Lecture Notes in Computer Science*, pages 76–91. Springer, 2004.
- [41] EJB, Enterprise JavaBeans technology. <http://java.sun.com/products/ejb/>. (Accessed July 2009).
- [42] D. Elenius and M. Ingmarsson. Ontology-based service discovery in p2p networks. In *Proceedings of the First International Workshop on Peer-to-Peer Knowledge Management, P2PKM'04*, 2004.
- [43] J. Farrell and H. Lausen. Semantic annotations for wsdli and xml schema. <http://www.w3.org/TR/sawsdli/>, August 2007.
- [44] C. Fellbaum, editor. *Language, Speech, and Communication*, MIT Press, Cambridge, chapter WordNet: An Electronic Lexical Database. MIT Press, 1998.
- [45] I. Foster. Globus toolkit version 4: Software for service-oriented systems. In *IFIP International Conference on Network and Parallel Computing*, number 3779 in *Lecture Notes in Computer Science*, pages 2–13. Springer-Verlag, 2005.
- [46] I. Foster, C. Kesselman, J. Nick, and S. Tuecke. The physiology of the Grid: An Open Grid Services Architecture for distributed systems integration. <http://www.globus.org/alliance/publications/papers/ogsa.pdf>, 2002. Open Grid Service Infrastructure WG, Global Grid Forum.
- [47] I. Foster, C. Kesselman, and S. Tuecke. The anatomy of the grid, enabling scalable, virtual organizations. *International Journal of Supercomputer Applications*, 15(3):200–222, 2001.



- [48] D. P. Friedman and M. Wand. Reification: Reflection without metaphysics. In *Proceedings of the 1984 ACM Symposium on LISP and Functional Programming*, pages 348–355. ACM, 1984.
- [49] D. Gale and L. S. Shapley. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1):9–15, 1962.
- [50] J. Gerke, P. Reichl, and B. Stiller. Strategies for service composition in P2P networks. In *Proceedings of the 2nd International Conference on E-Business and Telecommunication Networks, ICETE 2005*, pages 16–24, 2005.
- [51] Gnutella. <http://rfc-gnutella.sourceforge.net/>.
- [52] A. Gomez-Perez and R. Gonzalez-Cabero. SGSDesigner: A graphical interface for annotating and designing semantic grid services. In *Proceedings of the 15th International Conference on World Wide Web, WWW'06*, pages 867–868. ACM, 2006.
- [53] L. Gong. JXTA: A network programming environment. *IEEE Internet Computing*, 5(3):88–95, 2001.
- [54] T. Haselwanter H. Lausen. Finding web services. In *Proceedings of the First European Semantic Technology Conference, ESTC 2007*, 2007.
- [55] W. Hoschek. The Web Service Discovery Architecture. In *Proceedings of the 2002 ACM/IEEE Conference on Supercomputing, Maryland*, pages 1–15, 2002.
- [56] Java, <http://java.sun.com/>.
- [57] Java DataBase Connectivity, <http://java.sun.com/javase/technologies/database/>.
- [58] B. Jin, L. Zhang, and Z. Zang. A unified service discovery framework. In *Proceedings of the Sixth International Conference on Grid and Cooperative Computing, GCC 2007*, pages 203–209. IEEE Computer Society, 2007.
- [59] JSON, JavaScript Object Notation, <http://json.org>.
- [60] L. Juszczak, A. Michlmayr, C. Platzer, F. Rosenberg, A. Urbanec, and S. Dustdar. Large scale web service discovery and composition using high performance in-memory indexing. In *Proceedings of the 9th IEEE International Conference on E-Commerce Technology (CEC 2007) / 4th IEEE International Conference on Enterprise Computing, E-Commerce and E-Services (EEE 2007)*, pages 509–512. IEEE Computer Society, 2007.

- [61] Y. Kalfoglou and M. Schorlemmer. Ontology Mapping: The State of the Art. *The Knowledge Engineering Review*, 18(1):1–31, 2003.
- [62] S. Kang, D. Kim, Y. Lee, S. J. Hyun, D. Lee, and B. Lee. A semantic service discovery network for large-scale ubiquitous computing environments. *ETRI Journal*, 29(5):545–558, October 2007.
- [63] A. Keller and H. Ludwig. The wsla framework: Specifying and monitoring service level agreements for web services. *Journal of Network and System Management*, 11(1):57–81, 2003.
- [64] W. Kiessling and G. Koestler. Preference SQL - Design, Implementation, Experiences. In *VLDB*, pages 990–1001. Morgan Kaufmann, 2002.
- [65] M. Kifer, R. Lara, A. Polleres, C. Zhao, U. Keller, H. Lausen, and D. Fensel. A logical framework for web service discovery. In *Online Proceedings of the Semantic Web Services: Preparing to Meet the World of Business Applications Workshop, SWS 2004*, 2004.
- [66] M. Kifer, G. Lausen, and J. Wu. Logical foundations of object-oriented and frame-based languages. *Journal of the ACM*, 42(4):741–843, 1995.
- [67] M. Klein and A. Bernstein. Towards high-precision service retrieval. *IEEE Internet Computing*, 8(1):30–36, January-February 2004.
- [68] N. Kokash. A comparison of web service interface similarity measures. In *Proceedings of the European Starting AI Researcher Symposium (STAIRS)*, pages 220–231, 2006.
- [69] D. Kourtesis and I. Paraskakis. Combining SAWSDL, OWL-DL and UDDI for Semantically Enhanced Web Service Discovery. In *Proceedings of the 5th Annual European Semantic Web Conference, ESWC 2008*, volume 5021 of *Lecture Notes in Computer Science*, pages 614–628. Springer, 2008.
- [70] V. A. Koutsonikola and A. Vakali. LDAP: Framework, practices, and trends. *IEEE Internet Computing*, 8(5):66–72, 2004.
- [71] A. Kozlenkov, G. Spanoudakis, A. Zisman, V. Fasoulas, and F. Sanchez. Architecture-driven service discovery for service-centric systems. *International Journal of Web Services Research*, 4(2):82–113, 2007.
- [72] L. Kuang, S. Deng, Y. Li, W. Shi, and Z. Wu. Exploring semantic technologies in service matchmaking. In *Third IEEE European Conference on Web Services (ECOWS'05)*, pages 226–234, 2005.

- [73] Harold W. Kuhn. The Hungarian Method for the assignment problem. *Naval Research Logistics Quarterly*, pages 83–97, 1955.
- [74] E. Laure, S.M. Fisher, A. Frohner, C. Grandi, P. Kunszt, A. Krenek, O. Mulmo, F. Pacini, F. Prelz, J. White, M. Barroso, P. Buncic, F. Hemmer, A. Di Meglio, and A. Edlund. Programming the Grid with gLite. *Computational Methods in Science and Technology*, 12(1):33–45, 2006.
- [75] Y. Li, F. Zou, Z. Wu, and F. Ma. PWSD: A scalable web service discovery architecture based on peer-to-peer overlay network. In *Proceedings of the Sixth Asia Pacific Web Conference, APWeb 2004*, volume 3007 of *Lecture Notes in Computer Science*, pages 291–300. Springer, 2004.
- [76] K. Sycara M. Klusch, B. Fries. Automated semantic web service discovery with OWLS-MX. In *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS 2006*, pages 915–922. ACM Press, 2006.
- [77] D. Martin, M. Burstein, J. Hobbs, O. Lassila, D. McDermott, S. McIlraith, S. Narayanan, M. Paolucci, B. Parsia, T. Payne, E. Sirin, N. Srinivasan, and K. Sycara. OWL-S: Semantic markup for web services. <http://www.w3.org/Submission/OWL-S/>. W3C Member Submission, 22 November 2004.
- [78] S. Miles, J. Papay, V. Dialani, M. Luck, K. Decker, T. R. Payne, and L. Moreau. Personalized grid service discovery. *IEE Proceedings Software*, 150(4):252–256, 2003.
- [79] L. Miller, A. Seaborne, and A. Reggiori. Three implementations of SquishQL, a simple RDF query language. In *Proceedings of the First International Semantic Web Conference, ISWC 2002*, volume 2342 of *Lecture Notes in Computer Science*, pages 423–435. Springer, 2002.
- [80] N. Mitra. SOAP version 1.2 part 0: Primer (second edition). <http://www.w3.org/TR/soap12-part0/>, April 2007. W3C Recommendation.
- [81] MySQL, <http://www.mysql.com/>.
- [82] NAICS, North American Industry Classification System. <http://www.census.gov/eos/www/naics/>.
- [83] W. Nejdl, B. Wolf, Q. Changtao, M. Sintek, A. Naeve, M. Nilsson, M. Palmer, and T. Risch. EDUTELLA: A p2p networking infrastructure based on RDF. In *WWW02*, 2002.

- [84] Object Management Group. *OMG XML Metadata Interchange (XMI) Specification Version 1.1*, October 1999.
- [85] OMG. UML, Unified Modeling Language. <http://www.uml.org>.
- [86] S. A. Oundhakar, K. Verma, K. Sivashanmugam, A. P. Sheth, and J. A. Miller. Discovery of web services in a multi-ontology and federated registry environment. *International Journal of Web Services Research*, 2(3):1–32, 2005.
- [87] M. Pantazoglou and A. Tsalgatidou. A P2P Platform for Social Intelligent Web Service Publication and Discovery. In *Proceedings of the Third International Multi-Conference on Computing in the Global Information Technology, ICCGI 2008*, pages 271–276. IEEE Computer Society, 2008.
- [88] M. Pantazoglou, A. Tsalgatidou, and G. Athanasopoulos. Discovering Web Services and JXTA Peer-to-Peer Services in a Unified Manner. In *Proceedings of the 4th International Conference on Service-Oriented Computing, ICSOC '06*, volume 4294 of *Lecture Notes in Computer Science*, pages 104–115. Springer, 2006.
- [89] M. Pantazoglou, A. Tsalgatidou, G. Athanasopoulos, and T. Pilioura. A unified approach towards the discovery of web and peer-to-peer services. In *Proceedings of the IEEE International Conference on Web Services, ICWS 2006*, pages 901–902. IEEE Computer Society, 2006.
- [90] M. Pantazoglou, A. Tsalgatidou, and G. Spanoudakis. Behavior-aware, univied service discovery. In *Proceedings of the Service-Oriented Computing: A look at the inside (SOC@Inside) Workshop, co-located with ICSOC 2007*, 2007.
- [91] J. Pathak, S. Basu, R. Lutz, and V. Honavar. Parallel Web Service Composition in MoSCoE: A Choreography-Based Approach. In *Proceedings of the 4th IEEE European Conference on Web Services, ECOWS'06*, pages 3–12. IEEE Computer Society, 2006.
- [92] A. Patil, S. Oundhakar, and K. Verma A. Sheth. Meteor-s web service annotation framework. In *Proceedings of the 13th International Conference on World Wide Web, WWW'04*, pages 553–562, 2004.
- [93] D. Petcu. Between web and grid-based mathematical services. In *Proceedings of the 2006 International Multi-Conference on Computing in the Global Information Technology - Challenges for the Next Generation of IT & C, ICCGI 2006*, page 41. IEEE Computer Society, 2006.

- [94] T. Pilioura and A. Tsalgatidou. Unified publication and discovery of semantic web services. *ACM Transactions on the Web*, 3(3), June 2009.
- [95] D. Preuveneers and Y. Berbers. Pervasive Services on the Move: Smart Service Diffusion on the OSGi Framework. In *Proceedings of the 5th International Conference on Ubiquitous Intelligence and Computing, UIC08*, volume 5061 of *Lecture Notes in Computer Science*, pages 46–60. Springer, 2008.
- [96] S. Ran. A model for web services discovery with QoS. *ACM SIGecom Exchanges*, 4(1):1–10, 2003.
- [97] J. Rao and X. Su. A survey of automated web service composition methods. In *Proceedings of the First International Workshop on Semantic Web Services and Web Process Composition, SWSWPC 2004*, volume 3387 of *Lecture Notes in Computer Science*, pages 43–54. Springer, 2004.
- [98] RDF. Resource description framework. W3C, <http://www.w3.org/RDF/>.
- [99] C. M. F. A. Ribeiro, N.S. Rosa, and P. R. F. Cunha. An ontological approach for personalized services. In *20th International Conference on Advanced Information Networking and Applications, 2006. AINA 2006*, 2006.
- [100] T. Roelleke and J. Wang. TF-IDF uncovered: A study of theories and probabilities. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 435–442, 2008.
- [101] D. Roman, U. Keller, H. Lausen, R. L. Jos De Bruijn, M. Stollberg, A. Polleres, C. Feier, C. Bussler, and D. Fensel. Web Service Modeling Ontology. *Applied Ontology*, 1(1):77–106, 2005.
- [102] O. Sahin, C. Gerede, D. Agrawal, and A. El Abbadi. SPiDeR: P2P-Based Web Service Discovery. In *Proceedings of the Third International Conference on Service-Oriented Computing, ICSOC 2005*, volume 3826 of *Lecture Notes in Computer Science*, pages 157–169. Springer, 2005.
- [103] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 8(11):613–620, 1975.
- [104] SeCSE, Service Centric Systems Engineering. <http://www.secse-project.eu/>.
- [105] Seekda. <http://seekda.com>.
- [106] V. Shkapenyuk and T. Suel. Design and implementation of a high-performance distributed web crawler. In *In Proceedings of the the 18th International Conference on Data Engineering, ICDE 2002*, pages 357–368. IEEE Computer Society, 2002.

- [107] G. Spanoudakis, K. Mahbub, and A. Zisman. A platform for context aware runtime web service discovery. In *Proceedings of the 2007 IEEE International Conference on Web Services, ICWS 2007*, pages 233–240. IEEE Computer Society, 2007.
- [108] SPARQL. Query Language for RDF. W3C Recommendation, <http://www.w3.org/TR/rdf-sparql-query/>, January 2008.
- [109] N. Srinivasan, M. Paolucci, and K. Sycara. An efficient algorithm for OWL-S based semantic search in UDDI. In *SWSWPC 2005*, volume 3387 of *Lecture Notes in Computer Science*, pages 96–110. Springer, 2005.
- [110] N. Srinivasan, M. Paolucci, and K. Sycara. Semantic web service discovery in the OWL-S IDE. In *Proceedings of the 39th Annual Hawaii International Conference on System Sciences, HICSS-39*, page 109b. IEEE Computer Society, 2006.
- [111] M. Stal. Web services: Beyond component-based computing. *Communications of the ACM*, 45(10):71–76, 2002.
- [112] E. Stroulia and Y. Wang. Structural and semantic matching for assessing web service similarity. *International Journal of Cooperative Information Systems*, 14(4):407–438, 2005.
- [113] K. Sycara, M. Klusch, S. Widoff, and J. Lu. Dynamic service matchmaking among agents in open information environments. *ACM SIGMOD Record, Special Issue on Semantic Interoperability in Global Information Systems*, 28(1):47–53, 1999.
- [114] L. Taher, R. Basha, and H. El Khatib. Establishing association between QoS properties in service oriented architecture. In *International Conference on Next Generation Web Services Practices, NWeSP 2005*, 2005.
- [115] D. Talia, P. Trunfio, and O. Verta. WSRF Services for Composing Distributed Data Mining Applications on Grids: Functionality and Performance. In *Proceedings of the 2006 International Conference on Computational Science and its Applications, ICCSA 2006*, volume 3980 of *Lecture Notes in Computer Science*, pages 1080–1089. Springer-Verlag, 2006.
- [116] M. Tian, A. Gramm, H. Ritter, and J. Schiller. Efficient selection and monitoring of qos-aware web services with the ws-qos framework. In *Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence, WI '04*, pages 152–158, Washington DC, USA, 2004. IEEE Computer Society.
- [117] V. Tasic, K. Patel, and B. Pagurek. Wsol - web service offerings language. In *Proceedings of the CAiSE 2002 International Workshop on Web Services, E-Business*,

- and the Semantic Web, WES 2002*, volume 2512 of *Lecture Notes in Computer Science*, pages 57–67. Springer-Verlag, 2002.
- [118] A. Tsalgaidou, G. Athanasopoulos, and M. Pantazoglou. Interoperability among heterogeneous services: The case of integration of p2p services with web services. *International Journal of Web Services Research (IJWSR)*, 5(4):79–110, October-December 2008.
- [119] A. Tsalgaidou, G. Athanasopoulos, M. Pantazoglou, A. J. Berre, C. Pautasso, R. Gronmo, and H. Hoff. *Unified discovery and composition of heterogeneous services: The SODIUM approach*, chapter 4, pages 67–100. Information Systems. MIT Press, 2008.
- [120] A. Tsalgaidou, G. Athanasopoulos, M. Pantazoglou, C. Pautasso, T. Heinis, R. Gronmo, H. Hoff, A.-J. Berre, M. Glittum, and S. Topouzidou. Developing scientific workflows from heterogeneous services. *SIGMOD Record*, 35(2):22–28, June 2006.
- [121] V. Tsetsos, C. Anagnostopoulos, and S. Hadjiefthymiades. On the evaluation of semantic web service matchmaking systems. In *ECOWS*, pages 255–264. IEEE Computer Society, 2006.
- [122] S. Tuecke, K. Czajkowski, I. Foster, J. Frey, S. Graham, C. Kesselman, T. Maguire, T. Sandholm, P. Vanderbilt, and D. Snelling. Open Grid Services Infrastructure (OGSI) version 1.0. [http://www.globus.org/alliance/publications/papers/Final\\_OGSI\\_Specification\\_V1.0.pdf](http://www.globus.org/alliance/publications/papers/Final_OGSI_Specification_V1.0.pdf), June 2003.
- [123] UNSPSC, United Nations Standard Products and Services Code. <http://www.unspsc.org>.
- [124] C.J. van Rijsbergen. *Information Retrieval*. Butterworth-Heinemann, 1979.
- [125] S. Vinoski. RESTful Web Services Development Checklist. *IEEE Internet Computing*, 12(6):96–95, Nov.-Dec. 2008.
- [126] W. Vogels. Web services are not distributed objects. *IEEE Internet Computing*, 7(6):59–66, Nov.-Dec. 2003.
- [127] W3C. XML Schema, <http://www.w3.org/XML/Schema>.
- [128] WSRF, Web Services Resource Framework. <http://www.oasis-open.org/committees/wsrf/>. (Accessed July 2009).

- [129] L. Wu, X. Meng, and S. Liu. A Service-Oriented, Scalable Approach to Grid-Enabling of Manufacturing Resources. In *Proceedings of the 4th International Conference on Cooperative Design, Visualization, and Engineering, CDVE 2007*, volume 4674 of *Lecture Notes in Computer Science*, pages 175–183. Springer-Verlag, 2007.
- [130] L. Wu, X. Meng, and S. Liu. Service-oriented encapsulation of manufacturing resources. In *Proceedings of the 2007 IEEE International Conference on Services Computing*, pages 727–728. IEEE Computer Society, 2007.
- [131] XML Path Language version 1.0. <http://www.w3.org/TR/xpath>.
- [132] XQuery 1.0: An XML Query Language. <http://www.w3.org/TR/xquery/>, January 2007.
- [133] XSLT, XSL Transformations Version 1.0. <http://www.w3.org/TR/xslt>.
- [134] C. Zhou, L.-T. Chia, and B.-S. Lee. DAML-QoS ontology for web services. In *Proceedings of the IEEE International Conference on Web Services, ICWS 2004*, page 472. IEEE Computer Society, 2004.