



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

**ΤΜΗΜΑ ΔΙΔΑΚΤΙΚΗΣ ΤΗΣ ΤΕΧΝΟΛΟΓΙΑΣ ΚΑΙ ΨΗΦΙΑΚΩΝ
ΣΥΣΤΗΜΑΤΩΝ**

ΚΑΤΕΥΘΥΝΣΗ ΗΛΕΚΤΡΟΝΙΚΗΣ ΜΑΘΗΣΗΣ

Συστήματα ηλεκτρονικής μάθησης με χρήση υπηρεσιών ιστού και ροών εργασίας



**Διπλωματική εργασία της
Δεβράνη Αικατερίνης, ΜΕ 0511**

Επιβλέπων: κος Βασιλακόπουλος Γεώργιος

Πειραιάς, 2009

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΡΔΑΛΙΑ

Περίληψη

Με τον όρο υπηρεσίες ιστού αναφερόμαστε σε υπηρεσίες του διαδικτύου, οι οποίες βασίζονται στα πρότυπα του *Web*. Η ανάπτυξή τους αποτέλεσε καινοτομία στον τομέα της τεχνολογίας, καθώς ένα από τα πλεονεκτήματα χρήσης τους είναι ότι μπορούν να χρησιμοποιηθούν από όλους τους χρήστες, λόγω της διαλειτουργικότητας που τα χαρακτηρίζει. Ακόμα, κάποιος χρήστης μπορεί να χρησιμοποιήσει μια υπηρεσία και απλά, διαβάζοντας ένα κείμενο *wSDL* για να δει τη δομή των μηνυμάτων που απαιτεί η υπηρεσία και βάσει της πληροφορίας αυτής να στείλει τα αντίστοιχα κατάλληλα μηνύματα με *SOAP*.

Στην παρούσα εργασία, αναφερόμαστε στους μηχανισμούς με τους οποίους μπορούμε να ανακαλύψουμε υπηρεσίες ιστού με τις απαιτούμενες ερωτήσεις σε μια *UDDI registry*. Επιπρόσθετα, αναφερόμαστε στη δημιουργία νέων υπηρεσιών ως συνδυασμός υπαρχόντων, ορίζοντας *business processes* (ροές εργασιών) μέσω τυποποιημένης γλώσσας περιγραφής, η οποία είναι βασισμένη στην *XML*, την *BPEL*, και πώς όλα τα παραπάνω φέρνουν την καινοτομία στην ανάπτυξη Συστημάτων Ηλεκτρονικής Μάθησης. Εν κατακλείδι, γίνεται αναφορά στη χρήση υπηρεσιών ιστού και υπηρεσιοστρεφών αρχιτεκτονικών σε συστήματα ηλεκτρονικής μάθησης, καθώς και αναφορά και παρουσίαση τέτοιων συστημάτων.

Λέξεις κλειδιά

Service Oriented Architecture (SOA), Web Services, Web Service Description Language (WSDL), Simple Object Access Protocol (SOAP), Universal Description & Discovery Integration (UDDI), Business Process, Business Process Execution Language (BPEL), E – Learning Systems.

Πίνακας περιεχομένων

Περίληψη.....	- 3 -
Λέξεις κλειδιά.....	- 3 -
Πίνακας περιεχομένων.....	- 4 -
Ευρετήριο πινάκων.....	- 7 -
Ευρετήριο εικόνων.....	- 8 -
Εισαγωγή.....	- 9 -
Κεφάλαιο 1. Service-oriented architecture (soa).....	- 12 -
Κεφάλαιο 2. Υπηρεσίες Ιστού.....	- 18 -
2.1. Τί είναι οι υπηρεσίες ιστού.....	- 19 -
2.2. Service oriented architecture και υπηρεσίες ιστού.....	- 20 -
2.3. Παραδείγματα υπηρεσιών ιστού.....	- 21 -
2.3.1. E-centric v.3.....	- 21 -
2.3.2. Παροχέας υπηρεσιών ταξιδιωτικών κρατήσεων.....	- 23 -
2.4. Γιατί να χρησιμοποιήσουμε υπηρεσίες ιστού.....	- 24 -
2.4.1. Ευκολότερος χειρισμός δεδομένων.....	- 24 -
2.4.2. Απλό πρωτόκολλο επικοινωνίας.....	- 25 -
2.4.3. Απλότητα υποδομής.....	- 25 -
2.4.4. Παροχή ευκολίας στην επικοινωνία.....	- 25 -
2.4.5. Διαλειτουργικότητα και ευκολία ανάπτυξης νέων εφαρμογών.....	- 26 -
2.5. Η τεχνολογία των υπηρεσιών ιστού.....	- 26 -
2.5.1. Transport protocol.....	- 32 -
2.5.2. Platform-independent message description format.....	- 33 -
2.5.3. Γλώσσα περιγραφής.....	- 34 -
2.5.4. Registry.....	- 35 -
2.5.5. Composition.....	- 36 -
2.6. Εφαρμογές των υπηρεσιών ιστού.....	- 36 -
2.7. Αναπτυσσόμενες τεχνολογίες.....	- 37 -
2.7.1. Αξιοπιστία (reliability).....	- 37 -
2.7.2. Ασφάλεια (security).....	- 38 -
2.7.3. Συναλλαγές (transaction).....	- 39 -
2.8. Προγραμματιστικό μοντέλο.....	- 41 -
Κεφάλαιο 3. Soap (Simple Object Access Protocol) – πρωτόκολλο μεταφοράς.....	- 43 -

3.1.	Soap και xml	- 44 -
3.2.	Επικοινωνία μέσω xml μηνυμάτων	- 46 -
3.3.	Η ανάγκη για μια πρότυπη κωδικοποίηση	- 46 -
Κεφάλαιο 4.	WsdI (Web Service Description Language)	- 48 -
4.1.	Η wsdl προδιαγραφή	- 49 -
4.2.	Λειτουργίες wsdl	- 52 -
Κεφάλαιο 5.	UDDI (Universal Description & Discovery Integration).....	- 54 -
5.1.	Οργάνωση του UDDI	- 55 -
5.2.	Ανακάλυψη των υπηρεσιών ιστού στο UDDI.....	- 57 -
5.2.1.	Ανακάλυψη μέσω web	- 57 -
5.2.2.	Ανακάλυψη με το UDDI inquiry api.....	- 57 -
Κεφάλαιο 6.	Σημασιολογικός ιστός (semantic web)	- 59 -
6.1.	Παγκόσμιος ιστός και Σημασιολογικός ιστός	- 63 -
6.2.	Σημασιολογικές υπηρεσίες διαδικτύου.....	- 65 -
Κεφάλαιο 7.	Οντολογίες	- 66 -
7.1.	Τί είναι οι οντολογίες	- 66 -
Κεφάλαιο 8.	Ροή Εργασίας (Business Process)	- 71 -
Κεφάλαιο 9.	BPEL (Business Process Execution Language).....	- 73 -
9.1.	Υλοποίηση της ροής εργασίας.....	- 73 -
9.1.1.	Activities.....	- 74 -
9.1.2.	Partners (συνέταιροι) : Αλληλεπιδρώντας με άλλους.....	- 75 -
9.1.3.	Service link types.....	- 76 -
9.1.4.	Πώς αναφέρεται ο partner στην ροή εργασίας.....	- 76 -
9.1.5.	Διάρκεια ζωής των ροών εργασίας.....	- 76 -
9.1.6.	Εκκίνηση της ροής εργασίας.....	- 77 -
9.2.	Προβλήματα ροών εργασίας.....	- 77 -
9.2.1.	Αντιμετωπίζοντας τα προβλήματα και τα λάθη κατά την εκτέλεση	- 78 -
9.3.	Παράδειγμα ροής εργασίας.....	- 79 -
9.3.1.	Δομή της ροής εργασίας	- 80 -
9.3.2.	Περιγραφή των υπηρεσιών ιστού που θα συνεργαστούν για τη ροή εργασίας.....	- 80 -
9.3.3.	Δημιουργία της ροής εργασίας.....	- 83 -
9.3.4.	Αλληλεπιδρώντας με τη ροή εργασίας	- 91 -
9.3.5.	Συνολική θεώρηση της ροής εργασίας για την έγκριση του δανείου.....	- 93 -

Κεφάλαιο 10. Συστήματα Ηλεκτρονικής Μάθησης και Υπηρεσίες Ιστού	- 94 -
10.1. Πρότυπα ηλεκτρονικής μάθησης.....	- 95 -
10.1.1. <i>Ariadne metadata</i>	- 96 -
10.1.2. <i>IEEE LOM</i>	- 97 -
10.1.3. <i>AICC</i>	- 98 -
10.1.4. <i>Scorm Content Structure Format (csf)</i>	- 98 -
10.1.5. <i>IMS CP Information Model</i>	- 99 -
10.2. Αντικείμενα μάθησης και πρότυπα ανάπτυξης του μαθησιακού υλικού -	99
10.3. Παράδειγμα ηλεκτρονικής μάθησης στην ιατρική εκπαίδευση και χρήση υπηρεσιών ιστού	- 101 -
Κεφάλαιο 11. Η χρήση υπηρεσιών ιστού και υπηρεσιοστρεφών αρχιτεκτονικών σε συστήματα ηλεκτρονικής μάθησης.....	- 104 -
11.1. Ηλεκτρονική μάθηση και οντολογίες.....	- 106 -
11.2. Αρχιτεκτονική συστήματος ηλεκτρονικής μάθησης	- 107 -
11.2.1. <i>Παροχή πληροφοριών</i>	- 111 -
11.2.2. <i>Προσπέλαση των εγγράφων εκμάθησης</i>	- 111 -
11.3. Παρόμοιες εργασίες	- 112 -
11.4. Ένα e-learning framework.....	- 112 -
11.4.1. <i>Προτεινόμενα επίπεδα</i>	- 113 -
11.5. Εργαλεία υποστήριξης του εννοιολογικού ιστού	- 114 -
11.5.1. <i>Edutella</i>	- 114 -
11.5.2. <i>Conceptual modeling and knowledge manifolds</i>	- 115 -
11.5.3. <i>Conceptual browsing with konzilla</i>	- 115 -
11.5.4. <i>Digital portofolios</i>	- 115 -
11.5.5. <i>Application independence: semantic vwe</i>	- 116 -
11.6. Personalized adaptive learning interfaces (pali)	- 116 -
11.7. KNOSOS.....	- 118 -
Κεφάλαιο 12. Συμπεράσματα.....	- 122 -
Βιβλιογραφία – Αναφορές	124

Ευρετήριο πινάκων

Πίνακας 1: Βασικές Τεχνολογίες των Υπηρεσιών Ιστού.....	- 29 -
Πίνακας 2. Δομή WSDL κειμένου.....	- 51 -
Πίνακας 3. Προσφερόμενες υπηρεσίες του UDDI	- 57 -
Πίνακας 4. Βασικές λειτουργίες του UDDI Inquiry API	- 58 -
Πίνακας 5. Loan Definitions WSDL	- 81 -
Πίνακας 6. Loan Approver WSDL	- 82 -
Πίνακας 7. Loan Approval WSDL.....	- 83 -
Πίνακας 8. Namespaces ροής εργασίας	- 84 -
Πίνακας 9. Καθορισμός των partners.....	- 85 -
Πίνακας 10. Καθορισμός των variables.....	- 86 -
Πίνακας 11. Περιγραφή loan assessor.....	- 87 -
Πίνακας 12. Προσθήκη partnerLinkType με όνομα assessor.....	- 88 -
Πίνακας 13. Προσθήκη συνεταίρου.....	- 88 -
Πίνακας 14. Προσθήκη μεταβλητής	- 88 -
Πίνακας 15. Προσθήκη flows και link	- 89 -
Πίνακας 16. Ενέργεια assign.....	- 90 -
Πίνακας 17. Invoke και reply	- 90 -
Πίνακας 18. Ενέργεια sequence.....	- 91 -
Πίνακας 19. Ενέργεια invoke.....	- 92 -
Πίνακας 20. Μήνυμα απάντησης reply.....	- 92 -
Πίνακας 21. Συσχετισμός σημασιολογικού ιστού και ηλεκτρονικής μάθησης.....	- 105 -
Πίνακας 22. Οντολογία στο σενάριο ηλεκτρονικής μάθησης.....	- 110 -

Ευρετήριο εικόνων

Εικόνα 1. e.Centric.....	- 22 -
Εικόνα 2. Το μοντέλο και η χρήση των υπηρεσιών ιστού	- 30 -
Εικόνα 3: Η λειτουργία μιας απλής υπηρεσίας ιστού	- 31 -
Εικόνα 4: Οι σχέσεις μεταξύ των βασικών τεχνολογιών των υπηρεσιών ιστού.	- 32 -
Εικόνα 5: Transport protocol	- 33 -
Εικόνα 6. WSDL Specification.....	- 51 -
Εικόνα 7. Τα τέσσερα πρότυπα λειτουργιών που υποστηρίζονται από την wsdl..	- 53 -
Εικόνα 8. Αρχιτεκτονική μιας πύλης ηλεκτρονικής μάθησης.....	- 108 -
Εικόνα 9. Προτεινόμενα επίπεδα για το learning framework.....	- 113 -
Εικόνα 10. Αρχιτεκτονική PALI (1)	- 118 -
Εικόνα 11. Αρχιτεκτονική PALI (2)	- 118 -

Εισαγωγή

Σκοπός της διπλωματικής αυτής είναι η ανακάλυψη υπηρεσιών ιστού (web services) μέσα από ερωτήσεις στο UDDI registry, να τις συνδυάσουμε με ροές εργασίας (business processes) βασιζόμενοι στις WSDL περιγραφές τους, καθώς και πώς τα παραπάνω φέρνουν την καινοτομία στην ανάπτυξη Συστημάτων Ηλεκτρονικής Μάθησης.

Στόχος είναι η παρουσίαση των δυνατοτήτων των τεχνολογιών Υπηρεσιών Ιστού και τα πλεονεκτήματα που παρέχουν σε σχέση με τις υπάρχουσες τεχνολογίες, τεκμηριώνοντας την καταλληλότητά τους για την χρήση τους, με σκοπό την ανάπτυξη πιο ευέλικτων Συστημάτων Ηλεκτρονικής Μάθησης.

Στα επόμενα κεφάλαια, θα αναλύσουμε τις έννοιες που πρέπει να γνωρίζουμε, ώστε να γίνουν κατανοητές οι εξηγήσεις που δίνονται για την υλοποίηση των παραπάνω. Οι έννοιες που αναλύονται, αποβλέπουν στην εξοικείωση με την Service Oriented Architecture (SOA), τις υπηρεσίες ιστού, καθώς και τη γλώσσα περιγραφής των business processes, την BPEL.

Έτσι, στο πρώτο κεφάλαιο παρουσιάζουμε την αρχιτεκτονική που βασίζεται στις υπηρεσίες ιστού και αποτελεί τη βασική τους λειτουργία.

Στη συνέχεια, αναφερόμαστε στις υπηρεσίες ιστού. Αναλύουμε τις υπηρεσίες ιστού δίνοντας την ερμηνεία τους, την τεχνολογία που βρίσκεται πίσω από αυτά και εξετάζουμε το πώς διάφορες εφαρμογές υλοποιούνται με την βοήθεια αυτών.

Στο τρίτο κεφάλαιο, επεξηγούμε το πρωτόκολλο επικοινωνίας των web υπηρεσιών ιστού και παρουσιάζουμε το πώς γίνεται η ανταλλαγή μηνυμάτων.

Στο τέταρτο κεφάλαιο, παρουσιάζουμε τη γλώσσα περιγραφής των υπηρεσιών ιστού, αναλύοντας τα δομικά της στοιχεία.

Στο πέμπτο κεφάλαιο, αναφερόμαστε στο UDDI, τον κατάλογο καταχωρήσεων των υπηρεσιών ιστού, εξετάζοντας τη δομή του. Επιπρόσθετα, γίνεται και μια απλή αναφορά στους τρόπους με τους οποίους μπορούμε να δημοσιεύσουμε και να αναζητήσουμε κάποιες υπηρεσίες ιστού.

Στο έκτο κεφάλαιο, αναφερόμαστε στον σημασιολογικό ιστό και τη σχέση του με τις υπηρεσίες ιστού.

Στο έβδομο κεφάλαιο, παρουσιάζουμε την έννοια της οντολογίας.

Στο όγδοο κεφάλαιο, ορίζουμε τί είναι μια ροή εργασίας (business process) και προσδιορίζουμε το πώς περιγράφεται αυτή μέσω της γλώσσας περιγραφής της, την BPEL.

Στο ένατο κεφάλαιο, έχοντας εξηγήσει στα προηγούμενα κεφάλαια όλους τους όρους και έννοιες που χρειάζονται για να γίνει κατανοητή η υλοποίηση μιας ροής εργασίας, περιγράφουμε αναλυτικά την σχεδίαση και, τέλος, την υλοποίησή της.

Στη συνέχεια, στο δέκατο κεφάλαιο, αναφέρεται η έννοια του όρου «Ηλεκτρονική Μάθηση», τα μειονεκτήματα που παρουσιάζουν τα περισσότερα Συστήματα Ηλεκτρονικής Μάθησης και πώς οι Υπηρεσίες Ιστού μπορούν να συνεισφέρουν θετικά κατά την ανάπτυξη τέτοιων συστημάτων, παρέχοντας λύσεις στους τομείς που υστερούν.

Στο ενδέκατο κεφάλαιο, γίνεται αναφορά στη χρήση υπηρεσιών ιστού και υπηρεσιοστρεφών αρχιτεκτονικών σε συστήματα ηλεκτρονικής μάθησης, όπου παρατίθεται μια αρχιτεκτονική για ένα σενάριο ηλεκτρονικής μάθησης βασισμένο σε οντολογίες. Επιπρόσθετα, παρουσιάζονται παρόμοιες εργασίες – συστήματα που έχουν υλοποιηθεί και μπορούν να συγκριθούν με αυτό το σενάριο.

Εν κατακλείδι, στο δωδέκατο κεφάλαιο, αναφέρονται τα συμπεράσματα που προκύπτουν από την παρούσα εργασία.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΡΑΙΑ

Κεφάλαιο 1. Service-oriented architecture (soa)

Σήμερα, που οι σύγχρονες επιχειρήσεις καλούνται να ανταποκρίνονται άμεσα στις δυναμικά μεταβαλλόμενες επιχειρηματικές συνθήκες, γίνεται επιτακτική η αξιοποίηση μιας προηγμένης τεχνολογικής υποδομής που επεκτείνει και ενοποιεί τις υπάρχουσες εφαρμογές.

Οι υπηρεσίες ιστού βασίζονται στο σκεπτικό της service-oriented αρχιτεκτονικής (SOA). Η SOA είναι η τελευταία εξέλιξη των διανεμημένων συστημάτων υπολογισμού που δίνει τη δυνατότητα σε κομμάτια λογισμικού να παρουσιάζονται ως υπηρεσίες.

Η αρχιτεκτονική Service Oriented Architecture (SOA) επιτρέπει τη δυναμική δημιουργία σύνθετων εφαρμογών και ροών εργασιών σε πραγματικό χρόνο με μοναδικά οφέλη. Συγκεκριμένα, προσφέρει οφέλη στο κόστος ανάπτυξης και συντήρησης εφαρμογών, απλοποιημένη ενοποίηση σε ετερογενή περιβάλλοντα, μεγαλύτερη αξιοποίηση των υπάρχοντων πόρων πληροφορικής και αποτελεσματική διαχείριση και ασφάλεια. Η αρχιτεκτονική SOA αντιλαμβάνεται την επιχείρηση σαν ένα σύνολο από συνδεδεμένες υπηρεσίες και, με προσέγγιση βασισμένη σε ανοικτά πρότυπα, μετατρέπει τις ροές εργασιών σε πιο αποτελεσματικές, αποδοτικές και συνεργατικές.

Είναι δυνατή η μοντελοποίηση μιας συγκεκριμένης ροής εργασίας, η αποτύπωση του κόστους και του χρόνου καθώς και η προσομοίωση του πραγματικού φόρτου που ενέχει η διαδικασία αυτή με στόχο τη βελτιστοποίησή της. Με αυτό τον τρόπο, η επιχείρηση μπορεί να αποτιμήσει τα οικονομικά οφέλη που θα αποφέρει μια τέτοια αλλαγή.

Το γεγονός ότι η αγορά κινείται προς την αρχιτεκτονική SOA υποδηλώνει πιθανότατα, ότι οι εταιρίες υιοθετούν την συγκεκριμένη αρχιτεκτονική για να αποκτήσουν ένα στρατηγικό πλεονέκτημα. Η αρχιτεκτονική SOA επιτρέπει σε

μια εταιρία να διαχειρίζεται τις ροές εργασιών της, χρησιμοποιώντας επαναχρησιμοποιήσιμους, τυποποιημένους πόρους ή «υπηρεσίες» που βασίζονται σε ένα συνδυασμό κώδικα λογισμικού, πνευματικής ιδιοκτησίας και βέλτιστων πρακτικών που έχουν αποκτηθεί από την εκτέλεση παρόμοιων έργων, σε πολλούς επιχειρηματικούς κλάδους.

Στον πυρήνα της μια αρχιτεκτονική SOA είναι ένα πλαίσιο πληροφορικής που αναλύει τις επιχειρηματικές εφαρμογές σε μεμονωμένες επιχειρηματικές λειτουργίες και διαδικασίες, που ονομάζονται υπηρεσίες. Σε μια τέτοια αρχιτεκτονική, μια υπηρεσία θα μπορούσε να είναι η επεξεργασία μιας συναλλαγής με πιστωτική κάρτα, μιας αίτησης για δάνειο ή μιας αγοραπωλησίας μετοχών. Μέχρι πρόσφατα, οι υπηρεσίες αυτές περιορίζονταν στις διαδικασίες και λειτουργίες που προσέφερε μία μονολιθική εφαρμογή.

Η αρχιτεκτονική Service Oriented Architecture είναι ένα αρχιτεκτονικό πλαίσιο, που περιγράφει τα πρότυπα που πρέπει να ακολουθούνται, έτσι ώστε οι υπηρεσίες που υλοποιεί η τεχνολογία – και οι οποίες αντιστοιχούν σε ροές εργασιών – να είναι ευέλικτες, αξιόπιστες και συνεργάσιμες μεταξύ τους. Οι κατασκευή αυτών των services μπορεί να γίνει με πολλά από τα διαθέσιμα εργαλεία παραγωγής κώδικα και κάθε μια από αυτές έχει έναν καθορισμένο τρόπο επικοινωνίας με το περιβάλλον της. Αυτός περιγράφεται μέσα σε αρχεία που περιέχουν δηλώσεις της Web Services Description Language (WSDL), ένα ακόμα πρότυπο μιας περιγραφικής γλώσσας, βασισμένης στην eXtended Markup Language (XML).

Ωστόσο, οι υπηρεσίες πρέπει να δημιουργούνται, υλοποιούνται, ενοποιούνται και επαναχρησιμοποιούνται ανεξαρτήτως των εφαρμογών και των υπολογιστικών πλατφόρμων στις οποίες λειτουργούν. Μια υπηρεσία μπορεί να αναπτύσσεται μια φορά και να υλοποιείται σε πολλά σημεία.

Η αρχιτεκτονική SOA κάνει εφικτή τη δυνατότητα αυτή με την κοινή χρήση πληροφοριών όχι μόνο μέσα σε μια επιχείρηση, αλλά και με τους πελάτες, τους προμηθευτές και τους συνεργάτες.

Κάθε SOA περιλαμβάνει τρεις ρόλους και σχέσεις που καθορίζουν τους παροχείς και χρήστες των υπηρεσιών ιστού. Αυτοί είναι: ο **χρήστης** μιας υπηρεσίας (service requestor), ο **παροχέας** μιας υπηρεσίας (service provider) και ο **κατάλογος** με τις διαθέσιμες υπηρεσίες(service registry).

- ✚ **παροχέας** είναι υπεύθυνος για τη δημιουργία περιγραφής της υπηρεσίας, για τη δημοσίευση αυτής της περιγραφής σε έναν ή περισσότερους καταλόγους υπηρεσιών και για την λήψη υπηρεσίας ιστού από έναν ή περισσότερους χρήστες. Ένας παροχέας υπηρεσιών λοιπόν μπορεί να είναι οποιαδήποτε εταιρεία που φιλοξενεί μια διαθέσιμη σε ένα δίκτυο υπηρεσιών ιστού.
- ✚ Ο **χρήστης** της υπηρεσίας είναι υπεύθυνος για τον εντοπισμό της περιγραφής μιας υπηρεσίας δημοσιευμένη σε ένα ή περισσότερους καταλόγους υπηρεσιών και είναι επίσης υπεύθυνος για τη χρησιμοποίηση αυτών των περιγραφών έτσι ώστε να συνδέσει ή να καλέσει υπηρεσίες ιστού που φιλοξενούνται από τους διάφορους παροχείς.
- ✚ Ο **κατάλογος** υπηρεσιών είναι υπεύθυνος για τη διαφήμιση των περιγραφών των υπηρεσιών ιστού που έχουν δημοσιευτεί σε αυτόν από διάφορους παροχείς και επιτρέπει στους χρήστες να αναζητούν συλλογές περιγραφών υπηρεσιών που περιέχονται στον κατάλογο. Ο ρόλος του καταλόγου είναι απλός: κάνει το κατάλληλο ταίριασμα μεταξύ του αιτούντα μιας υπηρεσίας και του παροχέα αυτής. Από τη στιγμή που γίνει το ταίριασμα, αλληλεπιδρούν πλέον απ' ευθείας μεταξύ τους ο χρήστης και ο παροχέας.

Η SOA περιλαμβάνει ακόμη τρεις λειτουργίες: **δημοσίευση** (publish), **εύρεση** (find) και **σύνδεση** (bind). Αυτές οι λειτουργίες καθορίζουν τις συμβάσεις μεταξύ των SOA ρόλων:

✚ Η **δημοσίευση** είναι μια πράξη καταχώρησης μιας υπηρεσίας ή διαφήμισης αυτής. Δρα σαν μια σύμβαση μεταξύ του καταλόγου υπηρεσιών και του παροχέα. Όταν ένας παροχέας δημοσιεύει την περιγραφή της υπηρεσίας ιστού του σε έναν κατάλογο υπηρεσιών, τότε, αυτός ο κατάλογος, διαφημίζει τις λεπτομέρειες των υπηρεσιών ιστού σε μια κοινότητα χρηστών. Οι ακριβείς λεπτομέρειες για το πώς γίνεται η δημοσίευση εξαρτάται από την συγκεκριμένη υλοποίηση του καταλόγου. Σε μερικά σενάρια άμεσης δημοσίευσης, το ίδιο το δίκτυο παίζει το ρόλο του καταλόγου υπηρεσιών, καθιστώντας τη δημοσίευση μια απλή πράξη μεταφοράς της περιγραφής της υπηρεσίας σε μια δομή ευρετηρίου ενός server web εφαρμογών. Άλλες υλοποιήσεις καταλόγων, όπως το UDDI, καθορίζουν μια πολύπλοκη υλοποίηση διαδικασίας δημοσίευσης.

✚ Η διαδικασία **εύρεσης** είναι αντίθετη της δημοσίευσης. Η διαδικασία εύρεσης είναι το συμβόλαιο μεταξύ του αιτούντα μιας υπηρεσίας και του καταλόγου υπηρεσιών. Με την διαδικασία εύρεσης ο χρήστης δηλώνει κάποια χαρακτηριστικά, όπως τον τύπο της υπηρεσίας, την ποιότητα των εγγυήσεων που προσφέρει μια υπηρεσία και άλλα. Ο κατάλογος υπηρεσιών κάνει ταίριασμα των κριτηρίων που θέτει ο αιτών και των δημοσιευμένων υπηρεσιών ιστού περιγραφών. Το αποτέλεσμα της λειτουργίας εύρεσης είναι μια λίστα υπηρεσιών ιστού περιγραφών που πληρούν τα παραπάνω κριτήρια. Βέβαια, η διαδικασία εύρεσης ποικίλλει ανάλογα με την υλοποίηση του καταλόγου υπηρεσιών. Από τη μια μεριά, υπάρχουν απλοί κατάλογοι υπηρεσιών που παρέχουν διαδικασίες εύρεσης με τίποτα παραπάνω από μια απλή, χωρίς

παραμέτρους αίτηση HTTP GET. Αυτό βέβαια σημαίνει ότι η διαδικασία της εύρεσης θα επιστρέφει πάντα όλες τις υπηρεσίες που έχουν καταχωρηθεί στον κατάλογο και εναπόκειται στον χρήστη να βρει ποιες από αυτές πληρούν τις προϋποθέσεις του. Από την άλλη πλευρά, φυσικά, υπάρχουν πιο πολύπλοκοι κατάλογοι, όπως το UDDI, που προσφέρουν εξαιρετικές δυνατότητες εύρεσης.

- ✚ Η διαδικασία **σύνδεσης** ενσαρκώνει το μοντέλο client-server μεταξύ του αιτούντα και του παροχέα υπηρεσιών. Η διαδικασία της σύνδεσης μπορεί να είναι αρκετά περίπλοκη και δυναμική, όπως η δυναμική δημιουργία ενός πληρεξούσιου του client που βασίζεται στην περιγραφή της υπηρεσίας και χρησιμοποιείται για να καλέσει την υπηρεσία. Μπορεί, από την άλλη, να είναι στατική διαδικασία, όπου κάποιος μπορεί να γράψει σε κώδικα τον τρόπο με τον οποίο η εφαρμογή του πελάτη θα καλέσει την υπηρεσία ιστού.

Ανακεφαλαιώνοντας, αυτό που προσφέρει η αρχιτεκτονική SOA είναι η δυνατότητα να αντιληφθεί κανείς την επιχείρηση ως ένα σύνολο από συνδεδεμένες υπηρεσίες και να αποκτήσει μια προσέγγιση βασισμένη σε ανοικτά πρότυπα, μετατρέποντας τις ροές εργασιών σε πιο αποδοτικές, αποτελεσματικές και συνεργατικές. Είναι δυνατή η μοντελοποίηση μιας συγκεκριμένης διαδικασίας, η αποτύπωση του κόστους και του χρόνου, καθώς και η προσομοίωση του πραγματικού φόρτου που ενέχει η διαδικασία αυτή, με στόχο την βελτιστοποίησή της.

Το κλειδί στην SOA είναι η περιγραφή της υπηρεσίας. Γιατί η περιγραφή της υπηρεσίας είναι αυτή που δημοσιεύεται από τον παροχέα στον κατάλογο υπηρεσιών. Η περιγραφή της υπηρεσίας είναι αυτή που ανακτάται από τον αιτούντα ως αποτέλεσμα μιας διαδικασίας εύρεσης. Η περιγραφή της υπηρεσίας επίσης είναι αυτή που παρέχει στον αιτούντα ό,τι πληροφορίες θέλει για να μπορέσει να συνδεθεί με ή να καλέσει την υπηρεσία ιστού. Η

περιγραφή της υπηρεσίας, τέλος, είναι αυτή που καθορίζει τι πληροφορία επιστρέφεται στον αιτούντα ως αποτέλεσμα της κλήσης της υπηρεσίας ιστού.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΡΑΙΑ

Κεφάλαιο 2. Υπηρεσίες Ιστού

Η ανάπτυξη της πληροφορικής και των τηλεπικοινωνιών (ICT), σε συνδυασμό με τον κυβερνοχώρο και το διαδίκτυο, δημιουργούν όλο και περισσότερο μεγαλύτερες προσδοκίες για ποικίλες εφαρμογές, σχεδόν σε όλους τους τομείς.

Η έννοια της υπηρεσίας (service) αποτέλεσε κεντρικό σημείο μελέτης σε πολλά επιστημονικά πεδία. Στον τομέα της οικονομίας και του marketing, σύμφωνα με την ιστοσελίδα της Wikipedia, ο όρος χρησιμοποιείται για να δηλώσει την παροχή μη υλικού αντίτιμου ενός αγαθού. Χρησιμοποιείται επίσης, για να τονίσει την ανάλογη σπουδαιότητα μιας υπηρεσίας σε μια προσφορά προϊόντων. Στον επιχειρησιακό τομέα, Baida, Z, Gordijn, J, Omelayenko, B (2004, pp. 1-10), μια υπηρεσία αντικατοπτρίζει τη διάταξη μιας γενικής επιχειρησιακής δραστηριότητας, η οποία παρέχει μια συγκεκριμένη αξία στον πελάτη.

Στον χώρο της πληροφορικής, ο όρος της υπηρεσίας χρησιμοποιείται συχνά ως ταυτόσημος με τον όρο της υπηρεσίας ιστού (web service). Έχουν δοθεί αρκετοί ορισμοί για τις υπηρεσίες ιστού. Ένας αξιόλογος ορισμός ο οποίος δίνεται από την IBM, αναφέρει ότι οι υπηρεσίες ιστού είναι μια τεχνολογία που επιτρέπει τις εφαρμογές να επικοινωνούν μεταξύ τους, ανεξαρτήτως πλατφόρμας και γλώσσας προγραμματισμού. Μια υπηρεσία ιστού αποτελεί μια διεπαφή λογισμικού (software interface), η οποία περιγράφει μια συλλογή από λειτουργίες, οι οποίες και μπορούν να προσεγγιστούν από το δίκτυο μέσω πρότυπων μηνυμάτων XML. Για την περιγραφή μιας λειτουργίας (operation) προς εκτέλεση και τα δεδομένα προς ανταλλαγή με κάποια άλλη εφαρμογή, χρησιμοποιεί πρότυπα βασισμένα στη γλώσσα XML. Ένα σύνολο

από υπηρεσίες ιστού που αλληλεπιδρούν μεταξύ τους καθορίζουν μια εφαρμογή υπηρεσίας ιστού.

Οι υπηρεσίες ιστού αποτελούν μια σχετικά νέα και διαδεδομένη τεχνολογία. Η υλοποίησή τους με απώτερο σκοπό την επικοινωνία μεταξύ συστημάτων ανεξαρτήτως λειτουργικού συστήματος, πλατφόρμας και γλώσσας προγραμματισμού, αποτελούν ένα ιδιαίτερα ενδιαφέρον πεδίο έρευνας, καθώς προσφέρουν λύσεις σε κάποια από τα μειονεκτήματα των υπαρχόντων συστημάτων ηλεκτρονικής μάθησης, τα οποία θα αναφέρουμε παρακάτω.

2.1. Τί είναι οι υπηρεσίες ιστού

Οι υπηρεσίες ιστού αποτελούν πλέον μια διαδεδομένη τεχνολογία. Η υλοποίησή τους με απώτερο σκοπό την επικοινωνία μεταξύ συστημάτων ανεξαρτήτως λειτουργικού συστήματος, πλατφόρμας και γλώσσας προγραμματισμού, αποτελούν ένα ιδιαίτερα ενδιαφέρον πεδίο έρευνας.

Σύμφωνα με τον ορισμό που δίνεται από την IBM:

“Οι υπηρεσίες ιστού αποτελούν την καινούργια γενιά δικτυακής εφαρμογής. Είναι αυτοπεριεχόμενες, αυτοπεριγραφόμενες εφαρμογές που δημοσιεύονται και τοποθετούνται στο διαδίκτυο αλλά και καλούνται μέσα από αυτό. Οι υπηρεσίες ιστού υλοποιούν λειτουργίες που μπορεί να είναι από απλές συναρτήσεις μέχρι σύνθετες ροές εργασιών.”

Με πιο απλά λόγια μια υπηρεσία ιστού είναι μια υπηρεσία διαθέσιμη στο διαδίκτυο, η οποία χρησιμοποιεί ένα καθορισμένο σύστημα μηνυμάτων βασισμένο σε XML και δεν είναι δεμένη με κανένα λειτουργικό σύστημα ή γλώσσα προγραμματισμού.

Επιπλέον είναι επιθυμητό, παρότι όχι υποχρεωτικό, μια υπηρεσία ιστού να ικανοποιεί δύο επιπλέον ιδιότητες:

- ✚ Μια υπηρεσία ιστού πρέπει να είναι αυτο-περιγραφόμενη. Όταν δημοσιεύουμε μια υπηρεσία ιστού πρέπει να δημοσιεύουμε επίσης και μια διαπροσωπεία (interface) προς αυτήν, δηλαδή μια περιγραφή αυτής, έτσι ώστε να μπορούν αυτοί που αναπτύσσουν ολοκληρωμένα συστήματα να την ενοποιήσουν μαζί με άλλες και να είναι εύκολα υλοποιήσιμη. Αν πρόκειται για SOAP service τότε η περιγραφή μπορεί να γίνει με τη βοήθεια XML γραμματικής και να καθοριστούν όλες οι δημόσιες μέθοδοι, τα ορίσματά τους και οι επιστρεφόμενες τιμές (public methods, method arguments, and return values).
- ✚ Μια υπηρεσία ιστού πρέπει να είναι ανακαλύψιμη. Όταν δημιουργούμε μια υπηρεσία ιστού τότε πρέπει να μπορούμε να γνωστοποιήσουμε αυτό το γεγονός. Πρέπει δηλαδή να μπορούμε να δημοσιεύσουμε ένα interface αυτής ώστε να είναι σε θέση οι ενδιαφερόμενοι να την βρουν και να την καλέσουν.

Συνοπτικά μια ολοκληρωμένη υπηρεσία ιστού είναι μια υπηρεσία η οποία:

- ✚ Είναι διαθέσιμη στο διαδίκτυο ή σε ιδιωτικά (intranet) δίκτυα
- ✚ Χρησιμοποιεί ένα καθορισμένο σύστημα μηνυμάτων βασισμένο σε XML
- ✚ Δεν είναι δεμένη με κανένα λειτουργικό σύστημα ή γλώσσα προγραμματισμού
- ✚ Είναι αυτό-περιγραφόμενη μέσω κοινής XML γραμματικής
- ✚ Είναι ανακαλύψιμη μέσω μιας απλής διαδικασίας εύρεσης (find)

2.2. Service oriented architecture και υπηρεσίες ιστού

Παρόλο που η τεχνολογία των υπηρεσιών ιστού και η προαναφερθείσα αρχιτεκτονική SOA συχνά συναντώνται μαζί ή σε κάποιο συνδυασμό, πρέπει

να αναφερθεί ότι οι δύο αυτοί όροι είναι διαφορετικοί και πρέπει να ξεχωρίζουν. Η αρχιτεκτονική SOA είναι ένα σενάριο, ή μια προσέγγιση, για την οικοδόμηση συστημάτων, τα οποία επικεντρώνονται σε ένα σύνολο από συστατικά που μπορούν να συνεργασθούν δυναμικά. Οι υπηρεσίες ιστού από την άλλη μεριά, είναι μια προσέγγιση για τον τρόπο που μπορεί να αναπτυχθεί μια αρχιτεκτονική SOA. Το μοντέλο των υπηρεσιών ιστού παρέχει προδιαγραφές για ένα σύνολο από γλώσσες και τεχνολογίες βασισμένες σε XML, που μπορούν να χρησιμοποιηθούν για την οικοδόμηση συστημάτων με αρχιτεκτονική SOA.

2.3. Παραδείγματα υπηρεσιών ιστού

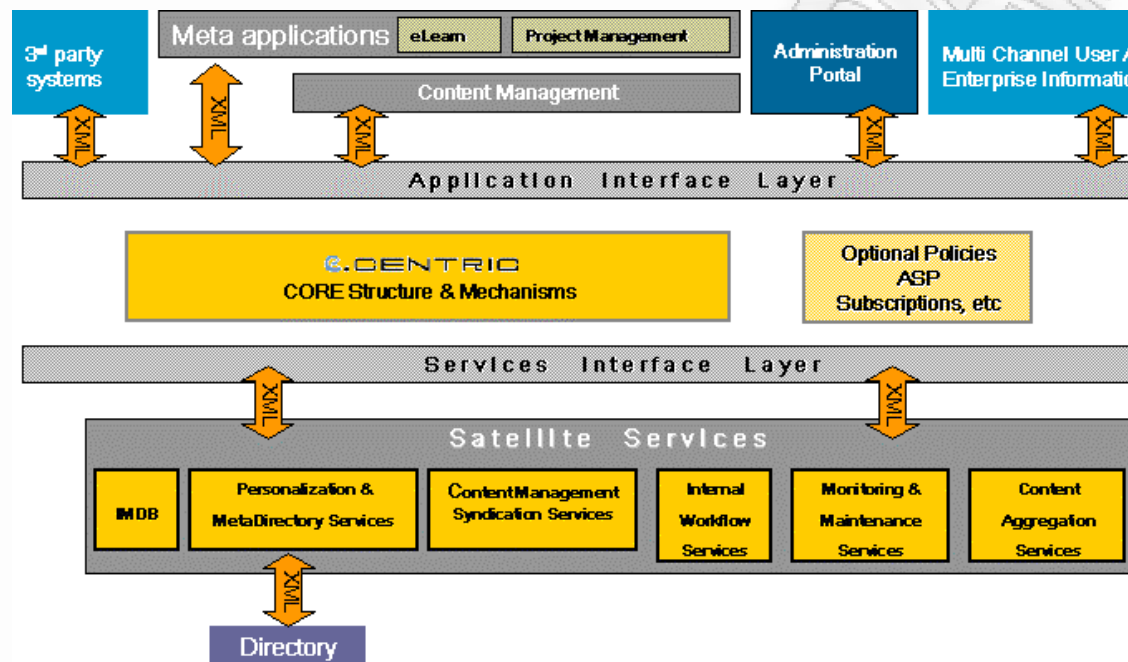
2.3.1. E-centric v.3

Το e.Centric v.3 επιτρέπει τη δημιουργία Enterprise Information Portals (Internet, Intranet, Extranet sites) και προσφέρει Content Management, Knowledge Management, διαθέσιμα modules e-learning, collaboration & messaging και Project Management. Ταυτόχρονα όμως, παρέχει τη δυνατότητα ανάπτυξης εφαρμογών εύκολα και γρήγορα με χρήση των παραπάνω ως υποδομή, ενώνοντας, ελέγχοντας και πολλαπλασιάζοντας τα οφέλη.

Η πλατφόρμα e.Centric έχει σχεδιαστεί ώστε να είναι εύκολα επεκτάσιμη και χρησιμοποιεί standards τελευταίας γενιάς, όπως XML, XSLT, XML Schema, WEBDAV, ICE, RSS και υπηρεσίες ιστού, ενώ λειτουργεί εξ' ολοκλήρου με τη χρήση του HTTP πρωτοκόλλου. Οι ASP επεκτάσεις του συστήματος δίνουν τη δυνατότητα σε ένα e.Centric Administration portal να φιλοξενεί πολλαπλά e.Centric sites. Επίσης δίνουν τα εργαλεία για την κεντρική και καθολική διαχείριση των φιλοξενούμενων (hosted) e.Centric sites.

Η αρχιτεκτονική του συστήματος επιτρέπει την κλιμάκωσή του σε πολλαπλές μηχανές (υποστηρίζοντας web farms, automatic content replication, scalable session management, κλπ.), για τη κατανομή του φόρτου εργασίας και την

αποφυγή οποιουδήποτε single point of failure. Ακόμη, έχει σχεδιαστεί ώστε να είναι εφικτή η διασύνδεσή του με την πλειοψηφία των συστημάτων βάσεων δεδομένων. Το περιεχόμενο που φυλάσσεται στον κοινό αποθηκευτικό χώρο, μπορεί να διανεμηθεί σε πολλαπλά κανάλια, όπως το Intranet, Internet ή ακόμη και WAP πύλες, σε ένα απεριόριστο αριθμό γλωσσών.



Εικόνα 1. E.centric

Επιπλέον, η πλατφόρμα e.Centric είναι ένα ανοιχτό σύστημα. Παρέχει εργαλεία παραμετροποίησης, προγραμματιστικό μοντέλο και τεκμηρίωση, στοιχεία που καθιστούν εύκολη την επέκταση του συστήματος. Το προγραμματιστικό μοντέλο του e.Centric (e.Centric API) παρέχει πρόσβαση σε όλες τις πλευρές του συστήματος: το περιεχόμενο, την διαχείριση χρηστών, τα δικαιώματα κλπ. Δίνετε σε μορφή COM API, Java API και Web Services.

Το COM API προσφέρει πρόσβαση στο e.Centric από κάθε εφαρμογή σε πλατφόρμα Windows (VB, VBScript, ASP, Windows Script Host, κλπ). Το Java API παρέχει παρόμοιες δυνατότητες προγραμματιστικής χρήσης του συστήματος σε non-Microsoft πλατφόρμες. Τέλος, το προγραμματιστικό

μοντέλο μέσω Web services παρέχει δυνατότητες πρόσβασης στο σύστημα πάνω από HTTP/HTTPS κανάλι, και υλοποιείται με την χρήση των standard προτύπων SOAP, WSDL, UDDI παρέχοντας δυνατότητες επικοινωνίας με οποιαδήποτε άλλη πλατφόρμα και σύστημα.

2.3.2. Παροχέας υπηρεσιών ταξιδιωτικών κρατήσεων

Έστω ότι ένας παροχέας υπηρεσιών ταξιδιωτικών κρατήσεων παρουσιάζει τις επιχειρηματικές εφαρμογές σαν υπηρεσίες ιστού που υποστηρίζουν πλήθος πελατών και των αιτήσεών τους. Αυτές οι επιχειρηματικές αιτήσεις παρέχονται από διαφορετικούς ταξιδιωτικούς οργανισμούς, που εδρεύουν σε διαφορετικά δίκτυα και γεωγραφικές θέσεις.

Παρακάτω αναλύεται ένα τυπικό σενάριο:

- ✚ Ο παροχέας ταξιδιωτικών υπηρεσιών αναπτύσσει τις δικές του υπηρεσίες ιστού εκθέτοντας τις επιχειρηματικές προτάσεις που εξασφάλισε από διάφορες ταξιδιωτικές επιχειρήσεις, όπως αυτές αερογραμμών, ενοικιάσεων αυτοκινήτων, διαμονής σε ξενοδοχεία, πληρωμής πιστωτικών καρτών και άλλα.
- ✚ Ο παροχέας των υπηρεσιών καταχωρεί τις υπηρεσίες που διαθέτει μαζί με τις περιγραφές αυτών χρησιμοποιώντας ιδιωτική ή δημόσια καταχώρηση. Ο κατάλογος των καταχωρήσεων αποθηκεύει τις σχετικές με τις υπηρεσίες πληροφορίες που εξέθεσε ο παροχέας των υπηρεσιών.
- ✚ Ο πελάτης ανακαλύπτει τις υπηρεσίες ιστού χρησιμοποιώντας μια μηχανή αναζήτησης ή εντοπίζοντάς αυτές απ' ευθείας από τον κατάλογο και στην συνέχεια τις καλεί για να πραγματοποιήσει ταξιδιωτικές κρατήσεις και άλλες λειτουργίες στο Internet χρησιμοποιώντας οποιαδήποτε πλατφόρμα ή άλλη συσκευή.

- ✚ Στην περίπτωση μεγάλων οργανισμών, οι επιχειρηματικές εφαρμογές χρησιμοποιούν αυτές τις υπηρεσίες ιστού για να προσφέρουν ταξιδιωτικές υπηρεσίες στους υπαλλήλους τους μέσω του εσωτερικού εταιρικού δικτύου.

2.4. Γιατί να χρησιμοποιήσουμε υπηρεσίες ιστού

Παραδοσιακά οι web εφαρμογές παρέχουν την δυνατότητα αλληλεπίδρασης μεταξύ του τελικού χρήστη και της δικτυακής σελίδας ενώ οι υπηρεσίες ιστού είναι service-oriented και παρέχουν τη δυνατότητα επικοινωνίας από εφαρμογή σε εφαρμογή μέσα στο διαδίκτυο και εύκολη πρόσβαση σε ετερογενείς εφαρμογές και συσκευές. Τα παρακάτω είναι οι βασικότεροι τεχνικοί λόγοι για να επιλέξουμε τις υπηρεσίες ιστού αντί των web εφαρμογών:

- ✚ Ευκολότερο χειρισμό δεδομένων
- ✚ Απλό πρωτόκολλο επικοινωνίας
- ✚ Απλότητα υποδομής
- ✚ Ευκολία στην επικοινωνία
- ✚ Διαλειτουργικότητα και ευκολία ανάπτυξης νέων εφαρμογών

2.4.1. Ευκολότερος χειρισμός δεδομένων

Όπως αναφέρει στην ιστοσελίδα της η IBM, ένα από τα σημαντικότερα προβλήματα στις κατανεμημένες τεχνολογίες ήταν το tight – coupling, όπου μια εφαρμογή που καλούσε μια άλλη απομακρυσμένη, ήταν αυστηρά δεμένη με αυτή από το function call που εκτελούσε και τις παραμέτρους που περνούσε. Στα περισσότερα συστήματα ήταν μια σταθερή επαφή, με ελάχιστη ή καθόλου ευελιξία στα περιβάλλοντα ή στις ανάγκες που συνεχώς μεταβάλλονταν, όπως και στα Συστήματα Ηλεκτρονικής Μάθησης. Η χρήση της γλώσσας XML από τις υπηρεσίες ιστού, μπορεί να περιγράψει

οποιαδήποτε δεδομένα, ανεξαρτήτως πλατφόρμας, για ανταλλαγή αυτών των δεδομένων μεταξύ των συστημάτων. Έτσι οδηγούμαστε σε εφαρμογές με «χαλαρή συνδεσιμότητα» (loosely coupled). Επιπρόσθετα, μπορούν να επαναξιολογήσουν, να τροποποιήσουν ή και να χειριστούν τύπους δεδομένων δυναμικά, κατά περίπτωση. Με τον τρόπο αυτό, οι υπηρεσίες ιστού, μπορούν να χειριστούν δεδομένα ευκολότερα και να επιτρέψουν στο λογισμικό να επικοινωνεί πιο ελεύθερα.

2.4.2. Απλό πρωτόκολλο επικοινωνίας

Οι παλιότερες τεχνολογίες χρησιμοποιούσαν πρωτόκολλα επικοινωνίας από τα κατανεμημένα περιβάλλοντα CORBA, RPC, DCOM. Όπως προαναφέρθηκε, οι υπηρεσίες ιστού χρησιμοποιούν ως πρωτόκολλο επικοινωνίας το SOAP, το οποίο είναι ένα νέο και ευκολότερο πρωτόκολλο από τα παραπάνω. Σήμερα, υφίστανται υλοποιήσεις του SOAP όχι μόνο από τις μεγαλύτερες εταιρίες πληροφορικής, αλλά και από μεμονωμένους προγραμματιστές, γεγονός αδιανόητο για παλιότερες τεχνολογίες. Επομένως, η δημιουργία μιας υλοποίησης SOAP που υπόκειται στα πρότυπα είναι ευκολότερη.

2.4.3. Απλότητα υποδομής

Οι υπηρεσίες ιστού λειτουργούν με πρότυπες γλώσσες και πρωτόκολλα, όπως η XML, το HTTP και το TCP/IP. Οι περισσότερες εταιρίες έχουν ήδη την κατάλληλη υποδομή και το εργατικό δυναμικό με την αντίστοιχη εμπειρία και γνώσεις των παραπάνω. Επομένως, το κόστος για την εφαρμογή των υπηρεσιών ιστού είναι σημαντικά μικρότερο αυτό των προηγούμενων τεχνολογιών.

2.4.4. Παροχή ευκολίας στην επικοινωνία

Οι παλιότερες τεχνολογίες δεν υποστήριζαν δυναμική συνεργασία μεταξύ εταιριών, καθώς οι τεχνολογίες όπως COBRA και DCOM χρησιμοποιούσαν μη πρότυπες πόρτες. Για να επιτευχθεί μια τέτοια συνεργασία, οι ενέργειες

που έπρεπε να πραγματοποιηθούν δεν ήταν συνήθως αποδεκτές από τις εταιρίες, καθώς έθεταν σε κίνδυνο την ασφάλεια των συστημάτων τους. Οι υπηρεσίες ιστού χρησιμοποιούν μεταξύ άλλων και το HTTP ως πρωτόκολλο μεταφοράς, οπότε τα περισσότερα firewalls επιτρέπουν την πρόσβαση μέσω της θύρας 80, η οποία και αποτελεί την πρότυπη θύρα για το HTTP. Με τον τρόπο αυτό, υποστηρίζονται ευκολότερες και δυναμικές συνεργασίες μεταξύ των συστημάτων εταιριών.

2.4.5. Διαλειτουργικότητα και ευκολία ανάπτυξης νέων εφαρμογών

Οι παλιότερες τεχνολογίες δεν ήταν διαλειτουργικές, καθώς κάθε vendor υλοποιούσε το δικό του πρότυπο για distributed object messaging. Με τη χρήση της XML ως μόνο πρότυπο στις υπηρεσίες ιστού, συστήματα που έχουν υλοποιηθεί από διαφορετικές τεχνολογίες όπως η Java και η .NET, μπορούν να επικοινωνήσουν μεταξύ τους. Επιπρόσθετα, λόγω του ότι η XML θεωρείται μια απλή «γλώσσα» είναι εύκολο να υλοποιηθούν νέες εφαρμογές σε σύντομο σχετικά χρονικό διάστημα.

2.5. Η τεχνολογία των υπηρεσιών ιστού

Οι τεχνολογίες υπηρεσιών ιστού, συνοπτικά, αποτελούνται κυρίως από τέσσερα βασικά συστατικά όπως αναφέρεται από τους Fensel, Lausen, Polleres, Bruijn, Stollberg, Roman, Domingue (2007) στο κεφάλαιό τους 'Web Services':

- ✚ Ένα συμφωνημένο πρωτόκολλο μεταφοράς (transport protocol)
- ✚ Πακέτα επικοινωνίας, ανεξάρτητης δομής (A platform-independent message description format)
- ✚ Μια γλώσσα για τις υπηρεσίες ιστού που να περιγράφει τη διεπαφή, τις διαδικασίες και με ποιόν τρόπο μπορεί η υπηρεσία να προσφερθεί

✚ Η παροχή ενός «μητρώου» των δημοσιευμένων και διαθέσιμων υπηρεσιών (registry)

Το πρώτο συστατικό μπορεί να πραγματοποιηθεί από οποιοδήποτε από τα κοινά πρωτόκολλα μεταφορών, όπως SMTP ή και FTP. Το πιο διαδεδομένο πρωτόκολλο που χρησιμοποιείται στα πλαίσια των υπηρεσιών ιστού είναι το HTTP, λόγω του πλεονεκτημάτος του ότι είναι και πιο διαδεδομένο στους χρήστες.

Οι υπηρεσίες ιστού παρέχουν χρήσιμη λειτουργικότητα στους χρήστες, μέσα από ένα πρότυπο δικτυακό πρωτόκολλο. Αυτό που χρησιμοποιείται περισσότερο σύμφωνα με το άρθρο του Wolter (2001) στην ιστοσελίδα της Microsoft είναι το SOAP (Simple Object Access Protocol).

Ακόμα, οι υπηρεσίες ιστού υποστηρίζουν έναν τρόπο ώστε να περιγράψουν τις διεπαφές τους με αρκετή λεπτομέρεια και να επιτρέψουν έτσι στον χρήστη να δημιουργήσει μια εφαρμογή – πελάτη, η οποία και να επικοινωνήσει μαζί τους. Η περιγραφή αυτή, συνήθως παρέχεται σε ένα έγγραφο XML, το οποίο ονομάζεται WSDL (Web Service Description Language).

Τέλος, οι υπηρεσίες ιστού καταχωρούνται με τέτοιο τρόπο, ώστε οι χρήστες να μπορούν να τις βρουν εύκολα. Αυτό γίνεται με το UDDI (Universal Discovery Description and Integration).

Οι Chitnis, Tiwari και Ananthamurthy (2006) στο άρθρο τους 'Introduction to Web Services Part 2: Architecture', παραθέτουν τον παρακάτω πίνακα με τις βασικές τεχνολογίες στις οποίες βασίζονται οι υπηρεσίες ιστού.

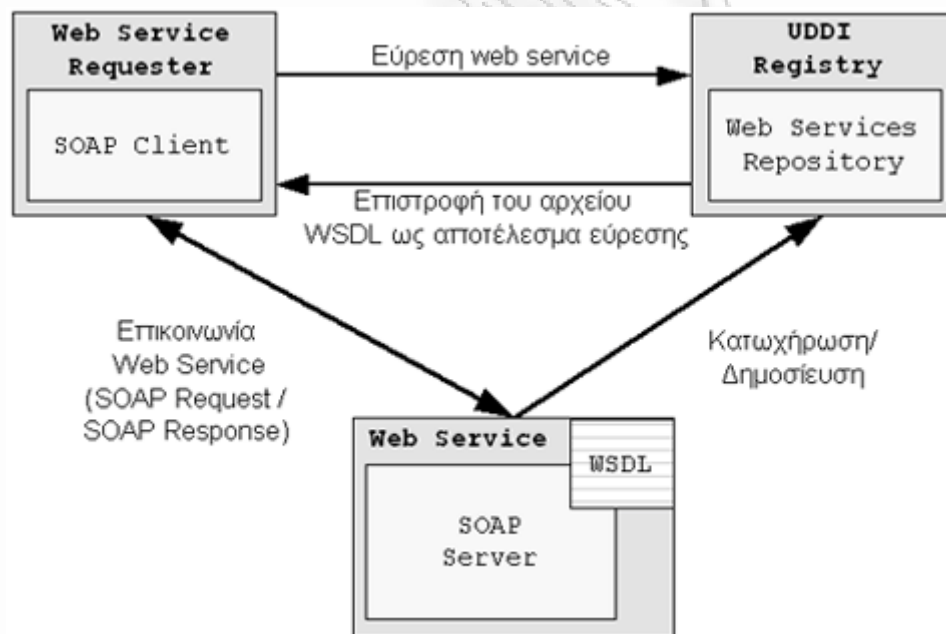
Βασικές Τεχνολογίες των Υπηρεσιών Ιστού		
Επίπεδο	Τεχνολογία	Περιγραφή
Ομοιόμορφη περιγραφή και ανταλλαγή δεδομένων	XML	Η Extended Markup Language (XML) είναι μια περιγραφική (μετα-) γλώσσα, η οποία έχει μια καλά καθορισμένη σύνταξη και σημασιολογία. Τα «περιγραφικά» χαρακτηριστικά της, την καθιστούν ως απλό αλλά δυνατό μηχανισμό για τη σύλληψη και ανταλλαγή στοιχείων μεταξύ των διαφορετικών εφαρμογών.
Πρότυπο κανάλι επικοινωνίας	SOAP	Το Simple Object Access Protocol είναι το κανάλι που χρησιμοποιείται για επικοινωνία μεταξύ μιας εφαρμογής που παρέχεται από μια υπηρεσία ιστού και μιας client-application. Η απλότητα του SOAP έγκειται στο ότι δεν καθορίζει κανένα νέο πρωτόκολλο μεταφοράς, αντίθετα, επαναχρησιμοποιεί το HyperText Transfer Protocol (HTTP) ή το Simple Mail Transfer Protocol (SMTP) για μεταφορά δεδομένων ως απλά μηνύματα. Η χρήση του HTTP και SMTP εξασφαλίζει την επικοινωνία μέσω του διαδικτύου. Η χρήση του SOAP είναι που πολλαπλασιάζει τις δυνατότητες των υπηρεσιών ιστού.
Πρότυπη περιγραφική γλώσσα για την περιγραφή των παρεχόμενων υπηρεσιών	WSDL	Οι εφαρμογές που παρέχουν υπηρεσίες ιστού διαφημίζουν τις υπηρεσίες τους χρησιμοποιώντας μια πρότυπη περιγραφική γλώσσα, την Web Services Description Languages (WSDL). Η WSDL

Βασικές Τεχνολογίες των Υπηρεσιών Ιστού		
		βασίζεται στην XML και χρησιμοποιεί ένα σύνολο ετικετών (tags) για την περιγραφή μιας υπηρεσίας ιστού, την υπηρεσία που παρέχεται, πού να εντοπιστεί κλπ. Οι client applications λαμβάνουν πληροφορίες για μια υπηρεσία ιστού πριν από την πρόσβασή τους σε αυτό και την χρήση του.
Καταχώριση και εντοπισμός των παρεχόμενων υπηρεσιών	UDDI	Το “ευρετήριο” των υπηρεσιών ιστού είναι το Universal Description Discovery and Integration (UDDI). Οι εφαρμογές που παρέχουν υπηρεσίες ιστού παρατίθενται σε έναν κατάλογο, από αυτούς που τις παρέχουν, χρησιμοποιώντας το UDDI. Όπως και στην WSDL, έτσι και το UDDI βασίζεται στην XML.

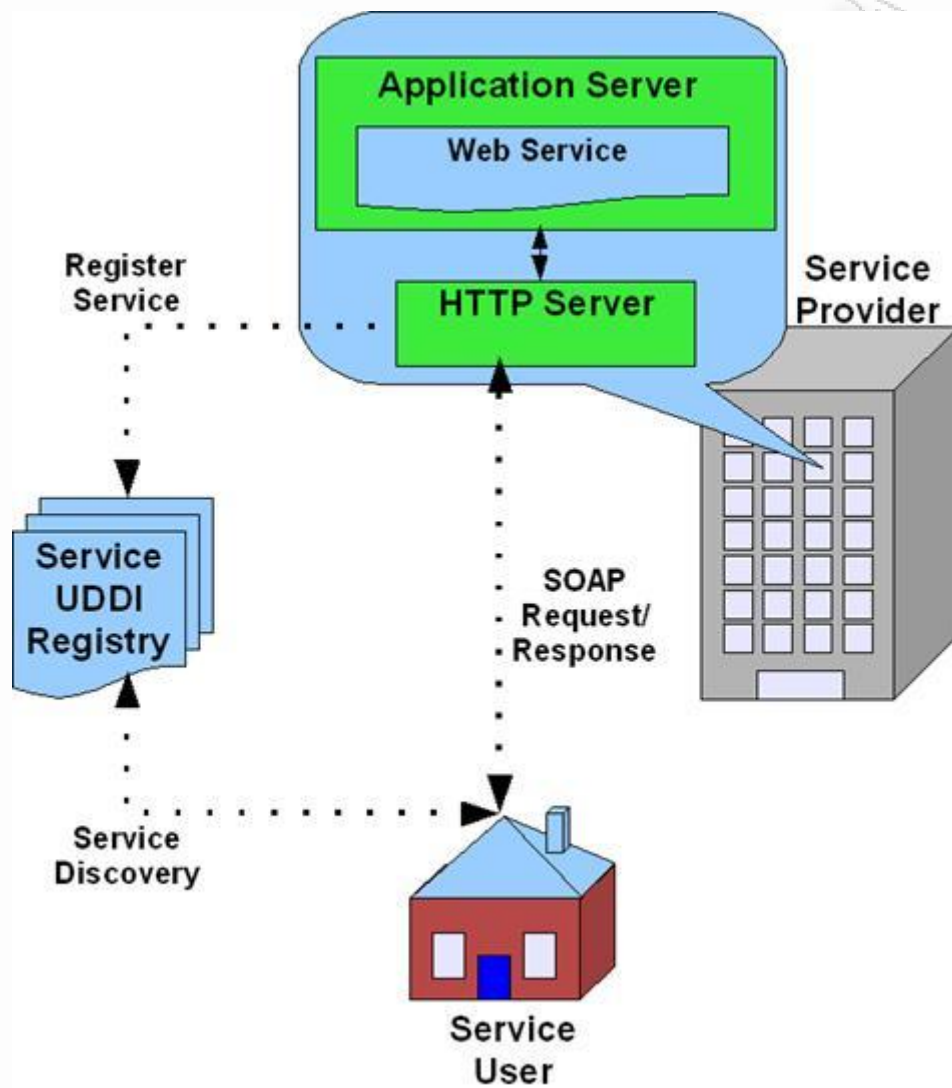
Πίνακας 1: Βασικές τεχνολογίες των υπηρεσιών ιστού

Επομένως, οι υπηρεσίες ιστού αποτελούν μια αρχιτεκτονική κατανεμημένων συστημάτων η οποία αποτελείται από αρκετά υπολογιστικά συστήματα, τα οποία προσπαθούν να επικοινωνήσουν μέσω του δικτύου, προς δημιουργία ενός συστήματος.

Αποτελούνται από ένα σύνολο προτύπων (Web Services - Web Services Tutorials), τα οποία και επιτρέπουν στους developers να υλοποιήσουν κατακευματισμένες εφαρμογές χρησιμοποιώντας διαφορετικά εργαλεία, για τη δημιουργία εφαρμογών που χρησιμοποιούν ένα συνδυασμό software modules, οι οποίες και καλούνται να τρέξουν από συστήματα διαφορετικών τμημάτων ενός οργανισμού, ή ακόμα και διαφορετικών οργανισμών.

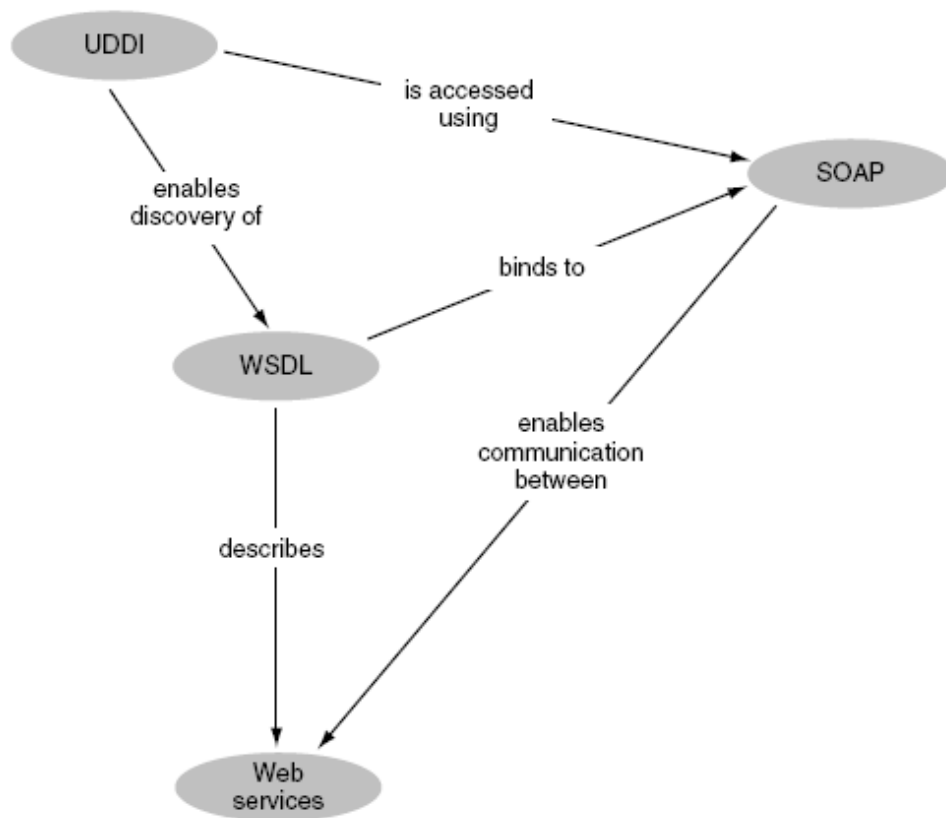


Εικόνα 2. Το μοντέλο και η χρήση των υπηρεσιών ιστού



Εικόνα 3: Η λειτουργία μιας απλής υπηρεσίας ιστού

Στο παρακάτω σχήμα περιγράφονται οι σχέσεις μεταξύ των βασικών τεχνολογιών, των υπηρεσιών ιστού:



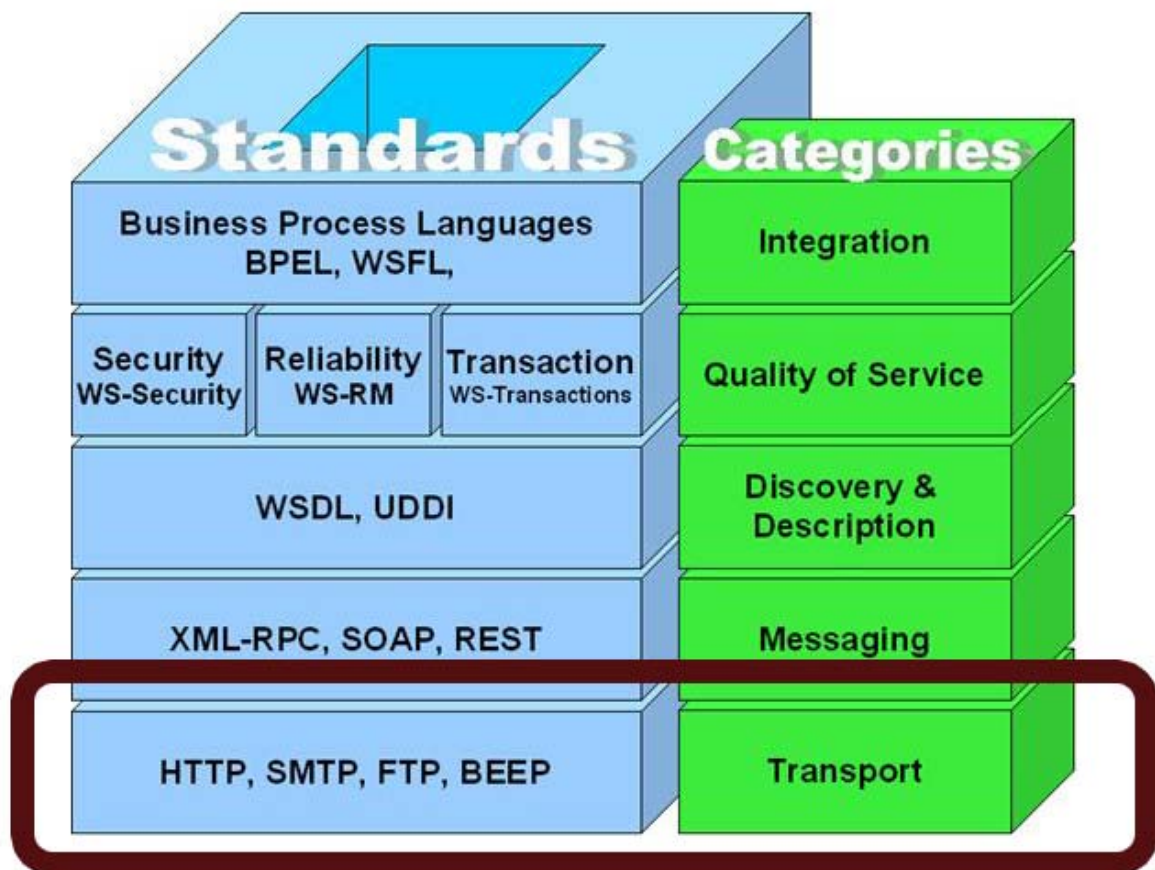
Εικόνα 4: Οι σχέσεις μεταξύ των βασικών τεχνολογιών των υπηρεσιών ιστού.

Μπορούμε να χρησιμοποιήσουμε υπηρεσίες ιστού σε πολλές εφαρμογές, οι οποίες εκτείνονται από οικιακά εργαλεία μέχρι βίντεο παιχνίδια, κι αυτό γιατί οι υπηρεσίες ιστού έχουν υλοποιηθεί για να λύσουν διαφορετικού τύπου προβλήματα. Παρόλο αυτό το εύρος εφαρμογών, οι υπηρεσίες ιστού κλέβουν την προσοχή περισσότερο των επιχειρήσεων οι οποίες τις χρησιμοποιούν για εμπορικές και ενδο-επιχειρηματικές ολοκληρωμένες εφαρμογές.

2.5.1. Transport protocol

Η κυριότερη λειτουργία του επιπέδου μεταφοράς είναι να μεταφέρει δεδομένα από ένα μηχάνημα σε ένα άλλο. Αποτελεί το πρωτόκολλο για τη μετάδοση δεδομένων των υπηρεσιών ιστού. Οι υπηρεσίες ιστού χρησιμοποιούν διάφορα πρωτόκολλα για να μεταφέρουν τα δεδομένα από υπηρεσία σε υπηρεσία, συμπεριλαμβανομένων των HTTP, SMTP, FTP ,χωρίς όμως να περιορίζονται μόνο σε αυτά.

Το πιο διαδεδομένο πρωτόκολλο μέχρι στιγμής που χρησιμοποιούν οι υπηρεσίες ιστού είναι το HTTP. Χρησιμοποιείται περισσότερο γιατί δεν μπλοκάρεται από firewalls και είναι έτσι το πρότυπο για διαλειτουργικά συστήματα.



Εικόνα 5: Transport protocol

2.5.2. Platform-independent message description format

Το επίπεδο μηνυμάτων περιγράφει σε τι μορφές θα είναι τα δεδομένα ώστε να μπορούν να μεταφερθούν από τη μια υπηρεσία στην άλλη. Η XML είναι η βασική μορφή που χρησιμοποιείται για τις υπηρεσίες ιστού. Τα δεδομένα που διαμορφώνονται με βάση την XML είναι σαν δέντρο (tree) με στοιχεία

(elements) και ολόκληρη η δενδρική μορφή καλείται έγγραφο (document). Η XML δεν έχει ξεχωριστή περιγραφή δεδομένων γιατί τα ίδια τα δεδομένα περιγράφουν τον εαυτό τους κι αυτό γιατί έχουν ειδικές ετικέτες που τους δίνουν ένα όνομα καθώς και την θέση τους μέσα στη δενδρική μορφή.

Το SOAP (Simple Object Access Protocol) είναι μια προδιαγραφή που λέει στον καταναλωτή μιας υπηρεσίας και στον προμηθευτή αυτής πώς να διαμορφώσουν και να διαβάσουν ένα μήνυμα γραμμένο σε XML που χρησιμοποιείται από μια υπηρεσία. Ένα SOAP μήνυμα έχει τρεις περιοχές: τον φάκελο, την επικεφαλίδα και το σώμα, τα οποία και αναλύουμε παρακάτω.

2.5.3. Γλώσσα περιγραφής

Το επίπεδο περιγραφής καθορίζει τρεις πλευρές μιας υπηρεσίας ιστού:

- ✚ Μεθόδους τις οποίες μια υπηρεσία ιστού κάνει διαθέσιμες
- ✚ Τα μηνύματα που δέχεται μια υπηρεσία ιστού
- ✚ Το πρωτόκολλο το, οποίο πρέπει να χρησιμοποιήσει ο πελάτης για να καλέσει τημ υπηρεσία ιστού

Οι υπηρεσίες ιστού χρησιμοποιούν μια γλώσσα περιγραφής την WSDL (Web Services Description Language) για να καθορίσουν μια σύμβαση υπηρεσίας. Η σύμβαση υπηρεσίας είναι μια περιγραφή ενός συνόλου από τελικά σημεία (endpoints) στα οποία στέλνονται ή λαμβάνονται μηνύματα μαζί με μια προδιαγραφή για το πώς πρέπει να διαμορφωθεί ένα XML μήνυμα για να σταλεί στο τελικό σημείο. Το τελικό σημείο (endpoint) είναι μια διεύθυνση στο διαδίκτυο η οποία δέχεται μηνύματα ειδικά διαμορφωμένα σύμφωνα με τη προδιαγραφή που καθορίζεται στην WSDL. Η WSDL χρησιμοποιεί τον όρο «λιμάνι» (port) για να περιγράψει το τελικό σημείο το οποίο θα δεχτεί το μήνυμα. Η WSDL περιγράφει τη σύμβαση της υπηρεσίας σαν μια συλλογή από ports τα οποία η υπηρεσία έχει κάνει διαθέσιμα.

Ο πελάτης μιας υπηρεσίας χρησιμοποιεί την περιγραφή αυτή με δύο τρόπους. Κατά την ώρα της ανάπτυξης της υπηρεσίας (development time), ο πελάτης μπορεί να δημιουργήσει ένα στέλεχος αυτής (stub) χρησιμοποιώντας την περιγραφή αυτής. Το στέλεχος (stub) είναι μια κλάση η οποία συμφωνεί με την περιγραφή της υπηρεσίας. Ο πελάτης συνδέεται με το στέλεχος κατά την ώρα της σύνταξης (compile time). Αυτό καλείται αρχικό δέσιμο (early binding). Οι υπηρεσίες διαδικτύου υποστηρίζουν επίσης και την ιδέα του κατοπινού δεσίματος (late binding), ένα δέσιμο που γίνεται μόνο κατά την ώρα που εκτελείται μια υπηρεσία μεταξύ του πελάτη και του παροχέα της υπηρεσίας. Αυτό γίνεται χρησιμοποιώντας ένα δυναμικό πληρεξούσιο (dynamic proxy) που διαμορφώνεται δυναμικά κατά την ώρα της εκτέλεσης χρησιμοποιώντας την WSDL περιγραφή της υπηρεσίας.

2.5.4. Registry

Οι υπηρεσίες διαδικτύου υποστηρίζουν την ιδέα της δυναμικής εύρεσης υπηρεσιών. Ένας πελάτης χρησιμοποιεί την αποθήκη υπηρεσιών για να ανακαλύψει υπηρεσίες που τον ενδιαφέρουν. Το UDDI (Universal Description, Discovery and Integration) είναι μια υπηρεσία ιστού από μόνη της που υποστηρίζει ένα σύνολο υπηρεσιών οι οποίες επιτρέπουν σε ένα χρήστη μιας υπηρεσίας να ανακαλύψει δυναμικά και να εντοπίσει την περιγραφή μιας υπηρεσίας διαδικτύου. Οι UDDI αποθήκες είναι από μόνες τους υπηρεσίες διαδικτύου που εκθέτουν ένα API ως ένα σύνολο καλά διαμορφωμένων SOAP μηνυμάτων. Τόσο ο παροχέας μιας υπηρεσίας όσο και ο καταναλωτής αυτής χρησιμοποιούν το SOAP και το HTTP για να δημοσιεύσουν μια υπηρεσία στην αποθήκη και να λάβουν πληροφορίες για όποιες άλλες ενδιαφέρονται. Οι δημόσιες αποθήκες υπηρεσιών παρέχουν πληροφορίες για το σημείο στο οποίο μπορείς να καλέσεις την υπηρεσία, τον ιδιοκτήτη της εταιρείας που παρέχει την υπηρεσία και κάποιες τεχνικές πληροφορίες για την υπηρεσία ιστού. Το UDDI υποστηρίζει δύο τύπους συνομιλίας:

- ✚ Ο παροχέας υπηρεσίας χρησιμοποιεί την UDDI αποθήκη για να δημοσιεύσει πληροφορίες σχετικά με τις υπηρεσίες που παρέχει.

- ✚ Ο καταναλωτής μιας υπηρεσίας στέλνει XML μηνύματα διαμορφωμένα σύμφωνα με το SOAP προς την αποθήκη υπηρεσιών για να παραλάβει μια λίστα από υπηρεσίες που ταιριάζουν στα κριτήριά του.

2.5.5. Composition

Η σύνθεση υπηρεσιών αναφέρεται στη δυνατότητα να συνδυάζουμε υπηρεσίες σε μια ροή εργασίας. Αυτό αναφέρεται επίσης ως «ενορχήστρωση υπηρεσιών» ή ως «χορογραφία υπηρεσιών» (service orchestration ή service choreography). Η σύνθεση υπηρεσιών αναφέρεται στην ικανότητα να βάζεις σε μια λογική σειρά και να κατευθύνεις συνομιλίες μεταξύ υπηρεσιών σε μια μεγαλύτερη συναλλαγή. Για παράδειγμα, μια συναλλαγή που προσθέτει έναν πελάτη σε μια υπηρεσία τραπεζικών λογαριασμών μπορεί κάλλιστα να δημιουργήσει περαιτέρω λογαριασμούς με το να παράσχει τις πληροφορίες του πελάτη στην υπηρεσία. Όλες αυτές οι αιτήσεις κατευθύνονται από ένα ευρύτερο επιχειρηματικό διαδικαστικό διάγραμμα το οποίο ή ολοκληρώνεται με επιτυχία ή αποτυγχάνει εξ' ολοκλήρου.

Υπάρχει πληθώρα προτεινόμενων γλωσσών για να περιγράψουν ένα επιχειρηματικό διαδικαστικό διάγραμμα. Οι δύο περισσότερο γνωστές είναι οι WSFL και XLANG. Πρόσφατα οι εταιρείες IBM, Microsoft και BEA συνδύασαν τις δύο παραπάνω γλώσσες σε μία προδιαγραφή που ονομάζεται BPEL4WS (Business Process Execution Language for Web Services).

2.6. Εφαρμογές των υπηρεσιών ιστού

Οι πρώτες υπηρεσίες ιστού είχαν ως σκοπό να αποτελούν πηγές πληροφορίας, τις οποίες θα μπορεί κάποιος εύκολα να ενσωματώσει στις εφαρμογές του, όπως προβλέψεις καιρού, αποτελέσματα αθλητικών παιχνιδιών, τιμές χρηματιστηρίου. Ο απώτερος στόχος ήταν η δημιουργία μιας ολόκληρης κατηγορίας εφαρμογών που να μπορεί να κατασκευαστεί, ώστε να αναλύει και να συνδυάζει την πληροφορία που ενδιαφέρει τον χρήστη και να

του την παρουσιάζει με ποικίλους τρόπους. Οι περισσότερες από αυτές τις πληροφορίες είναι ήδη διαθέσιμες στον παγκόσμιο ιστό, αλλά οι υπηρεσίες ιστού θα ενισχύσουν την προγραμματιστική πρόσβαση σε αυτές, κάνοντάς την πιο εύκολη και πιο αξιόπιστη. Χρησιμοποιώντας τις ήδη υπάρχουσες εφαρμογές ως υπηρεσίες ιστού, θα μπορούσαν να κατασκευαστούν πιο ισχυρές εφαρμογές, οι οποίες και θα χρησιμοποιούσαν τις υπηρεσίες ιστού ως δομικά στοιχεία.

Γεγονός αποτελεί η ανάπτυξη εφαρμογών υπηρεσιών ιστού στον τομέα της ηλεκτρονικής μάθησης, στους τομείς του ηλεκτρονικού εμπορίου (e-commerce), η παροχή e-government services, healthcare services και multimedia services.

2.7. Αναπτυσσόμενες τεχνολογίες

Οι βασικές τεχνολογίες των υπηρεσιών ιστού, το SOAP, WSDL και το UDDI, μαζί με την XML στην οποία βασίζονται, παρέχουν μόνο τις βασικές λειτουργίες των υπηρεσιών ιστού. Η αξιοπιστία (reliability), η ασφάλεια (security) και οι συναλλαγές (transactions), αποτελούν αναγκαία χαρακτηριστικά για μια ολοκληρωμένη κατανομημένη τεχνολογία. Η παροχή αυτών των υπηρεσιών στο περιβάλλον των υπηρεσιών ιστού βασίζονται σε κάποιες άλλες προδιαγραφές που είναι ακόμα υπό ανάπτυξη.

2.7.1. Αξιοπιστία (reliability)

Υπάρχει ένα σύνολο προδιαγραφών που πραγματεύεται ζητήματα που έχουν να κάνουν με την αξιόπιστη αποστολή μηνυμάτων (reliable messaging) και τις ειδοποιήσεις (notifications) (Newcomer και Lomow (2004):

- ✚ WS-ReliableMessaging και WS-Reliability. Καθορίζουν μηχανισμούς για την εξασφάλιση της παράδοσης των μηνυμάτων με τη σωστή σειρά και την εξάλειψη των διπλοτύπων.

- ✚ WS-Eventing και WS-Notification. Παρέχουν έναν μηχανισμό σχετικά με τα events, ώστε οι συνδρομητές να ειδοποιούνται ανεξάρτητα από τη σχέση μεταξύ αυτού που παρέχει την υπηρεσία και του αιτούντα.

2.7.2. Ασφάλεια (security)

Οι υπηρεσίες ιστού παρέχουν πολλά πλεονεκτήματα στις εφαρμογές, αλλά μπορούν να εκθέσουν σημαντικούς κινδύνους στην ασφάλεια. Η δημιουργία αλλά και η διατήρηση ενός περιβάλλοντος υπηρεσίας ιστού περιλαμβάνει και τη διαχείριση διαφόρων μηχανισμών για το διαδίκτυο, την XML, αλλά και τις ίδιες τις υπηρεσίες ιστού.

Η γενική αντιμετώπιση αφορά:

- ✚ Την ασφάλεια σε επίπεδο μεταφοράς, όπως τα τείχη προστασίας (firewalls), τα ιδεατά ιδιωτικά δίκτυα (VPNs), την επικύρωση (authentication), το non – repudiation και την κρυπτογράφηση (encryption).
- ✚ Την ασφάλεια σε επίπεδο μηνύματος, όπως τα στοιχεία επικύρωσης (authentication tokens) για να επικυρώσουν την ταυτότητα του αιτούντα, καθώς και οι δηλώσεις έγκρισης (authorization assertions) για να ελέγξουν την πρόσβαση στις υπηρεσίες του provider.
- ✚ Την ασφάλεια σε επίπεδο δεδομένων, όπως είναι η κρυπτογράφηση (encryption) και η ηλεκτρονική υπογραφή (digital signature), κυρίως για την προστασία ενάντια στις μη επιθυμητές αλλαγές των δεδομένων.
- ✚ Την ασφάλεια σε επίπεδο περιβάλλοντος, όπως είναι η διαχείριση (management), το logging και ο έλεγχος (auditing) για την αναγνώριση των προβλημάτων που πρέπει να διορθωθούν.

Για την αντιμετώπιση όσων προαναφέρθηκαν, δημιουργήθηκε ένα σύνολο προδιαγραφών, το οποίο βρίσκεται ακόμα υπό ανάπτυξη. Οι κυριότερες από αυτές τις προδιαγραφές είναι:

- ✚ WS-Security. Καθορίζει μια αρχιτεκτονική για ασφαλή επικοινωνία.
- ✚ WS-Policy. Οι σχετιζόμενες με αυτήν προδιαγραφές καθορίζουν μια πολιτική, σύνολο από κανόνες, για το πώς οι υπηρεσίες θα αλληλεπιδρούν μεταξύ τους.
- ✚ WS-Trust. Καθορίζει το μοντέλο εμπιστοσύνης για ασφαλείς συναλλαγές.
- ✚ WS-Privacy. Καθορίζει το πώς τηρείται η ιδιοτικότητα στις πληροφορίες.
- ✚ WS-Secure Conversation. Καθορίζει το πώς θα επιτευχθεί μια ασφαλή συνεδρία μεταξύ υπηρεσιών που ανταλλάσσουν δεδομένα με κανόνες που έχουν οριστεί στα WS-Policy, WS-Trust και WS-Privacy.
- ✚ WS-Federation. Καθορίζει τους κανόνες σχετικά με την ταυτότητα σε κατανεμημένο περιβάλλον.
- ✚ WS-Authorization. Χειρίζεται την επεξεργασία επικύρωσης που αφορά την πρόσβαση την ανταλλαγή δεδομένων.

2.7.3. Συναλλαγές (transaction)

Οι συναλλαγές στο περιβάλλον των υπηρεσιών ιστού, μας εξασφαλίζουν ότι ένα σύνολο αυτών πετυχαίνει ένα κοινό αποτέλεσμα. Οι υπηρεσίες ιστού συχνά εξαρτώνται η μια από την άλλη για την ολοκλήρωση πολύπλοκων αιτήσεων σε μια εφαρμογή, όπως η επεξεργασία μιας παραγγελίας, η οποία μπορεί να ενημερώνει πολλαπλές βάσεις δεδομένων αποθηκών, ή ακόμα και η ενημέρωση μιας εγγραφής πελάτη, η οποία μπορεί να ενημερώνει

πολλαπλές βάσεις δεδομένων με τα στοιχεία του πελάτη. Επομένως, αποτελούν ουσιαστικό χαρακτηριστικό τους. Οι πιο αξιόλογες και σημαντικότερες προσπάθειες που σχετίζονται με τις συναλλαγές στο περιβάλλον των υπηρεσιών ιστού είναι οι:

✚ WS-Transaction. Αποτελεί ένα σύνολο προδιαγραφών που περιλαμβάνει:

- WS-AtomicTransactions (WS-AT). Είναι ένα πρωτόκολλο που αφορά τη διαλειτουργικότητα των υπηρεσιών ιστού.
- WS-BusinessActivity (WS-BA). Είναι ένα ανοιχτό ενσωματωμένο πρωτόκολλο για μακροπρόθεσμες επιχειρησιακές διαδικασίες.
- WS-Coordination (WS-C). Αποτελεί ένα πλαίσιο συντονισμού που υποστηρίζει τα WS-AT και WS-C.

✚ WS-Composite Application Framework. Αποτελεί ένα σύνολο προδιαγραφών που περιλαμβάνει:

- WS-Context (WS-CTX). Αποτελεί ένα γενικό μηχανισμό διαχείρισης πλαισίου.
- WS-TransactionManagement (WS-TXM). Αποτελεί ένα πρωτόκολλο για τη διαλειτουργικότητα των υπηρεσιών ιστού (ACID), ένα πρωτόκολλο βασισμένο στο compensation για μακροπρόθεσμες διαδικασίες (LRA) και ένα πρωτόκολλο που αφορά διαχείριση επιχειρησιακών διαδικασιών (BP).
- WS-CoordinationFramework (WS-CF). Αποτελεί ένα πλαίσιο συντονισμού που υποστηρίζει τα WS-AT, WS-BA και τα τρία πρωτόκολλα του WS-TXM.

Το γεγονός ότι βρίσκονται ακόμα υπό ανάπτυξη, μπορεί να θεωρηθεί ως μειονέκτημα αυτών των υπηρεσιών.

2.8. Προγραμματιστικό μοντέλο

Το προγραμματιστικό μοντέλο των υπηρεσιών ιστού αποτελείται από μια συλλογή από τυποποιημένα πρωτόκολλα και διεπαφές εφαρμογών (application programming interfaces - APIs), οι οποίες χρησιμοποιούνται από ανθρώπους και εφαρμογές για τον εντοπισμό και τη χρήση των υπηρεσιών ιστού.

Βασική αρχή της δημιουργίας των υπηρεσιών ιστού είναι η τυποποίηση (standardization) απλών και ανοικτών (open) πρωτοκόλλων και διεπαφών, ώστε να είναι εφικτή η ευρεία διάδοση τους μέσω του διαδικτύου.

Τα επίπεδα του προγραμματιστικού μοντέλου των υπηρεσιών ιστού είναι:

- ✚ Το επίπεδο δικτύου (network) είναι το δομικό επίπεδο του προγραμματιστικού μοντέλου των υπηρεσιών ιστού. Όλες οι υπηρεσίες ιστού πρέπει να είναι διαθέσιμες μέσω του δικτύου. Το επίπεδο δικτύου βασίζεται συνήθως στο πρωτόκολλο HyperText Transport Protocol (HTTP protocol), αλλά μπορούν να χρησιμοποιηθούν και άλλα πρωτόκολλα όπως είναι το Internet Inter-ORB Protocol (IIOP) ή το IBM MQSeries.
- ✚ Πάνω από το επίπεδο δικτύου είναι το επίπεδο XML μηνυμάτων που παρέχει την επικοινωνία μεταξύ των υπηρεσιών ιστού και των πελατών που τα καλούν. Αυτό το επίπεδο επικοινωνίας βασίζεται στο πρωτόκολλο Simple Object Access Protocol (SOAP).

Το πρωτόκολλο SOAP είναι ένα XML πρωτόκολλο που παρέχει τις λειτουργίες δημοσίευσης (publish), εύρεσης (find) και σύνδεσης (bind) που περιγράφηκαν νωρίτερα.

- ✚ Πάνω από το επίπεδο XML μηνυμάτων είναι το επίπεδο περιγραφής υπηρεσιών. Αυτό το επίπεδο βασίζεται στην προδιαγραφή Web Services Description Language (WSDL), που περιγράφει σε μορφή XML εγγράφων διαθέσιμες υπηρεσίες ιστού για τους πελάτες.

Αυτά τα 3 επίπεδα είναι υποχρεωτικά για τη δημιουργία διαλειτουργικών (interoperable) υπηρεσιών ιστού και τη δημοσίευση τους στο Internet. Τα ακόλουθα επίπεδα είναι προαιρετικά και χρησιμοποιούνται όταν απαιτείται ανάλογα με τις επιχειρησιακές ανάγκες.

- ✚ Παρόμοια, στο επίπεδο ανακάλυψης μιας υπηρεσίας ιστού περιλαμβάνεται κάθε ενέργεια που εκτελείται ώστε ο αιτούντας υπηρεσίας να προσπελάσει το WSDL έγγραφο της υπηρεσίας.

Η ενέργεια μπορεί να είναι απλή, όπως η προσπέλαση του εγγράφου WSDL ή η επίσκεψη της διεύθυνσης που βρίσκεται το έγγραφο WSDL ή πολύπλοκη, όπως είναι η αναζήτηση σε μια αποθήκη UDDI και η χρήση των εγγράφων WSDL για την επιλογή ενός ή περισσότερων υπηρεσιών.

- ✚ Το επίπεδο ροής υπηρεσιών (service flow layer) παρέχει τη σύνθεση των υπηρεσιών ιστού σε ροές εργασίας (workflows) και την αναπαράσταση αυτής της συσσώρευσης των υπηρεσιών ως μια υπηρεσία ιστού υψηλότερου επιπέδου.

Για αυτό το επίπεδο η IBM έχει προτείνει την τυποποίηση Web Services Flow Language (WSFL) και η Microsoft την τυποποίηση XLANG. Η προδιαγραφή Business Process Execution Language for Web Services (BPEL4WS) αποτελεί μια προσπάθεια για ενοποίηση των προδιαγραφών WSFL και XLANG, συνδυάζοντας τα προτερήματα της κάθε μιας προσέγγισης, με στόχο την τυποποίηση (standard) της για τη σύνθεση υπηρεσιών web.

Κεφάλαιο 3. Soap (Simple Object Access Protocol) – πρωτόκολλο μεταφοράς

Το W3C από τον Ιούνιο του 2003 έχει συντάξει μία σύσταση (recommendation) η οποία είναι ό,τι πιο κοντινό υπάρχει στην προδιαγραφή (specification) του Simple Object Access Protocol v1.2. Μέσα στη σύσταση αυτή βρίσκεται ο παρακάτω ορισμός:

“Το SOAP στην έκδοση 1.2 είναι ένα ελαφρύ πρωτόκολλο προορισμένο για την ανταλλαγή δομημένων πληροφοριών σε ένα αποκεντρωμένο, διανεμημένο περιβάλλον. Χρησιμοποιεί τεχνολογίες XML για να καθορίσει ένα επεκτάσιμο πλαίσιο παρέχοντας μια δομή μηνυμάτων η οποία μπορεί να ανταλλαχθεί πάνω από ποικίλα δικτυακά πρωτόκολλα. Το πλαίσιο έχει σχεδιαστεί να είναι ανεξάρτητο από οποιοδήποτε προγραμματιστικό μοντέλο και σημασιολογία υλοποίησης.”

Δύο βασικοί στόχοι του σχεδιασμού του SOAP είναι η απλότητα και η επεκτασιμότητα. Το SOAP προσπαθεί να πετύχει αυτούς τους στόχους παραλείποντας, από το πλαίσιο μηνυμάτων, χαρακτηριστικά γνωρίσματα τα οποία συνήθως συναντούνται σε κατανεμημένα συστήματα. Μερικά από αυτά τα γνωρίσματα είναι η «αξιοπιστία» (reliability), η «ασφάλεια» (security), ο «συσχετισμός» (correlation), η «δρομολόγηση» (routing) και τα «σχέδια ανταλλαγής μηνυμάτων» (Message Exchange Patterns - MEPs).

Ενώ αναμένεται ότι πολλά χαρακτηριστικά θα καθοριστούν, η παρούσα προδιαγραφή παρέχει τις λεπτομέρειες μόνο για δύο MEPs : το HTTP και το SMTP. Τα υπόλοιπα χαρακτηριστικά γνωρίσματα αφήνονται να καθοριστούν από άλλες προδιαγραφές.

Συνοψίζοντας, μπορούμε να χαρακτηρίσουμε τη θέση του SOAP στο τεχνολογικό στρώμα των υπηρεσιών ιστού ως ένα τυποποιημένο πρωτόκολλο που «πακετάρει» τα μηνύματα τα οποία μοιράζονται και ανταλλάσσουν οι εφαρμογές. Η προδιαγραφή του SOAP καθορίζει τίποτα άλλο παρά έναν απλό φάκελο βασισμένο στην XML για να μεταφέρει τις διάφορες πληροφορίες μαζί με ένα σύνολο κανόνων για μετάφραση και μετατροπή των διαφόρων τύπων δεδομένων σε XML εκφράσεις. Ο σχεδιασμός του SOAP το κάνει κατάλληλο για μια μεγάλη ποικιλία εφαρμογών που χρησιμοποιούν μηνύματα για επικοινωνία και για πολλά ολοκληρωμένα μοντέλα. Κι αυτό είναι που το οδηγεί να γίνεται όλο και περισσότερο δημοφιλές.

3.1. Soap και xml

Όπως αναφέραμε, το SOAP είναι XML. Αυτό σημαίνει, ότι το SOAP είναι μια εφαρμογή της XML προδιαγραφής. Ο ορισμός και η λειτουργία της βασίζονται κυρίως στα πρότυπα της XML, όπως το XML Schema και τα XML Namespaces.

Ένα μήνυμα SOAP είναι ένα συνηθισμένο έγγραφο XML το οποίο περιέχει τα παρακάτω στοιχεία (elements):

- ✚ Ένα απαιτούμενο στοιχείο **Envelope** από το οποίο αναγνωρίζεται το έγγραφο XML ως μήνυμα SOAP.
- ✚ Ένα προαιρετικό στοιχείο **Header** που περιλαμβάνει βοηθητικές πληροφορίες.
- ✚ Ένα απαιτούμενο στοιχείο **Body** το οποίο περιλαμβάνει την κύρια πληροφορία του μηνύματος.
- ✚ Ένα προαιρετικό στοιχείο **Fault** το οποίο παρέχει πληροφορίες για σφάλματα που προκλήθηκαν κατά την επεξεργασία ενός μηνύματος.

Το πώς συντάσσουμε με XML ένα SOAP μήνυμα βασίζεται στο προεπιλεγμένο namespace για το SOAP envelope:

<http://www.w3.org/2003/05/soap-envelope>

Αυτό το XML namespace αναγνωριστικό δείχνει σε ένα XML Schema που καθορίζει τη δομή ενός SOAP μηνύματος.

Το προεπιλεγμένο namespace για την κωδικοποίηση του SOAP και τους τύπους δεδομένων είναι ο:

<http://www.w3.org/2001/12/soap-encoding>

```
<?xml version="1.0"?>
<soap:Envelope
xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
soap:encodingStyle=" http://www.w3.org/2001/12/soap-encoding">
<soap:Header>
...
...
</soap:Header>
<soap:Body>
<soap:Fault>
...
...
</soap:Fault> </soap:Body>
</soap:Envelope>
```

Μερικοί σημαντικοί κανόνες σύνταξης είναι οι παρακάτω:

1. Ένα μήνυμα SOAP **πρέπει** να είναι κωδικοποιημένο χρησιμοποιώντας XML.
2. Ένα μήνυμα SOAP **πρέπει** να χρησιμοποιεί το SOAP Envelope namespace.
3. Ένα μήνυμα SOAP **πρέπει** να χρησιμοποιεί το SOAP Encoding namespace.
4. Ένα μήνυμα SOAP **δεν πρέπει** να περιέχει αναφορά σε DTD.

5. Ένα μήνυμα SOAP **δεν πρέπει** να περιέχει XML Processing Instructions.

3.2. Επικοινωνία μέσω xml μηνυμάτων

Η επικοινωνία με XML μηνύματα γίνεται σε εφαρμογές που ανταλλάσσουν πληροφορίες χρησιμοποιώντας XML κείμενα. Παρέχει έναν ευέλικτο τρόπο επικοινωνίας για τις διάφορες εφαρμογές και αποτελεί τη βάση για το SOAP.

Επειδή η XML είναι ανεξάρτητη εφαρμογής, λειτουργικού συστήματος ή γλώσσας προγραμματισμού, τα XML μηνύματα μπορούν να χρησιμοποιηθούν σε όλα τα περιβάλλοντα. Μπορεί για παράδειγμα ένα πρόγραμμα στα Windows γραμμένο σε Perl να δημιουργήσει ένα XML κείμενο που αντιπροσωπεύει ένα μήνυμα, να το στείλει σε ένα πρόγραμμα των Unix γραμμένο σε Java και να επικοινωνήσουν έτσι τα δύο προγράμματα.

Η βασική ιδέα είναι ότι δύο εφαρμογές, ανεξαρτήτως λειτουργικού συστήματος, γλώσσας προγραμματισμού ή άλλης τεχνικής λεπτομέρειας υλοποίησης, μπορούν κάλλιστα να ανταλλάξουν πληροφορίες, χρησιμοποιώντας τίποτα άλλο παρά ένα απλό μήνυμα, κωδικοποιημένο με τέτοιο τρόπο, έτσι ώστε να γίνεται κατανοητό και από τις δυο εφαρμογές. Το SOAP παρέχει έναν πρότυπο τρόπο για την κατασκευή XML μηνυμάτων.

3.3. Η ανάγκη για μια πρότυπη κωδικοποίηση

Στην ανταλλαγή δεδομένων μεταξύ ετερογενών συστημάτων, πρέπει να υπάρχει συμφωνία ως προς την αναπαράσταση των δεδομένων αυτών. Ένα απλό δεδομένο, μπορεί να αναπαραστεί με πολλούς διαφορετικούς, αλλά εξίσου έγκυρους τρόπους με την XML.

Όμως δεν μπορούμε να πούμε με ακρίβεια ποιά είναι είναι η σωστή κωδικοποίηση, αντίθετα, σωστή θα θεωρήσουμε αυτή που θα δεχτεί η κάθε εφαρμογή. Το γεγονός αυτό μας οδηγεί στο συμπέρασμα ότι παρόλο που

τόσο ο server όσο και ο client και αν χρησιμοποιούν την XML για να επικοινωνήσουν δεν είναι αρκετό. Θα πρέπει να καθορίζονται:

- ✚ Οι τύποι των πληροφοριών που ανταλλάσσονται
- ✚ Ο τρόπος που οι πληροφορίες θα εκφράζονται στην XML
- ✚ Ο τρόπος διεξαγωγής της διαδικασίας αποστολής των πληροφοριών

Χωρίς αυτές τις προσυμφωνημένες συμβάσεις, τα προγράμματα δεν μπορούν να αποκωδικοποιήσουν τις πληροφορίες που λαμβάνουν ακόμα κι αν αυτές είναι γραμμένες σε XML. Το SOAP παρέχει αυτές τις συμβάσεις.

Κεφάλαιο 4. Wsdl (Web Service Description Language)

Πρωταρχικής σημασίας θεωρείται ο ορισμός μια συγκεκριμένης μεθόδου περιγραφής της κάθε δικτυακής υπηρεσίας. Για το σκοπό αυτό δημιουργήθηκε η περιγραφική γλώσσα Web Service Description Language (WSDL).

Σύμφωνα με τον ορισμό που δίνεται από το w3c, η WSDL είναι ένα σχήμα XML για την περιγραφή δικτυακών υπηρεσιών σαν ένα σύνολο από τελικά σημεία που λειτουργούν σε μηνύματα τα οποία περιέχουν πληροφορία είτε προσανατολισμένη στα έγγραφα είτε προσανατολισμένη στις διαδικασίες.

Με άλλα λόγια, η WSDL είναι μια προδιαγραφή που καθορίζει τον τρόπο περιγραφής των υπηρεσιών ιστού σε άλλη XML γραμματική. Η WSDL περιγράφει τέσσερα σημαντικά κομμάτια δεδομένων:

- ✚ Πληροφορία για όλες τις δημόσια διαθέσιμες λειτουργίες
- ✚ Πληροφορίες τύπων δεδομένων για όλα τα μηνύματα αίτησης και απάντησης
- ✚ Πληροφορίες σύνδεσης σχετικά με τα πρωτόκολλα μεταφοράς
- ✚ Πληροφορίες διευθύνσεων για τον εντοπισμό της συγκεκριμένης υπηρεσίας

Χρησιμοποιώντας την WSDL ένας πελάτης μπορεί να εντοπίσει μια υπηρεσία ιστού και να καλέσει οποιαδήποτε από τις δημόσιες λειτουργίες της.

Συμπερασματικά, μπορούμε να πούμε ότι η WSDL παρέχει ένα τρόπο στους παροχείς υπηρεσιών να περιγράψουν τη βασική μορφή των αιτήσεων και απαντήσεων των υπηρεσιών πάνω από διαφορετικά πρωτόκολλα και

κωδικοποιήσεις και χρησιμοποιείται για να περιγράψει **τί** μπορεί να κάνει μια υπηρεσία ιστού, **πού** βρίσκεται και **πώς** να το καλέσει κανείς.

4.1. Η wsdl προδιαγραφή

Όπως αναφέραμε παραπάνω, η WSDL είναι μια XML γραμματική για την περιγραφή των υπηρεσιών ιστού.

Ένα έγγραφο WSDL χρησιμοποιεί τα παρακάτω στοιχεία (elements) για τον ορισμό δικτυακών υπηρεσιών:

Types.

Το στοιχείο types περιγράφει όλους τους τύπους δεδομένων που χρησιμοποιούνται μεταξύ του πελάτη και του εξυπηρετητή. Η WSDL δεν ακολουθεί αποκλειστικά ένα συγκεκριμένο σύστημα δήλωσης τύπων, αλλά χρησιμοποιεί την W3C XML προδιαγραφή σχήματος ως προκαθορισμένη επιλογή της. Αν η υπηρεσία χρησιμοποιεί μόνο XML σχήμα δομημένο σε απλούς τύπους, όπως αλφαριθμητικά και ακεραίους, το στοιχείο types δεν είναι απαραίτητο.

Message

Το στοιχείο message περιγράφει ένα μήνυμα μιας κατεύθυνσης είτε αυτό είναι ένα μήνυμα αίτησης είτε απάντησης. Καθορίζει το όνομα του μηνύματος και περιέχει μηδέν ή περισσότερα στοιχεία 'part', τα οποία αναφέρονται στις παραμέτρους του μηνύματος ή στις επιστρεφόμενες τιμές του.

Operation

Το στοιχείο operation περιγράφει μια λειτουργία που υποστηρίζεται από μία υπηρεσία.

PortType

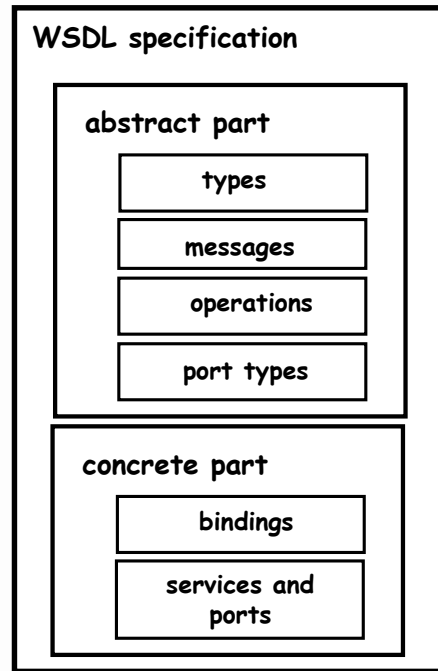
Το στοιχείο PortType συνδυάζει πολλαπλά στοιχεία 'message' για να συνθέσει μια απλή λειτουργία είτε μιας ή αμφίδρομης κατεύθυνσης. Για παράδειγμα, ένα PortType μπορεί να συνδυάζει ένα μήνυμα αίτησης και απάντησης σε μια μοναδική λειτουργία αίτησης-απάντησης η οποία χρησιμοποιείται πιο συχνά στις SOAP υπηρεσίες. Αξίζει να σημειωθεί ότι ένας PortType και συχνά καθορίζει πολύπλοκες λειτουργίες.

Binding

Το στοιχείο binding περιγράφει το συγκεκριμένο τρόπο υλοποίησης της υπηρεσίας πάνω στο καλώδιο.

Services and ports

Το στοιχείο service καθορίζει τη διεύθυνση για την κλήση της συγκεκριμένης υπηρεσίας και περιέχει το URL της υπηρεσίας.



Εικόνα 6. Wsdl specification

```
<definitions>
  <inport>
    <types>
      <schema> </schema>
    </types>
    <message>
      <part> </ part >
    </message>
    <PortType>
      <operation>
        <input></input>
        <output></ output >
        <fault></ fault >
      </ operation >
    </PortType>
    <binding>
      <operation>
        <input></input>
        <output></ output >
      </ operation >
    </binding>
    <service>
      <port></port>
    </service>
  </inport>
</definitions>
```

Πίνακας 2. Δομή wsdl κειμένου

4.2. Λειτουργίες wsdl

Η WSDL υποστηρίζει τέσσερα διαφορετικές λειτουργίες:

✚ Μιας κατεύθυνσης (One-way)

Η υπηρεσία λαμβάνει το μήνυμα. Έτσι το στοιχείο 'operation' έχει ένα μοναδικό στοιχείο 'input'.

✚ Αίτηση-Απάντηση (Request-response)

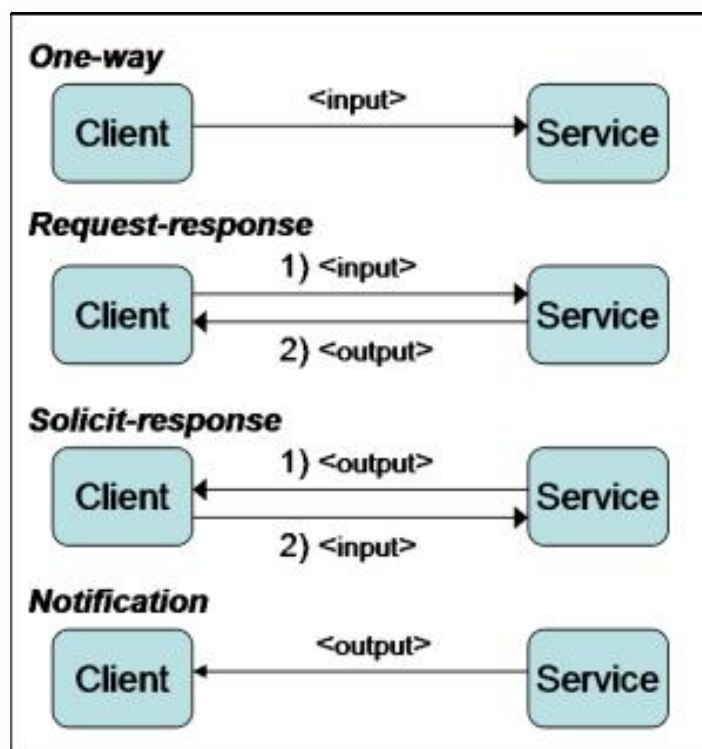
Η υπηρεσία λαμβάνει ένα μήνυμα αίτησης και στέλνει ένα μήνυμα απάντησης. Έτσι το στοιχείο 'operation' έχει ένα στοιχείο 'input' που ακολουθείται από ένα στοιχείο 'output'.

✚ Ζητώντας Απάντηση (Solicit response)

Η υπηρεσία στέλνει ένα μήνυμα και λαμβάνει έπειτα μια απάντηση. Έτσι, το στοιχείο 'operation' έχει ένα στοιχείο 'output' που ακολουθείται από ένα στοιχείο 'input'.

✚ Ειδοποίηση (Notification)

Η υπηρεσία στέλνει ένα μήνυμα. Έτσι, το στοιχείο 'operation' έχει ένα μοναδικό στοιχείο 'output'.



Εικόνα 7. Τα τέσσερα πρότυπα λειτουργιών που υποστηρίζονται από την wsdl

Κεφάλαιο 5. UDDI (Universal Description & Discovery Integration)

Το UDDI είναι μια τεχνική προδιαγραφή για την περιγραφή, ανακάλυψη και ολοκλήρωση των υπηρεσιών ιστού. Το UDDI είναι γι' αυτόν τον λόγο ένα σημαντικό μέρος του αναπτυσσομένου πρωτοκόλλου των υπηρεσιών ιστού, το οποίο διευκολύνει τις εταιρείες να δημοσιεύουν και να ανακαλύπτουν υπηρεσίες ιστού.

Ο σκοπός που δίνει το OASIS για το Universal Description, Discovery and Integration είναι ο παρακάτω:

"Οι υπηρεσίες ιστού έχουν νόημα μόνο όταν δυνητικοί χρήστες μπορούν να βρουν πληροφορίες ικανές ώστε να επιτρέψουν την εκτέλεσή τους. Το Universal Description Discovery & Integration (UDDI) εστιάζει στον καθορισμό ενός συνόλου από υπηρεσίες που θα υποστηρίζουν την περιγραφή και την ανακάλυψη:

- 1. Των εταιριών, των οργανισμών και άλλων παρόχων υπηρεσιών ιστού*
- 2. Των υπηρεσιών ιστού, που είναι διαθέσιμες*
- 3. Και των τεχνικών διεπαφών οι οποίες μπορούν να χρησιμοποιηθούν ώστε να έχει κάποιος χρήστης πρόσβαση σε αυτές τις υπηρεσίες.*

Βασισμένο σε ένα κοινό σύνολο από βιομηχανικά πρότυπα, συμπεριλαμβανομένων των HTTP, XML, XML Schema και SOAP το UDDI παρέχει μία διαλειτουργική, θεμελιώδη υποδομή για ένα περιβάλλον λογισμικού προσανατολισμένο στις υπηρεσίες τόσο για δημόσια διαθέσιμες υπηρεσίες όσο και για υπηρεσίες που εκτίθενται μόνο εσωτερικά ενός οργανισμού."

Με πιο απλά λόγια το UDDI είναι ο «χρυσός οδηγός» των υπηρεσιών ιστού. Όπως σε ένα χρυσό οδηγό, μπορούμε να αναζητήσουμε μια εταιρία που προσφέρει τις υπηρεσίες που χρειαζόμαστε, να διαβάσουμε για μια προσφερόμενη υπηρεσία και να επικοινωνήσουμε με κάποιον για περισσότερες λεπτομέρειες. Φυσικά μπορούμε να προσφέρουμε μια υπηρεσία ιστού χωρίς να το καταχωρίσουμε στο UDDI, όπως αν ανοίγαμε μία επιχείρηση στο υπόγειο του σπιτιού μας και βασιζόμασταν στη διαφήμιση από στόμα σε στόμα. Αλλά αν θέλουμε να αγγίξουμε το ευρύ κοινό, θα χρειαστούμε το UDDI ώστε οι δυνητικοί μας πελάτες να μπορέσουν να μας βρουν.

5.1. Οργάνωση του UDDI

Οι υπηρεσίες ιστού που αποθηκεύονται μέσα στο UDDI κατηγοριοποιούνται σε τρεις κατηγορίες, ανάλογες του τηλεφωνικού καταλόγου:

Λευκές σελίδες

Όπως και σε κάθε τηλεφωνικό κατάλογο οι λευκές σελίδες παρέχουν ένα ονομαστικό ευρετήριο έτσι και εδώ οι λευκές σελίδες παρέχουν ένα ονομαστικό ευρετήριο των επιχειρήσεων. Περιέχουν πληροφορίες όπως το όνομα της επιχείρησης, κάποια πληροφορία συνδέσμου, διεύθυνση και τηλέφωνο. Επίσης μπορεί να περιέχουν και αναγνωριστικά των επιχειρήσεων.

Κίτρινες σελίδες

Αυτές οι σελίδες κατηγοριοποιούν τις επιχειρήσεις ή τις υπηρεσίες ανάλογα με την κατηγορία στην οποία ανήκουν. Για παράδειγμα μπορεί να περιέχονται δεδομένα όπως βιομηχανίες, προϊόντα ή γεωγραφικούς κώδικες, όλα ομαδοποιημένα σύμφωνα με καθορισμένες κατηγορίες.

✚ Πράσινες σελίδες

Αυτές οι σελίδες περιλαμβάνουν την τεχνική περιγραφή μιας υπηρεσίας ιστού. Αυτό συνιστά έναν δείκτη σε μια περιγραφή της υπηρεσίας ιστού και μια διεύθυνση για να την καλέσεις.

Πληροφορία	Λειτουργίες	Λεπτομέρειες (που υποστηρίζονται από API)
White pages: Πληροφορίες όπως το όνομα, η διεύθυνση, το τηλέφωνο και άλλες πληροφορίες επικοινωνίας για μία επιχείρηση.	Publish: Πώς ο προμηθευτής μιας υπηρεσίας ιστού καταχωρεί των εαυτό του.	Business Information: Περιλαμβάνεται σε ένα αντικείμενο BusinessEntity , το οποίο με τη σειρά του περιλαμβάνει πληροφορίες για υπηρεσίες, κατηγορίες, επαφές, URLs, και άλλα αναγκαία στοιχεία για να αλληλεπιδράσουμε με μία επιχείρηση.
Yellow Pages: Πληροφορίες που κατηγοριοποιούν επιχειρήσεις. Βασίζονται σε υπάρχοντα πρότυπα κατηγοριοποίησης (μη ηλεκτρονικά).	Find: Πώς μία εφαρμογή βρίσκει μια συγκεκριμένη υπηρεσία ιστού.	Service Information: Περιγράφει μία ομάδα από υπηρεσίες ιστού. Αυτές περιλαμβάνονται σε ένα αντικείμενο BusinessService .
Green Pages: Τεχνικές πληροφορίες για τις υπηρεσίες ιστού που παρέχονται από μία	Bind: Πώς μία εφαρμογή συνδέεται, και αλληλεπιδρά με μια υπηρεσία ιστού	Binding Information: Οι απαραίτητες τεχνικές λεπτομέρειες για την κλήση μιας υπηρεσίας ιστού.

επιχείρηση.	αφού αυτό βρεθεί.	Περιλαμβάνουν τα URLs, πληροφορίες για ονόματα μεθόδων, τύπους ορισμάτων και ούτω καθ'εξής. Το αντικείμενο BindingTemplate αναπαριστά αυτά τα δεδομένα.
-------------	-------------------	--

Πίνακας 3. Προσφερόμενες υπηρεσίες του UDDI

Ο UDDI αναμένεται να αποτελέσει τη βάση για υψηλότερου επιπέδου υπηρεσίες που θα υποστηρίζονται από άλλα πρότυπα.

5.2. Ανακάλυψη των υπηρεσιών ιστού στο UDDI

5.2.1. Ανακάλυψη μέσω web

Υπάρχουν τρεις επιλογές για να ψάξει κανείς στο UDDI:

✚ Ο Internet Explorer υποστηρίζει την ανακάλυψη των υπηρεσιών μέσω του Real Names Keyword System. Μπορείς, δηλαδή να ανακαλύψεις τις υπηρεσίες ιστού γράφοντας “uddi” στη γραμμή διευθύνσεων, ακολουθούμενο από το όνομα της εταιρείας που σε ενδιαφέρει να βρεις την υπηρεσία ιστού.

✚ Κατ' ευθείαν στο site της Microsoft, στο <http://uddi.microsoft.com/>

✚ Κατ' ευθείαν στο site της IBM, στο <http://www-3.ibm.com/services/uddi/testregistry/>

5.2.2. Ανακάλυψη με το UDDI inquiry api

Το UDDI API είναι ένα πρωτόκολλο βασισμένο στο SOAP για να επικοινωνείς με το UDDI Business Registry. Το API είναι χωρισμένο σε δύο μέρη: το

Inquiry API για την ανακάλυψη των υπηρεσιών και το Publisher API για τη δημοσίευση και ενημέρωση αυτών.

Ο παρακάτω πίνακας παρουσιάζει τις κυριότερες λειτουργίες ανακάλυψης, οι οποίες χωρίζονται σε δύο κατηγορίες, τις *find_xxx* λειτουργίες που παρέχουν γενική ανακάλυψη υπηρεσιών και τις *get_xxx* λειτουργίες, που ανακτούν εγγραφές σύμφωνα με τις τιμές κλειδιών τους.

ΒΑΣΙΚΕΣ ΛΕΙΤΟΥΡΓΙΕΣ ΤΟΥ UDDI INQUIRY API	
Όνομα λειτουργίας	Περιγραφή
find_xxx λειτουργίες	
find_binding	Ψάχνει για bindings σχετικά με μια υπηρεσία
find_business	Ψάχνει για businesses που πληρούν κάποια κριτήρια
find_service	Ψάχνει για services που ανήκουν σε κάποια επιχείρηση
find_tModel	Ψάχνει για tModels που πληρούν κάποια κριτήρια
get_xxx λειτουργίες	
get_bindingDetail	Ανακτά μια εγγραφή bindingTemplate
get_businessDetail	Ανακτά μια εγγραφή businessEntity
get_serviceDetail	Ανακτά μια εγγραφή nusinessService
get_tModelDetail	Ανακτά μια εγγραφή tModel

Πίνακας 4. Βασικές λειτουργίες του UDDI inquiry api

Για να καλέσουμε αυτές τις συναρτήσεις μπορούμε να στείλουμε ένα SOAP μήνυμα απ' ευθείας στα παρακάτω URLs:

✚ Microsoft: <http://uddi.microsoft.com/inquire>

✚ IBM: <http://www-3.ibm.com/services/uddi/inquiryapi>

Κεφάλαιο 6. Σημασιολογικός ιστός (semantic web)

Ο σημασιολογικός ιστός (Semantic Web) σκοπεύει να αποτελέσει ένα καθολικό μέσο για την ανταλλαγή πληροφοριών δίνοντας νόημα (σημασιολογία) στο περιεχόμενο των εγγράφων στο Web, κατά τρόπο κατανοητό από τους υπολογιστές. Υπό την καθοδήγηση του δημιουργού του Web, Tim Berners-Lee του World Wide Web Consortium (W3C), ο σημασιολογικός ιστός επεκτείνει τις δυνατότητες του World Wide Web μέσω της χρήσης προτύπων, γλωσσών επισήμανσης (mark-up) και σχετικών εργαλείων επεξεργασίας.

Ο σημασιολογικός ιστός παρέχει ένα κοινό πλαίσιο που επιτρέπει στα δεδομένα να διαμοιράζονται και να επαναχρησιμοποιούνται πέρα από τα όρια μιας εφαρμογής, μιας επιχείρησης ή μίας κοινότητας. Είναι μια συλλογική προσπάθεια της οποίας ηγείται το W3C με τη συμμετοχή ενός μεγάλου αριθμού ερευνητών και βιομηχανικών συνεργατών. Είναι βασισμένο στο Resource Description Framework (RDF), το οποίο ενσωματώνει ποικίλες εφαρμογές χρησιμοποιώντας XML για τη σύνταξη και URIs για την ονοματοδοσία.

Το 2001 ο Tim Berners-Lee έγραψε:

"Το Semantic Web είναι μια επέκταση του τρέχοντος Web, στον οποίο δίνεται ένα καθορισμένο με σαφήνεια νόημα στις πληροφορίες, δίνοντας τη δυνατότητα σε υπολογιστές και ανθρώπους να συνεργάζονται καλύτερα."

Το μεγαλύτερο μέρος του περιεχομένου του Ιστού σήμερα σχεδιάζεται για να διαβαστεί από ανθρώπους, όχι για να το χειριστούν εννοιολογικά τα προγράμματα υπολογιστών. Βασίζεται κυρίως σε έγγραφα γλώσσας HTML η οποία χρησιμεύει στην περιγραφή ενός δομημένου κειμένου, με έμφαση στην οπτική απεικόνιση. Η HTML έχει περιορισμένες ικανότητες όσον αφορά την ταξινόμηση των τμημάτων κειμένου μιας σελίδας. Οι υπολογιστές μπορούν

αποδοτικά να αναλύσουν ιστοσελίδες ως προς την εμφάνιση και τις ρουτίνες, αλλά γενικά, οι υπολογιστές δεν έχουν συνήθως κάποιον αξιόπιστο τρόπο να επεξεργαστούν τη σημασιολογία (semantics) που περιέχεται στη σελίδα αυτή.

Για παράδειγμα, οι μηχανές αναζήτησης εκτελούν απλές λεξιλογικές αναλύσεις που βασίζονται σε συμβολοσειρές αναζήτησης και παράγουν λίστες αποτελεσμάτων που απαιτούν την εξαγωγή συμπερασμάτων από τους ανθρώπινους χρήστες, όσον αφορά το ποια από τα δεδομένα είναι τα κατάλληλα για την θεματική περιοχή που ερευνάται. Χρειάζεται δηλαδή η λογική ανάλυση εκ μέρους του χρήστη για να βρεθούν τα συναφή συμπεράσματα. Αυτό συμβαίνει διότι οι ιστοσελίδες δεν περιέχουν αρκετή πληροφορία σχετικά με τη σημασιολογία των περιεχομένων τους οπότε το λογισμικό του χρήστη δεν μπορεί να βγάλει συμπεράσματα από το περιεχόμενο της σελίδας. Για να αντιμετωπιστεί αυτό το πρόβλημα, οι τεχνολογίες του σημασιολογικού ιστού δίνουν τη δυνατότητα σε αυτοματοποιημένα εργαλεία να εξάγουν αυτά τα συμπεράσματα.

Ο σημασιολογικός ιστός επιχειρεί να δομήσει το σημασιολογικό περιεχόμενο των ιστοσελίδων, δημιουργώντας ένα περιβάλλον όπου οι πράκτορες λογισμικού που περιπλανώνται από σελίδα σε σελίδα μπορούν εύκολα να εκτελέσουν τους περίπλοκους στόχους των χρηστών τους. Έτσι, αποζητείται ο μετασχηματισμός του Web από ένα σύνολο στατικών σελίδων σε ένα δίκτυο δυναμικών παροχών υπηρεσιών (υπηρεσίες ιστού) που ανακαλύπτουν αυτόματα τις αναζητούμενες πληροφορίες, διαπραγματεύονται για αγαθά που ο χρήστης σκοπεύει να αγοράσει ή συγκεντρώνουν πληροφορίες από διαφορετικές πηγές και τις συνενώνουν σε ομοιογενείς μορφές.

Για να λειτουργήσει ο σημασιολογικός ιστός, πρέπει οι υπολογιστές να έχουν πρόσβαση σε δομημένες συλλογές πληροφοριών και σύνολα λογικών κανόνων που μπορούν να χρησιμοποιήσουν για να διεξάγουν την αυτοματοποιημένη εξαγωγή συμπερασμάτων.

Η ενσωμάτωση νοήματος στο περιεχόμενο είναι το βασικό χαρακτηριστικό του Σημασιολογικού Ιστού. Για την επίτευξη αυτού του χαρακτηριστικού, ο σημασιολογικός ιστός δομείται με βάση τα παρακάτω επίπεδα / τεχνολογίες:

- ✚ τα URIs (Universal Resource Identifier): συμβολοσειρές που ταυτοποιούν μοναδικά μία οντότητα (ένα Web site, μία ιδιότητα, έναν άνθρωπο, ένα πράγμα κλπ)
- ✚ γλώσσα επισημάνσεων XML: επιτρέπει στους χρήστες να προσθέτουν αυθαίρετη δομή στα έγγραφά τους, χωρίς να καθορίζει την σημασιολογία αυτής της δομής
- ✚ τεχνολογία RDF: χρησιμοποιείται για την αναπαράσταση δεδομένων και ανταλλαγή γνώσης στο διαδίκτυο
- ✚ τεχνολογία OWL: χρησιμοποιείται για τη δημιουργία και διανομή οντολογιών, υποστηρίζοντας προχωρημένη αναζήτηση στο διαδίκτυο, πράκτορες λογισμικού και διαχείριση γνώσης
- ✚ κανόνες και επίπεδο λογικής, που καθιστούν δυνατή την ευφυή συλλογιστική (intelligent reasoning) σε νοηματικά δεδομένα
- ✚ επίπεδο απόδειξης (proof), που καθιστά δυνατή την επικοινωνία ανάμεσα σε πράκτορες λογισμικού (agents).

Κάθε μία από τις παραπάνω τεχνολογίες βασίζεται σε εκείνες που αναφέρονται πριν από αυτή. Έτσι π.χ. η RDF βασίζεται στην XML και χρησιμοποιεί URIs. Οι τεχνολογίες αυτές συνδυάζονται ώστε να παρέχουν περιγραφές που συμπληρώνουν ή υποκαθιστούν το περιεχόμενο εγγράφων στο Web, όπως περιγράφεται παρακάτω. Αυτές οι machine-readable περιγραφές επιτρέπουν την προσθήκη νοήματος στο περιεχόμενο, διευκολύνοντας έτσι την αυτόματη αναζήτηση πληροφοριών από υπολογιστές.

Το νόημα του περιεχομένου εκφράζεται μέσω της RDF, η οποία κωδικοποιεί σύνολα από triples (τριάδες) που αντιπροσωπεύουν περίπου το υποκείμενο, το ρήμα και το αντικείμενο μιας πρότασης. Αυτές οι τριάδες μπορούν να γραφούν με XML. Ένα έγγραφο της RDF δηλώνει ότι κάποιες οντότητες (ιστοσελίδες, άνθρωποι ή πράγματα) έχουν κάποιες ιδιότητες με συγκεκριμένες τιμές (π.χ. ο Χ είναι συγγραφέας του βιβλίου Υ). Αυτός ο τρόπος αναπαράστασης μπορεί να εκφράσει τα περισσότερα δεδομένα που κατανοούν οι υπολογιστές. Το υποκείμενο, το ρήμα και το αντικείμενο προσδιορίζονται από ένα URI (Universal Resource Identifier), όπως ακριβώς γίνεται με τις ιστοσελίδες. Έτσι μπορεί ο καθένας να προσθέσει μία νέα οντότητα ή μία νέα ιδιότητα.

Ένα βασικό συστατικό του σημασιολογικού ιστού είναι οι οντολογίες. Οι οντολογίες ορίζουν με τυπικό τρόπο τις έννοιες και τις συσχετίσεις των εννοιών για κάποιο πεδίο. Περιέχουν ορισμούς κλάσεων αντικειμένων και σχέσεις ανάμεσα στις κλάσεις, καθώς και κανόνες εξαγωγής λογικών συμπερασμάτων. Μέσω των οντολογιών γίνεται δυνατόν να συνδυαστούν δεδομένα από διαφορετικές πηγές, οι οποίες μοιράζονται την ίδια οντολογία. Ακόμη, λύνονται προβλήματα ορολογίας καθώς η σημασία των όρων που εμφανίζονται σε μία σελίδα μπορεί να οριστεί με δείκτες προς την οντολογία. Οι οντολογίες μπορούν να βελτιώσουν την λειτουργία του Web αυξάνοντας την ακρίβεια των αναζητήσεων πληροφοριών, αφού αναζητούνται οι πληροφορίες που αναφέρονται μόνο σε μία συγκεκριμένη έννοια και όχι σε κάποια λέξη κλειδί. Μπορούν επίσης να χρησιμοποιηθούν για την συσχέτιση των πληροφοριών ενός site με αντίστοιχες δομές γνώσης και λογικούς κανόνες. Οι κοινές οντολογίες βοηθούν στην ανταλλαγή δεδομένων και νοημάτων μεταξύ διαφορετικών web-based υπηρεσιών.

Υπάρχουν πολλές αυτοματοποιημένες υπηρεσίες βασισμένες στο διαδίκτυο (υπηρεσίες ιστού) που δεν χρησιμοποιούν τη σημασιολογία, όμως άλλα προγράμματα (όπως είναι οι πράκτορες λογισμικού) δεν μπορούν να εντοπίσουν μόνα τους μία τέτοια υπηρεσία για κάποια συγκεκριμένη

λειτουργία. Αυτό μπορεί να συμβεί μόνο όταν υπάρχει μία κοινή γλώσσα περιγραφής παρεχόμενων υπηρεσιών σε ένα δίκτυο ούτως ώστε οι διάφοροι πράκτορες να διαφημίζουν τις υπηρεσίες τους σε μία υπηρεσία κίτρινων σελίδων.

Αφού εντοπιστεί ο επιθυμητός πράκτορας, οι δύο επικοινωνούντες πράκτορες λογισμικού μπορούν να κατανοήσουν ο ένας τον άλλο ανταλλάσσοντας οντολογίες. Ο σημασιολογικός ιστός παρέχει αυτή την ευελιξία. Ανακαλύπτοντας νέες οντολογίες οι πράκτορες λογισμικού αποκτούν νέες δυνατότητες εξαγωγής λογικών συμπερασμάτων.

Η ενοποιητική λογική γλώσσα του σημασιολογικού ιστού δίνει τη δυνατότητα να ενωθούν οι έννοιες που μπορεί να ορίσει ο καθένας (μέσω ενός URI) σε ένα καθολικό Web. Έτσι οι πράκτορες λογισμικού θα μπορούν να αναλύουν σημασιολογικά τη γνώση των ανθρώπων παρέχοντας μία νέα μορφή χρήσιμων εργαλείων.

6.1. Παγκόσμιος ιστός και Σημασιολογικός ιστός

Ο παγκόσμιος ιστός είναι σχεδιασμένος γύρω από διαθέσιμους πόρους, το μοναδικό τρόπο διευθυνσιοδότησης αυτών των πόρων και ένα σύνολο από εντολές. Είναι σχεδιασμένος να λειτουργεί πάνω σε μεγάλης κλίμακας και πολύπλοκα δίκτυα με κατανεμημένο τρόπο. Στην παράγραφο αυτή, όταν θα αναφερόμαστε στον όρο «πόρο» θα εννοούμε κάποιο πακέτο δεδομένων, το οποίο μπορεί να είναι κάποια έγγραφα, ιστοσελίδες ή οτιδήποτε άλλο δεδομένο. Στον παγκόσμιο ιστό αναφερόμαστε στους πόρους αυτούς μέσω των URIs (Uniform Resource Indicators) γενικότερα, ενώ μέσω των URLs (Uniform Resource Locators) αναφερόμαστε σε πόρους που μπορούν να διευθυνσιοδοτηθούν και να ανακτηθούν απευθείας μέσω πρωτοκόλλων πέραν του HTTP, όπως FTP.

Ο παγκόσμιος ιστός λειτουργεί πάνω από ένα τεράστιο δίκτυο υπολογιστών με ένα αστρονομικό αριθμό από ιστοσελίδες και ιστοχώρους και πρέπει να

συνεχίσει να λειτουργεί καθώς τα μεγέθη αυτά αυξάνονται. Σε ένα τέτοιο κατακεκομημένο δίκτυο που συνεχώς μεγαλώνει, οποιοσδήποτε διαθέτει υπολογιστή μπορεί με τη βοήθεια κάποιου εξυπηρέτη (server) να προσθέσει πόρους στο ήδη υπάρχον δίκτυο χωρίς να είναι αναγκασμένος να τις καταχωρήσει και σε κάποιο συγκεκριμένο μέρος ή κόμβο ή κατάλογο. Ο παγκόσμιος ιστός, λοιπόν, είναι ανοιχτός με την έννοια ότι νέες ιστοσελίδες, νέα δεδομένα και νέοι πόροι γενικότερα μπορούν να προστεθούν ελεύθερα από τον καθένα χωρίς κάποιο κεντρικό έλεγχο.

Είναι κατανοητό, λοιπόν, ότι ο παγκόσμιος ιστός είναι πιθανό να είναι ημιτελής, με την έννοια ότι δεν παρέχει καμία εγγύηση ότι όλοι οι πιθανοί σύνδεσμοι θα λειτουργούν και ότι όλη η πιθανή πληροφορία θα είναι διαθέσιμη.

Είναι προφανές πια, πως ο σημασιολογικός ιστός προκειμένου να συμβαδίζει με το τρέχον μοντέλο του παγκόσμιου ιστού πρέπει να ακολουθήσει «ιδέες» και προσεγγίσεις του τωρινού μοντέλου:

- ✚ Πρέπει να χρησιμοποιεί τρόπους διευθυνσιοδότησης μέσω URIs.
- ✚ Πρέπει να χρησιμοποιεί πρωτόκολλα με μικρό και κοινά κατανοητό σύνολο εντολών.
- ✚ Πρέπει να διατηρεί όσο το δυνατόν λιγότερο ή ακόμη και καθόλου ιστορικό των πληροφοριών του διαδικτύου.
- ✚ Πρέπει να είναι όσο το δυνατόν κατακεκομημένο.
- ✚ Πρέπει να μπορεί να λειτουργεί πάνω σε μεγάλης κλίμακας δίκτυα.
- ✚ Πρέπει να επιτρέπει την τοπική αποθήκευση των πληροφοριών ώστε να επιταχύνει την πρόσβαση σε αυτές και να ελαττώνει το φόρτο του δικτύου.

- ✚ Τέλος, μα εξίσου σημαντικό, πρέπει να είναι σε θέση να αντεπεξέλθει σε οποιαδήποτε ασυνέπεια συνδέσμων ή ημιτελών πληροφοριών.

6.2. Σημασιολογικές υπηρεσίες διαδικτύου

Ο όρος «Σημασιολογικές Υπηρεσίες Ιστού» (Semantic Web Services), αναφέρεται στον εμπλουτισμό του τρέχοντος μοντέλου υπηρεσιών ιστού με σημασιολογική πληροφορία (semantic information). Σκοπός της παραπάνω ενέργειας είναι πληρέστερη περιγραφή τους, με απώτερο στόχο την αρμονική ενσωμάτωση τους στο «Σημασιολογικό Ιστό». Με το τρόπο αυτό, η περιγραφή των υπηρεσιών θα είναι επαρκής για την αποδοτικότερη αναζήτηση τους, μέσα στο τεράστιο πλήθος των προσφερόμενων υπηρεσιών στο διαδίκτυο, αφού θα καθίστανται κατανοητές από αυτοματοποιημένο λογισμικό.

Με την μοντελοποίηση σε οντολογίες αντικειμένων του πραγματικού κόσμου και όχι απλώς δεδομένων, με τον τρόπο που υλοποιούνται και στον «Σημασιολογικό Ιστό», μια υπηρεσία ιστού π.χ. που προσφέρει πληροφορίες για αυτοκίνητα, θα μπορεί να συσχετίσει τον οδηγό με το αυτοκίνητο, χωρίς την ύπαρξη αναφοράς σε αυτοκίνητο ή ακόμα και την φωτογραφία ενός οδηγού με τον αριθμό άδειας κυκλοφορίας του, την φωτογραφία ενός αυτοκινήτου με τον αριθμό κυκλοφορίας του και την οδό κατοικίας του οδηγού. Τα παραπάνω επιτυγχάνονται με την περιγραφή του τί είναι αυτοκίνητο, οδηγός, τί είναι φωτογραφία και πώς συσχετίζονται μεταξύ τους. Η «Σημασιολογική Υπηρεσία Ιστού» θα αναλάβει να τα αναλύσει και να παράγει ένα εμπειριστατωμένο αποτέλεσμα, όπως το παραπάνω παράδειγμα.

Κεφάλαιο 7. Οντολογίες

Όπως αναφέραμε, ο σημασιολογικός ιστός θα φέρει δομή στο σημασιολογικό περιεχόμενο του Web, δημιουργώντας ένα περιβάλλον όπου οι πράκτορες λογισμικού που περιπλανώνται από σελίδα σε σελίδα θα μπορούν εύκολα να εκτελέσουν περίπλοκους στόχους για τους χρήστες. Ο σημασιολογικός ιστός θα στηριχτεί στη δυνατότητα της XML να καθορίζει προσαρμοσμένα σχήματα και στην ευελιξία της RDF (Resource Description Framework) να αναπαριστά δεδομένα. Το επόμενο στοιχείο που απαιτείται για το Semantic Web είναι μια γλώσσα οντολογίας Ιστού (Web Ontology Language) που μπορεί τυπικά να περιγράψει τη σημασιολογία των κλάσεων και των ιδιοτήτων που χρησιμοποιούνται στα έγγραφα του Web. Η γλώσσα αυτή πρέπει να υπερβεί τη βασική σημασιολογία του σχήματος RDF, προκειμένου οι μηχανές να εκτελούν χρήσιμους συλλογισμούς σε αυτά τα έγγραφα.

7.1. Τί είναι οι οντολογίες

Η λέξη "οντολογία" είναι ένας όρος δανεισμένος από τη φιλοσοφία, η οποία αναφέρεται στην επιστήμη της περιγραφής των διαφόρων ειδών οντοτήτων στον κόσμο και του τρόπου που αυτές συσχετίζονται. Συχνά, συγχέεται με την επιστημολογία που αφορά τη γνώση, ενώ έχει δημιουργήσει πολλές διαμάχες στις συζητήσεις για την τεχνητή νοημοσύνη.

Στα πλαίσια της κατανεμημένης γνώσης, οντολογία είναι ο προσδιορισμός μιας σύλληψης, δηλαδή, μια περιγραφή των εννοιών και των σχέσεων που μπορούν να υπάρξουν για έναν πράκτορα λογισμικού ή μια κοινότητα τέτοιων πρακτόρων. Είναι μια αναγνώσιμη από τον υπολογιστή περιγραφή της γνώσης των πόρων στο δίκτυο. Αυτός ο ορισμός είναι σύμφωνος με τη χρήση της οντολογίας ως σύνολο ορισμών εννοιών, αλλά γενικότερος. Λέμε ότι ένας πράκτορας λογισμικού "δεσμεύεται" σε μια οντολογία εάν οι ενέργειές του είναι σύμφωνες με τους ορισμούς στην οντολογία. Πρακτικά, μια κοινή οντολογία

καθορίζει το λεξιλόγιο με το οποίο ερωτήσεις και ισχυρισμοί ανταλλάσσονται μεταξύ των πρακτόρων λογισμικού. Οι οντολογικές δεσμεύσεις είναι συμφωνίες χρησιμοποίησης του κοινού λεξιλογίου κατά τρόπο κατανοητό και συνεπή. Οι πράκτορες λογισμικού που μοιράζονται ένα λεξιλόγιο δεν χρειάζονται να μοιράζονται μια βάση γνώσεων και καθένας από αυτούς δεν απαιτείται να απαντήσει σε όλες τις ερωτήσεις που μπορούν να διατυπωθούν στο κοινό λεξιλόγιο. Με τις οντολογίες οι πράκτορες λογισμικού γίνονται ευφυείς επειδή μπορούν να χρησιμοποιήσουν τη γνώση που περιλαμβάνεται στην οντολογία στο στάδιο της διαπραγμάτευσης και της λήψης αποφάσεων .

Στην επιστήμη των πληροφοριών, μια οντολογία καθορίζει τους όρους που χρησιμοποιούνται για να περιγράψουν και να αναπαραστήσουν έναν τομέα της γνώσης, με σκοπό τη διευκόλυνση της διανομής και της επαναχρησιμοποίησης της γνώσης αυτής. Μια οντολογία είναι το προϊόν μιας προσπάθειας να διατυπωθεί ένα εξαντλητικό και αυστηρό εννοιολογικό σχήμα για μια περιοχή. Είναι μία ιεραρχική δομή δεδομένων που περιέχει όλες τις σχετικές οντότητες, τις μεταξύ τους σχέσεις και κανόνες εξαγωγής συμπερασμάτων μέσα σε εκείνη την περιοχή. Οι οντολογίες χρησιμοποιούνται συνήθως στην τεχνητή νοημοσύνη και στην αναπαράσταση γνώσης. Τα προγράμματα υπολογιστών μπορούν να χρησιμοποιήσουν μια οντολογία για ποικίλους λόγους συμπεριλαμβανομένου του επαγωγικού συλλογισμού, της ταξινόμησης, των ποικίλων τεχνικών επίλυσης προβλημάτων, καθώς επίσης και για να διευκολύνουν την επικοινωνία και τη διανομή των πληροφοριών μεταξύ των διαφορετικών συστημάτων. Χρησιμοποιούνται από τους ανθρώπους, τις βάσεις δεδομένων, και τις εφαρμογές που πρέπει να μοιραστούν τις πληροφορίες περιοχών (μια περιοχή είναι ακριβώς μία συγκεκριμένη θεματική περιοχή ή τομέας της γνώσης, όπως η ιατρική, η κατασκευή εργαλείων, η ακίνητη περιουσία, η γεωγραφία, η οικονομική διαχείριση, κ.λπ.). Οι οντολογίες περιλαμβάνουν χρησιμοποιούμενους από υπολογιστές ορισμούς βασικών εννοιών μίας θεματικής περιοχής και τις σχέσεις μεταξύ τους. Κωδικοποιούν τη γνώση σε μια περιοχή, καθώς και τη

γνώση από περισσότερες από μία περιοχές. Έτσι, καθιστούν την γνώση αυτή επαναχρησιμοποιήσιμη.

Η οντολογική ανάλυση αποσαφηνίζει τη δομή της γνώσης. Δεδομένης μιας περιοχής, η οντολογία της διαμορφώνει τον πυρήνα οποιουδήποτε συστήματος αναπαράστασης γνώσης για αυτή την περιοχή. Χωρίς τις οντολογίες, ή τις συλλήψεις που κρύβονται κάτω από τη γνώση, δεν μπορεί να υπάρξει ένα λεξιλόγιο για την αναπαράσταση της γνώσης.

Η λέξη “οντολογία” έχει χρησιμοποιηθεί για να περιγράψει τεχνήματα διαφορετικών βαθμών δομής. Αυτά κυμαίνονται από τις απλές ταξονομίες, ως τα σχέδια μετα-δεδομένων και τις λογικές θεωρίες. Το Semantic Web χρειάζεται οντολογίες με έναν σημαντικό βαθμό δομής και οι οντολογίες αυτές πρέπει να διευκρινίζουν τις περιγραφές για τα ακόλουθα είδη εννοιών:

- ✚ Κλάσεις (γενικά πράγματα) στις πολλές περιοχές ενδιαφέροντος
- ✚ Οι σχέσεις που μπορούν να υπάρξουν μεταξύ των πραγμάτων
- ✚ Οι ιδιότητες (ή χαρακτηριστικά) των πραγμάτων (slots)

Οι οντολογίες εκφράζονται συνήθως σε μια βασισμένη στη λογική γλώσσα, έτσι ώστε λεπτομερείς, ακριβείς, συνεπείς και σημαντικές διακρίσεις να μπορούν να γίνουν μεταξύ των κλάσεων, των ιδιοτήτων, και των σχέσεων. Μερικά εργαλεία οντολογίας μπορούν να εκτελέσουν αυτοματοποιημένο συλλογισμό χρησιμοποιώντας τις οντολογίες, και να παρέχουν έτσι προηγμένες υπηρεσίες σε ευφυείς εφαρμογές όπως: σημασιολογική αναζήτηση και ανάκτηση, πράκτορες λογισμικού, υποστήριξη αποφάσεων, ομιλία και φυσική κατανόηση γλώσσας, διαχείριση γνώσης, ευφυείς βάσεις δεδομένων, και ηλεκτρονικό εμπόριο.

Οι οντολογίες φαίνεται ότι θα χρησιμοποιηθούν στο Semantic Web ως τρόπος αναπαράστασης της σημασιολογίας των εγγράφων και προσφοράς της ικανότητας σε web εφαρμογές και ευφυείς πράκτορες να χρησιμοποιήσουν τη

σημασιολογία αυτή. Οι οντολογίες μπορούν να αποδειχθούν πολύ χρήσιμες για μια κοινότητα ως τρόπος δόμησης και ορισμού της έννοιας των μετα-δεδομένων που συλλέγονται αυτήν την περίοδο και τυποποιούνται. Χρησιμοποιώντας τις οντολογίες, οι αυριανές εφαρμογές μπορούν να είναι “ευφυείς”, υπό την έννοια ότι μπορούν να λειτουργήσουν ακριβέστερα στο ανθρώπινο εννοιολογικό επίπεδο.

Οι οντολογίες μπορούν να ενισχύσουν τη λειτουργία του Web από πολλές απόψεις. Μπορούν να χρησιμοποιηθούν για να βελτιώσουν την ακρίβεια των Web αναζητήσεων - το πρόγραμμα αναζήτησης μπορεί να ψάχνει μόνο εκείνες τις σελίδες που αναφέρονται σε μια ακριβή έννοια αντί όλων των άλλων εννοιών των διαφορούμενων λέξεων κλειδιών. Οι πιο προηγμένες εφαρμογές μπορούν να χρησιμοποιήσουν τις οντολογίες για να συσχετίσουν τις πληροφορίες μιας σελίδας με τις σχετικές δομές γνώσης και κανόνες εξαγωγής συμπερασμάτων. Επιπλέον, καθιστούν πολύ ευκολότερη την ανάπτυξη προγραμμάτων που μπορούν να αντιμετωπίσουν περίπλοκες ερωτήσεις, των οποίων οι απαντήσεις δεν βρίσκονται σε μία ενιαία ιστοσελίδα.

Ακόμη, οι οντολογίες μπορούν να χρησιμοποιηθούν για να υποστηρίξουν συστήματα τεχνητής νοημοσύνης (AI), με την παροχή μιας βαθύτερης και σθεναρότερης αντιπροσώπευσης της περιοχής στην οποία κάποιος επιθυμεί να εξάγει συλλογισμούς και να λύσει προβλήματα. Σε αυτές εφαρμόζεται, αυστηρά, ο παραγωγικός συλλογισμός και όχι λογική ασαφής ή βασισμένη σε πιθανότητες, ή οποιοσδήποτε συλλογισμός που προσπαθεί να μιμηθεί τη συνείδηση.

Μια οντολογία διαφέρει από ένα σχήμα XML διότι είναι μια αναπαράσταση γνώσης, όχι ένα σχήμα μηνυμάτων. Τα περισσότερα βασισμένα στη βιομηχανία πρότυπα Ιστού αποτελούνται από έναν συνδυασμό σχημάτων μηνυμάτων και προδιαγραφών πρωτοκόλλου. Σε αυτά τα σχήματα έχει δοθεί μια λειτουργική σημασιολογία αλλά η προδιαγραφή δεν έχει ως σκοπό να υποστηρίξει το συλλογισμό έξω από τα πλαίσια αυτής της λειτουργίας.

Οι οντολογίες είναι κρίσιμες για τις εφαρμογές που θέλουν να αναζητήσουν ή να συγχωνεύσουν πληροφορίες από διαφορετικές κοινότητες. Αν και τα XML DTDs και XML σχήματα είναι επαρκή για την ανταλλαγή των στοιχείων μεταξύ των συμβαλλόμενων μερών που έχουν συμφωνήσει ορισμούς εκ των προτέρων, η έλλειψη σημασιολογίας τους αποτρέπει τις μηχανές από την αξιόπιστη εκτέλεση αυτής της εργασίας όταν τους δίνονται νέα λεξιλόγια XML. Ο ίδιος όρος μπορεί να χρησιμοποιηθεί με διαφορετική έννοια σε διαφορετικές εκφράσεις, και οι διαφορετικοί όροι μπορούν να χρησιμοποιηθούν για αντικείμενα που έχουν την ίδια έννοια. Η RDF και το RDF σχήμα αρχίζει να προσεγγίζει αυτό το πρόβλημα επιτρέποντας σε απλές σημασιολογίες να συνδεθούν με όρους. Με το σχήμα RDF, μπορούν να καθοριστούν κλάσεις, που μπορούν να έχουν πολλαπλές υποκλάσεις και υπερκλάσεις, και ιδιότητες, οι οποίες μπορούν να έχουν υπο-ιδιότητες, πεδία ορισμού και σύνολα τιμών. Από αυτή την άποψη, το σχήμα RDF είναι μια απλή γλώσσα οντολογίας. Εντούτοις, προκειμένου να επιτευχθεί η λειτουργικότητα μεταξύ πολυάριθμων, αυτόνομα αναπτυγμένων και διοικούμενων σχημάτων, απαιτείται πλουσιότερη σημασιολογία. Παραδείγματος χάριν, το σχήμα RDF δεν μπορεί να διευκρινίσει ότι οι κλάσεις προσώπων και αυτοκινήτων είναι διαχωρισμένες, ή ότι ένα κουαρτέτο έχει ακριβώς τέσσερις μουσικούς ως μέλη.

Μια οντολογία που είναι συνδεδεμένη με μια ιδιαίτερη περιοχή ονομάζεται domain ontology. Η αποκαλούμενη foundation ontology ή upper ontology είναι ένα είδος οντολογίας που προσπαθεί να περιγράψει γενικές οντότητες και δεν είναι τόσο συγκεκριμένη, οπότε είναι γενικά πιο εφαρμόσιμη. Περιέχει ένα βασικό γλωσσάριο, με τους όρους του οποίου οτιδήποτε άλλο σε μια ευρεία περιοχή μπορεί και πρέπει να περιγραφεί.

Κεφάλαιο 8. Ροή Εργασίας (Business Process)

Όπως αναπτύξαμε παραπάνω, οι υπηρεσίες ιστού είναι αυτόνομες εφαρμογές που περιγράφονται από την γλώσσα WSDL, δημοσιεύονται και ανακαλύπτονται στο UDDI και επικοινωνούν μεταξύ τους μέσω του πρωτοκόλλου SOAP. Διευκολύνουν τους χρήστες στο να συνδέουν διαφορετικά κομμάτια εργασιών διαμέσου διαφορετικών επιχειρηματικών μονάδων με έναν ανεξάρτητο πλατφόρμας και γλώσσας προγραμματισμού τρόπο.

Παρόλα αυτά κανένα από τα παραπάνω πρότυπα δεν καθορίζει μια περιγραφή για τον συσχετισμό των υπηρεσιών ιστού μεταξύ τους. Έτσι, οι υπηρεσίες ιστού παραμένουν απομονωμένες και αδιαφανείς. Το να “σπάσουμε” αυτήν την απομόνωση σημαίνει να συνδέσουμε τις υπηρεσίες ιστού μεταξύ τους και να καθορίσουμε πώς διάφορες συλλογές από υπηρεσίες ιστού μπορούν να συνδυαστούν μεταξύ τους και να δημιουργήσουν μια πιο πολύπλοκη λειτουργία, μια ροή εργασίας (business process).

Οι **ροές εργασιών** γενικά είναι μια έννοια που χρησιμοποιείται στον χώρο των επιχειρήσεων εδώ και αρκετές δεκαετίες. Το συγκεκριμένο ζήτημα αποτελεί πεδίο διαρκούς έρευνας και αφορά τόσο τις οικονομικές όσο και τις τεχνολογικές επιστήμες γενικότερα.

Μια ροή εργασίας (business process) καθορίζει ένα μοντέλο και μια γραμματική για την περιγραφή της συμπεριφοράς μιας ροής εργασίας. Καθορίζει τη σειρά εκτέλεσης των λειτουργιών σε μια συλλογή από υπηρεσίες ιστού, τα δεδομένα που ανταλλάσσονται μεταξύ τους, ποιοι συνέταιροι παίρνουν μέρος και πώς εμπλέκονται μέσα στη διαδικασία. Η αλληλεπίδραση μεταξύ κάθε συνεταίρου λαμβάνει χώρα μέσω των web services interfaces και

η δομή της σχέσης μεταξύ του συνεταιίρου και μιας διαδικασίας ενσωματώνεται σε ένα partner link.

Η ανάγκη για αναπαράσταση ρών εργασιών σε εφαρμογές με δικτυακές υπηρεσίες, όπως φάνηκε από την επιτυχία του RosettaNet, οδήγησε στη δημιουργία πολλών άλλων XML λεξιλογίων και προτύπων δικτυακών υπηρεσιών που αφορούν αυτά τα ζητήματα. Οι δύο πιο σημαντικές συμβολές είναι η Business Process Modeling Language και η Business Process Execution Language for Web Services.

Η **Business Processes Modeling Language** ή **BPML** αναπαριστά επιχειρησιακές διαδικασίες με τη βοήθεια μιας μεταγλώσσας βασισμένης στην XML. Η BPML ορίζει πάνω στις επιχειρησιακές διαδικασίες τις εξής λειτουργίες: ρή ελέγχου, ρή δεδομένων και ρή γεγονότων, ενώ παράλληλα προσθέτει και δυνατότητες για τον ορισμό επιχειρησιακών ρών, ρών ασφαλείας και ανταλλαγής μηνυμάτων. Σκοπός της BPML είναι επίσης να αναπτύξει κατάλληλο γραφικό περιβάλλον και γλώσσα ερωτήσεων.

Κεφάλαιο 9. BPEL (Business Process Execution Language)

Για να εκλείψουν πιθανά προβλήματα από την παραπάνω επικοινωνία, σημαντικές εταιρίες τεχνολογίας (όπως η IBM, η BEA, η SAP κλπ) δημιούργησαν την Business Process Execution Language for Web Services (BPEL4WS ή για συντομία BPEL) που ορίζει τη σημειολογία που πρέπει να χρησιμοποιείται για τον καθορισμό της συμπεριφοράς επιχειρησιακών υπηρεσιών, οι οποίες έχουν υλοποιηθεί σε υπηρεσίες ιστού. Η ιδέα πίσω από την BPEL4WS είναι ο συνδυασμός ενός ισχυρού τρόπου διαχείρισης επιχειρησιακών λειτουργιών με την ευελιξία και προσαρμοστικότητα των υπηρεσιών ιστού.

Ως γλώσσα που υλοποιεί εκτελέσιμες διαδικασίες, η BPEL ουσιαστικά πρέπει να καθορίζει μια καινούργια υπηρεσία ιστού συνθέτοντας ένα σύνολο υπαρχόντων υπηρεσιών ιστού. Έτσι η BPEL είναι βασικά η γλώσσα που μπορεί να υλοποιήσει αυτή τη σύνθεση. Η σύνθεση περιγράφεται από ένα σύνολο WSDL portTypes, όπως και κάθε άλλη υπηρεσία ιστού, και ονομάζεται ροή (process).

Η BPEL process καθορίζει επίσης πώς διευθετούνται οι διάφορες αλληλεπιδράσεις μεταξύ των συνεταίρων και των υπηρεσιών για να επιτύχουν ένα σκοπό. Ακόμη εισάγει μηχανισμούς για τον χειρισμό και την επεξεργασία των λαθών και τέλος εισάγει έναν μηχανισμό επανόρθωσης όταν συμβεί μια εξαίρεση ή όταν ο συνétairos ζητήσει αναίρεση των ενεργειών.

9.1. Υλοποίηση της ροής εργασίας

Αντίθετα με την παραδοσιακή γλώσσα υλοποίησης μιας WSDL υπηρεσίας, κάθε operation ενός portType δεν αντιστοιχεί σε ξεχωριστό κομμάτι λογικής

στην BPEL. Αντιθέτως, όλο το σύνολο των PortTypes της υπηρεσίας υλοποιείται από μια μοναδική BPEL διαδικασία. Έτσι, συγκεκριμένα σημεία εισόδου, με τα οποία μπορούν οι χρήστες εξωτερικά να καλούν τις operations της διαδικασίας, υποδεικνύονται μέσα στη ροή. Σε αυτά τα σημεία εισέρχονται ή εξέρχονται τα διάφορα μηνύματα των αντίστοιχων λειτουργιών μόνο-εισόδου ή εισόδου-εξόδου. Οι λειτουργίες μόνο-εξόδου (notification) και εξόδου-εισόδου(solicit-response) δεν απαιτούνται και δεν υποστηρίζονται.

9.1.1. Activities

Η BPEL διαδικασία είναι βασικά ένα διάγραμμα ροής. Κάθε βήμα στην διαδικασία ονομάζεται ενέργεια (activity). Υπάρχουν κάποιες βασικές ενέργειες, όπως:

- ✚ <invoke> : κλήση μιας λειτουργίας
- ✚ <receive> : αναμονή ενός μηνύματος να σταλεί από έξω
- ✚ <reply> : αποστολή της απάντησης από μια λειτουργία εισόδου-εξόδου
- ✚ <wait> : αναμονή για ένα χρονικό διάστημα
- ✚ <assign> : αντιγραφή δεδομένων από ένα σημείο σε ένα άλλο
- ✚ <throw> : υποδεικνύεις ότι έγινε κάποιο λάθος
- ✚ <terminate> : τερματισμός ολόκληρης της ροής
- ✚ <empty> : μην κάνεις τίποτα

Αυτές οι βασικές ενέργειες μπορούν να συνδυαστούν σε πιο περίπλοκους αλγόριθμους για να υλοποιήσεις την όλη ροή. Έτσι μπορούμε να δημιουργήσουμε

- ✚ μια ακολουθία από υπηρεσίες ιστού που θα εκτελούνται στη σειρά με τη βοήθεια της ενέργειας <invoke>
- ✚ διακλαδώσεις ροών με τη βοήθεια της ενέργειας <switch>
- ✚ βρόχους με το <while>
- ✚ επιλογή εκτέλεσης μιας ροής ανάμεσα σε πολλές με τη βοήθεια του <pick>

και τέλος

- ✚ παράλληλη ροή εργασιών με τη βοήθεια του <flow>. Στην παράλληλη ροή εργασιών μπορεί μια εργασία να καθορίζει πότε θα ξεκινήσει η επόμενη με τη βοήθεια των **links**.

9.1.2. Partners (συνέταιροι) : Αλληλεπιδρώντας με άλλους

Ως γλώσσα που συνθέτει ένα σύνολο από υπηρεσίες ιστού σε μία, η BPEL περιέχει κλήσεις σε άλλες υπηρεσίες ή δέχεται κλήσεις από τους πελάτες, που είναι οι χρήστες που στέλνουν μηνύματα.

Η κλήση σε άλλες υπηρεσίες γίνεται με τη βοήθεια του <invoke> και η αποδοχή κλήσεων με τη βοήθεια του <receive>. Η BPEL ονομάζει τις υπηρεσίες με τις οποίες αλληλεπιδρά συνεταίρους (partners).

Έτσι, ένας συνέταιρος μπορεί να είναι ένα από τα παρακάτω τρία:

- ✚ υπηρεσίες τις οποίες καλεί η ροή
- ✚ υπηρεσίες οι οποίες καλούν τη ροή
- ✚ ή υπηρεσίες τις οποίες καλεί η ροή και οι οποίες καλούν τη ροή (οποιοδήποτε από τα δύο μπορεί να συμβεί πρώτο)

9.1.3. Service link types

Το να περιγράψουμε τη σχέση του τρίτου είδους συνεταιίρου, θα πρέπει να τη δούμε είτε από την πλευρά των υπηρεσιών που καλούν τη ροή είτε από την πλευρά της ροής που καλεί τις υπηρεσίες. Για να το αποφύγουμε αυτό, με τη βοήθεια των service link types, μπορούμε να περιγράψουμε τη σχέση μεταξύ των δύο από ένα τρίτο πρόσωπο. Ένα service link type περιγράφει ένα σύνολο από ρόλους, όπου κάθε ρόλος αντιστοιχεί σε ένα σύνολο από portTypes. Η κεντρική ιδέα είναι όταν αλληλεπιδρούν δύο υπηρεσίες, τότε ο service link type καθορίζει πώς αυτές αλληλεπιδρούν, ιδιαίτερα τι υπηρεσίες προσφέρει κάθε πλευρά.

Η BPEL χρησιμοποιεί τους service link types για να καθορίσει συνεταιίρους. Για να ορίσουμε ένα συνέταιρο πρέπει να του δώσουμε ένα όνομα, μετά να προσδιορίσουμε σε ποιο service link type αναφέρεται και τέλος να καθορίσουμε το ρόλο που θα παίξει η ροή BPEL και το ρόλο που θα έχει η υπηρεσία ιστού.

9.1.4. Πώς αναφέρεται ο partner στην ροή εργασίας

Για να δουλέψει ένας συνέταιρος κατά την εκτέλεση, θα πρέπει να αναφέρεται σε μια πραγματική υπηρεσία ιστού. Για την ακρίβεια ο συνέταιρος είναι απλώς μια γραπτή αναφορά σε μια υπηρεσία (υπηρεσία ιστού). Η λέξη 'γραπτή' αναφέρεται στο ότι περιγράφουμε δύο στοιχεία για έναν συνέταιρο, ένα "<partnerLinkType>" και ένα "<role>" για να αναφέρουμε ποια υπηρεσία μπορεί να καλέσει και με ποια ιδιότητα.

9.1.5. Διάρκεια ζωής των ρών εργασίας

Όταν οι υπηρεσίες ιστού συνδυάζονται σε ροές εργασίας τότε η διάρκεια ζωής τους είναι στιγμιαία. Αυτό σημαίνει ότι ο πελάτης κάθε υπηρεσίας αλληλεπιδρά με ένα στιγμιότυπο της ροής εργασίας.

Αυτά τα στιγμιότυπα κάθε ροής εργασίας, δημιουργούνται σιωπηρά με την άφιξη μηνυμάτων για την ροή εργασίας. Δηλαδή, τα στιγμιότυπα αυτά δεν προσδιορίζονται από κάποιο 'instanceID', αλλά από κάποιες τιμές κλειδιά που περιλαμβάνονται μέσα στα μηνύματα. Για παράδειγμα αν μια ροή εργασίας περιγράφει την εκτέλεση μιας παραγγελίας, το νούμερο της παραγγελίας μπορεί να είναι το κλειδί για να προσδιορίσει το συγκεκριμένο στιγμιότυπο. Έτσι όταν σταλεί ένα μήνυμα για να εκκινήσει μια ροής εργασίας και δεν υπάρχει κάποιο στιγμιότυπο που να ταιριάζει με την τιμή-κλειδί που μεταφέρει το μήνυμα, τότε δημιουργείται ένα καινούργιο στιγμιότυπο το οποίο αυτόματα ταυτίζεται με την τιμή-κλειδί του μηνύματος.

Τα μηνύματα που αποστέλλονται σε μια ροή εργασίας είτε δημιουργούν ένα νέο στιγμιότυπο είτε παραλαμβάνονται εφόσον εντοπιστεί ένα υπάρχον κατάλληλο στιγμιότυπο, το οποίο περιμένει να λάβει κάποιο μήνυμα σε κάποιο άλλο σημείο μέσα στη ροή, εκτός του σημείου εκκίνησης. Η διαδικασία του να δημιουργήσουμε ένα νέο στιγμιότυπο ή του να εντοπίζουμε ένα κατάλληλο στιγμιότυπο για να παραδοθεί το μήνυμα ονομάζεται 'συσχέτιση μηνυμάτων'.

9.1.6. Εκκίνηση της ροής εργασίας

Μια ροή εργασίας μπορεί να εκκινηθεί όταν αποσταλεί σε αυτήν ένα μήνυμα. Για να παραλάβει αυτό το μήνυμα θα πρέπει να περιέχει μια πρώτη activity <receive>, η οποία θα περιλαμβάνει την ιδιότητα 'createInstance' με τιμή 'true'.

9.2. Προβλήματα ρών εργασίας

Κάποια από τα προβλήματα που μπορούν να εμφανιστούν είναι τα παρακάτω:

1. Πολλές επανεπεξεργασίες και επαναλήψεις εργασιών.

Στην περίπτωση αυτή επαναλαμβάνονται εργασίες που έχουν ήδη γίνει (π.χ. καταχώρηση που διορθώνεται ή συμπληρώνεται από την αρχή) λόγω αστοχίας.

Κύρια αιτία του παραπάνω συμπτώματος είναι συνήθως η έλλειψη έγκαιρης προειδοποίησης (ανάδρασης) στα στάδια μιας μακράς ροής εργασιών. Τα προβλήματα δεν αναγνωρίζονται στο στάδιο που δημιουργούνται, αλλά σε μεταγενέστερα στάδια, με αποτέλεσμα να απαιτείται επανάληψη των ενδιάμεσων σταδίων προκειμένου να διορθωθούν. Στόχος της αναδιοργάνωσης δεν είναι η αποτελεσματικότερη διαχείριση των επανεπεξεργασιών, αλλά η απάλλειψή τους.

2. *Υπαρξη πολλών ειδικών περιπτώσεων.*

Μπορούμε να σχεδιάσουμε ροές εργασιών σχετικά απλές. Η πολυπλοκότητά τους, όμως, αυξάνεται με την προσθήκη υποδιαδικασιών για την αντιμετώπιση εξαιρέσεων και ειδικών περιπτώσεων.

Κύρια αιτία αυτού του συμπτώματος είναι η ιστορική αγκύκλωση. Οι διαδικασίες αυτές μπορούν να απλοποιηθούν σημαντικά αν αποκατασταθεί η ροή εργασιών στην αρχική βασική της έκδοση, στην απλή της μορφή. Όσες από τις ειδικές περιπτώσεις είναι απαραίτητες μπορούν να αποτελέσουν χωριστές ροές εργασιών, ώστε με τον τρόπο αυτό, να μην προσθέτουν πολυπλοκότητα στη βασική διεργασία.

9.2.1. Αντιμετωπίζοντας τα προβλήματα και τα λάθη κατά την εκτέλεση

Οι άνθρωποι που σχεδιάζουν ροές εργασίας, θα πρέπει να έχουν κάποιο τρόπο να χειρίζονται τα λάθη που συμβαίνουν κατά την εκτέλεση της ροής εργασίας. Η γλώσσα BPEL έχει ενσωματωμένες κάποιες ενέργειες για να ρίχνει και να πιάνει εξαιρέσεις. Αυτό γίνεται μέσω των “throw” και “catch” στοιχείων.

Επιπλέον, η BPEL υποστηρίζει την έννοια της 'επανόρθωσης' (compensation), η οποία βοηθάει τον σχεδιαστή να υλοποιήσει εναλλακτικές πράξεις για κάποιες μη-αντιστρέψιμες ενέργειες. Για παράδειγμα, ας σκεφτούμε μια διαδικασία κράτησης ταξιδιωτικού εισιτηρίου. Από τη στιγμή που γίνει η κράτηση, που είναι μια πράξη η οποία δεν μπορεί να αντιστραφεί, για να γίνει η ακύρωση θα πρέπει να γίνουν κάποιες άλλες διαφορετικές ενέργειες. Αυτές οι ενέργειες καλούνται 'εναλλακτικές ενέργειες' για την ενέργεια της κράτησης.

Ο χειρισμός των λαθών και των εναλλακτικών ενεργειών υποστηρίζεται από την έννοια του 'score'. Το στοιχείο αυτό περιγράφει το περιεχόμενο κάθε ενέργειας. Περιλαμβάνει χειρισμούς λαθών, χειρισμούς επανορθώσεων, ορισμούς μεταβλητών που δέχονται δεδομένα, ορισμούς 'partner links' και ορισμούς συσχετίσεων (correlation sets).

9.3. Παράδειγμα ροής εργασίας

Θα παρουσιάσουμε ένα απλό παράδειγμα για να κατανοήσουμε πώς οι διάφορες ενέργειες μπορούν να δημιουργηθούν και να συνδυαστούν με την BPEL. Το παράδειγμα αναφέρεται στην εκτέλεση της διαδικασίας που ακολουθείται για την λήψη ενός δανείου. Θα παρουσιάσουμε τις βασικές ενέργειες που θα χρησιμοποιηθούν στη ροή εργασίας, μαζί με το πώς οι WSDL περιγραφές των υπηρεσιών ιστού σχετίζονται και θα χρησιμοποιηθούν στην όλη διαδικασία. Καθώς θα δημιουργούμε την ροή εργασίας θα εξηγήσουμε παράλληλα και την χρήση των συνεταίρων για την κλήση των υπηρεσιών, τη χρήση των variables οι οποίες δέχονται τα μηνύματα που ανταλλάσσονται και τέλος τη χρήση των ενεργειών (activities) με τις οποίες θα επικοινωνούμε με τον έξω κόσμο, οι οποίες είναι οι <receive>, <invoke>, <reply>.

Πιο συγκεκριμένα στο παράδειγμά μας ένας πελάτης στέλνει μια αίτηση για δάνειο. Η αίτηση επεξεργάζεται και ο πελάτης παίρνει απάντηση αν το δάνειο εγκρίθηκε ή όχι. Η επεξεργασία της αίτησης του πελάτη περιλαμβάνει το να σταλεί αυτή σε έναν οργανισμό ο οποίος παρέχει την υπηρεσία της

επεξεργασίας της αίτησης του δανείου και στέλνει πίσω την απάντηση. Ο πελάτης από την πλευρά του βλέπει μόνο ότι η διαδικασία θα λάβει την αίτησή του και θα του επιστρέψει την απάντηση.

Αν το ποσό δανείου είναι υψηλό τότε σίγουρα η αίτηση στέλνεται στον οργανισμό για να αποφασίσει. Αν όμως το ποσό δανείου είναι χαμηλό, τότε καλούμε μια νέα υπηρεσία ιστού, που ονομάζεται 'loan assessor', για να καθορίσει το ρίσκο του πελάτη. Αν ο εκτιμητής δανείου αποφασίσει ότι το ρίσκο είναι χαμηλό αν δώσει το δάνειο, τότε εγκρίνει την αίτηση, αλλιώς τη στέλνει στον οργανισμό για να αποφασίσει συνολικά.

9.3.1. Δομή της ροής εργασίας

Η ροή εργασίας για να εκκινηθεί, περιμένει να λάβει ένα μήνυμα, κατόπιν θα καλέσει την υπηρεσία του οργανισμού που θα επεξεργαστεί την αίτηση και τέλος θα στείλει την απάντηση πίσω στον πελάτη. Οι τρεις αυτές ενέργειες περιγράφονται στην BPEL με τις ενέργειες <receive>, <invoke>, <reply>. Χρειάζεται όμως επιπλέον να περιγράψουμε πώς σχετίζονται αυτές οι ενέργειες μεταξύ τους για να ξέρει η διαδικασία πώς να τις εκτελέσει. Το πώς σχετίζονται οι ενέργειες περιγράφεται στην BPEL με κάποιες ενέργειες οι οποίες επιβάλλουν περιορισμούς στο πώς θα κληθούν οι ενέργειες τις οποίες περιλαμβάνουν. Στο παράδειγμά μας, θέλουμε οι τρεις ενέργειες να γίνουν διαδοχικά, η μία μετά την άλλη. Αυτή η διαδοχή, μπορεί να επιτευχθεί με τη βοήθεια της ενέργειας <sequence>, η οποία θα περιείχε πρώτα την <receive> για να λάβει το μήνυμα, έπειτα την <invoke> για να καλέσει την υπηρεσία και τέλος την <reply> για να στείλει την απάντηση στο χρήστη.

9.3.2. Περιγραφή των υπηρεσιών ιστού που θα συνεργαστούν για τη ροή εργασίας

Η σύνθεση της ροής εργασίας βασίζεται πολύ στις WSDL περιγραφές των υπηρεσιών ιστού οι οποίες συμμετέχουν, με σκοπό να μπορεί να αναφερθεί στα μηνύματα που αυτές δέχονται και αποστέλλουν, στις μεθόδους (operations) που μπορεί να καλέσει και σε ποια interfaces (PortTypes) αυτές

ανήκουν. Στο παράδειγμά μας θα χρειαστούμε την περιγραφή της υπηρεσίας ιστού του οικονομικού οργανισμού που επεξεργάζεται την αίτηση για να δημιουργήσουμε τη ροή εργασίας. Ας υποθέσουμε ότι για να κάνεις την αίτηση δανείου παρέχεις όνομα, επώνυμο και το ποσό που επιθυμείς. Αυτές οι παράμετροι ενσωματώνονται σε ένα μήνυμα το οποίο ορίζεται στην περιγραφή της υπηρεσίας ιστού του οργανισμού (το `creditInformationMessage`) και επιστρέφεται ένα μήνυμα τύπου ακεραίου με όνομα `loanRequestErrorMessage`.

```
<definitions
targetNamespace="http://tempuri.org/services/loanassessor"
xmlns:tns="http://tempuri.org/services/loanassessor"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="http://schemas.xmlsoap.org/wsdl/">

  <message name="creditInformationMessage">
    <part name="firstName" type="xsd:string"/>
    <part name="name" type="xsd:string"/>
    <part name="amount" type="xsd:integer"/>
  </message>

  <message name="loanRequestErrorMessage">
    <part name="errorCode" type="xsd:integer"/>
  </message>

</definitions>
```

Πίνακας 5. Loan definitions wsdl

Ας υποθέσουμε ότι ο οικονομικός οργανισμός παρέχει μια υπηρεσία ιστού για την έγκριση του δανείου. Στην παρακάτω περιγραφή αυτής της υπηρεσίας ιστού, περιέχεται η μοναδική μέθοδος (operation), με όνομα "approve", η οποία αποφασίζει αν θα εγκριθεί το δάνειο ή όχι. Η operation παίρνει την πληροφορία του πελάτη σαν είσοδο και επιστρέφει την απάντηση σε ένα μήνυμα έγκρισης ή άρνησης, το οποίο περιγράφηκε στην παραπάνω περιγραφή.

```
<definitions
targetNamespace="http://tempuri.org/services/loanapprover"
xmlns:tns="http://tempuri.org/services/loanapprover"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:loandef="http://tempuri.org/services/loandefinitions"
xmlns="http://schemas.xmlsoap.org/wsdl/">

  <import
    namespace="http://tempuri.org/services/loandefinitions"
    location="http://localhost:8080/bpws-
samples/loanapproval/loandefinitions.wsdl"/>

  <message name="approvalMessage">
    <part name="accept" type="xsd:string"/>
  </message>

  <portType name="loanApprovalPT">
    <operation name="approve">
      <input message="loandef:creditInformationMessage"/>
      <output message="tns:approvalMessage"/>
      <fault name="loanProcessFault"
        message="loandef:loanRequestErrorMessage"/>
    </operation>
  </portType>

  <binding...>...</binding>
  <service name="LoanApprover">...</service>

</definitions>
```

Πίνακας 6. Loan Approver WSDL

Επιπλέον, πρέπει να ορίσουμε τα στοιχεία `partnerLinkTypes`, τα οποία καθορίζουν με ποιους ρόλους μπορούμε να καλέσουμε την υπηρεσία. Το στοιχείο `partnerLinkType` προσδιορίζει το πολύ μέχρι δύο ρόλους και αναφέρεται στα διαθέσιμα `PortTypes` (interfaces) των υπηρεσιών που συνδέονται μεταξύ τους. Μην ξεχνάμε ότι και η ροή εργασίας αποτελεί μια νέα υπηρεσία και αυτή. Άρα το να συνδέσουμε τη ροή εργασίας με μια υπηρεσία

ιστού αποτελεί και αυτό ένα partner link μεταξύ των δύο υπηρεσιών. Στο παράδειγμά μας, το partnerLinkType θα χρησιμοποιηθεί για να συνδέσει τον πελάτη με τη ροή εργασίας, καθώς επίσης και την διαδικασία με την υπηρεσία ιστού της επεξεργασίας του δανείου. Έχουμε δηλαδή δύο ρόλους, αλλά επειδή η ροή εργασίας και η υπηρεσία ιστού της επεξεργασίας του δανείου παρέχουν το ίδιο “approver” portType (interface) και καμιά δεν χρειάζεται να χρησιμοποιήσει άλλο portType, χρειάζεται τελικά μόνο ο ένας ρόλος.

```
<definitions
  targetNamespace="http://loans.org/wsd1/loan-approval"
  xmlns="http://schemas.xmlsoap.org/wsd1/"
  xmlns:slnk="http://schemas.xmlsoap.org/ws/2002/06/service-link/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:lns="http://loans.org/wsd1/loan-approval"
  xmlns:apns="http://tempuri.org/services/loanapprover">
  <import namespace="http://tempuri.org/services/loanapprover"
    location="http://localhost:8080/bpws-
samples/loanapproval/loanapprover.wsdl"/>
  <import
namespace="http://tempuri.org/services/loandefinitions"
  location="http://localhost:8080/bpws-
samples/loanapproval/loandefinitions.wsdl"/>
  <slnk:serviceLinkType name="loanApprovalLinkType">
    <slnk:role name="approver">
      <portType name="apns:loanApprovalPT"/>
    </slnk:role>
  </slnk:serviceLinkType>
  <service name="loanapprovalServiceBP"/>
</definitions>
```

Πίνακας 7. Loan Approval WSDL

9.3.3. Δημιουργία της ροής εργασίας

Όλες οι προαπαιτήσεις μας είναι τώρα διαθέσιμες και μπορούμε να προχωρήσουμε στην κατασκευή της ροής εργασίας. Ξεκινάμε την περιγραφή

με το στοιχείο <process> και αναφέρουμε και τα namespaces τα οποία θα χρειαστούν για να αναφερθούμε στα αντίστοιχα WSDL που περιγράφουν τις υπηρεσίες και τα μηνύματα που θα χρησιμοποιήσουμε. Τα namespaces που θα μας χρειαστούν είναι το target namespace της web service loan approver (http://.../loanapprover), το target namespace των partnerLinkTypes (http://.../loan-approval) και το target namespace των μηνυμάτων (http://.../loandefinitions). Η ροή εργασίας είναι τώρα έτοιμη να χρησιμοποιήσει την υπηρεσία ιστού σαν εξάρτημά της.

```
<process name="loanApprovalProcess"
  targetNamespace="http://acme.com/simpleloanprocessing"
  xmlns="http://schemas.xmlsoap.org/ws/2002/07/business-
process/"
  xmlns:lns="http://loans.org/wsd/loan-approval"
  xmlns:loandef="http://tempuri.org/services/loandefinitions"
  xmlns:apns="http://tempuri.org/services/loanapprover">
```

Πίνακας 8. Namespaces ροής εργασίας

Το επόμενο βήμα είναι να καθορίσουμε τους συμμετέχοντες στη ροή εργασίας. Τα ονόματα των συνεταιίρων και σε ποιο partnerLinkType ανήκουν καθορίστηκαν στην loan-approval.wsd/ περιγραφή. Για το παράδειγμά μας, οι συνέταιροι είναι ο πελάτης και ο οργανισμός που παρέχει το δάνειο. Η ιδιότητα myRole/partnerRole του στοιχείου <partner> καθορίζει πώς οι συνέταιροι και η ροή εργασίας θα αλληλεπιδράσουν δεδομένου του partnerLinkType. Η ιδιότητα myRole αναφέρεται στον ρόλο του partnerLinkType που θα παίξει η ροή εργασίας, ενώ η ιδιότητα partnerRole αναφέρεται στο ρόλο που θα παίξει ο κάθε συνέταιρος. Αυτά καθορίζονται στον ορισμό του partner.

```
<partners>
  <partner
    name="customer"
    serviceLinkType="lns:loanApproveLinkType"
    myRole="approver"/>
```

```
<partner
  name="approver"
  serviceLinkType="lns:loanApprovalLinkType"
  partnerRole="approver" />

</partners>
```

Πίνακας 9. Καθορισμός των partners

Η ροή εργασίας προσφέρει το loanApprovalPT interface στον πελάτη και ο οικονομικός οργανισμός προσφέρει αυτό το interface στη ροή εργασίας.

Αφού καθορίσουμε τους συνεταιίρους, είμαστε σχεδόν έτοιμοι να προσθέσουμε τις ενέργειες που θα απαρτίσουν τη ροή εργασίας. Ανακεφαλαιώνοντας, θα πρέπει να κάνουμε τα παρακάτω:

Για να ζητήσει ο πελάτης ένα δάνειο, θα πρέπει να στείλει ένα μήνυμα στη ροή εργασίας, η ροή εργασίας θα ζητήσει από τον οργανισμό αν δέχεται την αίτηση ή όχι και θα απαντήσει στον πελάτη στέλνοντάς του ένα άλλο μήνυμα αποδοχής ή άρνησης της αίτησής του. Πώς γίνεται αυτό στην BPEL; Πρώτα θα πρέπει να τοποθετήσεις το μήνυμα που θα έρθει από τον πελάτη σε έναν υποδοχέα (variable) τον οποίο μπορεί να προσπελάσει η ροή εργασίας. Στην BPEL, τα δεδομένα γράφονται και προσπελούνται από τους υποδοχείς δεδομένων, οι οποίοι μπορούν να κρατούν στιγμιότυπα των συγκεκριμένων τύπων μηνυμάτων που περιγράφονται στην WSDL.

Από τον ορισμό του customer partner και του loanApprovalPT, μπορούμε να καταλάβουμε ότι ο πελάτης θα στείλει ένα μήνυμα τύπου creditInformationMessage και θα λάβει μια απάντηση τύπου approvalMessage. Έτσι σχηματίζουμε και το σύνολο των υποδοχέων (variables).

```
<variables>
  <variable
    name="request" />
</variables>
```



```
messageType="loandef:CreditInformationMessage" />  
  
<variable  
name="approvalInfo"  
messageType="apns:approvalMessage" />  
  
</variables>
```

Πίνακας 10. Καθορισμός των variables

Επεκτείνουμε τη loan approval διαδικασία, προσθέτοντας links, conditions και assign, προσθέτοντας μια ακόμα ενέργεια. Τα Links συνδέουν δυο ενέργειες μεταξύ τους και επιτρέπουν η μία να καθορίζει αν θα εκτελεστεί μη άλλη. Οι conditions είναι εκφράσεις XPath και δηλώνουν πώς μπορούν να χειριστούν κάποιες συνθήκες σχετικά με τις τιμές κάποιων δεδομένων. Η <assign> ενέργεια δείχνει πώς μπορούμε να ανατιγράψουμε δεδομένα από μια μεταβλητή σε μια άλλη.

Στην BPEL αυτή η ροή εργασίας μοντελοποιείται χρησιμοποιώντας την ενέργεια <flow>, για να αποφασίσουμε ποιες ενέργειες θα γίνουν παράλληλα και υπό ποιες συνθήκες θα γίνουν ορισμένες. Αφήνουμε τα <receive> και <invoke> του προηγούμενου παραδείγματος όπως είναι και προσθέτουμε δύο <invoke>s –ένα για τον assessor κι ένα για τον approver. Προσθέτουμε και μια ενέργεια <assign> για να αντιγράψουμε το μήνυμα στο reply. Έπειτα συνδέουμε το <receive> με δύο links που προσδιορίζονται από δύο συνθήκες: αν το ποσό είναι λιγότερο από 10,000 τότε καλούμε τον assessor, αλλιώς τον approver. Στη συνέχεια συνδέουμε το <invoke> του assessor με αυτό του approver με την συνθήκη ότι το ρίσκο είναι υψηλό και με το <assign> με τη συνθήκη ότι το ρίσκο είναι χαμηλό. Τέλος, συνδέουμε το invoke του approver και το <assign> με το reply, χωρίς να προσδιορίζουμε κάποια συνθήκη. Η ροή που θα ακολουθηθεί θα είναι σύμφωνη με τις συνθήκες που θα ισχύουν στα links.

Η WSDL περιγραφή του loan assessor μπορεί να εκτελέσει μόνο μια ενέργεια, την 'check', η οποία επιστρέφει το επίπεδο ρίσκου.

```
<definitions
  targetNamespace="http://tempuri.org/services/loanassessor"
  xmlns:tns="http://tempuri.org/services/loanassessor"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:loandef="http://tempuri.org/services/loandefinitions"
  xmlns="http://schemas.xmlsoap.org/wsdl/">
  <import namespace="http://tempuri.org/services/loandefinitions"
    location=
      "http://localhost:8080/bpws4j-
samples/loanapproval/loandefinitions.wsdl"/>
  <message name="riskAssessmentMessage">
    <part name="risk" type="xsd:string"/>
  </message>
  <portType name="riskAssessmentPT">
    <operation name="check">
      <input message=
        "loandef:creditInformationMessage"/>
      <output message="tns:riskAssessmentMessage"/>
      <fault name="loanProcessFault" message=
        "loandef:loanRequestErrorMessage"/>
    </operation>
  </portType>
  <binding ...> ... </binding>
  <service name="LoanAssessor">...</service>
</definitions>
```

Πίνακας 11. Περιγραφή loan assessor

Προσθέτουμε ένα ακόμη partnerLinkType στον ορισμό της διαδικασίας, για να καθορίσουμε την αλληλεπίδρασή της με τον νέο συνέταιρο, τον assessor. Για να είναι κάποιος assessor θα πρέπει να έχει το riskAssessmentPT που ορίσαμε νωρίτερα.

```
<slnk:serviceLinkType name="riskAssessmentLinkType">
  <slnk:role name="assessor">
    <portType name="asns:riskAssessmentPT"/>
  </slnk:role>
```

```
</slnk:serviceLinkType>
```

Πίνακας 12. Προσθήκη partnerLinkType με όνομα assessor

Προσθέτουμε στους συνεταίρους έναν ακόμη

```
<partner name="assessor"  
  serviceLinkType="lms:riskAssessmentLinkType"  
  partnerRole="assessor" />
```

Πίνακας 13. Προσθήκη συνεταίρου

Και στους υποδοχείς μηνυμάτων έναν ακόμη για να λάβει την απάντηση του assessor:

```
<variable  
  name="riskAssessment"  
  messageType="asns:riskAssessmentMessage" />
```

Πίνακας 14. Προσθήκη μεταβλητής

Προσθέτουμε στη συνέχεια τα flows και links. Τα links έχουν μια αρχή (source) και ένα τέλος (target). Τα links έχουν πάνω τους μια συνθήκη μετάβασης, που παίρνει μια Boolean τιμή. Αν αυτή γίνει true τότε ακολουθείται το link, αλλιώς το link δεν εκτελείται.

```
<flow>  
  <links>  
    <link name="receive-to-assess" />  
    <link name="receive-to-approval" />  
    <link name="approval-to-reply" />  
    <link name="assess-to-setMessage" />  
    <link name="setMessage-to-reply" />  
    <link name="assess-to-approval" />  
  </links>  
  <receive name="receive1" partner="customer"  
    portType="apns:loanApprovalPT"  
    operation="approve" variable="request" />
```

```
        createInstance="yes">
        <source linkName="receive-to-assess"
            transitionCondition=
                "bpws:getVariableData('request',
'amount')<10000"/>
        <source linkName="receive-to-approval"
            transitionCondition=
                "bpws:getVariableData('request',
'amount')>=10000"/>
    </receive>
    <invoke name="invokeAssessor" partner="assessor"
        portType="asns:riskAssessmentPT"
        operation="check" inputVariable="request"
        outputVariable="riskAssessment">
        <target linkName="receive-to-assess"/>
        <source linkName="assess-to-setMessage"
            transitionCondition=
                "bpws:getVariableData('riskAssessment',
'risk')='low'"/>
        <source linkName="assess-to-approval"
            transitionCondition=
                "bpws:getVariableData('riskAssessment',
'risk')!='low'"/>
    </invoke>
```

Πίνακας 15. Προσθήκη flows και link

Αν προσέξουμε την συνθήκη μετάβασης `bpws:getVariableData('request', 'amount')<10000` θα δούμε ότι το 'request' είναι ο υποδοχέας ενός μηνύματος τύπου `loandef:CreditInformationMessage` το οποίο περιέχει τη μεταβλητή 'amount' τύπου `integer`. Άρα η συνθήκη αυτή εκτιμά την τιμή της μεταβλητής `amount` και αν είναι μικρότερη του 10000 τότε θα επιστρέψει `true` και θα ενεργοποιηθεί το link `receive-to-assess`.

Ας προσέξουμε επίσης και τη συνθήκη `bpws:getVariableData('riskAssessment', 'risk')='low'` η οποία ενεργοποιείται αν το ρίσκο είναι χαμηλό.

Αυτό που πρέπει να σταλεί πίσω στον πελάτη είναι η απάντηση, άρα θα χρησιμοποιήσουμε την ενέργεια <assign>:

```
<assign name="assign">
  <target linkName="assess-to-setMessage"/>
  <source linkName="setMessage-to-reply"/>
  <copy>
    <from expression="'yes'"/>
    <to variable="approvalInfo" part="accept"/>
  </copy>
</assign>
```

Πίνακας 16. Ενέργεια assign

Τέλος, προσθέτουμε τα <invoke> και <reply> όπως τα είχαμε και πριν.

```
<invoke name="invokeapprover"
  partner="approver" portType="apns:loanApprovalPT"
  operation="approve"
  inputVariable="request"
  outputVariable="approvalInfo">
  <target linkName="receive-to-approval"/>
  <target linkName="assess-to-approval"/>
  <source linkName="approval-to-reply" />
</invoke>
<reply name="reply" partner="customer"
  portType="apns:loanApprovalPT"
  operation="approve" variable="approvalInfo">
  <target linkName="setMessage-to-reply"/>
  <target linkName="approval-to-reply"/>
</reply>
</flow>
```

Πίνακας 17. Invoke και reply

Πρέπει να ξεκαθαρίσουμε τι γίνεται αν μια συνθήκη μετάβασης επιστρέψει false. Τότε αυτό θεωρείται ως λάθος στην BPEL και η ενέργεια που έχει αυτή τη συνθήκη ρίχνει ένα joinFailure λάθος, το οποίο αν δεν πιαστεί, μπλοκάρει όλη τη ροή εργασίας. Για να το αποτρέψουμε αυτό θα πρέπει να ορίσουμε μια

καθολική ιδιότητα `suppressJoinFailure` στο στοιχείο `<process>`. Αυτό θα σταματήσει την επέκταση των `joinExceptions` και σε άλλα σημεία.

9.3.4. Αλληλεπιδρώντας με τη ροή εργασίας

Η ροή εργασίας μπορεί να περιέχει μόνο μια ενέργεια, στην προκειμένη περίπτωση την `<sequence>`. Στην `<sequence>` μπορούμε να προσθέσουμε την ενέργεια `<receive>` η οποία θα περιμένει να λάβει από τον πελάτη το μήνυμα και να το τοποθετήσει στον αντίστοιχο υποδοχέα. Ο ορισμός της ενέργειας `<receive>` πρέπει να περιλαμβάνει το όνομα του συνεταίρου ο οποίος θα στείλει το μήνυμα, μαζί με το όνομα του `portType` και της `operation` στα οποία θέλει να στείλει το μήνυμα ο συνétaιρος. Σύμφωνα με αυτήν την πληροφορία, μόλις η ροή εργασίας λάβει το μήνυμα, θα ψάξει να βρει μια `<receive>` ενέργεια η οποία περιμένει ένα μήνυμα με την ίδια τριάδα `partner-portType-operation` και της παραδίδει το μήνυμα. Για να αποφευχθεί σύγχυση, η προδιαγραφή της BPEL δηλώνει ότι δεν πρέπει να υπάρχουν παραπάνω από μία `<receive>` ενέργειες που αντιστοιχούν στην ίδια τριάδα και οι οποίες δεν έχουν λάβει περιμένουν ταυτόχρονα να λάβουν το ίδιο μήνυμα. Η `<receive>` ενέργεια στη συνέχεια τοποθετεί το μήνυμα στον συγκεκριμένο υποδοχέα και ολοκληρώνεται. Ξεκινάμε λοιπόν με την ενέργεια `<sequence>` και προσθέτουμε την ενέργεια `<receive>` σε αυτήν.

```
<sequence>
  <receive name="receive1" partner="customer"
    portType="apns:loanApprovalPT"
    operation="approve" variable="request"
    createInstance="yes">
</receive>
```

Πίνακας 18. Ενέργεια `sequence`

Το επόμενο βήμα είναι να ρωτήσουμε την υπηρεσία ιστού του οικονομικού οργανισμού αν δέχεται την αίτηση ή όχι. Αυτό επιτυγχάνεται κάνοντας μια κλήση στην υπηρεσία ιστού με την ενέργεια `<invoke>`. Όταν τρέξει αυτή η

ενέργεια τότε θα καλέσει την υπηρεσία ιστού στέλνοντάς της το μήνυμα στον υποδοχέα εισόδου (inputVariable) της υπηρεσίας ιστού και αποθηκεύοντας την απάντηση στον υποδοχέα εξόδου (outputVariable) και ολοκληρώνεται. Μπορούμε να παρατηρήσουμε ότι την κλήση της operation “approve” θα κάνει ο partner “approver”.

```
<invoke name="invokeapprover"  
        partner="approver"  
        portType="apns:loanApprovalPT"  
        operation="approve"  
        inputContainer="request "  
        outputContainer="approvalInfo">  
  
</invoke>
```

Πίνακας 19. Ενέργεια invoke

Για να στείλει η ροή εργασίας την απάντηση στον πελάτη, χρησιμοποιεί την ενέργεια <reply>. Κάθε <reply> θα πρέπει να αντιστοιχίζεται σε ένα <receive>. Δηλαδή η τριάδα ‘partner-portType-Operation’ που περιέχει το <reply> θα πρέπει να ταυτίζεται με την τριάδα μιας <receive> ενέργειας. Το μήνυμα που θα σταλεί πίσω στον πελάτη θα περιέχει την απάντηση που έστειλε ο οικονομικός οργανισμός ενσωματωμένο στον υποδοχέα εξόδου (approvalInfo) της ενέργειας <invoke>. Μετά την ολοκλήρωση του <reply> η ροή εργασίας τελειώνει κλείνοντας τις ετικέτες <sequence> και <process>.

```
<reply  
name="reply" partner="customer"  
portType="apns:loanApprovalPT"  
operation="approve" container="approvalInfo">  
    </reply>  
</sequence>  
  
</process>
```

Πίνακας 20. Μήνυμα απάντησης reply

9.3.5. Συνολική θεώρηση της ροής εργασίας για την έγκριση του δανείου

Από τη στιγμή που δημιουργήθηκε η ροή εργασίας, θα περιμένει ένα μήνυμα για να εκτελεστεί. Αν παρατηρήσαμε σωστά, η ενέργεια <receive> περιείχε την ιδιότητα 'createInstance' με τιμή 'true'. Αυτό υποδηλώνει ένα σημείο εκκίνησης της ροής εργασίας.

Όταν ο πελάτης στείλει ένα μήνυμα στη ροή εργασίας με την συγκεκριμένη τριάδα partner-PortType-Operation τότε ένα νέο στιγμιότυπο δημιουργείται και εκτελείται. Στο παράδειγμά μας, η διαδικασία θα εκκινούσε τη <sequence>, η οποία με τη σειρά της θα εκκινούσε τη <receive>. Το μήνυμα θα ενσωματώνονταν στον υποδοχέα εισόδου της <invoke>, η οποία θα έκανε την κλήση προς την υπηρεσία ιστού και θα λάμβανε την απάντηση στον υποδοχέα εξόδου (arprovaInfo). Τέλος θα έπαιρνε αυτό το μήνυμα η <reply> και θα το έστελνε πίσω στον πελάτη και θα ολοκληρωνόταν αυτό το στιγμιότυπο της ροής εργασίας. Πολλαπλά στιγμιότυπα της ίδιας διαδικασίας μπορεί να τρέχουν ταυτόχρονα. Τα μηνύματα αυτά ξεχωρίζονται μεταξύ τους με τη βοήθεια του στοιχείου <correlation>, το οποίο χρησιμεύει για να δρομολογεί τα μηνύματα στον σωστό προορισμό τους.

Κεφάλαιο 10. Συστήματα Ηλεκτρονικής Μάθησης και Υπηρεσίες Ιστού

Όπως και με τον όρο «υπηρεσία ιστού», έτσι και με στον όρο «ηλεκτρονική μάθηση» έχουν δοθεί αρκετοί ορισμοί, οι οποίοι και μπορούν να ερμηνευτούν ανάλογα με το εκάστοτε πλαίσιο που θέλουμε να διευκρινίσουμε. Σύμφωνα με τον Derek Stockley, ηλεκτρονική μάθηση είναι η παράδοση ενός προγράμματος μάθησης ή εκπαίδευσης με ηλεκτρονικά μέσα. Η ηλεκτρονική μάθηση, αφορά στη χρήση κάποιου υπολογιστή ή άλλης μηχανής (για παράδειγμα, ενός κινητού τηλεφώνου), κατάλληλων να παρέχουν εκπαιδευτικό υλικό.

Ωστόσο, η αρχιτεκτονική των περισσότερων συστημάτων ηλεκτρονικής μάθησης παρουσιάζουν κάποια μειονεκτήματα, όπως η ανικανότητα ενσωμάτωσης νέων υπηρεσιών (lack of integration) σε υπάρχοντα συστήματα ηλεκτρονικής μάθησης και η ανικανότητα επαναχρησιμοποίησης τμήματος μαθησιακού περιεχομένου του, καθώς τα δομικά τμήματα του περιεχομένου από τα οποία αποτελούνται τα παραπάνω συστήματα, εξαρτώνται από την γλώσσα και την πλατφόρμα από την οποία υλοποιήθηκαν και δεν μπορούν να χρησιμοποιηθούν μεμονωμένα από κανέναν χρήστη. Επομένως, η αρχιτεκτονική τους δεν υποστήριζε την διαλειτουργικότητα, την επαναχρησιμοποίηση του μαθησιακού περιεχομένου ή και των δομικών τμημάτων τους, καθώς και υπήρχε ελάχιστη ευελιξία ως προς την προσαρμογή του συστήματος στις προσωπικές προτιμήσεις του εκάστοτε χρήστη.

Οι Υπηρεσίες Ιστού είναι οι πλέον αρμόδιες για εφαρμογές: οι οποίες πρέπει να λειτουργήσουν μέσω του Διαδικτύου, όπου η αξιοπιστία και η ταχύτητα δεν μπορούν να εγγωηθούν, όπου δεν υπάρχει καμιά δυνατότητα ρύθμισης κατά

την οποία οι χρήστες αλλά και οι προμηθευτές των υπηρεσιών να αναβαθμίζονται αμέσως, όπου τα δομικά τμήματα (components) και τα προϊόντα των vendors τρέχουν σε διαφορετικές πλατφόρμες και όπου μια υπάρχουσα εφαρμογή εκτελείται σε ένα δίκτυο και μπορεί να περατωθεί σε μια Υπηρεσία Ιστού.

Οι υπηρεσίες ιστού προσφέρουν λύσεις στα παραπάνω μειονεκτήματα των συστημάτων ηλεκτρονικής μάθησης, καθώς προσφέρουν διαλειτουργικότητα στο σύστημα, δηλώνουν ιδιότητες και σχέσεις μεταξύ των αντικειμένων (οντολογίες), παρέχουν πληροφορίες από και για τους συγγραφείς, δημιουργούν δυναμικά μαθήματα, παρέχουν στρατηγικές για την εκπλήρωση των απαιτήσεων των χρηστών.

10.1. Πρότυπα ηλεκτρονικής μάθησης

Για να είναι δυνατή η περαιτέρω αξιοποίηση μιας σειράς εφαρμογών/λογισμικού από άλλους οργανισμούς ή μεμονωμένους προγραμματιστές που δεν χρησιμοποιούν την ίδια γλώσσα προγραμματισμού με αυτή που αρχικά έχει υλοποιηθεί το σύστημα, έχουν αναπτυχθεί κάποια πρότυπα μοντελοποίησης της δομής περιεχομένου ηλεκτρονικών συστημάτων, η πλειονότητα των οποίων χρησιμοποιεί ως γλώσσα κωδικοποίησης των περιγραφών την XML.

Ουσιαστικά, αποτελούν τυπικές περιγραφές των περιγραφικών όρων που χρησιμοποιούνται για το σημασιολογικό σχολιασμό εκπαιδευτικού υλικού όλων των ειδών. Ένα πρότυπο μπορεί να αναφέρεται

- ✚ Είτε στη δομή ενός Αντικειμένου Μάθησης (Learning Object) με τη χρήση μεταδεδομένων
- ✚ Είτε στο διαμοιρασμό του περιεχομένου μέσω της χρήσης διαλειτουργικών Μοντέλων Δομής Περιεχομένου (Content Structure Models)

Τα βασικότερα πρότυπα που έχουν προταθεί είναι τα:

- ✚ **ARIADNE** (Alliance of Remote Instructional Authoring & Distribution Network for Europe)
- ✚ **DC** (Dublin Core)
- ✚ **IEEE/LOM** (Institute of Electrical and Electronic Engineers/Learning Object Metadata)
- ✚ **AICC** (Aviation Industry CBT Committee), όπου CBT στη θέση του Computer Based Training)
- ✚ **ADL/SCORM** (Advanced Distributed Learning/Sharable Content Object Reference Model)
- ✚ **IMS** (Instructional Management System)

Τα περισσότερα από αυτά συσχετίζονται μεταξύ τους και αναλύονται παρακάτω.

10.1.1. Ariadne metadata

Η ευρωπαϊκή εργασία ARIADNE διήρκεσε από το 1996-2000 και οδήγησε στη δημιουργία ενός συνόλου προτάσεων για εκπαιδευτικά μεταδεδομένα. Με το πέρας αυτής, όσοι εργάστηκαν στην ARIADNE ίδρυσαν το ARIADNE Foundation για να συνεχίσουν να εργάζονται βασιζόμενοι στα επιτεύγματα της αρχικής εργασίας. Το πρότυπο αυτό προσπαθεί να δώσει λύση στα εξής ζητήματα:

- ✚ Εύκολη και αποδοτική δεικτοδότηση των Αντικειμένων Μάθησης.
- ✚ Εύκολη εκμετάλλευση των μεταδεδομένων από χρήστες που αναζητούν συναφές παιδαγωγικό υλικό.

Η έκδοση 3.2 της περιγραφής του προτύπου καθορίζει ότι υπάρχουν κάποιες υποχρεωτικές κατηγορίες και μία προαιρετική, η κατηγορία σχολίων. Οι υποχρεωτικές κατηγορίες αποτελούν το ελάχιστο σύνολο περιγραφικών

γνωρισμάτων το οποίο καθιστά σχετικά ικανοποιητική την έρευνα, χωρίς να είναι ιδιαίτερα επιβαρυντική η δημιουργία της.

Το πρότυπο καθορίζει τις εξής κατηγορίες γνωρισμάτων:

1. Γενικές πληροφορίες πόρων
2. Σημασιολογία πόρων
3. Παιδαγωγικά γνωρίσματα
4. Τεχνικά χαρακτηριστικά
5. Συνθήκες χρήσης
6. Πληροφορία μετα-μεταδεδομένων
7. Σχόλια (Προαιρετικό)

10.1.2. IEEE LOM

Το IEEE LOM αποτελεί μία κοινή πρόταση των IMS και ARIADNE που δανείζεται κάποια στοιχεία από το DC. Τα κύρια χαρακτηριστικά του είναι τα εξής:

- ✚ XML DTDs χρησιμοποιούνται για τον καθορισμό της σύνταξης και της σημασιολογίας του
- ✚ Παρέχει τα κατάλληλα γνωρίσματα για την περιγραφή ενός Αντικειμένου Μάθησης (π.χ. όνομα στοιχείου, τύπος δεδομένου, ορισμός, λεξιλόγιο, μήκος πεδίου)
- ✚ Επικεντρώνεται σε ένα ελάχιστο σύνολο γνωρισμάτων για τη διαχείριση, τον τόπο και την αποτίμηση των αντικειμένων μάθησης
- ✚ Είναι ένα πολύ κατανοητό πρότυπο μεταδεδομένων και, οπότε, όλες σχεδόν οι υπάρχουσες υλοποιήσεις περιγραφών μεταδεδομένων βασίζονται σε αυτό

Το πρότυπο καθορίζει εννιά προαιρετικές κατηγορίες για πάνω από εβδομήντα στοιχεία μεταδεδομένων σχετιζόμενων με LOs με σεβασμό στην ακόλουθη σπυρωτή ιεραρχία (granularity hierarchy):

1. Curriculum
2. Course
3. Unit
4. Topic
5. Lesson
6. Fragment

10.1.3. AICC

Το πρότυπο αυτό παρέχει ένα μοντέλο δομής περιεχομένου με κύριο στόχο τη διαλειτουργικότητα. Διακρίνει τα παρακάτω τρία δομικά στοιχεία, με κεντρική ιδέα το ότι ένα μάθημα αποτελεί μία συλλογή των παρακάτω:

- ✚ **Assignable Units (AUs).** Τα μικρότερα εκπαιδευτικά στοιχεία τα οποία μπορούν να παρουσιασθούν σε ένα μαθητή, όπως μία HTML σελίδα.
- ✚ **Blocks.** Ένα block μπορεί να περιλαμβάνει AUs και/ή άλλα blocks (φωλιασμένα blocks)
- ✚ **Objectives.** Χρησιμοποιούνται για να καθορίσουν προαπαιτούμενα μαθημάτων και να αναπαραστήσουν τους στόχους ενός μαθήματος. Μπορεί να είναι απλά ή σύνθετα, ανάλογα με το αν περιέχουν απλά ή πολλαπλά AUs και blocks

Τα AICC CMI Guidelines for Interoperability παρέχουν μία σειρά από αρχεία για την αποθήκευση της στατικής και δυναμικής δομής των μαθημάτων που ανταλλάσσονται μεταξύ συστημάτων ηλεκτρονικής μάθησης. Η ανταλλαγή των αρχείων αυτών καθιστά δυνατή την ανακατασκευή του μαθήματος σε άλλες πλατφόρμες.

10.1.4. Scorm Content Structure Format (csf)

Το SCORM CSF προέρχεται από το AICC CMI. Διαφοροποιείται από αυτό σε δύο σημεία:

- ✚ Είναι εφαρμόσιμο όχι μόνο σε ολοκληρωμένα courses, αλλά και σε υποσύνολα των courses ή ομάδες των courses
- ✚ Έχει μετονομάσει τον όρο AU σε διαμοιραζόμενο αντικείμενο περιεχομένου, χωρίς να αλλάξει τη σημασία αυτού

10.1.5. IMS CP INFORMATION MODEL

Το μοντέλο αυτό, κύριο στοιχείο του οποίου είναι το πακέτο (package), έχει τα εξής χαρακτηριστικά:

- ✚ Βασίζεται σε ένα σύνολο δομών δεδομένων για να παράσχει διαλειτουργικότητα του περιεχομένου σε ένα διαδικτυακό περιβάλλον
- ✚ Ενσωματώνει σε ένα μοναδικό αρχείο ένα ολόκληρο course και τα σχετιζόμενο με αυτό μεταδεδομένα
- ✚ Είναι πολύ ευέλικτο στην αναπαράσταση μαθησιακού περιεχομένου
- ✚ Συνεργάζεται σημαντικά με άλλους οργανισμούς

10.2. Αντικείμενα μάθησης και πρότυπα ανάπτυξης του μαθησιακού υλικού

Τα τελευταία χρόνια γίνονται προσπάθειες από τους οργανισμούς προτυποποίησης, ώστε το μαθησιακό υλικό στο περιβάλλον του παγκόσμιου ιστού να αναπτύσσεται στη μορφή αντικειμένων μάθησης. Η έκταση, απο μαθησιακή άποψη, που καλύπτει ένα αντικείμενο μάθησης μπορεί να ποικίλει, αλλά πρακτικά θεωρείται ότι θα πρέπει να αντιστοιχεί σε μαθησιακό υλικό, τυπικού χρόνου μάθησης, περίπου δέκα έως δεκαπέντε λεπτών. Από άποψη περιεχομένου, το μέγεθος αυτό αντιστοιχεί σε ένα έγγραφο που αποτελείται από μία ή περισσότερες σελίδες εκπαιδευτικού υλικού.

Κάθε αντικείμενο μάθησης θα πρέπει να είναι αυτοτελές, έτσι ώστε να μην πρέπει απαραίτητα να συμπληρωθεί από οποιοδήποτε άλλο για να έχει νόημα η χρήση του. Συνακόλουθα, αυτό σημαίνει ότι θα μπορεί να

διαμοιραστεί και να επαναχρησιμοποιηθεί σε πολλαπλά μαθήματα και ενότητες μαθημάτων.

Τα αντικείμενα μάθησης θεωρούνται τα δομικά στοιχεία του ηλεκτρονικού εκπαιδευτικού υλικού που όταν συνδυαστούν κατάλληλα δημιουργούνται τα μαθήματα. Σε αυτό το σημείο, διαφαίνεται ο ρόλος της προτυποποίησης. Χωρίς την προτυποποίηση των αντικειμένων μάθησης, ο συνδυασμός, η διαχείριση και η επαναχρησιμοποίησή τους θα ήταν εξαιρετικά δύσκολη, ή ακόμα και αδύνατη.

Βασικό χαρακτηριστικό των αντικειμένων μάθησης είναι η δυνατότητα επαναχρησιμοποίησής τους σε πολλαπλά μαθησιακά περιβάλλοντα. Για να είναι δυνατός ο διαμοιρασμός (sharing) του μαθησιακού υλικού και η επαναχρησιμοποίησή του, κάθε αντικείμενο μάθησης συνοδεύεται από πρόσθετη πληροφορία, η οποία περιγράφει το είδος και τη μορφή του περιεχομένου του, τις έννοιες που διδάσκονται σε αυτό, το δημιουργό του, το επίπεδο δυσκολίας για έναν τυπικό μαθητή, το είδος και το βαθμό αλληλεπίδρασης που προσφέρει κλπ. Η πληροφορία αυτή υπάρχει σε συγκεκριμένη μορφή και κωδικοποιείται σε XML, με στόχο να είναι παγκόσμια κατανοητή από κάθε σύστημα σύνθεσης και διαχείρισης εκπαιδευτικού υλικού. Η πρακτική που προτείνεται από την πλειονότητα των ερευνητών είναι η μετα-πληροφορία αυτή, να διατηρείται σε ξεχωριστό έγγραφο, στο οποίο να μπορεί να έχει πρόσβαση το σύστημα ηλεκτρονικής μάθησης, χωρίς να είναι απαραίτητο το άνοιγμα ή η εμφάνιση του περιεχομένου του αντικειμένου μάθησης.

Αρκετοί οργανισμοί εμπλέκονται στην ανάπτυξη προτύπων του μαθησιακού υλικού. Τα πρότυπα αυτά καλύπτουν τόσο την ανάπτυξη του εκπαιδευτικού υλικού στη μορφή των αντικειμένων μάθησης, όσο και τους τρόπους με τους οποίους ένα σύστημα διαχείρισης τα αξιοποιεί.

Η έρευνα στα πλαίσια τεσσάρων βασικών οργανισμών προτυποποίησης, των ARIADNE, AICC, IMS και IEEE, έχει οδηγήσει την ADL στην ανάπτυξη της προτυποποίησης SCORM.

Η προτυποποίηση αυτή περιλαμβάνει τα εξής μέρη:

- ✚ Το μοντέλο περιεχομένου (Content Aggregation Model - CAM), το οποίο περιέχει προδιαγραφές για τα μεταδεδομένα των αντικειμένων μάθησης, τη συσκευασία τους (packaging) και την οργάνωσή τους.
- ✚ Το περιβάλλον πραγματικού χρόνου (Run-Time Environment - RTE), το οποίο περιέχει προδιαγραφές που καθορίζουν τον τρόπο με τον οποίο ένα σύστημα διαχείρισης εντοπίζει και καλεί ψηφιακό υλικό.

10.3. Παράδειγμα ηλεκτρονικής μάθησης στην ιατρική εκπαίδευση και χρήση υπηρεσιών ιστού

Η έννοια της ηλεκτρονικής μάθησης υποθέτει ότι σε κάποιες περιπτώσεις οι φοιτητές διδάσκονται μόνοι τους. Για την επιστήμη της ιατρικής, όμως, κάτι τέτοιο δεν μπορεί να συμβαίνει. Για να εξασφαλίσουμε ποιοτική ιατρική εκπαίδευση είναι απαραίτητη η ύπαρξη καθηγητή.

Δεδομένης της ιστορίας της ιατρικής εκπαίδευσης, η συνεχώς αυξανόμενη χρήση ιατρικού e-learning είναι απόλυτα κατανοητή. Η ιατρική εκπαίδευση βασίζεται σε ένα μοντέλο μεταφοράς δεδομένων, μέσω διαλέξεων και βιβλίων, εξέταση των φοιτητών σε αυτά τα δεδομένα καθώς και πραγματική εξάσκηση των φοιτητών σε ό,τι μάθανε πάνω σε πραγματικούς ασθενείς. Η αυτόνομη μάθηση λοιπόν πάντα υπήρχε στον κορμό της ιατρικής εκπαίδευσης, με την πολύτιμη όμως συμβολή του εξειδικευμένου εκπαιδευτικού προσωπικού.

Η ηλεκτρονική μάθηση όχι μόνο αποδεικνύεται συνεπής με την πιο ανθρώπινη προσέγγιση της ιατρικής εκπαίδευσης, αλλά μάλιστα αποδεικνύεται και πολύ πιο επαρκής. Μια γραφική αναπαράσταση της καρδιακής παθοφυσιολογίας, από κάποια δικτυακή βιβλιοθήκη μπορεί να

αποδειχθεί πολύ πιο διδακτική από μια ανιαρή παρουσίαση. Στα πλαίσια της ειδικότητας, μπορεί ο κάθε ειδικευόμενος, ο οποίος έχει ήδη αποκτήσει το βασικό κορμό γνώσης της ιατρικής επιστήμης, να μπορεί να αναζητά από μια βάση δεδομένων γνώσεων, απαντήσεις στα προβλήματα που του παρουσιάζονται, με συνέπεια η μελέτη του να είναι πιο παραγωγική από το να συμμετέχει σε ενιαίες ομαδικές παρουσιάσεις.

Η τεχνολογία έχει να επιδείξει διάφορες εφαρμογές, που στηρίζουν την ιατρική εκπαίδευση και τη συνεχιζόμενη ενημέρωση και κατάρτιση μέσα από το διαδίκτυο. Ένα σημαντικό μέρος τέτοιων εφαρμογών αφορούν σε πλήρη οργάνωση διαδικτυακών μαθημάτων σε πανεπιστημιακές σχολές επιστημών υγείας.

Η πρόσβαση σε εκπαιδευτικό υλικό και επιστημονικό τύπο είναι ανάμεσα στις πλέον δημοφιλείς χρήσεις του διαδικτύου για τους γιατρούς.

Το πιο χαρακτηριστικό ίσως παράδειγμα αποτελεί η δυνατότητα χρήσης μέσω διαδικτύου της βάσης δεδομένων MedLine της Εθνικής Βιβλιοθήκης Ιατρικής των ΗΠΑ (National Library of Medicine, USA). Έχουν αναπτυχθεί διάφορες διαδικτυακές εφαρμογές για πρόσβαση στα περιεχόμενα της βάσης MedLine, ωστόσο, ιδιαίτερα δημοφιλής είναι σήμερα η εφαρμογή PubMed, που προσφέρεται από την Εθνική Βιβλιοθήκη Ιατρικής των ΗΠΑ στη διεύθυνση <http://www.ncbi.nlm.nih.gov/entrez> και υποστηρίζεται και εμπλουτίζεται λειτουργικά από διάφορα εργαλεία. Στο διαδίκτυο διατίθεται ένας μεγάλος αριθμός από παρόμοιες εξειδικευμένες βάσεις δεδομένων σε διάφορα θέματα ιατρικής αλλά και βιολογίας. Ένα χαρακτηριστικό παράδειγμα είναι η βάση OMIM, που διαθέτει κατάλογο και σχετικές πληροφορίες για κληρονομικές ασθένειες.

Με τη χρήση της κατάλληλης υπηρεσίας ιστού, οι παραπάνω βάσεις δεν θα είναι προσβάσιμες μόνο από τους χρήστες που γνωρίζουν τη διεύθυνση τους και απευθύνουν άμεσα τις αιτήσεις τους στις συγκεκριμένες ιστοσελίδες. Κοινές μηχανές αναζήτησης ή ευρετήρια θα μπορούν γενικά να

προσπελάσουν αυτές τις βάσεις, έχοντας ως αποτέλεσμα το περιεχόμενό τους να περιέχεται σε μια γενική αναζήτηση του χρήστη, χρησιμοποιώντας κλασσικά εργαλεία για την αναζήτηση στο παγκόσμιο ιστό.

Το προγραμματιστικό παράδειγμα των υπηρεσιών ιστού έχει ήδη αρχίσει να εφαρμόζεται με επιτυχία στο χώρο της επιστήμης της Βιολογίας για την εξόρυξη, αναζήτηση και ανάκληση βιολογικών δεδομένων στο διαδίκτυο. Ιδιαίτερα πρόσφατα έχουν αρχίσει να εμφανίζονται και οι πρώτες καθαρά ιατρικές εφαρμογές υπηρεσιών ιστού για την ολοκλήρωση βιολογικών και κλινικών δεδομένων στην Ογκολογία, καθώς και για την διαχείριση ιατρικής εικόνας και ηλεκτρονικού φακέλου υγείας μέσα από το διαδίκτυο.

Κεφάλαιο 11. Η χρήση υπηρεσιών ιστού και υπηρεσιοστρεφών αρχιτεκτονικών σε συστήματα ηλεκτρονικής μάθησης

Έχουν αναπτυχθεί πολυάριθμες εκπαιδευτικές εφαρμογές τα τελευταία χρόνια στο πλαίσιο του διαδικτύου και ειδικότερα του παγκόσμιου ιστού. Οι εφαρμογές ποικίλουν, από απλές, που παρέχουν μικρές δυνατότητες αλληλεπίδρασης και προσαρμογής στις ανάγκες του μαθητή, έως ευφυή συστήματα, στα οποία γίνεται εξατομίκευση της μαθησιακής διαδικασίας, υπάρχει πλούσια αλληλεπίδραση με τους μαθητές, καθώς επίσης και υποστήριξη της συνεργασίας ανάμεσά τους. Όμως, η αποδοτική σχεδίαση και ανάπτυξη συστημάτων τα οποία ενσωματώνουν στοιχεία ευφυΐας, θέτει νέες, τεχνολογικές και άλλες απαιτήσεις. Η μετεξέλιξη του παγκοσμίου ιστού, ο σημασιολογικός ιστός, εμφανίζεται σαν ένα πολλά υποσχόμενο περιβάλλον για τον σχεδιασμό και την υλοποίηση συστημάτων ηλεκτρονικής μάθησης. Ο σημασιολογικός ιστός είναι το αποτέλεσμα των προσπαθειών για το χτίσιμο μιας νέας αρχιτεκτονικής, στην οποία οι δυνατότητες χειρισμού του περιεχομένου από υπολογιστικά συστήματα, αναβαθμίζονται με βάση την προσθήκη σημασιών σε αυτό.

Επομένως, ο σημασιολογικός ιστός μπορεί να χρησιμοποιηθεί ως πλατφόρμα υλοποίησης ενός συστήματος ηλεκτρονικής μάθησης, καθώς η αρχιτεκτονική του φαίνεται ότι ικανοποιεί τις απαιτήσεις της ηλεκτρονικής μάθησης. Η οντολογία παρέχει το υπόβαθρο για να μπορούν να γίνουν σημασιολογικές επερωτήσεις και πλοήγηση στα δεδομένα που περιγράφουν οι RDF και XML, με τη βοήθεια εργαλείων μάθησης.

Τομέας	Ηλεκτρονική Μάθηση	Σημασιολογικός Ιστός
Διανομή γνώσεων	Ο μαθητευόμενος αποφασίζει για την ύλη του (pull)	Το υλικό συγκεντρώνεται κατόπιν επερωτήσεων
Ανταπόκριση	Ανταποκρίνεται στις ανάγκες που εντοπίζει	Το υλικό παρέχεται μέσω προφίλ χρήστη
Πορεία	Μη γραμμική, επιτρέπει απ' ευθείας πρόσβαση σε οποιοδήποτε κομμάτι γνώσης	Πλοήγηση στις πληροφορίες του ενδιαφέροντος του χρήστη
Πρόοδος	Συνεχής, η μάθηση δε σταματάει ποτέ	Η βάση των πληροφοριών ανανεώνεται
Αρχή	Κατανεμημένη, διδάσκων και διδασκόμενος συναποφασίζουν	Αναγκαία η αλληλεπίδραση χρήστη-συστήματος
Εξατομίκευση	Προσωποποιημένη, σχεδιασμένη για τις ανάγκες κάθε μαθητευόμενου	Ο χρήστης αναζητεί πληροφορίες και η οντολογία συνδέει το πληροφοριακό υλικό με τις συγκεκριμένες αναζητήσεις του
Προσαρμοστικότητα	Δυναμική, το περιεχόμενο της γνώσης αλλάζει σύμφωνα με την εμπειρία του μαθητευόμενου	Όλα τα παραπάνω!

Πίνακας 21. Συσχετισμός σημασιολογικού ιστού και ηλεκτρονικής μάθησης

Οι δυνατότητες που εμπεριέχονται στις τεχνολογίες του σημασιολογικού ιστού, τον καθιστούν ένα περιβάλλον ακόμα πιο πλούσιο και κατάλληλο για την ανάπτυξη συστημάτων ηλεκτρονικής μάθησης με δυνατότητες προσαρμογών στις ανάγκες του χρήστη. Το μαθησιακό υλικό, έχοντας τη μορφή αντικειμένων μάθησης (learning objects), δηλαδή ψηφιακών οντοτήτων με αυτοδύναμη μαθησιακή αξία και περιέχοντας σημασιολογική πληροφορία προσαρτησμένη σε αυτά, μπορεί να αξιοποιηθεί στα πλαίσια ενός

προσαρμοζόμενου συστήματος. Σύμφωνα με τους μαθησιακούς στόχους, τις προτιμήσεις, την επίδοση, την προηγούμενη γνώση και ενδεχομένως και άλλες παραμέτρους που χαρακτηρίζουν τον εκπαιδευόμενο, ένα προσαρμοστικό σύστημα μπορεί να του δώσει, με μεγαλύτερη ακρίβεια, το μαθησιακό υλικό που είναι κατάλληλο για αυτόν, ή τις επιλογές πλοήγησης που ανταποκρίνονται καλύτερα στις ανάγκες του.

11.1. Ηλεκτρονική μάθηση και οντολογίες

Οι οντολογίες παίζουν όλο και πιο σημαντικό ρόλο στην ανάπτυξη συστημάτων ηλεκτρονικής μάθησης (e-learning systems). Στα σύγχρονα συστήματα ηλεκτρονικής μάθησης οντολογίες χρησιμοποιούνται για την αναπαράσταση του μαθησιακού υλικού (περιεχόμενο, μορφή, δομή), των μαθησιακών εννοιών και οντοτήτων (πχ. μάθημα, τάξη, μαθητής) και των μεταξύ τους σχέσεων. Πρότυπα μεταδεδομένων (Dublin Core, IEEE LOM) και ειδικές οντολογίες για την αναπαράσταση μαθησιακού υλικού έχουν προταθεί και χρησιμοποιηθεί σε ευφυή συστήματα elearning, ενώ εκπαιδευτικές εφαρμογές όπως οι CIPHER, Connections, Conzilla, Edutella και άλλες βασίζουν τη λειτουργία τους σε οντολογίες.

Για την αξιολόγηση οντολογιών μαθησιακού υλικού υλοποιούμε με τη μορφή οντολογίας ένα μοντέλο αξιολόγησης λαμβάνοντας υπ' όψιν *συντακτικές* (syntactic), *σημασιολογικές* (semantic), *πραγματιστικές* (pragmatic) και *κοινωνικές* (social) *μετρικές*, καθώς και *μετρικές συνάφειας* (cohesion metrics).

Η *συντακτική* ανάλυση μιας οντολογίας αφορά τις συσχετίσεις μεταξύ των συμβόλων των (λέξεων, φράσεων, προτάσεων). Η *σημασιολογική* ανάλυση μιας οντολογίας μελετά τις σχέσεις ανάμεσα στα σύμβολα και τις έννοιες που αυτά αναπαριστούν. Η *πραγματιστική* ανάλυση αναφέρεται στη χρησιμότητα της οντολογίας για τους χρήστες και τους πράκτορες λογισμικού που την προσπελαίνουν. Οι *κοινωνικές μετρικές* λαμβάνουν υπ' όψιν το γεγονός ότι πράκτορες λογισμικού (software agents) και οντολογίες συνυπάρχουν και επικοινωνούν μεταξύ τους σχηματίζοντας κοινότητες. Τέλος, οι *μετρικές*

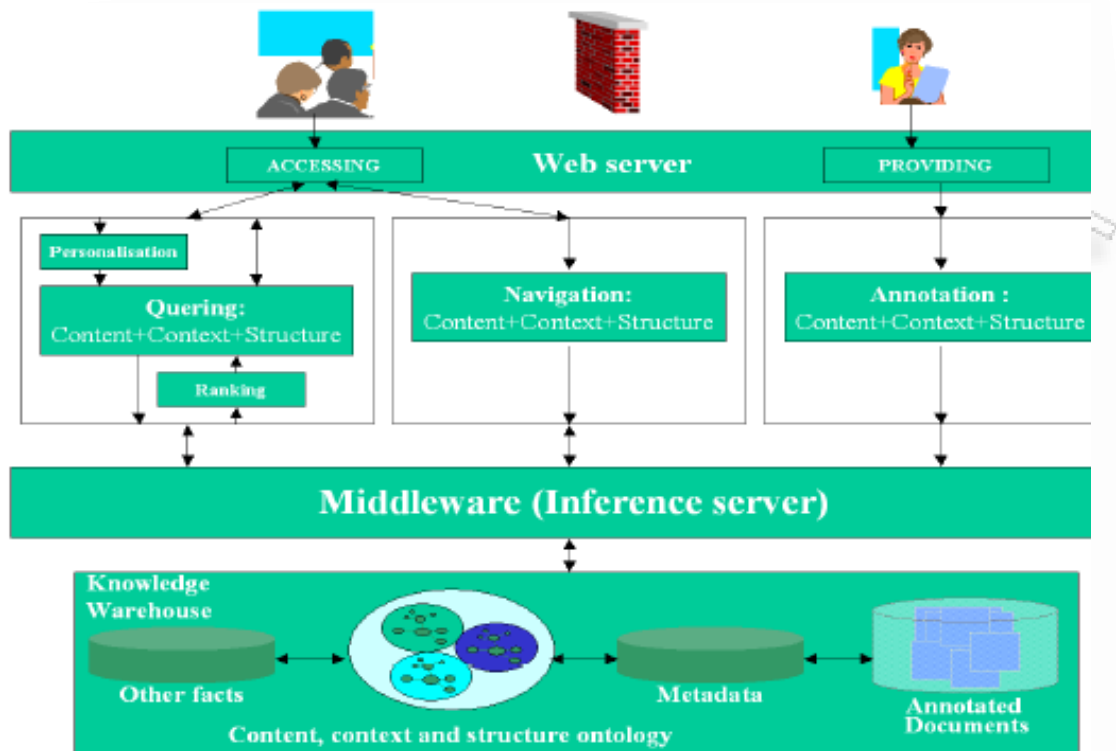
συνάφειας αναφέρονται στο βαθμό που οι κλάσεις μιας οντολογίας συσχετίζονται μεταξύ τους μέσω των ιδιοτήτων τους.

Αν και όλες οι μετρικές \square εωρούνται εξίσου σημαντικές και μπορούν να έχουν ίσα βάρη κατά τον υπολογισμό της ολικής ποιότητας, τα βάρη κάποιων μετρικών μπορούν να μεταβάλλονται ανάλογα με την εφαρμογή. Αφού καθοριστούν τα βάρη, το μοντέλο μπορεί να χρησιμοποιηθεί για την αξιολόγηση μιας ή περισσότερων οντολογιών μαθησιακού υλικού. Το μοντέλο περιλαμβάνει ένα τμήμα εξαγωγής συμπερασμάτων που περιέχει κανόνες που μετρούν τις επιμέρους ποιότητες μιας οντολογίας και μπορεί να υλοποιηθεί εύκολα σε OWL.

Ειδικές οντολογίες για την αναπαράσταση μαθησιακού υλικού έχουν προταθεί και χρησιμοποιηθεί σε ευφυή συστήματα e-learning. Κάποια από τα πιο γνωστά συστήματα e-learning που χρησιμοποιούν οντολογίες για την αναπαράσταση μαθησιακού υλικού είναι τα Edutella, Conzilla, SchollOnto και RichODL.

11.2. Αρχιτεκτονική συστήματος ηλεκτρονικής μάθησης

Παρακάτω παρατίθεται μια αρχιτεκτονική για ένα σενάριο ηλεκτρονικής μάθησης, βασισμένο σε οντολογίες. Η αρχιτεκτονική του συστήματος απεικονίζεται στην παρακάτω εικόνα.



Εικόνα 8. Αρχιτεκτονική μιας πύλης ηλεκτρονικής μάθησης

Το βασικό μέρος του συστήματος είναι μία οντολογία που περιγράφει τα μαθήματα, η οποία και παρουσιάζεται στον παρακάτω πίνακα. Ο ορισμός της οντολογίας περιέχει μια is-a κληρονομικότητα μεταξύ σχετικών αντικειμένων (concepts), σχέσεις μεταξύ αντικειμένων, δηλώσεις ιδιοτήτων και παραγωγικών κανόνων για την αναφορά νέας γνώσης. Όπως φαίνεται στην εικόνα, η πρώτη στήλη περιέχει την δήλωση των αντικειμένων και της is-a κληρονομικότητας που έχουν. Για παράδειγμα ο «PhDStudent» είναι υποαντικείμενο του αντικειμένου «Student», το οποίο είναι υποαντικείμενο του «Person».

Διαπιστώνουμε ότι η οντολογία για ένα μάθημα δομείται από τις επιμέρους οντολογίες περιβάλλον (content), περιεχόμενο (context) και δομή (structure). Η οντολογία περιβάλλον (content) περιγράφεται από αντικείμενα όπως «Protocol», «Service», «Topology». Η οντολογία περιεχόμενο (context) στηρίζεται σε ένα παιδαγωγικό μοντέλο και περιγράφεται από τα αντικείμενα «Introduction», «Explanation», «Example». Το πιο σημαντικό μέρος της

οντολογίας δομής είναι οι σχέσεις μεταξύ εκπαιδευτικών εγγράφων, όπως «prevDocument», «nextDocument», «IsBasedOn», «IsBasisFor».

Σύμφωνα με την μεσαία στήλη του πίνακα, τα εκπαιδευτικά έγγραφα δομούνται με μορφή δένδρου. Δηλαδή οι σχέσεις «prevDocument», «nextDocument» περιγράφουν έγγραφα τα οποία ανήκουν στο ίδιο επίπεδο και οι σχέσεις «parentDocument», «firstChildDocument» συνδέουν έγγραφα διαφορετικών επιπέδων. Τα αντικείμενα «Course», «Module» και «Atom» χρησιμοποιούνται για να δηλώσουν τη δυσκολία και την πολυπλοκότητα των εγγράφων. Όλα τα υπόλοιπα στοιχεία της οντολογίας, τα οποία παρουσιάζονται στον πίνακα -όπως το «name», «title» και «path»-, αναφέρονται σε κάποια συγκεκριμένη τιμή και περιγράφουν ένα συγκεκριμένο έγγραφο.

Λόγω της ύπαρξης της κληρονομικότητας, οι ιδιότητες και οι σχέσεις που είναι δηλωμένες σε ένα αντικείμενο κληρονομούνται από τα υποαντικείμενα αυτής. Η δήλωση των ιδιοτήτων και των σχέσεων ενός αντικειμένου παρουσιάζονται στην μεσαία στήλη του πίνακα. Σύμφωνα με τον πίνακα μπορούμε να δούμε ότι ένα έγγραφο μπορεί να έχει ένα όνομα, ένα τίτλο, ένα συγγραφέα, καθώς και μία σχέση η οποία αναφέρει τα έγγραφα στα οποία έχει βασιστεί το έγγραφο.

Στην τρίτη στήλη του πίνακα περιγράφονται οι κανόνες που έχουμε ορίσει στην οντολογία. Με την χρήση αυτών των κανόνων έχουμε την δυνατότητα να θέσουμε κάποιους περιορισμούς την οντολογία. Για παράδειγμα ο πρώτος κανόνας του πίνακα καθορίζει την μεταβατική ιδιότητα της σχέσης «hasTopic». Οπότε, σύμφωνα με αυτόν και με τα γεγονότα ότι «e-Learning hasTopic TeleTeaching» και «TeleTeaching hasTopic WebBasedLearning», προκύπτει το γεγονός «e-Learning hasTopic WebBasedLearning».

Concept	Relation	Role
Object []. Document :: Object. Content :: Object. Protocol :: Content. Service :: Content. Topology :: Content. Bustopology ::Topology. CircleTopology ::Topology. Context :: Object. Introduction :: Context. Explanation :: Context. Example :: Context. Figure :: Example. Structure :: Object. Course :: Structure. Module :: Structure. Atom :: Structure. Person :: Object. Author :: Person. Student :: Person. PhDStudent :: Student.	Document [Name ==> String; Title ==> String; Path ==> String; has Author ==> Author; content ==> Content; context ==> Context; structure ==> Structure; prevDocument==>Docum ent ; nextDocument==>Docum ent ; firstchildDocument==>Doc ument ; parentDocument==>Docu ment ; relatedDocuments==>Doc ument ; IsBasedOn==>Document ; IsBasisFor==>Document ; ...]. Content [hasTopic==>Content].	FORALL A,B,C A[hasTopic->>C]<-A:Content and A[hasTopic->>B] and B:Content and B[hasTopic->>C] and C:Content. FORALL D, c1, c2 D: Document[content->.C1]<- C1:Content and C2:Content:Content and D:Document[content->>C2] and C1[hasTopic->>C2]. FORALL D1, D2 D1:Document[prevDocument->>D2]<- EXISTS E1, E2, C C:Content and D2:Document[context->>E2] and E2 :Example and D1[context->>E1] and E1 :Explanation and D1[content->>C] and D2[content->>C]. FORALL D1, D2 D1 :Document[parentDocument- ->>D2]<- D2 :Document[firstchildDocument- ->>D1]. FORALL D1, D2 D1 :Document[parentDocument- ->>D2]<- D2 :Document[nextDocument->>D1]. FORALL D, S D :Document[structure->>S :Course]<- Exists D1, S1, D1 :Document and (S1 :Course or S1 :Module) and D1[structure->>S1] and D1[parent->>D].

Πίνακας 22. Οντολογία στο σενάριο ηλεκτρονικής μάθησης

Τα υπόλοιπα κεντρικά κομμάτια που απεικονίζονται στην Εικόνα 8, ανταποκρίνονται σε ενέργειες που πραγματοποιούνται σε ένα περιβάλλον ηλεκτρονικής μάθησης:

 παροχή πληροφοριών από το σύστημα εκπαίδευσης

- ✚ προσπέλαση των εγγράφων εκμάθησης από τους χρήστες ή από τους συγγραφείς, με επερωτήσεις.

Στη συνέχεια, θα περιγράψουμε τις παραπάνω ενέργειες.

11.2.1. Παροχή πληροφοριών

Αρχικός στόχος είναι να δημιουργηθεί το υλικό εκμάθησης, που θα χρησιμοποιηθεί για την ανάπτυξη κύκλων εκπαίδευσης. Το υλικό εκμάθησης θα πρέπει να περιγραφεί σύμφωνα με την οντολογία της προηγούμενης ενότητας. Συγκεκριμένα, θα πρέπει να περιγραφούν τα παρακάτω:

- ✚ Σε τι αναφέρεται το έγγραφο (content annotation)
- ✚ Ποια είναι τα περιεχόμενα του (context annotation)
- ✚ Πώς συνδέεται με άλλα έγγραφα (structure annotation)

Με αυτό τον τρόπο δημιουργούμε δεδομένα, τα οποία αναφέρονται στην οντολογία (course ontology) και αποθηκεύονται σε knowledge warehouse. Για την απάντηση των επερωτήσεων χρησιμοποιείται η μηχανή του συστήματος Ontobroker, η οποία παράγει καινούργια γνώση με ένα έξυπνο συνδυασμό από γεγονότα που υπάρχουν στο knowledge warehouse και από το ορισμό μιας οντολογίας.

11.2.2. Προσπέλαση των εγγράφων εκμάθησης

Για να προσπελάσουμε τα έγγραφα, χρησιμοποιείται η οντολογία με την οποία δίνουμε σημασιολογία στα δεδομένα. Σε πρώτη φάση, τη χρησιμοποιούμε για σημασιολογική αναζήτηση σύμφωνα με το περιβάλλον, τα περιεχόμενα, και την δομή. Σε δεύτερη φάση, με την χρήση της οντολογίας μπορούμε να πλοηγηθούμε μεταξύ των εγγράφων εκμάθησης, τα οποία πήραμε από την αναζήτηση. Η σημασιολογική πλοήγηση στηρίζεται στις σχέσεις που έχουν οριστεί μεταξύ των αντικειμένων των οντολογιών content και context.

11.3. Παρόμοιες εργασίες

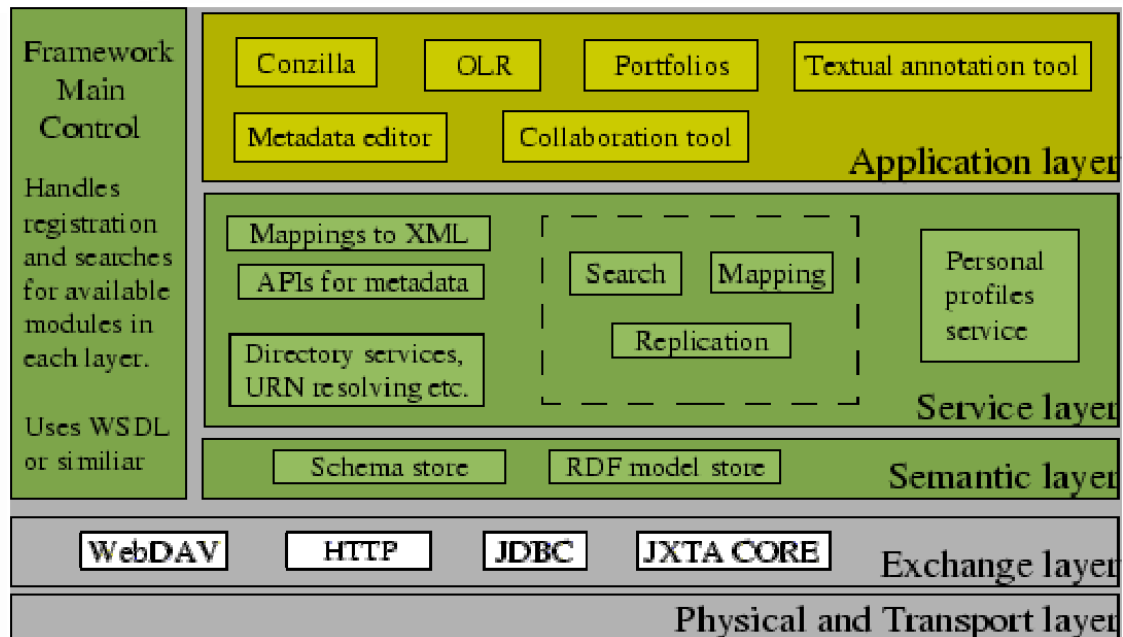
Υπάρχουν κι άλλα συστήματα που θα μπορούσαν να συγκριθούν με αυτό το σενάριο. Ένα από αυτά είναι το Karina, το οποίο παρέχει τη δυνατότητα να δημιουργούνται τα μαθήματα δυναμικά, σύμφωνα με τις προτιμήσεις του χρήστη. Για την περιγραφή των μαθημάτων χρησιμοποιούνται εννοιολογικοί γράφοι και περιέχει στρατηγικές οι οποίες εκπληρώνουν τις απαιτήσεις του χρήστη όσον αφορά στην αναζήτηση και στην πλοήγηση. Το σύστημα Sybil, το οποίο προέρχεται από το Karina, χρησιμοποιεί οντολογίες για να περιγράψει το περιεχόμενο των μαθημάτων. Ωστόσο, και τα δύο δεν περιγράφουν με σαφή τρόπο την δομή του μαθήματος.

Το Collaborative Courseware Generating σύστημα χρησιμοποιεί καινούργιες τεχνολογίες για την περιγραφή των μαθημάτων (XML, XSLT, WebDAV), αλλά χωρίς τη σαφή ύπαρξη οντολογίας. Επίσης, δεν ορίζει το περιεχόμενο και την δομή με σαφή τρόπο. Όπως μπορούμε να καταλάβουμε, τα παραπάνω συστήματα δεν χρησιμοποιούν το Σημασιολογικό Ιστό σε αντίθεση βέβαια με το σενάριο που παρουσιάσαμε αναλυτικά.

11.4. Ένα e-learning framework

Στην συνέχεια παρουσιάζεται ένα learning framework, στο οποίο οι υπηρεσίες μπορούν να αναπτυχθούν και να επικοινωνούν μεταξύ τους όπως τα συστήματα. Το βασικότερο μέρος αυτού του framework είναι το Σημασιολογικό Επίπεδο, στο οποίο κατασκευάζεται η δομή του Σημασιολογικού Ιστού. Επίσης, το learning framework αποτελείται από τεχνικές του Σημασιολογικού Ιστού και peer-to-peer συστήματα για αναζήτηση, ανάκτηση, δημοσίευση, αντιγραφή και σχεδίαση των δεδομένων.

Η αρχιτεκτονική του framework αποτελείται από επίπεδα τα οποία παρουσιάζονται αναλυτικότερα παρακάτω. Κάθε επίπεδο είναι καλά ορισμένο έτσι ώστε να μπορεί να επικοινωνεί με αποτελεσματικό τρόπο με τα γειτονικά του.



Εικόνα 9. Προτεινόμενα επίπεδα για το learning framework

11.4.1. Προτεινόμενα επίπεδα

Στη συνέχεια, παρουσιάζονται κάποια από τα επίπεδα που προτείνονται για ένα Learning Framework.

- ✚ **Φυσικό Επίπεδο (Physical and Transport layer).** Σ' αυτό το επίπεδο τα bits μεταφέρονται μέσω κάποιου πρωτοκόλλου όπως TCP/IP.
- ✚ **Επίπεδο Συναλλαγών (Exchange layer).** Το επίπεδο exchange layer περιέχει εκείνες τις εφαρμογές που χρησιμοποιούνται για την ανάκτηση πληροφορίας. Όπως μπορούμε να δούμε στην παραπάνω εικόνα, το http είναι ένα από αυτά. Το WebDAV είναι μια επέκταση το οποίο επιτρέπει το γράψιμο, την προσπέλαση, και την προσαρμογή των πηγών καθώς και των δεδομένων τους.
- ✚ **Σημασιολογικό Επίπεδο (Semantic layer).** Το σημασιολογικό επίπεδο, όπου αποθηκεύονται προσωρινά τα μοντέλα που χρησιμοποιούνται για την αναπαράσταση της πληροφορίας, θα πρέπει να είναι συμβατό με το Σημασιολογικό Δίκτυο (Semantic Web) και γι' αυτό χρησιμοποιεί RDF/RDFS και όλες τις επεκτάσεις σε σχέση με τις οντολογίες και τα σχήματα.

- ✚ **Επίπεδο Εφαρμογών (Application layer).** Σε αυτό το επίπεδο συναντιέται ο άνθρωπος και η μηχανή, μέσω εφαρμογών που είναι φιλικές και εύχρηστες στον χρήστη.
- ✚ **Κεντρικός Έλεγχος Framework (Framework main control).** Αυτό το επίπεδο αποτελεί την κύρια δομή της αρχιτεκτονικής που όλα τα μοντέλα δηλώνουν τις δυνατότητες τους με την χρήση των γλωσσών WSDL ή OIL. Γενικότερα μια εφαρμογή δηλώνει ποιο σχήμα, ποια πρωτόκολλα θα χρησιμοποιήσει.

11.5. Εργαλεία υποστήριξης του εννοιολογικού ιστού

Ο Σημασιολογικός Ιστός μπορεί να παρομοιαστεί με ένα τεράστιο, αλληλοσυνδεδεμένο γράφο ο οποίος δεν είναι εύκολα κατανοήσιμος από τον άνθρωπο. Για το λόγο αυτό προτάθηκε από το KMR (Knowledge Management Research) group, μια δομή ονομαζόμενη «εννοιολογικού ιστού», η οποία θα βρίσκεται πάνω από το Σημασιολογικό Ιστό και θα έχει σκοπό να τον κάνει πιο κατανοητό και προσπελάσιμο από τους χρήστες. Στο KMR group έχουν αναπτυχθεί διάφορες εφαρμογές, απαραίτητες για την πραγματοποίηση των απαιτήσεων της αρχιτεκτονικής του εννοιολογικού ιστού. Σε αυτή την αρχιτεκτονική χρησιμοποιείται RDF λεξικό, όπως τα Dublin Core και IMS/IEEE LOM για την αναπαράσταση δεδομένων. Στη συνέχεια, θα παρουσιάσουμε αυτά τα εργαλεία τα οποία μπορούν να χρησιμοποιηθούν για να υλοποιήσουν σημαντικά σενάρια για την ηλεκτρονική μάθηση.

11.5.1. Edutella

Η Edutella είναι μια υπηρεσία αναζήτησης για peer-to-peer συστήματα τα οποία διευκολύνουν την ανταλλαγή εκπαιδευτικών πηγών. Το Edutella αποτελείται από ένα σύνολο από υπηρεσίες υλοποιημένες με το σύστημα JXTA και οι οποίες περιλαμβάνουν δυνατότητες για αναζήτηση, σχεδίαση και απάντηση. Οι αναζητήσεις θα αποστέλλονται σε όλους που έχουν την δυνατότητα να απαντήσουν. Ο σχεδιασμός θα επιτρέπει την μετάφραση σχημάτων σε ένα σχήμα το οποίο θα είναι κατανοήσιμο από την εφαρμογή, με

αποτέλεσμα οι πληροφορίες να μπορούν να χρησιμοποιηθούν πιο εύκολα, ανεξάρτητα από τον τύπο τους.

11.5.2. Conceptual modeling and knowledge manifolds

Το βασικότερο μέρος του εννοιολογικού ιστού είναι η εννοιολογική μοντελοποίηση, η οποία παρέχει σημασιολογία στις πηγές και είναι κατανοητή από τον άνθρωπο. Για το λόγο αυτό, χρησιμοποιείται μια τεχνική, η Unified Language Modeling (ULM), η οποία είναι μια έκδοση της UML (Unified Modeling Language). Η UML παρέχει ένα καλά πιστοποιημένο λεξιλόγιο για εννοιολογική μοντελοποίηση. Με την χρήση των παραπάνω τεχνολογιών ο εννοιολογικός ιστός σχεδιάζεται σαν γνώση πολλαπλών επιπέδων.

11.5.3. Conceptual browsing with conzilla

Ένα από τα εργαλεία του εννοιολογικού ιστού, ο εννοιολογικός φυλλομετρητής (conceptual browser), δίνει την δυνατότητα στον χρήστη να ψάχνει για περιεχόμενα με την μορφή UML διαγραμμάτων. Ο χρήστης με αυτή την σύνθεση μπορεί να έχει πλήρη εικόνα του περιεχομένου, ενώ εξερευνεί διάφορα είδη από περιβάλλοντα. Ο Conzilla είναι ένας εννοιολογικός φυλλομετρητής σχεδιασμένος στο KMR, διαθέσιμος μέσω του Διαδικτύου. Είναι ένα πολύτιμο εργαλείο ανασκόπησης υλικού με πολύπλοκη δομή ή σημασία. Τέλος, αυτός ο φυλλομετρητής έχει την δυνατότητα να γίνει ένα πολύ χρήσιμο εργαλείο για αναπαράσταση για οποιοδήποτε RDF δεδομένα τα οποία έχουν εννοιολογικό περιεχόμενο.

11.5.4. Digital portfolios

Για την διαχείριση του περιβάλλοντος (content) χρησιμοποιείται ένα ψηφιακό χαρτοφυλάκιο αναπτυγμένο επίσης από την KMR. Το ψηφιακό χαρτοφυλάκιο είναι μία Διαδικτυακή αποθήκη με προσωπικές πληροφορίες οι οποίες χρησιμοποιούνται από τους καθηγητές και τους μαθητές για αποθήκευση και δημοσίευση για e-learning σεναρίων. Αυτό το χαρτοφυλάκιο χρησιμοποιεί RDF suite για την περιγραφή των δεδομένων και της δομής. Το ψηφιακό

χαρτοφυλάκιο μπορεί να εφοδιαστεί με ένα Edutella peer interface, με την βοήθεια του οποίου μετατρέπεται σε ένα σύστημα διαχείρισης περιβάλλοντος (content management system), το οποίο παρέχει δυνατότητα όχι μόνο δημοσίευσης των εγγράφων, αλλά και διασποράς των εγγράφων και των δομών των μαθημάτων.

11.5.5. Application independence: semantic vwe

Όπως αναφέραμε, ένα σημαντικό πλεονέκτημα της χρήσης του Σημασιολογικού Διαδικτύου, ως βάση του εννοιολογικού ιστού, είναι η ανεξαρτησία των εφαρμογών. Άλλη μια εφαρμογή στην οποία εισήγαμε την RDF για διαλειτουργικότητα είναι το Virtual Workspace Environment (VWE), το οποίο αναπτύχθηκε στο KMR με την επίβλεψη του Fredrik Paulsson. Το VWE είναι ένα κατακευματισμένο σύστημα για διαχείριση ηλεκτρονικής μάθησης (Learning Management System) και σχεδιάστηκε για να υποστηρίζει την κατασκευή προσωπικών περιβαλλόντων για learning με την σύνθεση learning πηγών. Το VWE μπορεί να τρέξει από οποιοδήποτε φυλλομετρητή, με αποτέλεσμα ο χρήστης να μπορεί να προσπελάσει το δικό του περιβάλλον εκμάθησης.

11.6. Personalized adaptive learning interfaces (pali)

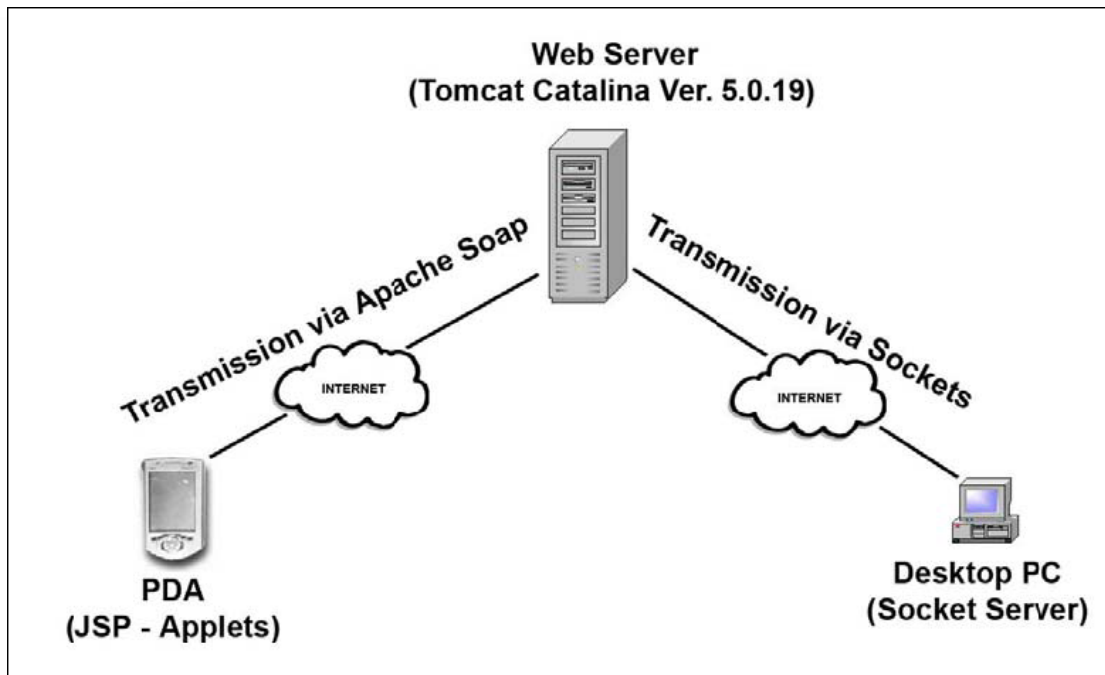
Το project PALI είναι ένα πρόγραμμα χρηματοδοτούμενο από το Υπουργείο Παιδείας και το ΕΠΕΑΕΚ, στα πλαίσια του προγράμματος χρηματοδότησης έρευνας των τεχνολογικών εκπαιδευτικών ιδρυμάτων Αρχιμήδης Ι. Σκοπός του PALI είναι η παροχή προσωποποιημένων υπηρεσιών στους χρήστες μέσα από ετερογενείς συσκευές, με χρήση σημασιολογικών τεχνικών και τεχνητής νοημοσύνης. Για περισσότερες πληροφορίες σχετικά με το project PALI, ο επίσημος διαδικτυακός τόπος είναι <http://users.epp.teicrete.gr/kathigites/amalamos/pali>.

Το PALI είναι ένα πολυεπίπεδο προσωποποιημένο σύστημα που ενσωματώνει έναν αριθμό από τεχνολογίες με σκοπό την προσωποποιημένη

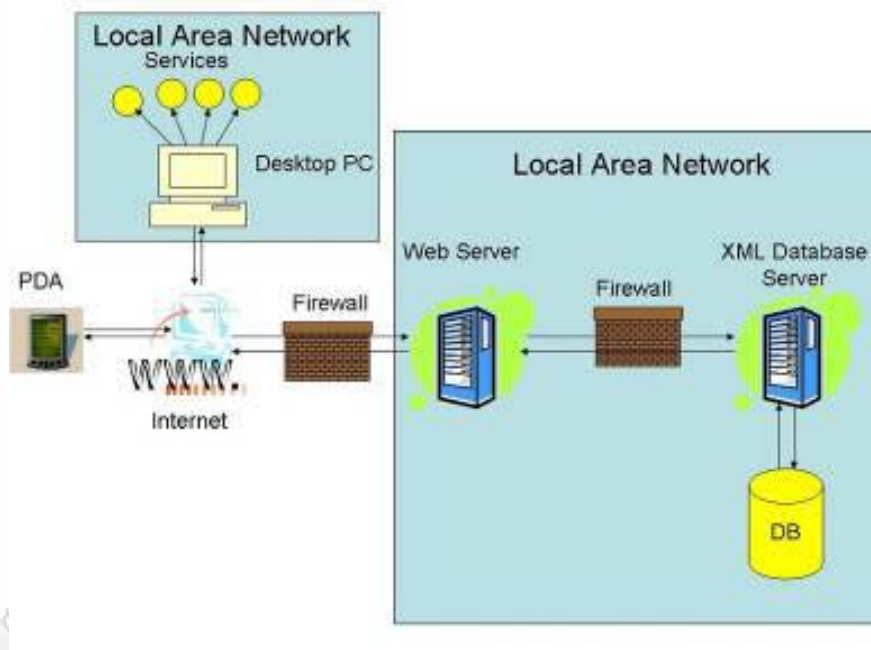
παροχή υπηρεσιών στον τελικό χρήστη. Το σύστημα αποτελείται από μία εφαρμογή που εγκαθίσταται σε φορητές συσκευές και σε συστήματα ηλεκτρονικής μάθησης, μέσω τις οποίας ο τελικός χρήστης αποκτά πρόσβαση στις υπηρεσίες που παρέχει το σύστημα. Το κυρίως σύστημα είναι ένας web server με δυνατότητα εκτέλεσης υπηρεσιών ιστού που αναλαμβάνει μέσω μίας ενδιάμεσης εφαρμογής να δρομολογεί τα μηνύματα των χρηστών στις κατάλληλες υπηρεσίες ιστού μέσω ανταλλαγής SOAP μηνυμάτων. Οι υπηρεσίες ιστού αναλαμβάνουν να προωθήσουν το τελικό μήνυμα μέσω sockets, στην end-user εφαρμογή που εκτελείται σε απομακρυσμένους υπολογιστές που έχει πρόσβαση ο χρήστης για την ολοκλήρωση των αιτήσεων. Οι παρεχόμενες υπηρεσίες είναι μεταξύ άλλων:

1. Απομακρυσμένη πρόσβαση σε personal information manager εφαρμογές όπως το MS Outlook.
2. Απομακρυσμένη πρόσβαση σε πόρους τοπικών δικτύων όπως εκτυπωτές και fax.
3. Απομακρυσμένη πρόσβαση σε πληροφορίες (έγγραφα, βάσεις δεδομένων)
4. Απομακρυσμένη προσωποποιημένη διαδικτυακή αναζήτηση σε μηχανές αναζήτησης
5. Απομακρυσμένη ανταλλαγή αρχείων

Η διαδικασία προσωποποίησης των υπηρεσιών περιλαμβάνει την ύπαρξη ενός παραμετροποιήσιμου προφίλ χρηστών. Το προφίλ αυτό είναι αποθηκευμένο σε δομημένη μορφή (XML), σε έναν profile database server (eXist). Το προφίλ ανανεώνεται μέσω μίας κατάλληλης διεπαφής που παρακολουθεί την επικοινωνία ανάμεσα στην ενδιάμεση εφαρμογή και τις υπηρεσίες ιστού και ανανεώνει χρησιμοποιώντας Xqueries το προφίλ του κάθε χρήστη με τις αλλαγές που συνέβησαν. Μέχρι το στάδιο ολοκλήρωσης του έργου προβλέπεται η ενσωμάτωση τεχνολογιών για την σημασιολογική αναζήτηση των δεδομένων του χρήστη καθώς και τεχνικές τεχνητής νοημοσύνης στα στάδια προσωποποίησης των υπηρεσιών.



Εικόνα 10. Αρχιτεκτονική pali (1)



Εικόνα 11. Αρχιτεκτονική PALI (2)

11.7. KNOSOS

Το πρόγραμμα KNOSOS εντάσσεται στο ευρωπαϊκό πρόγραμμα Leonardo Da Vinci. Εκκινεί από τη διαπίστωση της ταχείας σύγκλισης των τεχνολογιών διανομής πληροφοριών μέσω Παγκόσμιου Ιστού και μέσω εκπομπής

(ψηφιακή τηλεόραση). Επιχειρεί να αξιοποιήσει τις ευκαιρίες παροχής εκπαιδευτικών υπηρεσιών με το συνδυασμό αυτών των δύο βασικών τεχνολογιών και να συγκεράσει ευρέως αποδεκτά μοντέλα όπως το TV-Anytime που αναφέρεται στην περιγραφή προγραμμάτων ψηφιακής τηλεόρασης και του SCORM.

Υποστηρίζει την παροχή αποτελεσματικών εκπαιδευτικών υπηρεσιών σε ειδικούς των ΤΠΕ καθώς και στο προσωπικό των τηλεοπτικών οργανισμών και επιχειρήσεων ώστε να μπορέσουν να αξιοποιήσουν τις τεχνολογίες και τα πρότυπα της σύγχρονης ψηφιακής τηλεόρασης. Για το σκοπό αυτό έχει εγκαταστήσει ένα δίκτυο παροχής εκπαιδευτικού υλικού και ηλεκτρονικής μάθησης το οποίο αποσκοπεί στην ανάπτυξη δεξιοτήτων από μέρος του εμπλεκόμενου ανθρώπινου δυναμικού στον τομέα της ψηφιακής τηλεόρασης.

Επιπλέον, επεξεργάζεται και προτείνει ενοποιημένα μοντέλα ηλεκτρονικής μάθησης σε περιβάλλον ψηφιακής τηλεόρασης αξιοποιώντας τα υφιστάμενα και τα αναπτυσσόμενα διεθνή πρότυπα στους τομείς αυτούς. Με τον τρόπο αυτό επιτυγχάνει αφενός τη δημιουργία των προϋποθέσεων για την ανάπτυξη ενός ιδιαίτερα αποτελεσματικού καναλιού διανομής εκπαιδευτικών υπηρεσιών που απευθύνεται στο ευρύ κοινό και αφετέρου τον εμπλουτισμό του παραδοσιακού τηλεοπτικού τομέα με νέες επικερδείς και κοινωνικά χρήσιμες δραστηριότητες.

Κύριο αποτέλεσμα του έργου είναι η ανάπτυξη μιας κατανεμημένης ψηφιακής βιβλιοθήκης επαναχρησιμοποιήσιμων εκπαιδευτικών πόρων που αναφέρονται σε τεχνικές λύσεις, λογισμικό, διεθνή πρότυπα, νομικά θέματα, καινοτόμες εφαρμογές ηλεκτρονικής μάθησης κ.λ.π. στο πεδίο των τεχνολογιών κατανεμημένων συστημάτων παγκόσμιου ιστού και ψηφιακής τηλεόρασης.

Βασικό εργαλείο για την επίτευξη των τεχνικών στόχων του έργου αποτελεί η χρήση του διεθνούς προτύπου επαναχρησιμοποιήσιμου εκπαιδευτικού περιεχομένου και ηλεκτρονικής μάθησης SCORM της ADL, το οποίο

αποσκοπεί στην επίτευξη διαλειτουργικότητας και επαναχρησιμοποίησης εκπαιδευτικού υλικού –τόσο εντός του δικτύου εκπαιδευτικών κόμβων του KNOSOS όσο και με εξωτερικά συστήματα μάθησης– και είναι αυτή τη στιγμή παγκοσμίως το πλέον αποδεκτό πρότυπο για την ανάπτυξη διαλειτουργικών συστημάτων ηλεκτρονικής μάθησης επιλύοντας τα μέχρι πρότινος προβλήματα ασυμβατότητας μεταξύ τους.

Η κατανεμημένη ψηφιακή βιβλιοθήκη εκπαιδευτικού υλικού του KNOSOS, όπως αναφέραμε και παραπάνω, βασίζεται στο SCORM. Τα τοπικά κέντρα κατάρτισης του δικτύου του KNOSOS (Ελλάδα, Βουλγαρία, Γαλλία και Ουγγαρία), αναπτύσσουν εκπαιδευτικό υλικό συμβατό με το πρότυπο SCORM, το οποίο στη συνέχεια μπορεί να μεταφερθεί από το ένα κέντρο στο άλλο ή ακόμη και εκτός του δικτύου του KNOSOS σε άλλα δίκτυα που ακολουθούν τις προδιαγραφές του SCORM. Οι εκπαιδευόμενοι, μπορούν να χρησιμοποιούν εργαλεία συμβατά με το πρότυπο για να αλληλεπιδράσουν με το εκπαιδευτικό υλικό και για να έχουν, εν γένει, πρόσβαση στις υπηρεσίες διαβίου μάθησης που παρέχει το KNOSOS. Υπάρχει μια κεντρική βιβλιοθήκη υλικού που είναι υλοποιημένη σε ένα σύστημα διαχείρισης σχεσιακών και XML native βάσεων δεδομένων ανοικτού λογισμικού που περιλαμβάνει ειδικούς συντάκτες (editors) και εργαλεία διαχείρισης, καθώς και εφαρμογές υποστήριξης ομαδικών εκπαιδευτικών δραστηριοτήτων. Το εκπαιδευτικό υλικό που βρίσκεται στην κεντρική βιβλιοθήκη και ακολουθεί τις προδιαγραφές του SCORM, μπορεί να λάβει και να χρησιμοποιήσει οποιοσδήποτε χρήστης του παγκόσμιου ιστού. Η βάση δεδομένων της βιβλιοθήκης υποστηρίζει δομές που είναι πλήρως εναρμονισμένες με το SCORM και τα σχετικά πρότυπα μεταδεδομένων, λαμβάνοντας υπόψη τα δικαιώματα χρήσης του υλικού, όπως έχουν καθοριστεί από τους δημιουργούς του.

Η κύρια θετική επίπτωση του έργου KNOSOS είναι η ανάπτυξη νέων δεξιοτήτων στο πεδίο της ολοκληρωμένης χρήσης τεχνολογιών παγκόσμιου ιστού και ψηφιακής τηλεόρασης από τις ομάδες στόχους δίνοντας τη δυνατότητα σε πρόσωπα που προέρχονται από τις ανθρωπιστικές και

παιδαγωγικές επιστήμες να δημιουργήσουν και να εφαρμόσουν στην πράξη νέες μορφές ψηφιακού περιεχομένου για εκπαιδευτική χρήση. Ως θετική μακροπρόθεσμη επίπτωση του έργου επισημαίνεται η συνεισφορά του στην πρόσβαση, μέσω σύγχρονων τηλεπικοινωνιακών δικτύων, σε εκπαιδευτικές πληροφορίες σε διάφορα επίπεδα ξεκινώντας από εκπαιδευτικούς οργανισμούς, και καταλήγοντας σε οικιακή χρήση ή εν κινήσει, πράγμα που αποτελεί ένα μεγάλο βήμα προς την υλοποίηση του παραδείγματος της «πανταχού παρούσας» μάθησης δηλαδή της εξατομικευμένης δια βίου μάθησης για όλους, οποτεδήποτε, και οπουδήποτε.

Η διάχυση των αποτελεσμάτων του έργου πραγματοποιείται σε δύο συμπληρωματικούς άξονες: Αφενός μέσω των δραστηριοτήτων των εθνικών κόμβων ηλεκτρονικής μάθησης που έχουν αναπτυχθεί στις τέσσερις συμμετέχουσες χώρες. Αφετέρου, μέσω της ελεύθερης πρόσβασης στις ηλεκτρονικές υπηρεσίες μάθησης και στο εκπαιδευτικό υλικό προς τρίτους, μέσω δράσεων προώθησης, μέσω ειδικών ημερίδων, καθώς και μέσω ειδικών δράσεων που λαμβάνουν χώρα ιδίως στις χώρες της Ανατολικής Ευρώπης με έμφαση στις χώρες της Βαλκανικής στις οποίες υπάρχει τεράστιο ενδιαφέρον για τις τεχνολογίες που καλύπτει το έργο.

Κεφάλαιο 12. Συμπεράσματα

Ενώ με την αρχιτεκτονική των υπηρεσιών ιστού αντιμετωπίστηκαν τα περισσότερα προβλήματα συμβατότητας και διαλειτουργικότητας μεταξύ των δικτυακών λογισμικών συστημάτων, υπάρχουν ακόμα αρκετά ανοικτά θέματα που έχουν να κάνουν με την ασφάλεια, την ποιότητα επικοινωνίας και με τον αυτόματο τρόπο αναζήτησης και ενσωμάτωσης έτοιμων υπηρεσιών σε ένα λογισμικό σύστημα.

Η υιοθέτηση των τεχνολογιών υπηρεσιών ιστού και της αρχιτεκτονικής της οποίας υλοποιούν, παρέχουν το υπόβαθρο για την ανάπτυξη συστημάτων ηλεκτρονικής μάθησης, τα οποία είναι πιο ευέλικτα, διαλειτουργικά, υποστηρίζουν την επαναχρησιμοποίηση υπαρχόντων υπηρεσιών, αλλά και την αναβάθμιση ή και αντικατάστασή τους από άλλες. Όμως, λόγω του γεγονότος ότι αρκετές από τις προδιαγραφές των αναπτυσσόμενων τεχνολογιών βρίσκονται ακόμα υπό ανάπτυξη, όπως την ασφάλεια των υπηρεσιών ιστού, θα πρέπει να δοθεί ιδιαίτερη μνεία, περισσότερη έμφαση και προσοχή σε θέματα που σχετίζονται με τις υπό ανάπτυξη τεχνολογίες.

Ακόμα, για να είναι αποδοτική η υλοποίηση μιας πλατφόρμας e-Learning είναι απαραίτητη η ύπαρξη προτύπων, ώστε να επιτυγχάνεται η διάρκεια, η διαλειτουργικότητα, η προσβασιμότητα και η επαναχρησιμοποίηση του μαθησιακού περιεχομένου. Πάνω στα πρότυπα αυτά, αναπτύσσονται και αρχιτεκτονικές για τις διάφορες πλατφόρμες.

Έτσι η τεχνολογία των υπηρεσιών ιστού εξελίσσεται συνεχώς προτείνοντας και θεσπίζοντας νέα standards πάνω στον τρόπο ανάπτυξης και λειτουργίας των υπηρεσιών δίνοντας μάλιστα ιδιαίτερη έμφαση σε αυτό που οι ξένοι επιστήμονες ονομάζουν «semantic web» δηλαδή στη σημασιολογική ερμηνεία όλων των κόμβων πληροφορίας του παγκόσμιου ιστού.

Βιβλιογραφία – Αναφορές

Βιβλία

- [1]. Erl, T 2004, 'Introduction to Web services technologies', in Service-Oriented Architecture: A Field Guide to Integrating XML and Web Services, Prentice Hall, New Jersey.
- [2]. Fensel, D, Lausen, H, Polleres, A, Bruijn, J, Stollberg, M, Roman, D, Domingue, J 2007, 'Web Services', in Enabling Semantic Web Services, Springer Berlin Heidelberg, New York.
- [3]. Sanjiva Weerawarana, Francisco Curbera, Frank Leymann, Tony Storey, Donald F. Ferguson, March 22, 2005, 'Web Services Platform Architecture: SOAP, WSDL, WS-Policy, WS-Addressing, WS-BPEL, WS-Reliable Messaging, and More'
- [4]. Jorge Cardoso, 2007, 'Semantic Web Services: Theory, Tools, and Applications', Information Science Reference, Hershey, New York
- [5]. Steve Graham, Simeon Simeonov, Toufic Boubez, Doug Davis, Glen Daniels, Yuichi Nakamura, Ryo Neyama, 2001, 'Building Web Services with Java™: Making Sense of XML, SOAP, WSDL, and UDDI, Sams Publishing

Επιστημονικά άρθρα

- [6]. Baida, Z, Gordijn, J, Omelayenko, B 2004, 'A Shared Service Terminology for Online Service Provisioning', ICEC '04, Sixth International Conference on Electronic Commerce, ACM Press, New York, USA, pp. 1-10.
- [7]. Brusilovsky, P 2004, 'KnowledgeTree: A Distributed Architecture for Adaptive E-Learning', Proceedings of the 13th international World Wide Web conference, ACM Press, New York, USA, pp. 104-113.
- [8]. Exodus Solutions, Τεχνική περιγραφή e.Centric v.3, pp. 2-7.
- [9]. K. Gottschalk, S. Graham, H. Kreger, J. Snell, 2002, 'Introduction to Web services architecture', IBM Systems Journal, Vol 41, No 2.
- [10]. M. Nichols, 2003, "A theory for eLearning". Educational Technology & Society.
- [11]. Καλδούδη, Ε, 2004, 'Διαδίκτυο και υγεία – Σύγχρονες εξελίξεις'
- [12]. L. Stojanovic, S. Staab, R. Studer, 2001, 'eLearning based on the Semantic Web'
- [13]. Naeve, M. Nilsson, M. Palmer, 'eLearning in the Semantic Age'
- [14]. Αράπη, Π, Μουμουτζής, Ν, Μαραγκουδάκης, Ι, Χριστοδουλάκης, Σ, 2005, 'Το έργο KNOSOS: Ένα καταναμημένο περιβάλλον ηλεκτρονικής μάθησης που χρησιμοποιεί το πρότυπο SCORM για την υποστήριξη διαλειτουργικότητας ανάμεσα σε τοπικά κέντρα ηλεκτρονικής μάθησης'

Ιστοσελίδες Διαδικτύου

- [1]. Business Process with BPEL4WS: Understanding BPEL4WS, Part 1, <http://www-106.ibm.com/developerworks/webservices/library/ws-bpelcol1/>
- [2]. Business Process with BPEL4WS: Learning BPEL4WS, Part 2, <http://www-106.ibm.com/developerworks/webservices/library/ws-bpelcol2/>
- [3]. Business Process with BPEL4WS: Learning BPEL4WS, Part 3, <http://www-106.ibm.com/developerworks/webservices/library/ws-bpelcol3/>
- [4]. Business Process with BPEL4WS: Learning BPEL4WS, Part 5, <http://www-106.ibm.com/developerworks/webservices/library/ws-bpelcol5/>
- [5]. Business Process with BPEL4WS: Learning BPEL4WS, Part 7, <http://www-106.ibm.com/developerworks/webservices/library/ws-bpelcol7/>
- [6]. Business processes in a Web services world, <http://www-128.ibm.com/developerworks/webservices/library/ws-bpelwp/>
- [7]. Web Services Business Process Execution Language - Working Draft 01, 08 September 2004, <http://www.oasis-open.org/apps/org/workgroup/wsbpel/>
- [8]. Semantic Web, http://semanticweb.org/wiki/Main_Page
- [9]. IBM (2007), “New to SOA and Web services” <http://www-128.ibm.com/developerworks/webservices/newto/websvc.html>
- [10]. MSDN (2001), “XML Web Services Basics” <http://msdn2.microsoft.com/en-us/library/ms996507.aspx>
- [11]. Roseindia.net (2007), “Web Services” <http://www.roseindia.net/webservices/webservices.shtml>
- [12]. Developer.com (2003), “Web Services Tutorial: Understanding XML and XML Schema”
Part 1: <http://www.developer.com/services/print.php/2195981>
Part 2: http://www.developer.com/services/print.php/10928_2195981_2
- [13]. Developer.com (2002), “Introduction to Web Services”
Part 1: <http://www.developer.com/services/article.php/1485821>
Part 2: Architecture” <http://www.developer.com/services/article.php/1495091>
Part 3: Understanding XML” <http://www.developer.com/services/article.php/1557871>
- [14]. W3C (2001), “Web Service Definition Language (WSDL)” <http://www.w3.org/TR/wsd/>
- [15]. XML.com (2001), “A Web Services Primer” <http://webservices.xml.com/lpt/a/760>
- [16]. UDDI.org (2004), “UDDI Version 3.0.2” http://uddi.org/pubs/uddi_v3.htm

- [17]. W3C (2006), “SOAP Version 1.2”
Part 0: Primer” <http://www.w3.org/TR/2006/PER-soap12-part0-20061219/>
Part 1: Messaging Framework <http://www.w3.org/TR/soap12-part1/>
Part 2: Adjuncts <http://www.w3.org/TR/2003/REC-soap12-part2-20030624/>
- [18]. E-learning Definition and Explanation (Elearning, Online Training, Online Learning)
<http://derekstockley.com.au/elearning-definition.html>
- [19]. Newcomer και Lomow (2004), Understanding SOA with Web Services, Addison Wesley Professional <http://safari.awprofessional.com/0321180860>
- [20]. Wikipedia, Service economy
http://en.wikipedia.org/wiki/Service_economy
- [21]. η-επιχειρείν, “Το μέλλον στις υπηρεσίες στο internet και η κατάσταση στην Ελλάδα”
http://www.go-online.gr/ebusiness/specials/article.html?article_id=219
- [22]. Semantic Web Metadata for e-Learning - Some Architectural Guidelines,
<http://www2002.org/CDROM/alternate/744/>
- [23]. Οι οντολογίες στην πληροφορική - Μερικές εφαρμογές των οντολογιών,
http://www.ontologyportal.net/index.php?option=com_content&view=article&id=83:e-learning-ontology-&catid=79:merikes-efarmoges&Itemid=3
- [24]. “Μεθοδολογία και πλαίσιο ανάπτυξης προσαρμοζόμενου συστήματος ηλεκτρονικής μάθησης σε περιβάλλον σημασιολογικού ιστού”, Γιουβανάκης, Αθανάσιος
<http://dspace.lib.uom.gr/bitstream/2159/3556/1/Giouvvanakis.pdf>