

Πανεπιστήμιο Πειραιώς
Τμήμα : Ψηφιακών Συστημάτων
Πρόγραμμα Μεταπτυχιακών Σπουδών
Κατεύθυνση : Ψηφιακών επικοινωνιών και δικτύων



Πτυχιακή εργασία

Του Σπουδαστή ΧΡΙΣΤΟΦΟΡΟΥ Β. ΚΑΡΑΧΡΗΣΤΟΥ

Πειραιάς 2009

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΡΔΑΛΙΑΣ

Πειραιάς 2009

Πρόλογος

Η παρούσα διπλωματική εργασία ασχολείται με ένα από τα πιο ενδιαφέροντα πεδία της σύγχρονης επιχειρησιακής έρευνας. Μελετά την ανάπτυξη web services με τη βοήθεια του εργαλείου websphere της IBM κυρίως στο πρώτο επίπεδο, στο επίπεδο σχεδιασμού της επιχειρησιακής διαδικασίας του .

Η διπλωματική εργασία εκπονήθηκε στο Χαροκόπειο Πανεπιστήμιο Αθηνών.

Η εργασία έγινε υπό την επίβλεψη της Καθηγήτριας του τμήματος Πληροφορικής και Τηλεματικής Νικολαΐδου Μαρίας την οποία θα ήθελα να ευχαριστήσω προσωπικά για την μεγάλη βοήθεια που μου προσέφερε καθόλη την διάρκεια της προσπάθειάς μου με πολύτιμες συμβουλές αλλά και παρέχοντάς μου ιδανικές συνθήκες εργασίας και μελέτης.

Καθοριστική για την περάτωση του συγκεκριμένου έργου ήταν η συμβολή της Δρ. Νανσυς Αλεξοπούλου η οποία καθοδήγησε την όλη προσπάθεια. Σε αυτήν οφείλεται τόσο η επιλογή του θέματος όσο και η υπέρβαση πολλών από τις δυσκολίες που συνάντησα. Θα ήθελα να την ευχαριστήσω για την πολύτιμη βοήθειά της τόσο σε επιστημονικό επίπεδο όσο και στο επίπεδο της προσωπικής στήριξης και διαρκούς ενθάρρυνσης.

Πειραιάς, Μάρτιος 2009,

Καραχρήστος Β. Χριστόφορος

Περιεχόμενα

<u>Κεφάλαιο 1</u>	7
<u>1.1 Αντικείμενο της διπλωματικής</u>	8
<u>1.1.1 Τι είναι επιχειρησιακή διαδικασία;</u>	8
<u>1.1.2 Τι είναι διαχείριση επιχειρησιακής διαδικασίας ;</u>	8
<u>1.2 Οργάνωση του τόμου</u>	9
<u>Κεφάλαιο 2</u>	11
<u>2.1 Εισαγωγή</u>	11
<u>2.2 Επιχειρησιακές Διαδικασίες (Business Processes) βασισμένες στις δικτυακές υπηρεσίες (web services)</u>	12
<u>2.3 Εισαγωγή στις δικτυακές υπηρεσίες</u>	12
<u>2.3.1 Αρχιτεκτονική των δικτυακών υπηρεσιών.</u>	13
<u>2.3.2 Πρότυπα για δικτυακές υπηρεσίες</u>	15
<u>2.3.2.1 Γλώσσα Περιγραφής Υπηρεσιών Διαδικτύου – WSDL</u>	15
<u>2.3.2.2 Πρωτόκολλο SOAP</u>	18
<u>2.3.2.3 Universal Description, Discovery and Integration (UDDI)</u>	21
<u>2.3.2.4 Πρότυπα περιγραφής Επιχειρησιακών Διαδικασιών που βασίζονται σε υπηρεσίες.</u>	23
<u>2.3.2.4.1 BPEL</u>	24
<u>2.3.2.4.2 BPMN</u>	27
<u>2.3.2.4.3 Σύγκριση των δύο γλωσσών</u>	28
<u>Κεφάλαιο 3</u>	30
<u>3.1 Εισαγωγή</u>	30
<u>3.2 Βασικές έννοιες της BPEL</u>	31
<u>3.2.1 Partner Link Types</u>	31
<u>3.2.2 Partner link</u>	33
<u>3.3 ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ</u>	34
<u>3.3.1 Βασικές δραστηριότητες</u>	34
<u>3.3.2 Δομημένες δραστηριότητες</u>	35
<u>3.4 Μεταβλητές (Variables)</u>	36
<u>3.4.1 Επισκόπηση των μεταβλητών</u>	36
<u>3.4.2 Τύποι μηνυμάτων WSDL</u>	37

<u>3.5 Τύπος σχημάτων XML</u>	38
<u>3.5.1 XML Schema Element</u>	38
<u>3.5.2 Τι είναι ένα link ;</u>	39
<u>3.6 Compensation</u>	39
<u>3.6.1 Τι είναι Compensation (αποκατάσταση);</u>	39
<u>3.7 Συσχέτιση (Correlation)</u>	40
<u>3.7.1 Τι είναι Correlation ;</u>	40
<u>3.7.2 Τι είναι Correlation set</u>	41
<u>3.7.2.1 Ιδιότητες</u>	41
<u>3.8 Event Handling</u>	42
<u>3.8.1 Τι είναι Χειριστές γεγονότων (Event Handlers)</u>	42
<u>3.8.2 Παραδείγματα των δραστηριοτήτων χειριστών γεγονότος</u>	42
<u>3.9 Fault Handling</u>	43
<u>3.9.1 Τι είναι BPEL Fault Handling;</u>	43
<u>Κεφάλαιο 4</u>	44
<u>4.1 Εισαγωγή</u>	44
<u>4.2 Διαδικασία εκτέλεσης έργου</u>	45
<u>4.2.1 ΜΟΝΤΕΛΟΠΟΙΗΣΗ ΤΟΥ PROJECT ΣΕ BPEL ΜΕΣΩ WEBSHERE BUSINESS MODELER</u>	48
<u>4.2.2 Επιχειρησιακά στοιχεία</u>	49
<u>4.2.2.1 Δημιουργία επιχειρησιακών στοιχείων.</u>	49
<u>4.2.2.2 Δημιουργώντας ένα κατάλογο δεδομένων (data catalog) ..</u>	50
<u>4.2.2.3 Προσθέτοντας χαρακτηριστικά στο επιχειρησιακό στοιχείο.</u>	51
<u>4.2.2.4 Δημιουργία μιας περίπτωσης επιχειρησιακών στοιχείων (business item instance)</u>	52
<u>4.2.2.5 Συσχέτιση του επιχειρησιακού στοιχείου με την διαδικασία.</u>	52
<u>4.2.2.6 Δημιουργία ενός προτύπου επιχειρησιακών στοιχείων.</u>	54
<u>4.2.2.7 Προσθέτοντας μια τοπική εργασία.</u>	54
<u>4.2.2.8 Ορίζοντας τις εισόδους και εξόδους της τοπικής εργασίας.</u>	54
<u>4.2.2.9 Δημιουργία τοπικής εργασίας .</u>	55
<u>4.2.2.10 Δημιουργώντας ένα resource catalog</u>	57
<u>4.2.2.11 Δημιουργώντας ένα resource definition καθορισμός πόρων</u>	58
<u>4.2.2.12 Προσθέτοντας χαρακτηριστικά σε ένα resource definition</u>	59
<u>4.2.2.13 Δημιουργία ενός πόρου</u>	61
<u>4.2.2.14 Προσθήκη των attribute values σε έναν πόρο</u>	62

<u>4.2.2.15 Δημιουργία ενός ρόλου.....</u>	<u>63</u>
<u>4.2.2.16 Προσθήκη των δαπανών σε έναν πόρο.....</u>	<u>64</u>
<u>4.2.2.17 Δημιουργία ενός χρονοδιαγράμματος.....</u>	<u>65</u>
<u>4.2.2.18 Προσθήκη των ιδιοτήτων σε ένα χρονοδιάγραμμα.....</u>	<u>66</u>
<u>4.3 Process Diagramms</u>	<u>66</u>
<u>4.3.1 Δημιουργία των διαγραμμάτων διαδικασίας (process diagramms)</u>	<u>66</u>
<u>Κεφάλαιο 5.....</u>	<u>69</u>
<u>5.1 Εισαγωγή</u>	<u>69</u>
<u>5.2 Ο Bpel κώδικας της διαδικασίας</u>	<u>69</u>
<u>5.3 Ο WSDL ορισμός της διαδικασίας</u>	<u>79</u>
<u>5.4 Σχηματική αναπαράσταση.....</u>	<u>86</u>
<u>Σύνοψη και συμπεράσματα.....</u>	<u>91</u>
<u>ΒΙΒΛΙΟΓΡΑΦΙΑ.....</u>	<u>93</u>

Κεφάλαιο 1

Στην εποχή μας ο χώρος των *Web Services* αποτελεί ένα τομέα που εξαπλώνεται συνεχώς και αποτελεί πεδίο εντατικότητας και συνεχούς έρευνας. Όλο και περισσότερες είναι οι νέες τεχνολογίες που αναπτύσσονται προκειμένου να βελτιώσουν τις ήδη υπάρχουσες εφαρμογές και να ορίσουν νέες προοπτικές, βελτιώνοντας τις υπηρεσίες που παρέχονται.

1.1 Αντικείμενο της διπλωματικής

1.1.1 Τι είναι επιχειρησιακή διαδικασία;

Εάν ψάξουμε στο διαδίκτυο για έναν ορισμό της επιχειρησιακής διαδικασίας θα βρούμε πολλούς ορισμούς. Όλοι αυτοί οι ορισμοί περιέχουν κοινά στοιχεία :

- Μια επιχειρησιακή διαδικασία πυροδοτείται από ένα επιχειρησιακό γεγονός
- Μια επιχειρησιακή διαδικασία δέχεται ία είσοδο και δημιουργεί μια έξοδο
- Μια επιχειρησιακή διαδικασία συντίθεται από συγγενής δομικές δραστηριότητες
- Μια επιχειρησιακή διαδικασία μπορεί να είναι μέρος μιας μεγαλύτερης διαδικασίας και μπορεί να εξαρτάται από άλλες επιχειρησιακές διαδικασίες
- Μια επιχειρησιακή διαδικασία συνήθως εξαρτάται από πολλές επιχειρησιακές συναρτήσεις για παράδειγμα προσωπικό, στέγαση, τμήμα IT.

1.1.2 Τι είναι διαχείριση επιχειρησιακής διαδικασίας ;

Η διαχείριση επιχειρησιακής διαδικασίας οδηγεί στην επιχειρησιακή βελτιστοποίηση και καινοτομία εφαρμόζοντας επιχειρησιακή στρατηγική μέσω της μοντελοποίησης, ανάπτυξης και διαχείρισης σε όλο τον κύκλο ζωής της.

Η διαχείριση επιχειρησιακής διαδικασίας συνδυάζει επιχειρησιακές διαδικασίες, πληροφορία και IT πηγές ευθυγραμμίζοντας όλα τα εταιρικά στοιχεία όπως άνθρωποι, πληροφορίες, τεχνολογία και διαδικασίες για να δημιουργήσουν μια ενιαία άποψη. Αυτή η ενοποίηση των πηγών επιτρέπει στον οργανισμό να έχουν πρόσβαση στην επιχειρησιακή πληροφορία γρηγορότερα και να ανταποκριθεί πιο γρήγορα στην αγοραστική ζήτηση.

Για το σκοπό αυτό η εταιρία IBM έχει αναπτύξει ένα ολοκληρωμένο πακέτο προϊόντων διαχείρισης επιχειρησιακών διαδικασιών που ονομάζεται WebSphere Business Integration.

Η WebSphere Business Integration είναι μια ομάδα προϊόντων που δίνουν τη δυνατότητα σε εταιρίες να καθορίσουν, δημιουργήσουν, συγχωνεύσουν και να βελτιστοποιήσουν επιχειρησιακές διαδικασίες χρησιμοποιώντας εφαρμογές που τρέχουν σε server IT διεπαφές.

Ο WebSphere® Business Modeler είναι μέρος της ομάδος προϊόντων WebSphere Business Integration.

Σκοπός της συγκεκριμένης πτυχιακής είναι η παροχή ενός οδηγού για την προετοιμασία μιας επιχειρησιακής μοντελοποίησης με τη βοήθεια του WebSphere® Business

Modeler. Με την χρήση ενός παραδείγματος σε αυτή την πτυχιακή παρουσιάζονται οι μηχανισμοί που μπορεί να χρησιμοποιηθούν για να :

- Μοντελοποιήσουμε και να προσομοιώσουμε μια επιχειρησιακή διαδικασία (business process)
- Καθορίσουμε όποιες επιχειρησιακές μετρήσεις θέλουμε

Μέσω της παρακάτω εργασίας παρουσιάζεται επιπλέον καθοδήγηση για τη χρήση του WebSphere® Business Modeler.

1.2 Οργάνωση του τόμου

Παραπάνω κάναμε μια σύντομη εισαγωγή που αφορούσε κυρίως το αντικείμενο της διπλωματικής εργασίας γενικά καθώς και το γενικό πλαίσιο των αναγκών της σύγχρονης τεχνολογίας που μας οδήγησαν στο να μελετήσουμε και να εμβαθύνουμε όσο το δυνατό περισσότερο στο συγκεκριμένο θέμα.

Παρακάτω θα κάνουμε μια πολύ σύντομη περιγραφή στα κεφάλαια που ακολουθούν προκειμένου να διευκολυνθεί ο αναγνώστης στην καλύτερη κατανόηση της μελέτης μας.

Στο **κεφάλαιο 2** βλέπουμε τι είναι επιχειρησιακές διαδικασίες πάνω σε Web Services καθώς και μια εισαγωγή στα Web Services. Έπειτα γίνεται αναφορά στα πρότυπα των διαδικτυακών υπηρεσιών όπως η *γλώσσα περιγραφής υπηρεσιών διαδικτύου – WSDL* , το *πρωτόκολλο SOAP* καθώς και το *Universal Description, Discovery and Integration (UDDI)*.

Στο **κεφάλαιο 3** γίνεται μια εισαγωγή στη **bpel** και στις βασικές της έννοιες. Ποιο συγκεκριμένα αναφέρονται οι ορισμοί των partner link types καθώς και των partner links, επίσης γίνεται αναφορά στις βασικές δραστηριότητες της **bpel** , στις μεταβλητές της, στα event handlings καθώς και στα fault handlings.

Στο **κεφάλαιο 4** περιγράφουμε το εργαλείο Websphere της IBM. Ξεκινώντας γίνεται η περιγραφή μιας διαδικασίας εκτέλεσης έργου ενός σεναρίου. Συγκεκριμένα περιγράφουμε την **μοντελοποίηση** του project **εισαγωγής ασθενούς σε νοσοκομείο και διαδικασία διάγνωσης** σε **bpel** μέσω **websphere business modeler**. Ποιο κάτω στο ίδιο κεφάλαιο γίνεται και περιγραφή χρήσης καθώς και των δυνατοτήτων του εργαλείου websphere business modeler.

Τέλος στο **κεφάλαιο 5** παρατίθεται ο κώδικας **bpel** και ο κώδικας **wsdl** που περιγράφει την διαδικασία που αναφέρθηκε και γίνεται επεξήγηση των επιμέρους κομματιών που την απαρτίζουν.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΡΑΙΑ

Κεφάλαιο 2

2.1 Εισαγωγή

Τα web services είναι μια τεχνολογία που επιτρέπει στις εφαρμογές να επικοινωνούν μεταξύ τους ανεξαρτήτως πλατφόρμας και γλώσσας προγραμματισμού. Ένα web service είναι μια διεπαφή λογισμικού (software interface) που περιγράφει μια συλλογή από λειτουργίες οι οποίες μπορούν να προσεγγιστούν από το δίκτυο μέσω πρότυπων μηνυμάτων XML. Χρησιμοποιεί πρότυπα βασισμένα στη γλώσσα XML για να περιγράψει μία λειτουργία (operation) προς εκτέλεση και τα δεδομένα προς ανταλλαγή με κάποια άλλη εφαρμογή. Μια ομάδα από web services οι οποίες αλληλεπιδρούν μεταξύ τους καθορίζει μια εφαρμογή web services.

2.2 Επιχειρησιακές Διαδικασίες (Business Processes) βασισμένες στις δικτυακές υπηρεσίες (web services)

Στόχος του παρόντος κεφαλαίου είναι να παρουσιάσει τις δικτυακές υπηρεσίες (web services) καθώς και μέσα από ποια πρότυπα αυτές υλοποιούνται.

Η ανάγκη για αναπαράσταση επιχειρησιακών διαδικασιών σε εφαρμογές με δικτυακές υπηρεσίες, οδήγησε στη δημιουργία πολλών XML λεξιλογίων και προτύπων δικτυακών υπηρεσιών που αφορούν αυτά τα ζητήματα. Έτσι θα δούμε τις δύο βασικότερες περιγραφικές γλώσσες που αναπτύχθηκαν για την περιγραφή των δικτυακών υπηρεσιών την Business Process Modeling Language και Business Process Execution Language .

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: ροή ελέγχου, ροή δεδομένων και ροή γεγονότων, ενώ παράλληλα προσθέτει και δυνατότητες για τον ορισμό επιχειρησιακών ρόλων, ρόλων ασφαλείας και ανταλλαγής μηνυμάτων. Σκοπός της BPML είναι επίσης να αναπτύξει κατάλληλο γραφικό περιβάλλον και γλώσσα ερωτήσεων.

Μια συνεργασία μεταξύ της IBM, της Microsoft και της BEA, ανέπτυξε την Business Process Execution Language for Web Services, με το ακρωνύμιο BPEL4WS. Η συγκεκριμένη γλώσσα έχει αναπτυχθεί ειδικά για να δουλεύει με δικτυακές υπηρεσίες και έχει ενσωματώσει τις εργασίες της Microsoft's XLANG και IBM's Web Services Flow Language. Η BPEL4WS διαχώρισε τις επιχειρησιακές διαδικασίες με τα επιχειρησιακά πρωτόκολλα. Οι επιχειρησιακές διαδικασίες αναπαριστούν έναν συμμετέχοντα σε μία επιχειρησιακή συναλλαγή. Τα επιχειρησιακά πρωτόκολλα περιγράφουν την συναλλαγή, έτσι ώστε να είναι ορατή σε όλα τα εμπλεκόμενα μέρη, και εφόσον δεν αποκαλύπτουν την εσωτερική συμπεριφορά των τελευταίων, ονομάζονται αφηρημένες διαδικασίες.

Για να κατανοήσουμε καλύτερα τις απαιτήσεις μιας περιγραφικής γλώσσας επιχειρησιακών διαδικασιών που εντάσσει σε αυτή και την τεχνολογία των δικτυακών υπηρεσιών παραθέτουμε στη συνέχεια μια σύντομη ανάλυση των τελευταίων, καθώς και των απαιτήσεων που προκύπτουν όσον αφορά τη γλώσσα στην οποία αναφερθήκαμε.

2.3 Εισαγωγή στις δικτυακές υπηρεσίες

Ο όρος <<δικτυακές υπηρεσίες>> όπως υπονοεί το όνομά του αναφέρεται σε υπηρεσίες προσβάσιμες μέσω του διαδικτύου. Στον όρο όμως εμπεριέχονται πολύ περισσότερες έννοιες απ' ό,τι υπονοεί η ονομασία του. Με την έννοια δικτυακές

υπηρεσίες αναφερόμαστε και στην αρχιτεκτονική, τα πρότυπα, την τεχνολογία και τα επιχειρησιακά μοντέλα που απαιτούνται για να υπάρξουν και να υλοποιηθούν οι δικτυακές υπηρεσίες.

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

Με άλλα λόγια οι δικτυακές υπηρεσίες είναι κατασκευαστικές μονάδες για τη δημιουργία εφαρμογών. Σαν παράδειγμα θα μπορούσαμε να θεωρήσουμε μια κατανεμημένη ψηφιακή βιβλιοθήκη κατασκευασμένη από δικτυακές υπηρεσίες που η κάθε μια τους υλοποιεί λειτουργίες όπως κατανεμημένη αναζήτηση, πιστοποίηση, αίτημα ενοικίασης, πληρωμή, μετάφραση κ.α.. Αυτές οι δικτυακές υπηρεσίες θα συνδυαστούν με την βοήθεια μιας εφαρμογής ηλεκτρονική βιβλιοθήκης, η οποία δημιουργεί το απαραίτητο περιβάλλον για να γίνουν προσβάσιμες οι διάφορες πληροφοριακές πηγές από μια κλειστή εξουσιοδοτημένη κοινότητα χρηστών.

Η σύγχρονη μορφή του διαδικτύου επιτρέπει στους χρήστες να συνδεθούν σε διάφορες εφαρμογές στο διαδίκτυο. Οι δικτυακές υπηρεσίες επιτρέπουν σε εφαρμογές να συνδεθούν με άλλες εφαρμογές. Με άλλα λόγια από το πρότυπο επιχείρηση προς πελάτη, περνάμε στο πρότυπο επιχείρηση προς επιχείρηση.

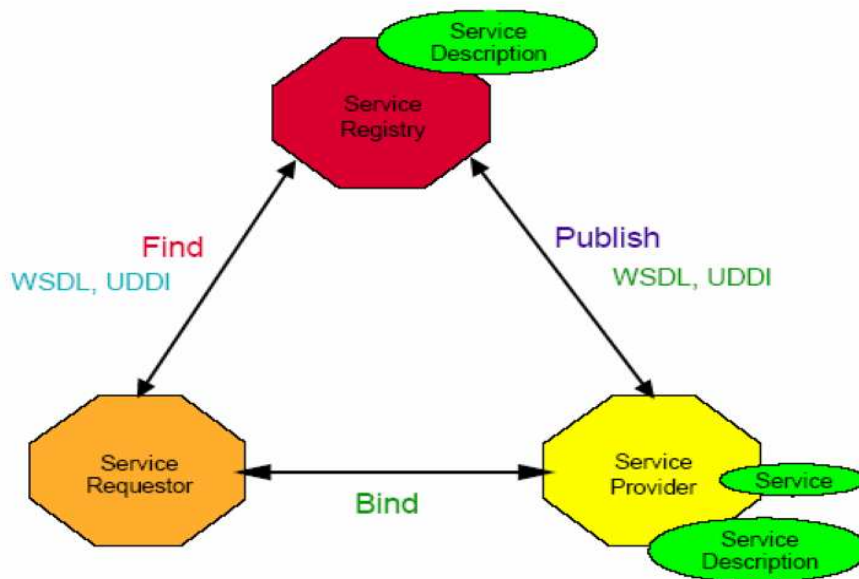
2.3.1 Αρχιτεκτονική των δικτυακών υπηρεσιών.

Επιλέξαμε πιο πάνω το παράδειγμα της ηλεκτρονικής βιβλιοθήκης για να περιγράψουμε τις λειτουργίες των δικτυακών υπηρεσιών. Η επιλογή του συγκεκριμένου παραδείγματος θα μας βοηθήσει να καταλάβουμε την αρχιτεκτονική που απαιτεί η τεχνολογία των δικτυακών υπηρεσιών για την υλοποίησή της αφού αυτή είναι αρκετά όμοια με την αρχιτεκτονική του παραδείγματος της βιβλιοθήκης. Για την ακρίβεια, τα μεταδεδομένα που χρησιμοποιεί η εφαρμογή της ηλεκτρονικής βιβλιοθήκης για την πιστοποίηση, αναζήτηση κλπ είναι επίσης εφαρμόσιμα στην τεχνολογία των δικτυακών υπηρεσιών.

Η τεχνολογία των δικτυακών υπηρεσιών έχει το πρόβλημα της επικοινωνίας με τις υπηρεσίες αφού αποκτηθεί πρόσβαση σε αυτές. Με βάση τα σύγχρονα δεδομένα το πρόβλημα της επικοινωνίας δεν υφίσταται αφού π.χ. στην περίπτωση που η επικοινωνία γίνεται μέσω κειμένων, τα κείμενα μπορούν να σταλούν μέσω του πρωτοκόλλου MIME και αφού πραγματοποιηθεί η λήψη τους, να επιλεγεί η κατάλληλη

εφαρμογή προκειμένου να γίνει η ανάγνωσή τους. Το πρόβλημα έχει να κάνει με τις μελλοντικές επεκτάσεις των δικτυακών υπηρεσιών που η επικοινωνία θα πρέπει να γίνεται όχι με κείμενα αλλά με τύπους δεδομένων που δεν έχουν οριστεί ακόμα. Χρειάζεται λοιπόν ο χρήστης μιας υπηρεσίας να γνωρίζει εξ' αρχής τον τύπο της υπηρεσίας στην οποία προσπαθεί να αποκτήσει πρόσβαση. Έτσι μια εφαρμογή που δημιουργήθηκε πριν από τη δικτυακή υπηρεσία να μπορεί να τη χρησιμοποιεί.

Η αρχιτεκτονική της IBM έχει προτείνει την εσωτερική δομή που απαιτείται προκειμένου να υποστηριχθούν οι δικτυακές υπηρεσίες στα πλαίσια τριών ρόλων, οι οποίοι είναι: ο παροχέας υπηρεσίας, ο αιτών χρήσης υπηρεσίας και το ευρετήριο υπηρεσιών. Οι τρεις πράξεις που ορίζονται προκειμένου να υλοποιηθεί η επικοινωνία μεταξύ των τριών ρόλων είναι η δημοσίευση, η εύρεση και η σύνδεση. Η σύνδεση είναι η λειτουργία εκείνη που επιτρέπει στις διάφορες εφαρμογές να συνδεθούν σε μια υπηρεσία που είναι δημοσιευμένη στο διαδίκτυο και να αρχίσουν να ανταλλάσσουν μηνύματα μαζί της.



Εικόνα 1: Αρχιτεκτονική web services

Ο παροχέας της υπηρεσίας δημοσιεύει την περιγραφή της σε ένα ευρετήριο. Ο πιθανός χρήστης στη συνέχεια, βρίσκει την περιγραφή της υπηρεσίας μέσω του ευρετηρίου. Η περιγραφή περιέχει αρκετά στοιχεία για τον πιθανό χρήστη προκειμένου να συνδεθεί στον παροχέα και να χρησιμοποιήσει την υπηρεσία.

2.3.2 Πρότυπα για δικτυακές υπηρεσίες

Παρόλο που η αρχιτεκτονική που περιγράψαμε παραπάνω μπορεί να θεωρηθεί ανεξάρτητη από κάποιο πρότυπο είναι σαφές ότι η ανεξαρτησία εφαρμογών είναι απαραίτητη για να γίνει η παραπάνω (και η οποιαδήποτε άλλη) αρχιτεκτονική ευρέως αποδεκτή. Για να επιτευχθεί ο συγκεκριμένος σκοπός, ένα μεγάλο ποσοστό εκπροσώπων σημαντικών επιχειρήσεων προσπαθούν να ορίσουν ένα πρότυπο στηριγμένο στη γλώσσα XML που θα καταστήσει εφικτή την υλοποίηση της αρχιτεκτονικής των δικτυακών υπηρεσιών.

Πρωταρχικής σημασίας θεωρείται ο ορισμός μια συγκεκριμένης μεθόδου περιγραφής της κάθε δικτυακής υπηρεσίας. Για το σκοπό αυτό δημιουργήθηκε η περιγραφική γλώσσα WSDL.

2.3.2.1 Γλώσσα Περιγραφής Υπηρεσιών Διαδικτύου – WSDL

Η ευρέως χρησιμοποιούμενη και καθιερωμένη γλώσσα περιγραφής υπηρεσιών διαδικτύου είναι η WSDL (Web Services Description Language). Είναι μια γλώσσα βασισμένη στην XML και περιγράφει τρεις σημαντικές ιδιότητες της υπηρεσίας:

- Τι κάνει η υπηρεσία – Περιγραφή των μεθόδων της υπηρεσίας, οι παράμετροί της και τα αποτελέσματα που επιστρέφει.
- Πώς γίνεται η πρόσβαση στην υπηρεσία – Παρέχονται όλες οι απαραίτητες πληροφορίες για τη σύνταξη των SOAP μηνυμάτων που θα ανταλλάγουν, καθώς και τα πρωτόκολλα που υποστηρίζονται από τον παροχέα.
- Που βρίσκεται η υπηρεσία – Εδώ παρέχονται πληροφορίες για τη δικτυακή διεύθυνση της υπηρεσίας. Συνήθως είναι ένα URL.

Όπως κάθε καλά ορισμένη γλώσσα βασισμένη σε XML, έτσι και η WSDL ορίζει κάποια βασικά στοιχεία που συναντώνται σε ένα WSDL έγγραφο. Τα κυριότερα από αυτά είναι:

- portType – Είναι ένας αφαιρετικός ορισμός της διασύνδεσης της υπηρεσίας διαδικτύου. Ουσιαστικά, σκοπός του είναι η περιγραφή αυτής της διασύνδεσης. Όπως γίνεται σε έναν κώδικα Java, που η διασύνδεση αποτελείται μόνο από τις υπογραφές των μεθόδων, έτσι κι εδώ ο ορισμός του portType είναι η συλλογή των λειτουργιών που παρέχει η υπηρεσία διαδικτύου. Ένα τέτοιο στοιχείο μπορεί να έχει ένα απλό όνομα, το οποίο πιθανόν να σχετίζεται με τη λειτουργία που προσφέρει η υπηρεσία. Εάν ένα έγγραφο WSDL περιέχει πολλά τέτοια στοιχεία, τότε το κάθε ένα οφείλει να έχει διαφορετικό όνομα.
- Message – Καθορίζει τη μορφή του μηνύματος, καθώς και το σύνολο των παραμέτρων που σχετίζονται με τις μεθόδους της υπηρεσίας. Το στοιχείο

αυτό μπορεί να αποσυντεθεί σε πολλά μέρη, που ονομάζονται parts. Κάθε στοιχείο message μέσα σε ένα WSDL έγγραφο πρέπει να έχει ένα μοναδικό όνομα ώστε να ξεχωρίζει από τα υπόλοιπα. Ουσιαστικά, τα στοιχεία αυτά δεν παρουσιάζουν ιδιαίτερο ενδιαφέρον αφού δεν είναι τίποτε άλλο από μια συλλογή από parts και κάθε part έχει ένα όνομα, που συνήθως αντικατοπτρίζει την πληροφορία που περιέχεται στο part.

- **Types** – Ορίζει τη συλλογή όλων των τύπων δεδομένων που χρησιμοποιούνται από την υπηρεσία διαδικτύου. Το εξ ορισμού σύστημα των τύπων δεδομένων στην WSDL είναι το XML σχήμα (XML Schema – XSD). Το σχήμα αυτό μπορεί να χρησιμοποιηθεί για να περιγράψει όλους τους τύπους δεδομένων που μπορεί να χρησιμοποιηθούν στα διάφορα μηνύματα που ανταλλάσσονται για την κλήση της υπηρεσίας. Το στοιχείο types είναι ένα μέρος για το WSDL έγγραφο στο οποίο ο χρήστης ορίζει τύπους δεδομένων που θα χρησιμοποιηθούν σε άλλα στοιχεία του εγγράφου.
- **Binding** – Περιέχει λεπτομέρειες για το πώς τα στοιχεία σε μια αφαιρετική δομή (όπως το portType) μπορούν να μετασχηματισθούν σε μια συμπαγή αναπαράσταση των δεδομένων και πρωτοκόλλων που θα χρησιμοποιηθούν για την επικοινωνία μεταξύ του πελάτη και της υπηρεσίας. Ουσιαστικά είναι το στοιχείο που θα πει στον πελάτη πώς ακριβώς να μορφοποιήσει το μήνυμα που θα στείλει για την κλήση της υπηρεσίας. Κάθε στοιχείο portType μπορεί να έχει ένα ή περισσότερα στοιχεία binding συσχετισμένα με αυτό. Για ένα συγκεκριμένο στοιχείο portType, ένα στοιχείο binding περιγράφει τον τρόπο που πρέπει να γίνει η κλήση των μεθόδων της υπηρεσίας χρησιμοποιώντας κάποιο καθορισμένο πρωτόκολλο, όπως το SOAP, πάνω σε κάποιο επίσης καθορισμένο πρωτόκολλο μεταφοράς, όπως το HTTP. Το όνομα του binding πρέπει να είναι μοναδικό καθώς πολλά τέτοια στοιχεία μπορεί να υπάρχουν σε ένα WSDL έγγραφο.
- **Port** – Περιγράφει πως ακριβώς εκφράζεται ένα binding σε μια φυσική θύρα του δικτύου. Ο ρόλος της είναι πολύ απλός αφού δεν κάνει τίποτε άλλο από μια απλή αντιστοιχία. Τα στοιχεία αυτά, όπως και τα προηγούμενα, οφείλουν να φέρουν ένα όνομα, μοναδικό μέσα στο WSDL έγγραφο. Συνήθως, τα στοιχεία αυτά υποδεικνύουν το URL στο οποίο πρέπει να σταλούν τα μηνύματα SOAP ώστε να γίνει η κλήση της

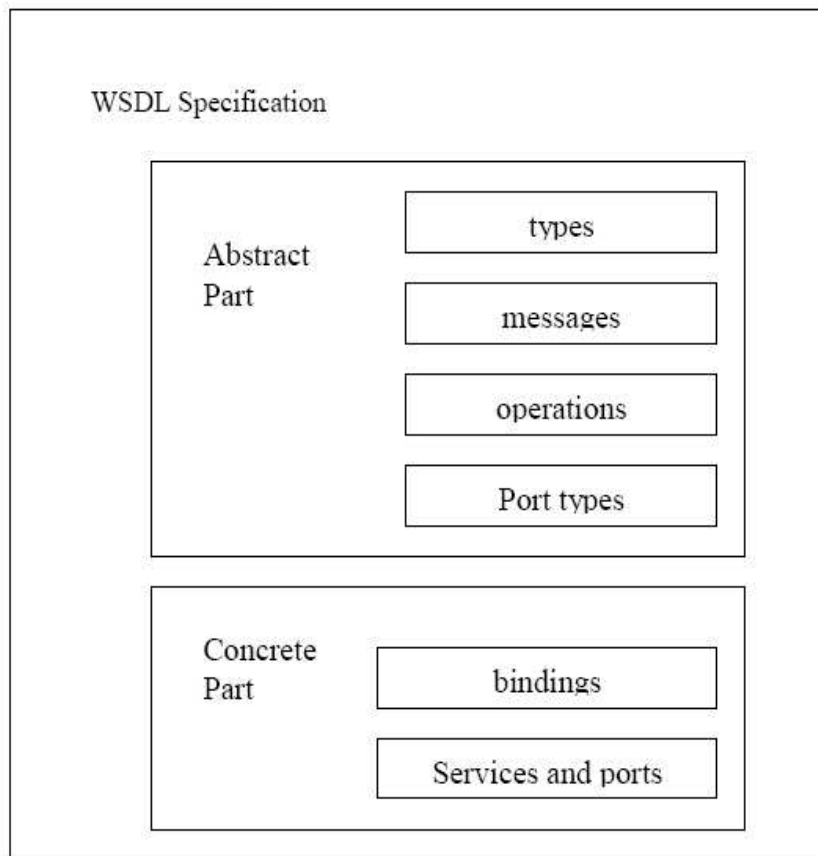
υπηρεσίας. Τα στοιχεία αυτά δε συναντούνται μόνο τους μέσα στο έγγραφο της περιγραφής της υπηρεσίας. Είναι στοιχεία – παιδιά του συστατικού service.

- Service – Ένα τέτοιο στοιχείο είναι μια συλλογή από ports. Το στοιχείο αυτό μπορεί να έχει ένα μοναδικό όνομα μέσα στο έγγραφο. Παρόλο που ο ρόλος του δεν έχει ιδιαίτερη σημασία, είναι καλό τα στοιχεία port να ομαδοποιούνται κάτω από ένα στοιχείο service. Συνήθως όμως ένα μόνο στοιχείο port συναντάται και έτσι καταλήγουμε στο να έχουμε ένα στοιχείο service, που να περιέχει ένα στοιχείο port.

Η δομή ενός WSDL εγγράφου ξεκινά με ένα στοιχείο ορισμών, το οποίο καλείται definitions. Το στοιχείο αυτό περιέχει όλα τα υπόλοιπα στοιχεία που μπορεί να συναντηθούν μέσα στο έγγραφο, δηλαδή μπορεί να περιέχει:

- Κανένα ή οσαδήποτε στοιχεία τεκμηρίωσης. Τα στοιχεία αυτά καλούνται documentation elements και χρησιμοποιούνται για να παρέχουν χρήσιμες πληροφορίες κατανοητές από τον άνθρωπο γύρω από την υπηρεσία διαδικτύου.
- Κανένα ή οσαδήποτε στοιχεία εισαγωγών. Τα στοιχεία αυτά καλούνται import elements. Τα στοιχεία αυτά δίνουν τη δυνατότητα επαναχρησιμοποίησης ήδη υπαρχόντων WSDL εγγράφων επιτρέποντας σε διάφορα στοιχεία να έχουν αναφορές σε άλλα ήδη υπάρχοντα που βρίσκονται σε διαφορετικό έγγραφο σε διαφορετικό αρχείο.
- Προαιρετικά ένα στοιχείο types.
- Κανένα ή οσαδήποτε στοιχεία message.
- Κανένα ή οσαδήποτε στοιχεία portType. Συνήθως μόνο ένα συναντάται.
- Κανένα ή οσαδήποτε στοιχεία binding. Συνήθως μόνο ένα συναντάται, το οποίο έχει να κάνει με το στοιχείο portType.
- Κανένα ή οσαδήποτε στοιχεία service. Και πάλι συνήθως μόνο ένα είναι αρκετό.

Τέλος, η έκδοση που χρησιμοποιείται ευρέως είναι η 2.0 της WSDL. στην οποία υπάρχει ο ισχυρισμός ότι υποστηρίζει κάποιου είδους σημασιολογικής πληροφορίας σε μικρό βαθμό.



Εικόνα 2 : ΔΟΜΗ WSDL

2.3.2.2 Πρωτόκολλο SOAP

Στα τέλη της δεκαετίας του 90, και συγκεκριμένα το 1997, μεγάλες εταιρείες, όπως η Microsoft, άρχισαν να διερευνούν κατά πόσο ο κατανεμημένος υπολογισμός μπορεί να βασιστεί στη γλώσσα XML. Ο σκοπός της έρευνας αυτής ήταν να γίνει εφικτή η επικοινωνία μεταξύ των εφαρμογών μέσω απομακρυσμένων κλήσεων διαδικασιών (Remote Procedure Calls – RPCs), χρησιμοποιώντας απλά πρωτόκολλα δικτύου, όπως το HTTP. **Error! Reference source not found.** Το 1999 έκανε την εμφάνισή του το SOAP, ένας RPC **Error! Reference source not found.** μηχανισμός βασισμένος σε XML. Από το 2000 ο οργανισμός W3C ασχολείται με την ιδέα αυτή και ύστερα από αρκετές αλλαγές, βελτιώσεις και τροποποιήσεις δυο ολόκληρων χρόνων, το 2003 δηλαδή, το SOAP με την έκδοση 1.2 γίνεται η προτεινόμενη προδιαγραφή – πρωτόκολλο για τις υπηρεσίες διαδικτύου.

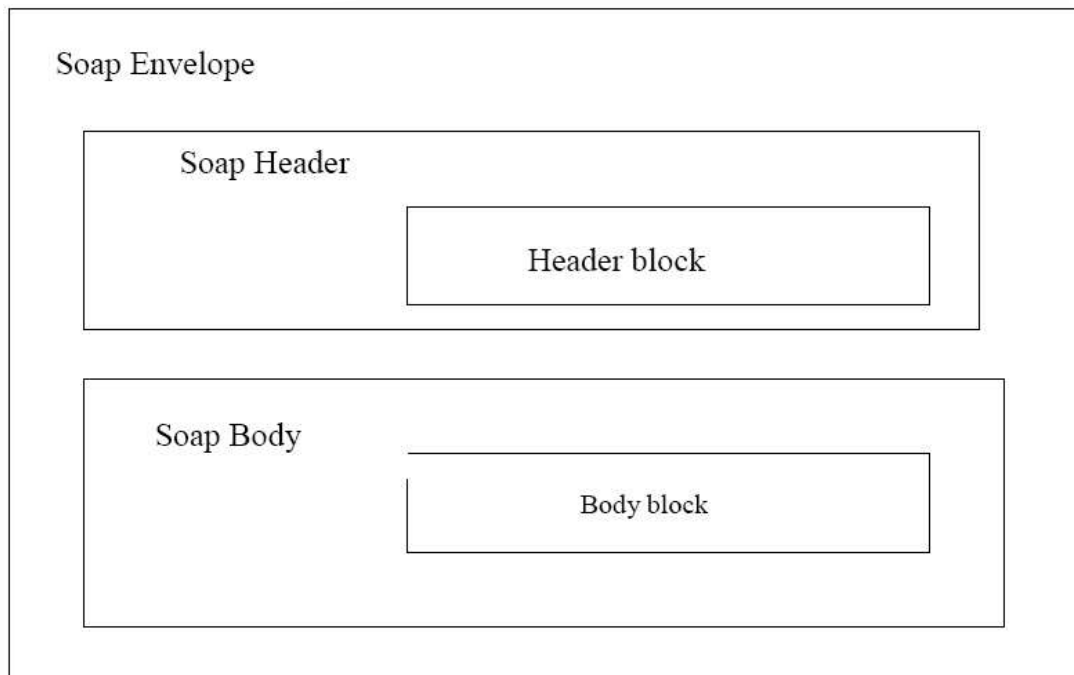
Στον πυρήνα του, το SOAP, είναι μια προδιαγραφή για ένα απλό αλλά ταυτόχρονα ευέλικτο XML πρωτόκολλο δεύτερης γενιάς. Το σημαντικό είναι ότι καθώς η έρευνα ξεκίνησε από τον κατανεμημένο υπολογισμό, το SOAP είναι το πλέον κατάλληλο πρωτόκολλο αφού ειδικεύεται σε τέτοια περιβάλλοντα. Το SOAP παρέχει τα κάτωθι χαρακτηριστικά και μηχανισμούς:

- Μηχανισμός για τον ορισμό της πληροφορίας στην επικοινωνία – Στο SOAP, όλη η πληροφορία είναι καταχωρημένη σε ένα ξεκάθαρο και ταυτοποιήσιμο SOAP μήνυμα (SOAP message). Αυτό γίνεται μέσω ενός SOAP φακέλου (SOAP envelope), που περικλείει όλες τις απαραίτητες πληροφορίες. Ένα μήνυμα SOAP, μπορεί να έχει σώμα (body), το οποίο περιέχει πληροφορία σε XML δομή. Επίσης, μπορεί να διαθέτει έναν αριθμό από επικεφαλίδες (headers), οι οποίες ενσωματώνουν επιπλέον πληροφορίες έξω από το σώμα του μηνύματος.
- Διεργασιακό μοντέλο – Αυτό το μοντέλο ορίζει ένα σύνολο κανόνων βάσει των οποίων γίνονται οι διαπραγματεύσεις μεταξύ των SOAP μηνυμάτων και του λογισμικού. Χαρακτηριστική είναι η απλότητά του.
- Μηχανισμός για αντιμετώπιση σφαλμάτων – Το SOAP παρέχει το μηχανισμό αυτό με τη μορφή των SOAP faults, τα οποία όταν χρησιμοποιούνται μπορεί να προσδιοριστεί η πηγή που προκάλεσε το σφάλμα. Επιπλέον, παρέχουν τη δυνατότητα ανταλλαγής διαγνωστικών πληροφοριών μεταξύ των μελών της επικοινωνίας.
- Μοντέλο επεκτασιμότητας – Το μοντέλο αυτό χρησιμοποιεί SOAP επικεφαλίδες για την υλοποίηση προεκτάσεων πάνω στο SOAP. Οι επικεφαλίδες περιέχουν κομμάτια από δεδομένα με δυνατότητα επέκτασης, τα οποία ταξιδεύουν μαζί με το μήνυμα και μπορούν να γίνουν στόχος για επέκταση σε συγκεκριμένους κόμβους του δικτύου.
- Ευέλικτος μηχανισμός για αναπαράσταση δεδομένων – Ο μηχανισμός αυτός επιτρέπει στα δεδομένα σε οποιαδήποτε σειριακή μορφή κι αν βρίσκονται να αναπαρασταθούν σε XML μορφή.
- Σύμβαση για αναπαράσταση των RPCs και των απαντήσεών τους σαν SOAP μηνύματα – Οι απομακρυσμένες κλήσεις διαδικασιών είναι αρκετά διαδεδομένες στον κατανεμημένο προγραμματισμό/ υπολογισμό και μπορούν να αναπαρασταθούν καλά μέσω του μηχανισμού αυτού σαν SOAP μηνύματα.
- Πρωτόκολλο εγκαθίδρυσης σύνδεσης – Το πρωτόκολλο αυτό ορίζει μια αρχιτεκτονική για την οικοδόμηση συνδέσεων επικοινωνίας ώστε να είναι εφικτή η ανταλλαγή SOAP μηνυμάτων πάνω σε μέσα μεταφοράς και επικοινωνιακά κανάλια. Χρησιμοποιείται το HTTP πρωτόκολλο, καθώς είναι το πιο διαδεδομένο και ευρέως χρησιμοποιούμενο στο διαδίκτυο.

Όσον αφορά στα μηνύματα SOAP, εφόσον το πρωτόκολλο είναι βασισμένο στην XML, είναι αναμενόμενο τα μηνύματα να έχουν μια τέτοια μορφή και ουσιαστικά να μην είναι τίποτε άλλο, παρά XML έγγραφα. Το στοιχείο-ρίζα του μηνύματος είναι το `soapenv:Envelope`, το οποίο περικλείει το στοιχείο `soapenv:Body`, που περιέχει πληροφορία σχετιζόμενη με το σκοπό του μηνύματος. Το στοιχείο `soapenv:Body` με τη σειρά του μπορεί να περιέχει άλλα στοιχεία, τα οποία να σχετίζονται ή να αναπαριστούν την απομακρυσμένη κλήση διαδικασίας που πρόκειται να εκτελεστεί. Το όνομα της μεθόδου που θα κληθεί και το όνομα του στοιχείου αυτού περιέχονται μέσα στο σώμα του μηνύματος.

Τα μηνύματα αυτά στέλνονται μέσω πρωτοκόλλου `http 1.1` με τη μέθοδο `POST`. Η απάντηση σε ένα τέτοιο μήνυμα, που και πάλι μεταφέρεται μέσω `HTTP` είναι και αυτή ένα SOAP μήνυμα. Το μήνυμα της απάντησης περικλείεται κι αυτό από το συστατικό `soapenv:Envelope`, που περιέχει το `soapenv:Body`, το οποίο ενσωματώνει και τη βασική πληροφορία του μηνύματος, στην οποία συγκαταλέγεται και μια κωδικοποιημένη αναπαράσταση του αποτελέσματος της απομακρυσμένης κλήσης που πραγματοποιήθηκε.

Το SOAP παρέχει, όπως ήδη αναφέρθηκε, τη δυνατότητα σε ένα μήνυμα να συμπεριλαμβάνονται στοιχεία – επικεφαλίδες. Ο ρόλος τους είναι και ο σημαντικότερος λόγος που χρησιμοποιούμε τα μηνύματα SOAP και όχι απλά XML έγγραφα. Τα στοιχεία αυτά (headers) έχουν την ιδιότητα να αναπαριστούν ένα επιπρόσθετο και επεκτάσιμο είδος πληροφορίας το οποίο μεταφέρεται μαζί με το υπόλοιπο μήνυμα, χωρίς να τροποποιείται ο κυρίως πυρήνας του μηνύματος. Για να γίνει κατανοητό αυτό ας δούμε ένα παράδειγμα της αληθινής ζωής ακριβώς αντίστοιχο με την ιδέα των SOAP μηνυμάτων. Ας υποθέσουμε ότι θέλουμε να στείλουμε ένα έγγραφο αλλά και κάποιες επιπρόσθετες πληροφορίες χωρίς όμως να θέλουμε να μαρκάρουμε το έγγραφο με αυτές. Τοποθετούμε, λοιπόν, το έγγραφο στο φάκελο και στη συνέχεια προσθέτουμε μία ή δύο σελίδες χαρτί, οι οποίες περιγράφουν την επιπλέον πληροφορία που θέλαμε να στείλουμε. Η αντιστοιχία του παραδείγματος αυτού με το πρωτόκολλο που περιγράφουμε είναι ότι ο φάκελος είναι το στοιχείο `soapenv:Envelope`, το έγγραφο που θέλουμε να στείλουμε είναι το στοιχείο `soapenv:Body` και οι σελίδες με την επιπλέον πληροφορία είναι τα στοιχεία επικεφαλίδες (headers).



Εικόνα 3 :Δομή SOAP

Ένα εργαλείο για την πραγματοποίηση SOAP συναλλαγών είναι ο Apache Axis, μια μηχανή SOAP ανοιχτού λογισμικού.

2.3.2.3 Universal Description, Discovery and Integration (UDDI)

Στις αρχές του 2000, και καθώς η ιδέα των υπηρεσιών ιστού άρχισε να κερδίζει έδαφος μέσα στην κοινότητα, έγινε σαφές ότι η καταχώρηση σε καταλόγους των υπηρεσιών ιστού ήταν κάτι απαραίτητο για να μπορέσει να εφαρμοστεί πρακτικά η ιδέα. Επιπλέον, λόγω της γενικότερης στροφής προς τα ανοικτά πρότυπα, οτιδήποτε εκτός από έναν πρότυπο μηχανισμό αλληλεπίδρασης και αναζήτησης δε θα μπορούσε να γίνει αποδεκτό. Ένα τέτοιο πρότυπο καταλόγου, θα έπρεπε να υποστηριχθεί και από τις περισσότερες, αν όχι από όλες τις μεγάλες εταιρίες παροχής λογισμικού, και να υιοθετηθεί από τις διάφορες βιομηχανίες. Έτσι η πρωτοβουλία του UDDI, που ήταν το αποτέλεσμα πολλών μηνών συνεργασίας μεταξύ αντιπροσώπων από τις Arriba, IBM, και Microsoft, που ξεκίνησε την άνοιξη του 2000, γεννήθηκε και ανακοινώθηκε επίσημα στις 6 Σεπτεμβρίου 2000. Η υποστήριξη για το UDDI επεκτάθηκε και πέρα από τις τρεις εταιρίες που συμμετείχαν αρχικά και αυτή τη στιγμή, το έργο UDDI περιλαμβάνει μια κοινότητα που αριθμεί πάνω από 310 εταιρίες.

Ο σκοπός του UDDI είναι να διευκολύνει την ανακάλυψη υπηρεσιών και κατά το χρόνο σχεδίασης, και δυναμικά κατά το χρόνο εκτέλεσης. Συνεπώς το UDDI λειτουργεί ως ένας κοινός άμεσα συνδεδεμένος κατάλογος (και των αντιστοίχων υπηρεσιών), το οποίο λειτούργησε για πρώτη φορά στις 2 Μαΐου 2001. Ο κατάλογος αυτός, συνήθως

αναφέρεται ως κατάλογος επιχειρήσεων UDDI (UDDI Business Registry). Ο κατάλογος αυτός, στην πραγματικότητα αποτελείται από δυο πανομοιότυπους καταλόγους, που διατηρούνται αυτή τη στιγμή από δύο εταιρίες (IBM και Microsoft), οι οποίοι και αποκαλούνται διαχειριστές του UDDI.

Για να γίνει κάποιος διαχειριστής ενός τέτοιου καταλόγου, πρέπει να ακολουθήσει αυστηρές συμφωνίες για την αντιγραφή των δεδομένων, το απόρρητο των δεδομένων και τις διάφορες πολιτικές. Από την οπτική μιας καταχωρημένης επιχείρησης αλλά και από την οπτική του απλού χρήστη, η απαίτηση είναι να μην υπάρχει διαφορά ανάμεσα στα διαφορετικά αντίγραφα του καταλόγου. Οι επιχειρήσεις μπορούν να κάνουν εγγραφή σε οποιονδήποτε διαχειριστή UDDI, και τα δεδομένα τους θα αντιγραφούν σε όλα τα άλλα αντίγραφα του καταλόγου στους άλλους διαχειριστές. Επομένως, οι χρήστες μπορούν να πραγματοποιήσουν αναζήτηση στον κατάλογο οποιουδήποτε διαχειριστή για να βρουν τις επιχειρήσεις που αναζητούν, άσχετα με το διαχειριστή που έχει επιλέξει η συγκεκριμένη επιχείρηση για να εγγραφεί. Υπάρχουν όμως κάποια λεπτά θέματα που έχουν σχέση με την απόφαση σχετικά με το ποιον διαχειριστή να χρησιμοποιήσει μια επιχείρηση για να εγγραφεί. Για παράδειγμα, κάποιοι διαχειριστές μπορεί να ζητούν επιπρόσθετες προαιρετικές πληροφορίες που δεν απαιτούνται από το UDDI. Αυτή η πληροφορία δεν αντιγράφεται στα αντίγραφα του καταλόγου στους άλλους διαχειριστές. Επιπλέον, αυτό που πρέπει να σημειωθεί είναι ότι από τη στιγμή που μια επιχείρηση επιλέξει κάποιο διαχειριστή για να εγγραφεί, η οποιαδήποτε ενημέρωση και τροποποίηση των στοιχείων θα πρέπει να γίνει από τον ίδιο διαχειριστή. Αυτό είναι αναγκαίο διότι ο κάθε διαχειριστής ακολουθεί διαφορετικές πολιτικές πιστοποίησης και ασφάλειας, και επομένως η πληροφορία πιστοποίησης δεν είναι εύκολο να αντιγραφεί στους άλλους διαχειριστές.

Το UDDI είναι κάτι παραπάνω από ένας κατάλογος επιχειρήσεων και υπηρεσιών. Ορίζει ένα σύνολο από δομές δεδομένων και προδιαγραφές API, για την προγραμματική αναζήτηση και καταχώρηση εταιριών, υπηρεσιών, δεσμών, και ειδών υπηρεσιών. Στα τυπικά σενάρια υπηρεσιών διαδικτύου, οι παροχείς υπηρεσιών διαδικτύου θα θέλουν να δημοσιεύουν τις περιγραφές των εταιριών και των υπηρεσιών τους σε έναν κατάλογο, και όσοι αναζητούν υπηρεσίες, είτε κατά τη σχεδίαση, είτε κατά την εκτέλεση, θα θέλουν να αναζητήσουν στον κατάλογο περιγραφές υπηρεσιών. Οι προδιαγραφές API του UDDI παρέχουν γι' αυτό το λόγο ένα σύνολο από προγραμματιστικές διεπαφές (API) δημοσίευσης για την καταχώρηση υπηρεσιών, και αντίστοιχες διεπαφές ερωτήσεων API για την εύρεση υπηρεσιών.

Επιπλέον από την παροχή μιας προγραμματιστικής διεπαφής, οι διαχειριστές του καταλόγου UDDI, παρέχουν ένα σύστημα αλληλεπίδρασης με το χρήστη, βασισμένο στο διαδίκτυο, για την καταχώρηση, διαχείριση, ανεύρεση εταιριών και υπηρεσιών μέσα στον κατάλογο.

2.3.2.4 Πρότυπα περιγραφής Επιχειρησιακών Διαδικασιών που βασίζονται σε υπηρεσίες.

Στόχος των δικτυακών υπηρεσιών είναι η επίτευξη καθολικής ανεξαρτησίας εφαρμογών μεταξύ των διαφόρων δικτυακών προτύπων. Σε πρώτη φάση, η ανεξαρτησία των εφαρμογών εξασφαλίζεται πλήρως χάρη στα τα τρία αυτά πρότυπα. Η χρήση πολλών διαφορετικών συστημάτων, όμως, απαιτεί περισσότερα από τη δυνατότητα της διεξαγωγής απλών συναλλαγών μεταξύ συγκεκριμένων και ανεξάρτητων πρωτοκόλλων. Η πλήρης δυναμική των δικτυακών υπηρεσιών ως πλατφόρμα χρήσης ανεξάρτητων εφαρμογών θα επιτευχθεί μόνο αν οι εφαρμογές και οι επιχειρησιακές διαδικασίες γίνουν ικανές να υλοποιήσουν τις πολύπλοκες συναλλαγές τους με ένα πρότυπο ανεξάρτητο από τις ίδιες.

Γενικά τα πρότυπα για επιχειρησιακές διαδράσεις χρησιμοποιούν peer to peer ανταλλαγές μηνυμάτων τόσο σύγχρονων όσο και ασύγχρονων. Για την περιγραφή αυτών των επιχειρησιακών διαδράσεων χρειάζεται μια επίσημη περιγραφή των πρωτοκόλλων ανταλλαγής μηνυμάτων. Η περιγραφή τέτοιων επιχειρησιακών πρωτοκόλλων περιλαμβάνει τον ακριβή προσδιορισμό της συμπεριφοράς των μηνυμάτων που ανταλλάσσονται έτσι ώστε να μην αποκαλύπτεται ο ακριβής τρόπος υλοποίησης των εσωτερικών διαδικασιών που χρησιμοποιεί η μια επιχείρηση από την άλλη.

Υπάρχουν δύο καλοί λόγοι για να διαχωρίσουμε το δημόσιο από το ιδιωτικό μέρος μιας επιχειρησιακής διαδικασίας. Ο πρώτος είναι ότι οι επιχειρήσεις προφανώς δεν θέλουν να αποκαλύπτουν όλη την εσωτερική διαδικασία λήψης αποφάσεων και διαχειρίσεις των δεδομένων τους στους συνεργάτες τους. Ο άλλος είναι ότι ο διαχωρισμός μεταξύ του δημόσιου και του ιδιωτικού μέρους παρέχει την ελευθερία της αλλαγής του ιδιωτικού μέρους της υλοποίησης χωρίς να επηρεάζεται το πρωτόκολλο του δημοσίου μέρους.

Τα επιχειρησιακά πρωτόκολλα πρέπει να περιγράφονται με σαφήνεια με τρόπο ανεξάρτητο της εκάστοτε πλατφόρμας και να περιλαμβάνουν όλες τις πτυχές της συμπεριφοράς που έχουν διεπιχειρησιακή σημασία. Κάθε συμμετέχων μπορεί να σχεδιάζει επιχειρησιακές διαδικασίες συμβατές με το παραπάνω πρωτόκολλο χωρίς να

εντάσσεται στο σχεδιασμό η οποιαδήποτε ανθρώπινη συμφωνία. Με τον τρόπο αυτό αποφεύγεται η κύρια δυσκολία που υπάρχει σήμερα στη δημιουργία διεπιχειρησιακών διαδικασιών.

Προκειμένου να προσδιορίσουμε τις κύριες έννοιες που απαιτούνται για να περιγράψουμε ένα επιχειρησιακό πρωτόκολλο όπως το περιγράψαμε παραπάνω, πρέπει να θεωρήσουμε τα εξής:

- Τα επιχειρησιακά πρωτόκολλα περιλαμβάνουν συμπεριφορά εξαρτώμενη από τα δεδομένα που χρησιμοποιούν.
- Η ικανότητα να προσδιορίζουμε οριακές συνθήκες εκτός του πλαισίου λειτουργίας, περιλαμβανομένων και τρόπων ανάνηψης του συστήματος από αυτές, είναι εξίσου σημαντική με την υπό φυσιολογικές συνθήκες καλή λειτουργία του.
- Οι μακράς διάρκειας διαδράσεις περιλαμβάνουν πολλαπλές, συχνά εμφωλευμένες μονάδες εργασίας, η καθεμιά με τις δικές της απαιτήσεις δεδομένων. Για το λόγο αυτό τα επιχειρησιακά πρωτόκολλα απαιτούν την ευκολία στον προσδιορισμό των συντεταγμένων, στις οποίες βρίσκονται τα αποτελέσματα της εκάστοτε μονάδας εργασίας (διαδικασίας).

Αν θέλαμε, λοιπόν, να ορίσουμε ακριβείς περιγραφές των διαφόρων υπηρεσιών για κάποιο διεπιχειρησιακό Business Process πρωτόκολλο, χρειαζόμαστε μια πλούσια περιγραφικά γλώσσα, η οποία θα περιέχει και πολλά στοιχεία γλώσσας εκτέλεσης. Το κλειδί για την επιτυχημένη υλοποίηση μιας τέτοιας γλώσσας είναι ο διαχωρισμός μεταξύ των δημόσιων μηνυμάτων που ανταλλάσσονται και των εσωτερικών διαδικασιών που εκτελούνται έτσι ώστε οι διαδικασίες αυτές να μπορούν να χειρίζονται δεδομένα, χωρίς να χρειάζεται να περιγραφούν από δημόσιο πρωτόκολλο.

Παρακάτω παραθέτουμε μια αναλυτική περιγραφή των δύο γλωσσών που ήδη υπάρχουν στο συγκεκριμένο τομέα: της BPEL4WS και της BPML.

2.3.2.4.1 BPEL

Η BPEL4WS ακολουθεί πλήρως το πρότυπο που περιγράψαμε παραπάνω. Όσον αφορά τα μηνύματα, τα ανταλλασσόμενα μεταξύ διαδικασιών και επιχειρήσεων, χρησιμοποιεί τη λογική των ιδιοτήτων μηνύματος προκειμένου να αναγνωρίσει τα σχετικά με το πρωτόκολλο δεδομένα που εμπεριέχονται σ' αυτά. Οι ιδιότητες αυτές μπορεί να θεωρηθούν ότι είναι διαφανή δεδομένα σε μια προοπτική δημόσιας ανταλλαγής δεδομένων, η οποία αντιτίθεται στα αδιαφανή δεδομένα που ανταλλάσσονται μεταξύ εσωτερικών / ιδιωτικών συναρτήσεων που χρησιμοποιούνται.

Τα διαφανή δεδομένα επηρεάζουν το επιχειρησιακό πρωτόκολλο με έναν άμεσο τρόπο, ενώ τα αδιαφανή δεδομένα το επηρεάζουν μόνο δημιουργώντας μη-ντετερμινισμό, γιατί ο τρόπος, με τον οποίο επηρεάζουν τις αποφάσεις είναι αδιαφανής. Η BPEL4WS θεωρεί ότι κάθε δεδομένο πρέπει να είναι διαφανές και γι' αυτό το λόγο το βλέπει σαν ιδιότητα του μηνύματος.

Οι έννοιες που χρειάζονται για να ορίσουμε ένα επιχειρησιακό πρωτόκολλο, καθώς και οι έννοιες που χρειάζονται για να ορίσουμε ένα εκτελέσιμο Business Process δημιουργούν μια εννοιολογική αλυσίδα και η BPEL4WS είναι σχεδιασμένη για να καλύψει όλο το εύρος της. Η γλώσσα αυτή ορίζει ένα μοντέλο και μια γραμματική προκειμένου να περιγράψει τη συμπεριφορά μιας Business Process βασισμένη σε συναλλαγές μεταξύ της διαδικασίας και των επιμέρους χρηστών της. Η αλληλεπίδραση με τον κάθε χρήστη υλοποιείται μέσω διεπαφών που αντιστοιχούν σε δικτυακές υπηρεσίες, και η δομή της σχέσης στο επίπεδο της διεπαφής είναι ενθυλακωμένη σε αυτό που ονομάζουμε σύνδεσμο υπηρεσιών (Service Link). Η BPEL4WS διαδικασία ορίζει το πως διαφορετικές υπηρεσίες μπορούν να αλληλεπιδράσουν με τους χρήστες της προκειμένου να πετύχουν έναν επιχειρησιακό στόχο, καθώς και την λογική και την κατάσταση που απαιτείται για να πραγματοποιηθεί ένας τέτοιος προσανατολισμός. Η γλώσσα αυτή, επίσης, εισάγει συστηματικούς μηχανισμούς για την επίτευξη του ελέγχου πιθανών επιχειρησιακών απρόβλεπτων οριακών συνθηκών και σφαλμάτων της διαδικασίας. Τέλος, η BPEL4WS εισάγει ένα μηχανισμό για τον καθορισμό του πως ατομικές ή σύνθετες λειτουργίες εντός μιας διαδικασίας, μπορούν να αποζημιωθούν στην περίπτωση που οι παραπάνω εξαιρέσεις εμφανιστούν ή ένας συνεργάτης / χρήστης ζητήσει ακύρωση της συμμετοχής του στην διαδικασία.

Η βασική ιδέα της BPEL4WS μπορεί να εφαρμοστεί με δύο τρόπους. Μια διαδικασία της μπορεί να ορίσει ένα ρόλο επιχειρησιακού πρωτοκόλλου χρησιμοποιώντας την ιδέα της αφηρημένης διαδικασίας (abstract process). Π.χ. σε ένα πρωτόκολλο εμπορικής συναλλαγής, ο πωλητής και ο αγοραστής είναι δύο διακριτοί ρόλοι, ο καθένας εκ των οποίων αποτελεί μια τέτοια αφηρημένη διαδικασία. Η σχέση τους είναι τυπικά μοντελοποιημένη σαν ένας σύνδεσμος υπηρεσίας (Service Link). Οι αφηρημένες διαδικασίες χρησιμοποιούν όλο το εύρος των ιδεών της BPEL4WS, αλλά προσεγγίζουν τον χειρισμό των δεδομένων με τρόπο που αντανάκλα το επίπεδο της αφαίρεσης που χρειάζεται για να περιγράψουμε τις διάφορες προοπτικές ενός επιχειρησιακού πρωτοκόλλου. Πιο συγκεκριμένα, οι αφηρημένες διαδικασίες διαχειρίζονται μόνο δεδομένα που σχετίζονται με το επιχειρησιακό πρωτόκολλο. Επιπρόσθετα, οι

αφηρημένες διαδικασίες χρησιμοποιούν μη-ντετερμινιστικές τιμές δεδομένων προκειμένου να κρύψουν τα ιδιωτικά μέρη της συμπεριφοράς τους.

Είναι επίσης δυνατό να χρησιμοποιήσουμε την BPEL4WS για να ορίσουμε μια εκτελέσιμη Business Process. Η λογική και η κατάσταση της διαδικασίας ορίζουν τη φύση και την ακολουθία των αλληλεπιδράσεων μεταξύ δικτυακές υπηρεσίες που διεξάγονται σε κάθε χρήστη / συνεργάτη της και άρα και στα πρωτόκολλα αλληλεπίδρασης. Όσο ένας ορισμός μιας διαδικασίας σε BPEL4WS δε χρειάζεται να είναι ολοκληρωμένος όσον αφορά το ιδιωτικό μέρος της υλοποίησης, η γλώσσα αποτελεσματικά ορίζει μια μεταφέρσιμη μορφή εκτέλεσης για Business Processes, η οποία στηρίζεται αποκλειστικά στους πόρους της δικτυακής υπηρεσίας και στα δεδομένα σε μορφή XML. Επιπλέον, τέτοιες διαδικασίες εκτελούνται και αλληλεπιδρούν με τους χρήστες / συνεργάτες με ένα συνεπή τρόπο ανεξάρτητα από την πλατφόρμα που τις υποστηρίζει ή το προγραμματιστικό μοντέλο που χρησιμοποιείται για την υλοποίηση του περιβάλλοντος που τη φιλοξενεί.

Ακόμα και όταν τα ιδιωτικά μέρη της υλοποίησης χρησιμοποιούν λειτουργίες που εξαρτώνται από τις πλατφόρμες στις οποίες έχουν δημιουργηθεί, κάτι που είναι πιθανό στις περισσότερες πραγματικές περιπτώσεις, η λογική συνέχεια του βασικού μοντέλου μεταξύ αφηρημένων και επιχειρησιακών διαδικασιών στην BPEL4WS κάνει δυνατή την εξαγωγή και εισαγωγή των δημόσιων παραμέτρων, που είναι ενσωματωμένες στο επιχειρησιακό πρωτόκολλο σαν διαδικασίες, ή σαν ρόλοι, διατηρώντας ταυτόχρονα το σκοπό και τη δομή των πρωτοκόλλων. Αυτό είναι κατά γενική ομολογία η πιο ελκυστική προοπτική για να χρησιμοποιήσει κανείς την BPEL4WS από την οπτική ότι απελευθερώνει τις δυνατότητες των δικτυακών υπηρεσιών εφόσον επιτρέπει την ανάπτυξη εργαλείων και άλλων τεχνολογιών που αυξάνουν το επίπεδο της αυτοματοποίησης και συνεπώς μειώνουν το κόστος εγκατάστασης διεπιχειρησιακών αυτόματων επιχειρησιακών διαδικασιών.

Η BPEL4WS έχει διαστρωματωθεί στην κορυφή διάφορων XML προτύπων: WSDL 1.1, XML Schema και XPath 1.0. Τα WSDL μηνύματα και οι ορισμοί τύπων στο XML Schema παρέχουν το μοντέλο δεδομένων που χρησιμοποιείται από τις διαδικασίες της BPEL4WS. Το XPath παρέχει υποστήριξη στη διαχείριση των δεδομένων. Όλοι οι εξωτερικοί πόροι και συνεργάτες αναπαρίστανται σαν WSDL υπηρεσίες. Η BPEL4WS παρέχει επεκτασιμότητα προκειμένου να συμπεριλάβει στο μέλλον πιθανές επεκτάσεις των συγκεκριμένων προτύπων και πιο ειδικά το XPath και τα συσχετιζόμενα πρότυπα που χρησιμοποιούνται στη δημιουργία αρχείων XML.

Μεταξύ των παραπάνω προτύπων η WSDL έχει την καλύτερη επιρροή πάνω στην γλώσσα BPEL4WS. Το μοντέλο μιας BPEL4WS διαστρωματώνεται στην κορυφή του υπηρεσιακού μοντέλου που ορίζει η WSDL 1.1. Στο κέντρο του BPEL4WS μοντέλου διαδικασίας είναι η peer-to-peer αλληλεπίδραση μεταξύ υπηρεσιών που περιγράφονται από την WSDL 1.1. Τόσο οι διαδικασίες όσο και οι χρήστες / συνεργάτες μοντελοποιούνται σαν WSDL υπηρεσίες. Μια Business Process ορίζει τον τρόπο με τον οποίο μπορεί να γίνει ο προσανατολισμός των αλληλεπιδράσεων μεταξύ ενός στιγμιότυπου μιας διαδικασίας και των χρηστών / συνεργατών της. Ένας ορισμός μιας BPEL4WS διαδικασίας παρέχει και / ή χρησιμοποιεί μία ή περισσότερες υπηρεσίες, και χρησιμοποιεί την περιγραφή της συμπεριφοράς και της αλληλεπίδρασης ενός στιγμιότυπου μιας διαδικασίας με τους συνεργάτες της και τους πόρους μέσω διεπαφών δικτυακών υπηρεσιών. Δηλαδή η BPEL4WS ορίζει πρωτόκολλα ανταλλαγής μηνυμάτων ακολουθούμενα από μία επιχειρησιακή διαδικασία ενός συγκεκριμένου ρόλου στη διάδραση.

Ο ορισμός μιας BPEL4WS διαδικασίας ακολουθεί επίσης και το μοντέλο WSDL όσον αφορά το διαχωρισμό ανάμεσα σε αφηρημένα περιεχόμενα μηνυμάτων, όπως αυτά χρησιμοποιούνται σε μια επιχειρησιακή διαδικασία, καθώς πληροφορίες για την υλοποίησή της (μηνύματα και τύπους θυρών ενάντια σε πληροφορίες διεύθυνσης και σύνδεσης). Συγκεκριμένα, μια BPEL4WS αναπαριστά τους συνεργάτες / χρήστες, καθώς και τις αλληλεπιδράσεις μεταξύ αυτών στα πλαίσια των WSDL διεπαφών (τύπους θυρών και λειτουργιών). Καμιά αναφορά δε γίνεται στις πραγματικές υπηρεσίες που χρησιμοποιούνται από το συγκεκριμένο στιγμιότυπο της διαδικασίας. Μια BPEL4WS διαδικασία είναι ένας επαναχρησιμοποιούμενος ορισμός που μπορεί να υλοποιηθεί με πολλούς τρόπους και πολλά σενάρια, ενώ διατηρεί μια ενιαία συμπεριφορά στο επίπεδο εφαρμογής σε καθένα από αυτά τα σενάρια. Αξίζει να σημειώσουμε ότι η περιγραφή της υλοποίησης μιας BPEL4WS διαδικασίας ξεφεύγει από τους σκοπούς του συγκεκριμένου κεφαλαίου.

2.3.2.4.2 BPML

Η BPML έχει παρόμοια λογική με την BPEL4WS. Το σημείο στο οποίο διαφέρουν κυρίως είναι ο τρόπος υλοποίησης.

Σκοπός λοιπόν της BPML είναι να ορίσει ένα αφαιρετικό μοντέλο και ένα συντακτικό βασισμένο σε XML προκειμένου να περιγράψει εκτελέσιμες επιχειρησιακές διαδικασίες και τις επιμέρους οντότητες που αυτές υποστηρίζουν. Η ίδια η BPML δεν ορίζει κάποια ειδική εφαρμογή που θα αποτελείται από επιχειρησιακές διαδικασίες αλλά περιορίζεται

μόνο στη περιγραφή τους. Πιο συγκεκριμένα ορίζει το αφαιρετικό μοντέλο και τη γραμματική που απαιτείται για να περιγράψει κανείς μια εκτελέσιμη επιχειρησιακή διαδικασία. Το γεγονός αυτό δίνει στη BPML τη δυνατότητα να χρησιμοποιείται για πολλούς σκοπούς πέραν της περιγραφής απλών επιχειρησιακών διαδικασιών, όπως σύνθετες δικτυακές υπηρεσίες(επιχειρησιακές διαδικασίες αποτελούμενες από δικτυακές υπηρεσίες) και διαδικασίες που περιλαμβάνουν την συνεργασία πολλών διαφορετικών και ανεξάρτητα μεταξύ τους μερών.

Η BPML στηρίζει το πρότυπό της σε μια σειρά άλλων προτύπων. Τα πρότυπα αυτά αναφέρονται παρακάτω.

- XML 1.0
- XML – namespaces
- XML- schema
- Xpath 1.0
- WSDL 1.1

Η BPML προβλέπει για την υλοποίησή της τους εξής τύπους οντοτήτων:

- BPML constructs: Είναι οι βασικές κατασκευαστικές μονάδες που περιγράφουν μια διαδικασία σε BPML. Οι μονάδες αυτές περιγράφονται με την βοήθεια ειδικής γραμματικής στηριγμένη σε XML.
- BPML definitions: Είναι μια συλλογή από τα ονόματα των BPML constructs που χρησιμοποιούνται για την περιγραφή της συγκεκριμένης διαδικασίας. Τα ίδια αποτελούν και αυτά ένα BPML construct.
- BPML package: Είναι μια συλλογή από BPML definitions που μπορεί να περιέχουν και definitions από άλλες γλώσσες όπως XML schema ή WSDL 1.1
- BPML documents: Είναι XML αναπαραστάσεις ενός BPML package βασισμένα στο συντακτικό που ορίζει η ίδια η BPML.

Όπως είδαμε η BPML μοιάζει πολύ με την BPEL4WS τόσο στη σχεδίαση όσο και στην υλοποίηση . Δεν κρίνεται σκόπιμο να εμβαθύνουμε περισσότερο στις δυο γλώσσες. Αυτό όμως που επιβάλλεται να γίνει είναι μια προσπάθεια σύγκρισής τους.

2.3.2.4.3 Σύγκριση των δύο γλωσσών

Ο ανταγωνισμός μεταξύ των δύο γλωσσών είναι πολύ μεγάλος αφού όλες οι μεγάλες εταιρίες στο χώρο της ανάπτυξης λογισμικού ενδιαφέρονται για την ενιαία περιγραφή

των επιχειρησιακών διαδικασιών και έχουν πολωθεί προς τη μια ή την άλλη κατεύθυνση. Εμείς θα προσπαθήσουμε μια σύγκριση μεταξύ των όσο το δυνατό πιο αντικειμενικά και αμερόληπτα.

Προκειμένου να πετύχουμε κάτι τέτοιο θα απαριθμήσουμε ονομαστικά κάποια κοινά χαρακτηριστικά και διαφορές που θεωρούνται αποδεκτά και από τις δύο πλευρές. Τα χαρακτηριστικά αυτά είναι τα εξής :

- Η BPML περιλαμβάνει πιο γενικευμένες οντότητες από την BPEL4WS και μπορεί γενικώς να περιγράψει περισσότερες μορφές επιχειρησιακών διαδικασιών.
- Και οι δύο γλώσσες χρησιμοποιούν τα ίδια ιδιώματα και παρόμοιο συντακτικό.
- Και οι δύο έχουν την ικανότητα να περιγράφουν πλήρως σύνθετες όσο και απλές επιχειρησιακές διαδικασίες.
- Και οι δύο γλώσσες στηρίζουν το συντακτικό τους σε επιμέρους προγραμματιστικά σύνολα το καθένα από τα οποία περιγράφει όμοιες οντότητες.
- Και οι δύο γλώσσες παρέχουν δυνατότητες για επαναληπτικές, συγχρονισμένες και δυναμικές διαδικασίες.
- Ενώ η BPML περιγράφει πιο γενικευμένες οντότητες από την BPEL4WS, η BPEL4WS περιγράφει με πιο πλήρη τρόπο διαδικασίες που περιλαμβάνουν δικτυακές υπηρεσίες.

Κεφάλαιο 3

3.1 ΕΙΣΑΓΩΓΗ

Αυτό το κεφάλαιο συζητά το Business Process Execution Language για Web Service (BPEL4WS) Version 1.1 πρότυπο, και παρέχει μια λεπτομερή περιγραφή της.

Σε αυτό το τμήμα συζητάμε τις επιχειρησιακά διαδικασίες και τα πρωτόκολλα τους καθώς και τη σχέση τους με τα ανοικτά πρότυπα βιομηχανίας BPEL.

Συζητάμε επίσης τα σχετικά πρότυπα:

- WSDL
- XSD
- XPath

Επισκόπηση

Η Business Process Execution Language προδιαγραφή καθορίζει πρότυπα για τον καθορισμό επιχειρησιακών διαδικασιών στη χρήση Υπηρεσιών Ιστού. Μία business process είναι ένα σύνολο δραστηριοτήτων που αντιπροσωπεύουν μία πραγματική λειτουργία. Η BPEL υποστηρίζει τη σύνθεση διαδικασιών με την κατασκευή και τη σύνδεση των διαφορετικών δραστηριοτήτων για να δημιουργήσει μια business process.

3.2 Βασικές έννοιες της BPEL

Ένα πολύ σημαντικό, εάν όχι το πιο σημαντικό στοιχείο για την χρήση της BPEL είναι η περιγραφή των επιχειρησιακών αλληλεπιδράσεων στις οποίες κάθε επιχειρησιακό στοιχείο αλληλεπιδρά μέσω των web service διεπαφών με άλλα επιχειρησιακά στοιχεία. Μια σημαντική απαίτηση για ρεαλιστική μοντελοποίηση των επιχειρησιακών διαδικασιών σε αυτό το περιβάλλον είναι η ικανότητα να μοντελοποιήσουμε τις απαραίτητες σχέσεις μεταξύ των στοιχείων. Η σχέση μιας επιχειρησιακής διαδικασίας με μια partner της είναι σχέση peer-to-peer που απαιτεί μια αμφίδρομη εξάρτηση τους. Με άλλα λόγια ένας partner είναι ένας καταναλωτής μιας υπηρεσίας που παρέχεται από την επιχειρησιακή διαδικασία αλλά και πάροχος μίας υπηρεσίας στην επιχειρησιακή διαδικασία. Ο ορισμός των Partner links χρησιμοποιείται για να μοντελοποιηθούν αμφίδρομες σχέσεις επικοινωνίας μεταξύ των partners. Τα Partner links καθορίζουν τη δομή της σχέσης των partners μέσω του καθορισμού των μηνυμάτων και των port types που χρησιμοποιούνται.

Κατά τη διαμόρφωση των επιχειρησιακών διαδικασιών, πρέπει να περιγράψουμε τη σχέση μεταξύ μιας επιχειρησιακής διαδικασίας και των διαδικασιών των συνεργατών της επιχείρησης.

Ένας συνεργάτης μπορεί να είναι:

- Ένας καταναλωτής μιας υπηρεσίας που παρέχεται μέσω μιας επιχειρησιακής διαδικασίας
- Ένας παροχέας μιας υπηρεσίας, η οποία χρησιμοποιείται με μια επιχειρησιακή διαδικασία
- Υπηρεσίες που μια επιχειρησιακή διαδικασία επικαλείται και που επικαλούνται επίσης τη διαδικασία

3.2.1 Partner Link Types

Ένας Partner Link Type περιγράφει την σχέση επικοινωνίας μεταξύ δύο υπηρεσιών μέσω του καθορισμού των ρόλων που παίζει κάθε υπηρεσία στην επικοινωνία που οι δύο υπηρεσίες WSDL σκοπεύουν να πραγματοποιήσουν. Ένας Partner Link Type χαρακτηρίζει αυτήν την επικοινωνία μέσω του καθορισμού των ρόλων που διαδραματίζονται από κάθε υπηρεσία. Είναι υπεύθυνος για τον καθορισμό των port types που παρέχεται από την υπηρεσία μέσω των οποίων θα πραγματοποιηθεί η ανταλλαγή μηνυμάτων με κάποια άλλη partner υπηρεσία.

```

<partnerLinkType name="BuyerSellerLink"
  xmlns="http://schemas.xmlsoap.org/ws/2003/05/partner-link/">
  <role name="Buyer">
    <portType name="buy:BuyerPortType"/>
  </role>
  <role name="Seller">
    <portType name="sell:SellerPortType"/>
  </role>
</partnerLinkType>

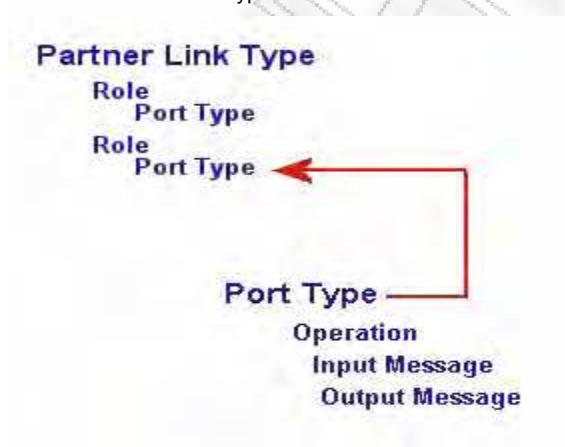
```

Εικόνα 4: Βασική σύνταξη μιας Partner Link Type

Ένας Partner Link Type μπορεί να περιλάβει έναν ρόλο ή δύο ρόλους. Στην περίπτωση όπου ένας Partner Link Type περιέχει δύο ρόλους, κάθε μια από τις υπηρεσίες πρέπει να εφαρμόσει το ρόλο της με την παροχή του port type που διευκρινίζεται. Η χρήση δύο ρόλων δείχνει ότι υπάρχει μια απαίτηση για την καλούμενη υπηρεσία να λάβει κάποιο τύπο επανάκλησης από την υπηρεσία στόχων κατά τη διάρκεια της συνομιλίας.

Στην περίπτωση όπου ένας Partner Link Type περιέχει μόνο έναν ρόλο, δεν υπάρχει κανένας περιορισμός που τοποθετείται στην καλούμενη web service σχετικά με τους ρόλους. Η υπηρεσία που περιγράφεται από τον ένα ρόλο είναι σε θέση να ολοκληρώσει τη συνομιλία της χωρίς την ανάγκη για μια επανάκληση. Η ακόλουθη απεικόνιση παρουσιάζει ένα παράδειγμα ενός τύπου συνδέσεων συνεργατών καθορίζοντας δύο ρόλους.

Εικόνα 5 :Partner Link Type Definition



Εικόνα 6 : Partner Link Type Example



Παραδείγματος χάριν, μια partner link type που ονομάζεται AuctionHouse_Seller ν περιγράφει δύο ρόλους: AuctionHouse και Seller. Ο ρόλος AuctionHouse

υποστηρίζει έναν port type Seller ο οποίος αναμένει ένα μήνυμα εισαγωγής ενός στοιχείου για πώληση. Ο ρόλος Seller υποστηρίζει έναν port type Seller_Answer που αναμένει ένα μήνυμα εισαγωγής από το AuctionHouse, περιμένοντας εάν το στοιχείο έχει πουληθεί ή όχι. Το output message είναι το όνομα του στοιχείου προς πώληση.

Σε αυτό το παράδειγμα, τα port types είναι από την ίδια υπηρεσία εντούτοις, τα port types μπορούν να είναι από δύο διαφορετικές υπηρεσίες.

Ο Partner Link Type είναι μια επέκταση WSDL. Μπορεί να διευκρινίσει έναν ή δύο ρόλους. Τα port types μπορούν να είναι από το ίδιο ή από τα διαφορετικά αρχεία WSDL. Ένας καθορισμός Partner Link Type που χρησιμοποιείται με τη διαδικασία σας μπορεί να προέλθει από τις ακόλουθες πηγές:

- Από ένα αρχείο WSDL, όπου καθορίζεται ήδη, ή πού μπορείτε να προσθέσετε έναν νέο καθορισμό
- Από ένα χωριστό αρχείο WSDL με το namespace του, για την περίπτωση όπου υπάρχουν δύο port types, και είναι από τις διαφορετικές υπηρεσίες

3.2.2 Partner link

Οι υπηρεσίες με τις οποίες μια επιχειρησιακή διαδικασία αλληλεπιδρά μοντελοποιούνται σαν Partner links στην BPEL. Κάθε partner link χαρακτηρίζεται από ένα Partner Link Type. Περισσότερα από ένα partner links μπορούν να χαρακτηρίζονται από το ίδιο partner link type.

```
<partnerLinks>
  <partnerLink name="ncname" partnerLinkType="qname"
    myRole="ncname"? partnerRole="ncname"?>+
</partnerLink>
</partnerLinks>
```

Εικόνα 7 : Partner Link

Ένας καθορισμός partner link αποτελείται από:

- Ένα όνομα
- Ένα partner link type
- Τουλάχιστον ένας ρόλος και μπορεί να περιλάβει δύο ρόλους
- Η προαιρετική ιδιότητα αποκαλούμενη Initialize Partner Role

- Προαιρετικές ιδιότητες για Προσθήκη των σχολίων σε μια διαδικασία (Adding Comments to a Process) και Προσθήκη της τεκμηρίωσης σε μια διαδικασία (Adding Documentation to a Process)
- Προαιρετικές ιδιότητες για Δήλωση των στοιχείων και των ιδιοτήτων επέκτασης (Declaring Extension Elements and Attributes)

Σε μια διαδικασία BPEL, χρησιμοποιούμε την partner link κατά την καθορισμό των δραστηριοτήτων αλληλεπίδρασης υπηρεσιών Ιστού. Παραδείγματος χάριν, μια receive δραστηριότητα διευκρινίζει μια partner link, το port type και τη λειτουργία του. Το port type προέρχεται από το σχετικό partner link type. Αυτό σημαίνει ότι η διαδικασία μπορεί να δεχτεί τα εισερχόμενα στοιχεία μηνυμάτων από την υπηρεσία που διαδραματίζει το ρόλο που καθορίζεται στην partner link.

Τουλάχιστον, μια διαδικασία πρέπει να έχει μια partner link για κάθε ιδιαίτερη ανταλλαγή επικοινωνίας που συμμετέχει. Η BPEL επιτρέπει τις πολλαπλές partner link του ίδιου τύπου partner link type να υπάρξει μέσα σε μια διαδικασία.

Εξ ορισμού, μια partner link δηλώνεται στο επίπεδο διαδικασίας. Μπορείτε να δηλώσετε μια partner link σε επίπεδο πεδίου. Το όνομα των partner link πρέπει να είναι μοναδικό μεταξύ του συνόλου των partner links που καθορίζονται από τη διαδικασία ή το πεδίο.

3.3 ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ

3.3.1 Βασικές δραστηριότητες

Οι βασικές δραστηριότητες χωρίζουν σε δύο είδη: εκείνοι που αλληλεπιδρούν με τους Web service partners και εκείνοι που χρησιμοποιούνται εσωτερικά σε μια διαδικασία BPEL.

Ο ακόλουθος πίνακας παρέχει τους συνοπτικούς ορισμούς και τις συνδέσεις με περισσότερες λεπτομέρειες.

Όνομα δραστηριότη	Περιγραφή
Δραστηριότητες αλληλεπίδρασης υπηρεσιών Ιστού	
Receive	Δέχεται τα στοιχεία μηνυμάτων από έναν partner υπηρεσ Προαιρετικά αρχίζει μια διαδικασία με τη δημιουργία περίπτωσης της διαδικασίας.
Invoke	Κατευθύνει μια Web service για να εκτελέσει μια λειτουργία
Reply	Στέλνει μια απάντηση σε έναν partner

Εσωτερικές δραστηριότητες διαδικασίας	
Validate	Επικυρώνει τις τιμές των μεταβλητών ενάντια στον καθορισμό σχετικών στοιχείων XML και WSDL τους
Assign	Χειρίζεται τις μεταβλητές διαδικασίας με τη δημιουργία αντιγράφου από/προς τις διαδικασίες για τους
Throw	Επισημαίνει ένα σφάλμα.
Exit	Σταματά μια εκτελέσιμη διαδικασία αμέσως
Wait	Η εκτέλεση διαδικασίας σταματά για ένα καθορισμένο χρόνο διάστημα
Empty	Μια δραστηριότητα που δεν κάνει τίποτα όταν εκτελεί. Χρήσιμος τις καταστάσεις όπου χρειάζεστε μια δραστηριότητα αλλά θέλουμε τίποτα να συμβεί πραγματικά
Break	Σταματά μία δραστηριότητα. Η επεξεργασία συνεχίζεται κανονικά την επόμενη δραστηριότητα.
Continue	Συνεχίζεται με την επόμενη επανάληψη σε έναν βρόχο.
Suspend	Αναστέλλει τη διαδικασία. Χρήσιμη σε μια σύλληψη ή ένα γεγονός catchAll για να πιάσει τα απροσδόκητα λάθη.

3.3.2 Δομημένες δραστηριότητες

Όνομα	Περιγραφή
Sequence	Εκτελεί τις δραστηριότητες σε έναν διαταγμένο κατάλογο
Flow	Εκτελεί όλες τις δραστηριότητες παράλληλα
While	Εκτελεί μια δραστηριότητα επανειλημμένα έως ότου αξιολογεί ο όρος σε false
Repeat Until	Εκτελεί μια δραστηριότητα τουλάχιστον μία φορά και επαναλαμβάνει ότου αξιολογεί ο όρος του σε True
Pick	Εκτελεί μια επιλεγμένη δραστηριότητα βασισμένη σε ένα εισερχόμενο μήνυμα. Περιλαμβάνει τη Create ιδιοκτησία περίπτωσης για να αρχίσει διαδικασία.

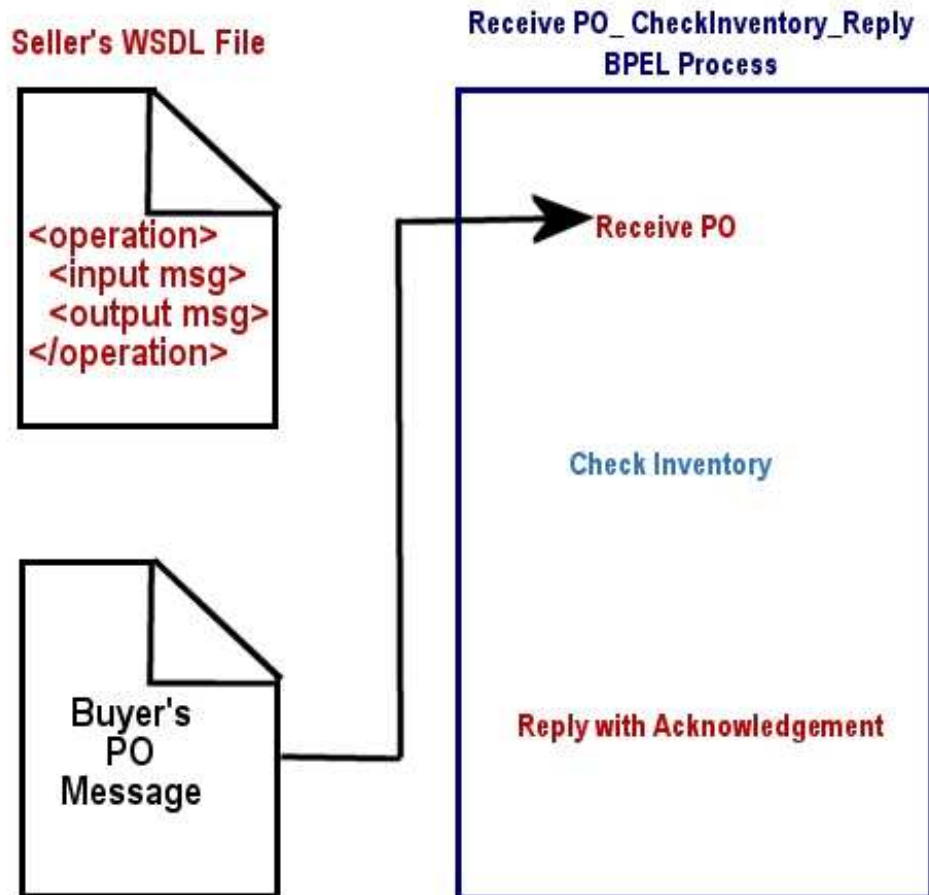
If	Εκτελεί μια επιλεγμένη δραστηριότητα βασισμένη στους όρους κλάδων
Scope	Παρέχει ένα πλαίσιο για το σύνολο δραστηριοτήτων, με το συσχετισμό, αποζημίωση, το χειρισμό ελαττωμάτων, το χειρισμό γεγονότος, το χειρισμό λήξης
For Each	Επαναλαμβάνει πέρα από έναν κατάλογο στοιχείων εγγράφων

3.4 Μεταβλητές (Variables)

3.4.1 Επισκόπηση των μεταβλητών

Μια διαδικασία BPEL λαμβάνει, χειρίζεται, και στέλνει τα στοιχεία μέσω των μεταβλητών. Οι μεταβλητές κρατούν τα μηνύματα ανταλλαγμένα μεταξύ των επιχειρησιακών συνεργατών και των στοιχείων που χρησιμοποιούνται μέσα στη διαδικασία.

Παραδείγματος χάριν, μια διαδικασία που λαμβάνει ένα μήνυμα εντολής αγοράς από έναν αγοραστή βάζει το μήνυμα σε μια μεταβλητή εισαγωγής. Από εκεί τη μεταβλητή το περιεχόμενο μπορεί να προσεγγιστεί από άλλες διαδικασίες. Η ακόλουθη απεικόνιση παρουσιάζει ένα παράδειγμα.



Εικόνα 8 παράδειγμα χρήσης μεταβλητών

Οι μεταβλητές καθορίζονται με έναν από τους ακόλουθους τρόπους:

- WSDL Message Types
- XML Schema Element
- XML Schema Type

3.4.2 Τύποι μηνυμάτων WSDL

Μια μεταβλητή τύπων μηνυμάτων WSDL χρησιμοποιεί έναν τύπο μηνυμάτων που δηλώνεται σε ένα WSDL namespace στη διαδικασία.

Η ακόλουθη απεικόνιση παρουσιάζει ένα παράδειγμα μιας μεταβλητής που ορίζεται ως ένας τύπος μηνυμάτων WSDL. Ο namespace δηλώνεται στη διαδικασία BPEL.

BPEL Process

```
<variable name = "BuyerPO"  
messageType = "Ins:POMsg"/>
```

WSDL File

```
<portType name = "poPT">  
  <operation name = "PO">  
    <input message = "Ins:POMsg"/>  
  </operation>  
</portType>
```

Εικόνα 9 τύπος μηνύματος WSDL

3.5 Τύπος σχημάτων XML

Μια μεταβλητή τύπου XML χρησιμοποιεί έναν απλό ή σύνθετο τύπο στοιχείου που χτίζεται στο schema του στα συγκεκριμένο namespace. Παραδείγματος χάριν:

```
<variable  
xmlns:props="http://example.com/shipProps"  
name="itemsShipped" type="props:itemCountType"/>
```

3.5.1 XML Schema Element

Μια μεταβλητή στοιχείων σχημάτων XML χρησιμοποιεί ένα στοιχείο που καθορίζεται στο schema του σ αυτό το συγκεκριμένο namespace (σ), όπως φαίνεται στο ακόλουθο παράδειγμα.

BPEL Process

```
<variable name="buyer_PO">  
  <element="pOrd"/>
```

WSDL Schema

```
<xsd:element name="pOrd">  
  <xsd:complexType>  
    <xsd:sequence>  
      <xsd:element ref="pO"/>  
    </xsd:sequence>  
  </xsd:complexType>  
</xsd:element>
```

Εικόνα 10 : XML Schema Element

3.5.2 Τι είναι ένα link ;

Μια σύνδεση (link) είναι μια δομή που συνδέει δύο δραστηριότητες και ελέγχει τη διαταγή της εκτέλεσης των δύο δραστηριοτήτων. Μια σύνδεση μπορεί να περιλάβει μια συνθήκη , περαιτέρω έλεγχος πότε, ή εάν, μια δραστηριότητα εκτελεί. Η συνθήκη είναι μια έκφραση XPath.

Μπορούμε να προσθέσουμε πολλαπλές συνδέσεις μεταξύ των βασικών δραστηριοτήτων και μεταξύ των δομημένων δραστηριοτήτων. Μπορούμε επίσης να προσθέσουμε συνδέσεις σε μια βασική δραστηριότητα που περιλαμβάνεται από μια δομημένη δραστηριότητα.

Σύνταξη συνδέσεων XML

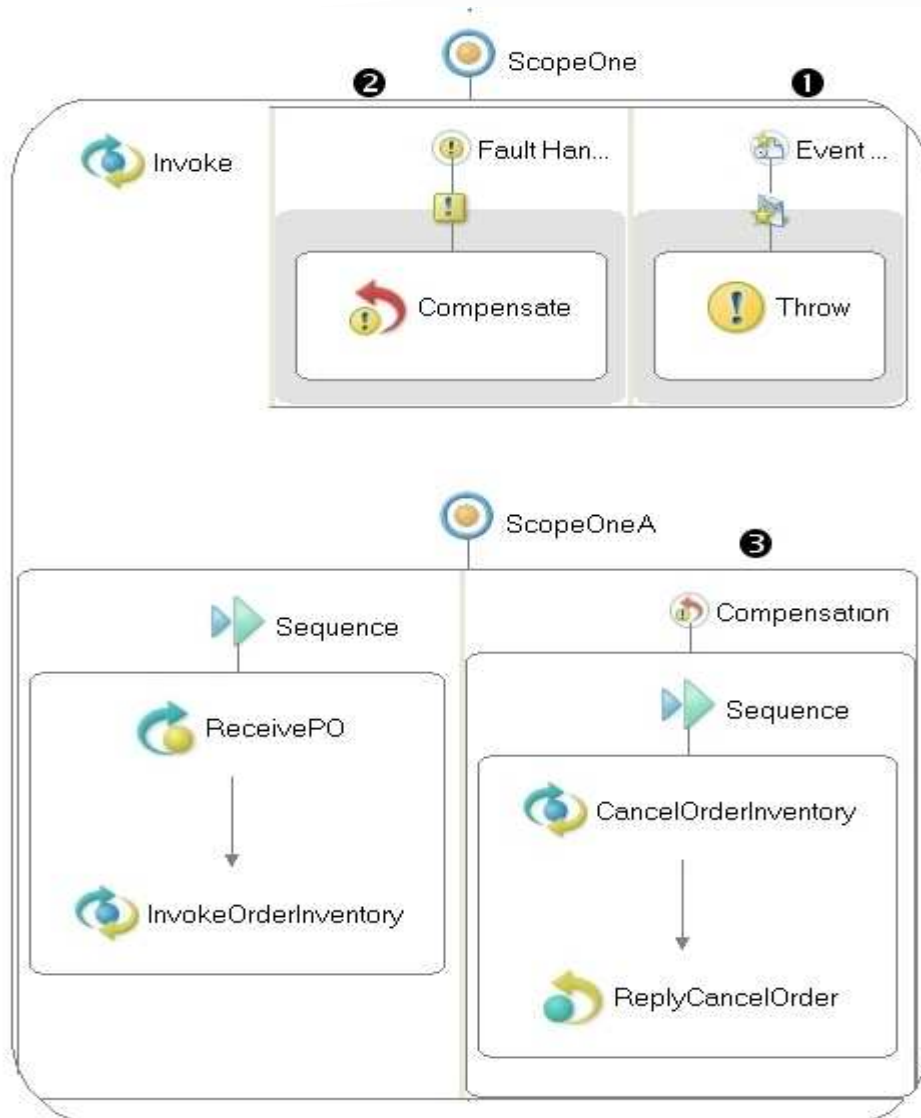
```
< ... >
< "..... name="L1/>
  activity1
< "..... linkName="L1/>
  ...
  activity2
< "..... linkName="L1/>
  ...
</... >
```

3.6 Compensation

3.6.1 Τι είναι Compensation (αποκατάσταση);

Η αποκατάσταση επιτρέπει σε μια επιχειρησιακή διαδικασία να καθορίσει μια δραστηριότητα του οποίου η εκτέλεση χρησιμεύει στο να αντιστρέψει κάποια προηγουμένως εκτελεσμένη λογική εφαρμογή.

Μια επιχειρησιακή διαδικασία μπορεί να συνεχίσει ακόμα κι αν μέρος της πρέπει να αντιστραφεί. Η ακόλουθη απεικόνιση παρουσιάζει ένα παράδειγμα ενός μέρους οηEvent ενός χειριστή γεγονόςτος αναγκάζοντας την επιχειρησιακή διαδικασία να ακυρώσει μια διαταγή καταλόγων και να στείλει μια ειδοποίηση ακύρωσης διαταγής. Οι δραστηριότητες ενός ήδη ολοκληρωμένου πεδίου είναι αυτές που αντισταθμίζονται.



Εικόνα 11 : onEvent ενός χειριστή γεγονότος

1	Όταν ο χειριστής γεγονότος λαμβάνει ένα μήνυμα ακύρωσης διαταγής, στέλνει ένα fault στον fault handler
2	Ο fault handler εκτελεί αντισταθμίζει τη δραστηριότητα για το προηγουμένως ολοκληρωμένο πεδίο στο οποίο συνδέεται
3	Ο compensation handler κυλά πίσω την εργασία της υπηρεσίας InvokeOrderInventory

3.7 Συσχέτιση (Correlation)

3.7.1 Τι είναι Correlation ;

Correlation είναι μια κατασκευή για την παρακολούθηση μιας ομάδας μηνυμάτων που ανήκουν σε μια ιδιαίτερη αλληλεπίδραση business partner .Ταιριάζει τα μηνύματα με τις business partner για τις οποίες προορίζονται.

Όταν μια μηχανή BPEL λαμβάνει ένα μήνυμα, ψάχνει μια διαδικασία που μπορεί να χειριστεί το μήνυμα. Μερικά μηνύματα προκαλούν νέα στιγμιότυπα διαδικασίας να

δημιουργηθούν. Άλλα μηνύματα πρέπει να σταλούν σε μια ήδη-τρέχουσα διαδικασία. Πώς εκείνη η διαδικασία βρίσκεται; Βρίσκεται με το ταίριασμα του περιεχομένου ενός συνόλου Correlation . Τα δεδομένα μέσα στο σύνολο Correlation είναι η υπογραφή που αφήνει τη μηχανή να ταιριάζει το μήνυμα με τη διαδικασία που περιμένει εκείνο το μήνυμα.

Τα βήματα για την προσθήκη του συσχετισμού σε μια διαδικασία παρουσιάζονται στην ακόλουθη απεικόνιση.



Εικόνα 12 : προσθήκη του συσχετισμού σε μια διαδικασία

3.7.2 Τι είναι Correlation set

Ένα correlation set είναι ένα σύνολο κοινών ιδιοτήτων στα μηνύματα. Ο σκοπός του correlation set είναι να ενεργήσει ως προσδιοριστικό συνομιλίας: κρατά μαζί όλα τα μηνύματα που προορίζονται για την ίδια συνομιλία.

Ένα μήνυμα μπορεί να έχει διάφορες ιδιότητες, κάθε μια που ορίζεται ως μια property alias.

3.7.2.1 Ιδιότητες

Μια ιδιότητα μηνυμάτων που χρησιμοποιείται για το συσχετισμό μηνυμάτων πρέπει να είναι ένας απλός τύπος σχημάτων όπως ένα xsd:int ή ένα xsd:string. Παραδείγματος χάριν, μια ιδιότητα μπορεί να είναι ένας αριθμός εντολής αγοράς ή μια ταυτότητα πελατών. Μια ιδιοκτησία υπάρχει μέσα σε ένα μήνυμα WSDL που διαβιβάζεται κατά τη διάρκεια μιας αλληλεπίδρασης Web services. Ένα μήνυμα μπορεί

να περιέχει πολλαπλάσιες ιδιότητες και μια ιδιότητα μπορεί να υπάρξει σε διάφορα μηνύματα.

3.8 Event Handling

3.8.1 Τι είναι Χειριστές γεγονότων (Event Handlers)

Ο χειριστής γεγονότων είναι μια δραστηριότητα που τρέχει ταυτόχρονα με ένα πεδίο. Τα γεγονότα μπορούν να είναι ένα από δύο τύπους: ένα γεγονός μηνυμάτων ή ένας συναγερμός. Όταν ένα γεγονός ή ένας συναγερμός μηνυμάτων εμφανίζεται, ο χειριστής γεγονότος που είναι συνδεδεμένος με το γεγονός επικαλείται μια δραστηριότητα.

Οι χειριστές γεγονότος είναι ιδιαίτερα χρήσιμοι για τα γεγονότα και τα αιτήματα που δεν μπορούν να συσχετιστούν με την κύρια δραστηριότητα, αλλά μπορούν να εμφανιστούν σε απρόβλεπτους χρόνους. Παραδείγματος χάριν, ένας πελάτης μπορεί να ακυρώσει μια διαταγή που υποβάλλεται σε επεξεργασία.

Οι χειριστές γεγονότος θεωρούνται μέρος της κανονικής συμπεριφοράς του πεδίου, αντίθετα από τους fault handlers και τους Compensation handlers οι οποίοι συμβαίνουν πέρα της κανονικής επεξεργασίας.

Ένας χειριστής γεγονότος μπορεί να συνδεθεί με τη διαδικασία συνολικά ή με ένα πεδίο.

Ένα ή περισσότερα γεγονότα μπορούν να εμφανιστούν και να αντιμετωπιστούν οποιαδήποτε στιγμή ενώ το αντίστοιχο πεδίο είναι ενεργό.

Ένας χειριστής γεγονότος δεν μπορεί να ενεργοποιηθεί έως ότου μια περίπτωση διαδικασίας δημιουργηθεί, σημαίνοντας δεν μπορεί να δημιουργήσει η ίδια την περίπτωση διαδικασίας.

Τύποι γεγονότος

Ένα γεγονός μπορεί να είναι ένα εισερχόμενο μήνυμα που αντιστοιχεί σε ένα αίτημα/μία απάντηση ή μια μονόδρομη λειτουργία σε ένα WSDL.

Ένα γεγονός μπορεί επίσης να είναι ένας συναγερμός που σταματά μετά από έναν καθορισμένο χρόνο ή διαρκεί για έναν καθορισμένο χρόνο, και επαναλαμβάνει προαιρετικά.

3.8.2 Παραδείγματα των δραστηριοτήτων χειριστών γεγονότος

Όταν ένα μήνυμα που προκαλεί ένα γεγονός παραλαμβάνεται, ένα από τα εξής μπορεί να είναι η πιθανή απάντηση από έναν χειριστή γεγονότος:

- Στέλνετε μια απάντηση

- Ολοκληρώνεται η process instance . Παραδείγματος χάριν, μια διαταγή ακυρώνεται, έτσι η process instance πρέπει να ακυρωθεί, και δεν υπάρχει καμία τρέχουσα εργασία που ανατρέπεται και που αντισταθμίζεται.
- Στέλνετε ένα Fault για να αναγκαστεί η τρέχουσα εργασία να αναστραφεί.

3.9 Fault Handling

3.9.1 Τι είναι BPEL Fault Handling;

Ο χειρισμός ελαττωμάτων Fault handling σε μια διαδικασία BPEL είναι αντίστροφη εργασία, που ανατρέπει τη μερική και ανεπιτυχή εργασία ενός πεδίου στο οποίο ένα ελάττωμα έχει εμφανιστεί. Ο χειρισμός ελαττωμάτων διαφέρει από το χειρισμό αποζημιώσεων στο ότι ο χειρισμός ελαττωμάτων δρα όταν ένα ελάττωμα εμφανίζεται σε ένα πλαίσιο ενώ η αποζημίωση αντιστρέφει την εργασία σε ένα επιτυχώς ολοκληρωμένο πεδίο.

Ένα ελάττωμα μπορεί να εμφανιστεί για τους ακόλουθους λόγους:

- Μια λειτουργία web service δεν μπορεί να ολοκληρωθεί επιτυχώς, και η υπηρεσία επιστρέφει ένα ελάττωμα
- Ένα εσωτερικό λάθος διαδικασίας εμφανίζεται, και ένα τυποποιημένο ελάττωμα BPEL εμφανίζεται
- Μία < throw > ή < rethrow > δραστηριότητα βγάζει ένα ελάττωμα

Ο χειρισμός ελαττωμάτων μπορεί να είναι γενικός ή τοπικός: μπορείτε να προσθέσετε τους χειριστές ελαττωμάτων στη διαδικασία συνολικά ή σε ένα πεδίο μέσα στη διαδικασία.

Όταν ένα ελάττωμα εμφανίζεται, η κανονική επεξεργασία ολοκληρώνεται, και ο έλεγχος μεταφέρεται στον αντίστοιχο χειριστή ελαττωμάτων, όπως καθορίζεται στο τμήμα < faultHandlers > της διαδικασίας ή του πεδίου.

Σημειώστε ότι η διαδικασία δεν επιτρέπει την αποζημίωση για ένα πεδίο στο οποίο ένας χειριστής ελαττωμάτων επικαλείται.

Κεφάλαιο 4

4.1 ΕΙΣΑΓΩΓΗ

Σε αυτό το κεφάλαιο παρέχεται μια επισκόπηση του WebSphere Business Modeler που αποτελεί προϊόν της οικογένειας WebSphere process integration. Τα προϊόντα αυτής της οικογένειας υποστηρίζουν την ολοκληρωμένη διαχείριση επιχειρησιακών διαδικασιών σε όλο τον κύκλο ζωής τους,

Τα προϊόντα WebSphere process integration δίνουν τη δυνατότητα να :

- Μοντελοποιήσουμε, προσομοιώσουμε και να αναλύσουμε περίπλοκα επιχειρησιακά σενάρια γρήγορα και αποδοτικά πριν υλοποιηθούν
- Μετατρέψουμε υπάρχουσες διαδικασίες ώστε να είναι δυναμικές και εύκολα αποροφούμενες ώστε να παρέχουν αποδοτικότητα ως προς το κόστος
- Επιτρέπουν στους χρήστες να παρακολουθούν τις επιχειρησιακές διαδικασίες ώστε να μπορούν να τις βελτιώνουν ανά πάσα στιγμή

Το WebSphere process integration αποτελείται από τα εξής προϊόντα :

- WebSphere Business Modeler
- WebSphere Integration Developer
- WebSphere Process Server
- WebSphere Business Monitor

Αυτό το κεφάλαιο έχει ως σκοπό να μας εισαγάγει στα βασικές στοιχεία, τα εργαλεία, τις λειτουργίες, και τις ικανότητες που είναι διαθέσιμες στον WebSphere

Business Modeler της IBM και να επιδείξει πώς μπορούμε γρήγορα να δημιουργήσουμε ρεαλιστικά μοντέλα για τις επιχειρησιακές διαδικασίες μας.

Σε αυτό το κεφάλαιο παρέχονται βαθμιαίες οδηγίες, που μας καθοδηγούν μέσα από τα βασικά στοιχεία που πρέπει να καταλάβουμε για να αρχίσουμε να μοντελοποιούμε τις δικές μας processes, resources και organizations. Παρουσιάζονται εκτεταμένα τα εξής σημεία:

- Create projects and catalogs (Δημιουργία projects και καταλόγων)
- Create and define resources (Δημιουργία και καθορισμός πόρων)
- Create definitions (Δημιουργία Definitions)
- Create and define business items (Δημιουργία επιχειρησιακών στοιχείων)
- Add tasks and other elements to a process diagram (πρόσθεση στοιχείων σε ένα διάγραμμα)
- Run simulations (Εκτέλεση simulations)
- Adjust simulation settings (Προσαρμογή ρυθμίσεων simulations)
- Analyze simulation results (Ανάλυση των αποτελεσμάτων του simulation)

4.2 Διαδικασία εκτέλεσης έργου

Στο παρακάτω project θα δημιουργήσουμε ένα έργο επιχειρησιακής μοντελοποίησης που θα αναπαριστά την εισαγωγή ενός ασθενούς σε νοσοκομειακή μονάδα και η διαδικασία διάγνωσης της πάθησης του καθώς και του καθορισμού της αγωγής που πρέπει να ακολουθήσει.

Καταρχήν γίνεται η εισαγωγή του ασθενούς στο νοσοκομείο. Εκεί δίνει τα στοιχεία εισαγωγής του (ονοματεπώνυμο , φύλλο , ηλικία και είδος ασφάλισης). **(1)**

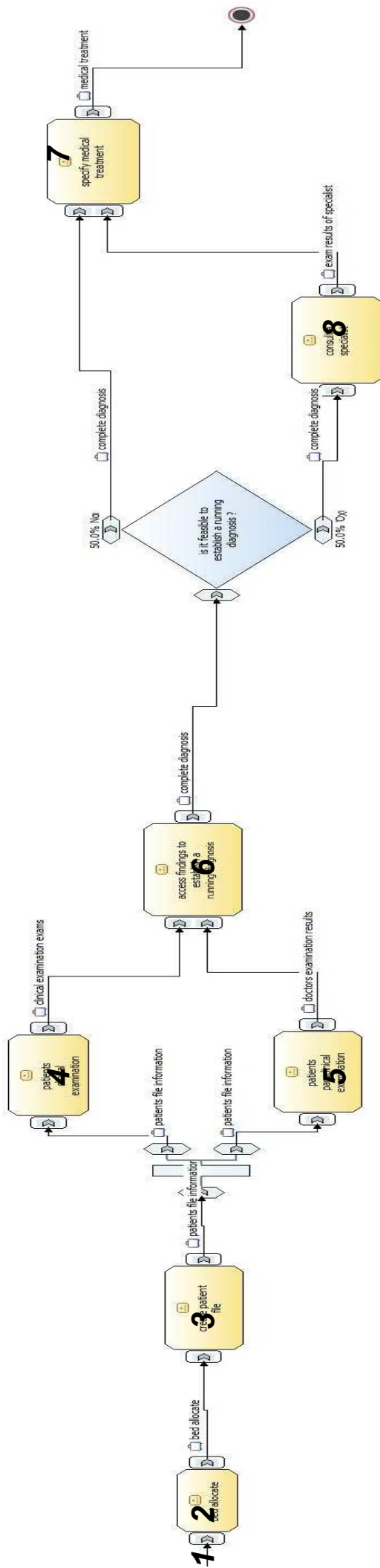
Στην συνέχεια η διαδικασία προχωρά με την τοποθέτηση του ασθενούς σε κάποια κλίνη. Αυτό για να γίνει θα πρέπει να εξεταστούν μερικοί παράγοντες όπως το φύλλο του ασθενούς καθώς σε άλλα κρεβάτια τοποθετούνται οι άνδρες ασθενείς και σε άλλα οι γυναίκες. Άλλος παράγοντας που εξετάζεται είναι το είδος ασφάλισης του ασθενούς δηλαδή αν είναι δημόσια ασφάλιση ή ιδιωτική. Θα μπορούσε να εξεταστεί αυτός ο παράγοντας και σε μεγαλύτερο βάθος δηλαδή ως προς το είδος της ασφάλισης του ασθενούς ανάλογα με το ταμείο στο οποίο είναι γραμμένος (π.χ ΙΚΑ , ΟΓΑ κ.τ.λ.) αλλά δεν κρίθηκε αναγκαίο. Τέλος ένας τρίτος παράγοντας που εξετάζεται είναι η διαθεσιμότητα της κλίνης που ζητάτε. **(2)**

Εφόσον ο ασθενής τοποθετηθεί σε κλίνη προχωρά στο επόμενο στάδιο της διαδικασίας το οποίο είναι η δημιουργία του φακέλου του ασθενούς. Στον φάκελο προστίθενται διάφορα στοιχεία εκτός αυτών που δηλώθηκαν κατά την εισαγωγή του όπως το ιστορικό του ασθενούς (παλαιότερες ασθένειες που έχει περάσει , χρόνιες ασθένειες κτλ). Αυτό είναι βασικό χαρακτηριστικό καθώς δίνεται η δυνατότητα για καλύτερη λήψη απόφασης από τους γιατρούς. Άλλο στοιχείο που προστίθεται είναι αν ο ασθενής κατά την περίοδο εισαγωγής του ακλουθεί και κάποιες άλλες φαρμακευτικές αγωγές. **(3)**

Εφόσον δημιουργηθεί ο φάκελος του ασθενούς αυτός προχωρά στις απαραίτητες εξετάσεις που πρέπει να γίνουν. Εδώ ακολουθούνται δυο στάδια παράλληλα. Στο ένα στάδιο ο ασθενής πραγματοποιεί ιατρικές εξετάσεις (αιματολογικές , υπέρηχοι κ.τ.λ.). **(4)** Στο άλλο στάδιο η εξέταση περιλαμβάνει εξετάσεις από ιατρό επιτόπου. Σε αυτό το στάδιο απαιτείται η συμμετοχή στο έργο και φυσικού πρόσωπου ώστε να προχωρήσει η διαδικασία. Το ρόλο του φυσικού προσώπου στο έργο μας θα παίξει ο γιατρός που θα εξετάσει τον ασθενή. **(5)**

Εφόσον συγκεντρωθούν τα αποτελέσματα και από τα δυο στάδια εξετάσεων ελέγχονται αυτά ως προς το αν είναι ικανά ώστε να πραγματοποιηθεί διάγνωση. **(6)** Εάν είναι ικανά να πραγματοποιηθεί διάγνωση τότε προχωρούμε στον καθορισμό φαρμακευτικής αγωγής για τον ασθενή. **(7)** Εάν όμως τα αποτελέσματα δεν επαρκούν για να πραγματοποιηθεί διάγνωση τότε ο ασθενής θα πρέπει να εξεταστεί και από κάποιο δεύτερο γιατρό με συγκεκριμένη ειδικότητα άρα θα γίνει εξέταση σε μεγαλύτερο βάθος. **(8)** Εφόσον πραγματοποιηθεί και σε αυτή την περίπτωση διάγνωση προχωρούμε σε φαρμακευτική περίθαλψη οπότε τελειώνει ουσιαστικά και η διαδικασία.

ΔΙΑΓΡΑΜΜΑ ΜΟΝΤΕΛΟΠΟΙΗΣΗΣ ΤΟΥ ΕΡΓΟΥ e-health

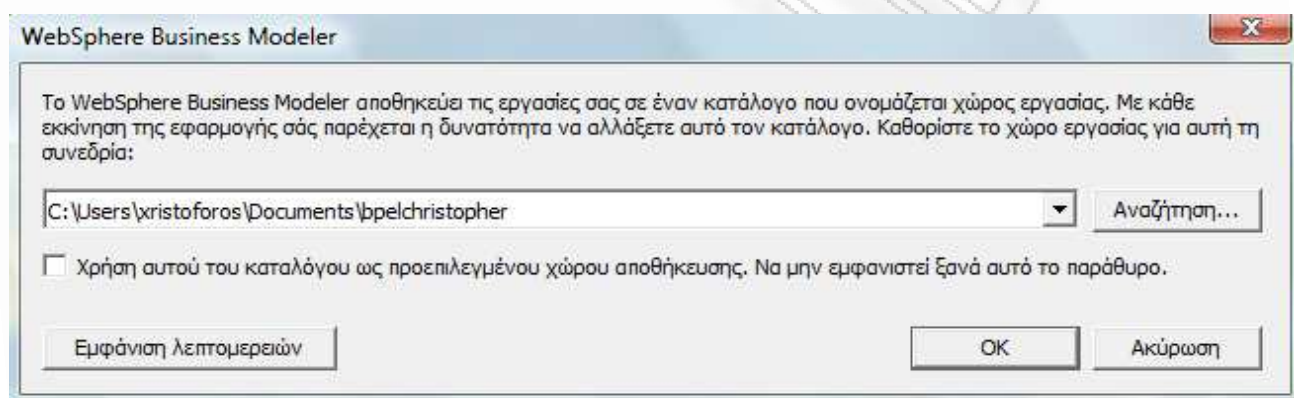


Εικόνα 13 : Διαδικασία εκτέλεσης έργου

4.2.1 ΜΟΝΤΕΛΟΠΟΙΗΣΗ ΤΟΥ PROJECT ΣΕ BPEL ΜΕΣΩ WEBSHERE BUSINESS MODELER

Σε αυτό το μέρος θα περιγράψουμε πώς μοντελοποιούμε τη διαδικασία μέσω μιας επιχειρησιακής προοπτικής. Επίσης παρέχουμε βήμα προς βήμα οδηγίες για το πώς αυτό μπορεί να επιτευχτεί στο επιχειρησιακό μας σενάριο.

Εκκινούμε τον WEBSHERE BUSINESS MODELER και δημιουργούμε ένα νέο workspace όπως φαίνεται στην εικόνα :

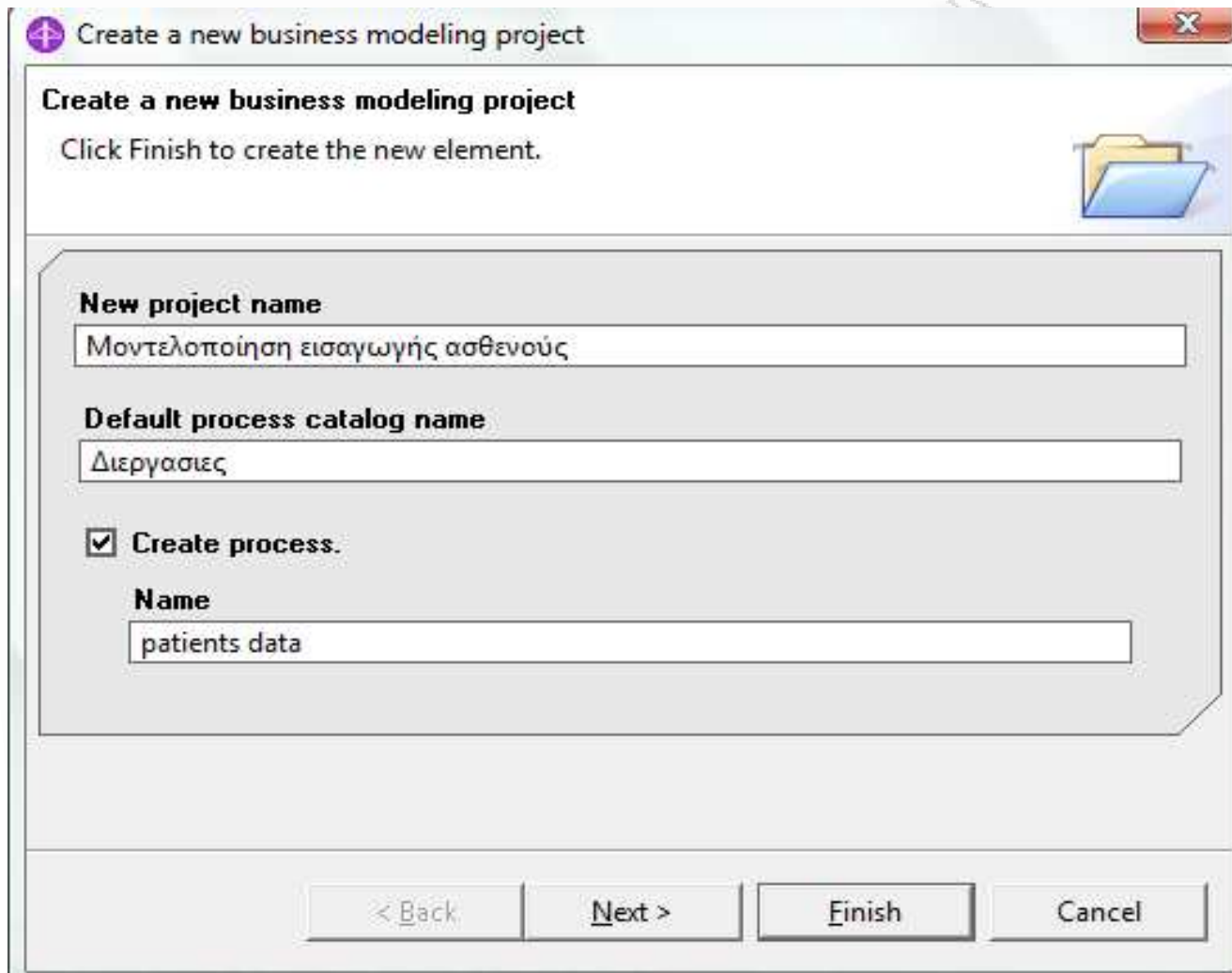


Εικόνα 14: Δημιουργία Workspace

Καθορίζουμε το μονοπάτι στο οποίο θα εργαζόμαστε και θα αποθηκεύεται το project. Στη συνέχεια μέσω των Μοντελοποίηση → Δημιουργία → Έργο Επιχειρησιακής Μοντελοποίησης ξεκινά το wizard Δημιουργία νέου έργου επιχειρησιακής μοντελοποίησης στο οποίο δημιουργούμε το project το οποίο ονομάζουμε “ Μοντελοποίηση εισαγωγής ασθενούς”.


Εισάγουμε τις ακόλουθες τιμές στα παρακάτω πεδία:

- ορίζουμε το όνομα του έργου σε Μοντελοποίηση εισαγωγής ασθενούς
- ορίζουμε το προεπιλεγμένο όνομα καταλόγου διεργασιών σε Διεργασίες
- ορίζουμε το όνομα της διεργασίας patients data.



Εικόνα 15 : Δημιουργία νέου Project

Πατάμε επόμενο , επιλεγούμε την επιλογή ελεύθερη διάταξη και πατάμε Ολοκλήρωση .Έτσι ολοκληρώνεται η δημιουργία του workspace του project μας.

Εάν δεν μπορούμε να δούμε το Project Tree view κάνουμε κλικ στο Apply four-pane layout κουμπί  στη γραμμή εργασιών.

4.2.2 Επιχειρησιακά στοιχεία

4.2.2.1 Δημιουργία επιχειρησιακών στοιχείων.

Επιχειρησιακά στοιχεία είναι εκείνα τα στοιχεία τα οποία περνούν από την μια δραστηριότητα στην επόμενη κατά τη διάρκεια της εκτέλεσης του project. Μπορούμε να

μοντελοποιήσουμε σαν επιχειρησιακό στοιχείο καθετί το οποίο δημιουργείται, επιβλέπεται , δοκιμάζεται ή εκτελείται.

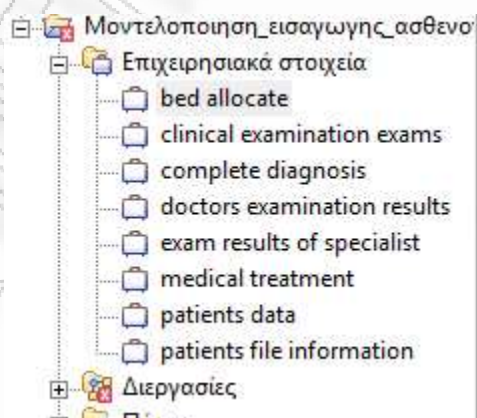
Τα επιχειρησιακά στοιχεία είναι διαφορετικά από τους πόρους. Ενώ οι πόροι είναι οι άνθρωποι, ο εξοπλισμός, ή το υλικό που χρησιμοποιείται την εργασία για να εκτελέσει έναν στόχο, επιχειρησιακά στοιχεία είναι οι οντότητες που οι πόροι εκτελούν. Παραδείγματος χάριν, μια διαταγή πελατών είναι ένα επιχειρησιακό στοιχείο, ενώ οι πωλήσεις συνδέουν αυτές τις πράξεις στην παραγγελία πελάτη σαν πόρο.

4.2.2.2 Δημιουργώντας ένα κατάλογο δεδομένων (data catalog)

Ένας κατάλογος δεδομένων είναι ένας κατάλογος όπου μπορούμε να αποθηκεύσουμε επιχειρησιακά στοιχεία και στιγμιότυπα επιχειρησιακών στοιχείων.

Καταρχήν καθώς ο ασθενής εισάγεται στο νοσοκομείο δημιουργείται ένα επιχειρησιακό στοιχείο με τα στοιχεία του εκάστοτε ασθενούς. Για να δημιουργήσουμε ένα επιχειρησιακό στοιχείο ακολουθούμε την παρακάτω διαδικασία:

1. Στο project tree κάνουμε δεξί κλικ στο πεδίο Επιχειρησιακά στοιχεία και επιλέγουμε NEO → Επιχειρησιακό στοιχείο.
2. Στο πεδίο Name of the business item δίνουμε το όνομα patient data.
3. Εάν θέλουμε στο πεδίο Description of new business item δίνουμε κάποια περιγραφή.
4. Πατάμε finish και έτσι δημιουργείται το επιχειρησιακό στοιχείο.



Εικόνα 16 : Project tree επιχειρησιακών στοιχείων

Τέλος κλείνουμε το πλαίσιο : Γνωρίσματα επιχειρησιακού στοιχείου και σώζουμε τις αλλαγές.

Αυτή είναι η διαδικασία που ακολουθείται για τη δημιουργία ενός νέου επιχειρησιακού στοιχείου στον Websphere business modeler. Παρακάτω θα χρειαστεί να δημιουργήσουμε και άλλα επιχειρησιακά στοιχεία με τον ίδιο τρόπο.

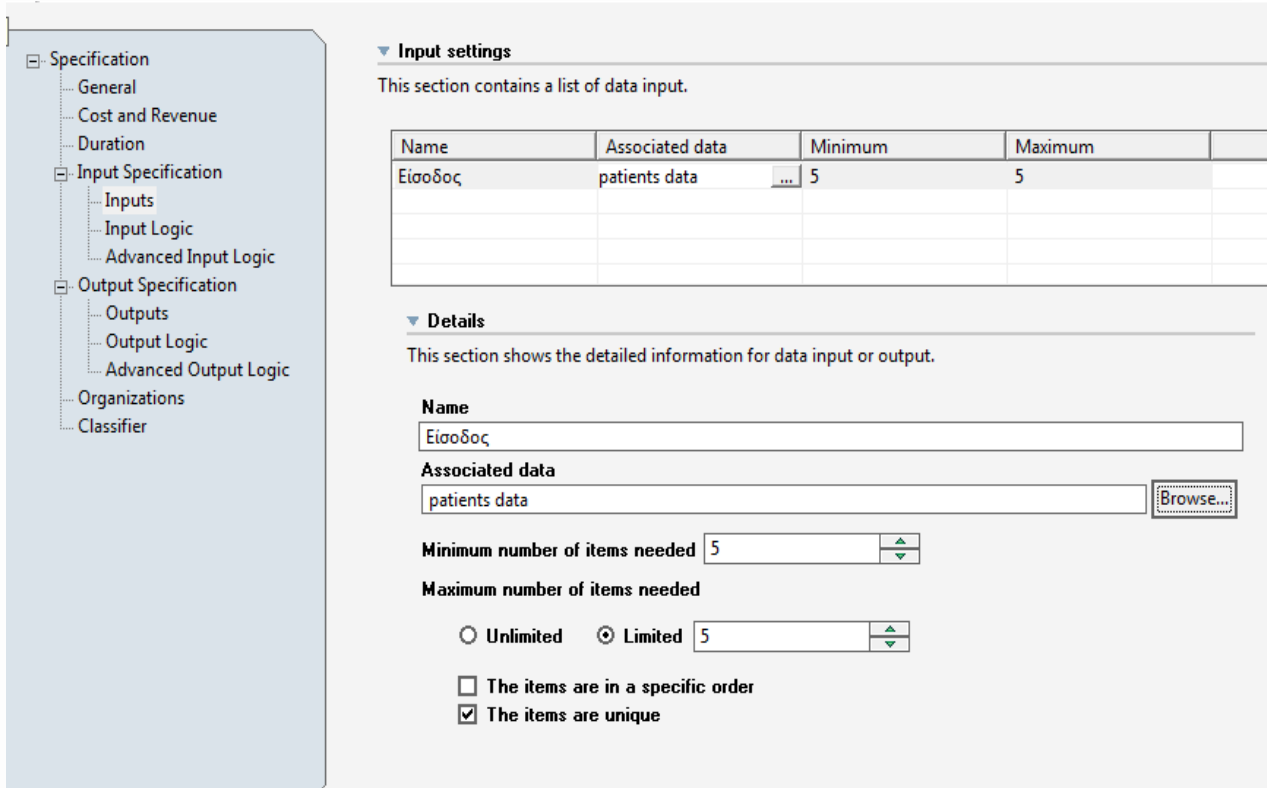
4.2.2.4 Δημιουργία μιας περίπτωσης επιχειρησιακών στοιχείων (business item instance)

Μια περίπτωση επιχειρησιακών στοιχείων είναι ένα συγκεκριμένο περιστατικό ή ένα παράδειγμα ενός επιχειρησιακού στοιχείου. Μια περίπτωση επιχειρησιακών στοιχείων μπορεί να χρησιμοποιηθεί ως στοιχεία δειγμάτων για τα επιχειρησιακά στοιχεία μέσα στις προσομοιώσεις. Όταν δημιουργείτε μια περίπτωση ένα από τα επιχειρησιακά στοιχεία σας, εκείνη η περίπτωση θα κληρονομήσει όλες τις ιδιότητες που καθορίσατε για το επιχειρησιακό στοιχείο. Έχετε έπειτα επιλέξει την επιλογή τις τιμές ιδιοτήτων για εκείνη την συγκεκριμένη περίπτωση.

4.2.2.5 Συσχέτιση του επιχειρησιακού στοιχείου με την διαδικασία.

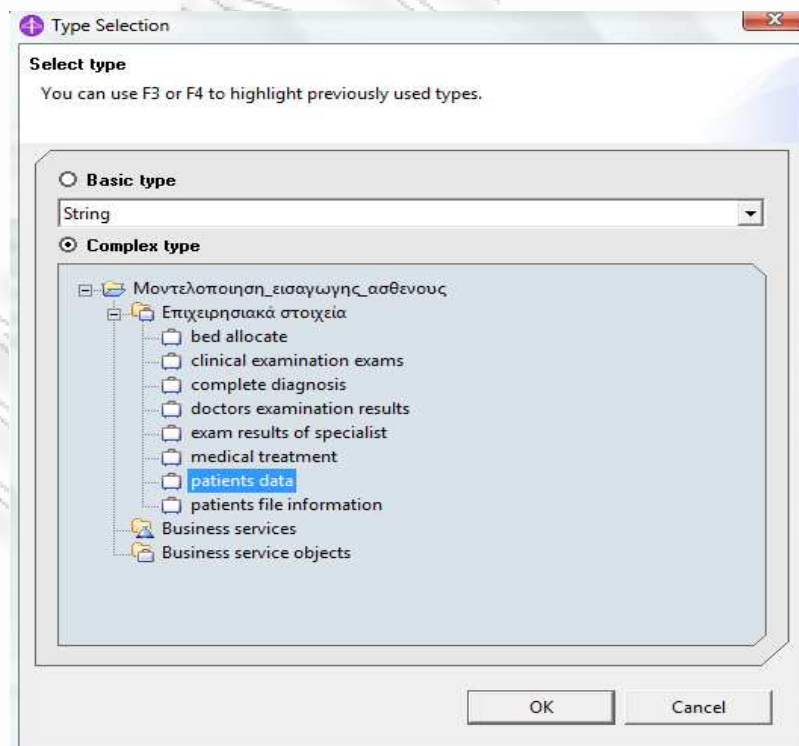
Όλες οι διαδικασίες πρέπει να έχουν μια ή περισσότερες συσχετίσεις επιχειρησιακών στοιχείων με τις εισόδους και τις εξόδους της διαδικασίας. Ακολουθούμε τα επόμενα βήματα για να συσχετίσουμε τα δεδομένα με τη διαδικασία.

1. Εκτείνουμε τον κατάλογο Διεργασίες και διπλό-κλικάρουμε στο Μοντελοποίηση εισαγωγής ασθενούς.
2. Η διαδικασία Μοντελοποίηση εισαγωγής ασθενούς περιέχει προειδοποιήσεις. Στο πεδίο Σφάλματα βλέπουμε να παρουσιάζονται 2 σφάλματα τα οποία διορθώνουμε αφαιρώντας το start και stop node από τον Process Editor πατώντας delete.
3. Επιλέγουμε το πεδίο Προδιαγραφές στον Process Editor.
4. Ακολουθούμε το εξής μονοπάτι Προδιαγραφές→Προδιαγραφές εισόδου→Είσοδοι.
5. Στις Ρυθμίσεις Εισόδου πατάμε Προσθήκη. Στο πεδίο Όνομα αλλάζουμε την τιμή σε patient data.



Εικόνα 18 : Κάρτα ρυθμίσεων δεδομένων

6. Στο πεδίο λεπτομεριών πατάμε Αναζήτηση → Σύνθετο Είδος → patient data και πατάμε ok.



Εικόνα 19 : type selection

7. Έπειτα επιλεγούμε το μονοπάτι Προδιαγραφές→Προδιαγραφές εισόδου →Λογική εισόδου. Αλλάζουμε την τιμή στο πεδίο όνομα σε patient data.

4.2.2.6 Δημιουργία ενός προτύπου επιχειρησιακών στοιχείων

Τα πρότυπα επιχειρησιακών στοιχείων είναι κατηγορίες που χρησιμοποιούνται για να διαμορφώσουν τις ομάδες επιχειρησιακών στοιχείων που μοιράζονται τις κοινές ιδιότητες. Μια επιχείρηση ή μια οργάνωση μπορεί να έχει πολλά διαφορετικά επιχειρησιακά στοιχεία στις διαδικασίες της. Μια μεγαλύτερη οργάνωση μπορεί να έχει εκατοντάδες ή ακόμα και χιλιάδες επιχειρησιακά στοιχεία. Πολλά από αυτά τα επιχειρησιακά στοιχεία να έχουν τα κοινά χαρακτηριστικά ή ιδιότητες, καθώς επίσης και να περιέχουν τις ιδιότητες που είναι μοναδικές. Ένα πρότυπο επιχειρησιακών στοιχείων επιτρέπει σε σας να δημιουργήσετε ένα σύνολο ιδιοτήτων μία φορά, και να εφαρμόσει έπειτα εκείνες σε πολλά επιχειρησιακά στοιχεία

4.2.2.7 Προσθέτοντας μια τοπική εργασία.

Όλα τα παραπάνω ορίζονται σαν στοιχεία εισόδου του ασθενούς στο νοσοκομείο. Εφόσον εισαχτεί ο ασθενής θα πρέπει να τοποθετηθεί σε κλίνη. Για να γίνει αυτό δημιουργείται μια καινούργια τοπική εργασία από τα στοιχεία του panel στην οποία δίνουμε το όνομα bed allocate (Τοποθέτηση σε κλίνη).

4.2.2.8 Ορίζοντας τις εισόδους και εξόδους της τοπικής εργασίας.

Κάνοντας κλικ πάνω στην τοπική εργασία εμφανίζονται τα διάφορα προς επεξεργασία γνωρίσματα της τοπικής εργασίας (π.χ. είσοδοι έξοδοι κτλ).Οπότε από την καρτέλα γνωρίσματα επιλέγουμε είσοδοι και έπειτα Προσθήκη (δηλ. Γνωρίσματα→Είσοδοι→Προσθήκη).Στο πεδίο Ρυθμίσεις εισόδου δίνουμε όνομα ΕΙΣΟΔΟΣ και σαν αντίστοιχα δεδομένα επιλέγουμε από την αναζήτηση το patient data. Αυτό γίνεται γιατί θεωρούμε πως τα στοιχεία που δίνει ο ασθενής κατά την εισαγωγή του στο νοσοκομείο είναι επαρκή ώστε να επεξεργαστούν και να τοποθετηθεί σε κλίνη. Έτσι στο πεδίο Λογική εισόδου δίνουμε το όνομα ΕΙΣΟΔΟΣ.

Στη συνέχεια καθώς τα στοιχεία εισόδου εισέρχονται στην τοπική εργασία γίνεται η επεξεργασία τους ώστε να επιδεχθεί η κλίνη του ασθενούς. Για να επιδεχθεί η κλίνη

εξετάζονται τρία κριτήρια οποία αφορούν τρία από τα ιδιοχαρακτηριστικά της εισόδου και αυτά είναι:

- A) φύλλο ασθενούς (sex)
- B) ασφάλιση (insurance)
- Γ) διαθεσιμότητα(availability)

Εφόσον τα στοιχεία αυτά εισέρχονται στην τοπική εργασία και εξετάζονται ως προς τις διάφορες περιπτώσεις επιλέγεται κλίνη για τον ασθενή και δημιουργείται ένα νέο επιχειρησιακό στοιχείο στην έξοδο το οποίο ονομάζεται bed allocate και περιέχει όλα τα προηγούμενα χαρακτηριστικά συν ένα ακόμη χαρακτηριστικό που ονομάζεται bed id.

Όνομα	Είδος	Ελάχιστη...	Μέγιστη ...	Για ανάγ...
firstname	Αλφαριθμητική σειρά	1	1	Αληθές (..
lastname	Αλφαριθμητική σειρά	1	1	Αληθές (..
sex	Αλφαριθμητική σειρά	1	1	Αληθές (..
age	Ακέραιος	1	1	Αληθές (..
insurance	Αλφαριθμητική σειρά	1	1	Αληθές (..
bedid	Αλφαριθμητική σειρά	1	1	Αληθές (..

Εικόνα 20 : ιδιοχαρακτηριστικά επιχειρησιακού στοιχείου

Για να δημιουργήσουμε αυτό το επιχειρησιακό στοιχείο στην διακλάδωση έργων ακολουθούμε τη διαδικασία που ακολουθήθηκε και ποιο πάνω .

Στη συνέχεια θα πρέπει να συνδέσουμε αυτό το επιχειρησιακό στοιχείο με την έξοδο της τοπικής εργασίας bed allocate. Έτσι στην καρτέλα Γνωρίσματα→Έξοδοι του bed allocate δίνουμε όνομα ΕΞΟΔΟΣ και συσχετίζουμε τα δεδομένα στην ενότητα αντίστοιχα δεδομένα με το bed allocate.Επίσης στην Λογική εξόδου δίνουμε το όνομα ΕΞΟΔΟΣ.

4.2.2.9 Δημιουργία τοπικής εργασίας .

Επόμενο βήμα του project είναι η δημιουργία του προσωπικού φάκελου κάθε ασθενούς. Η διαδικασία ολοκλήρωσης της τοπικής αυτής εργασίας είναι ίδια με την προηγούμενη τοπική εργασία. Έτσι έχουμε τα εξής :

Δημιουργούμε από το panel μια τοπική εργασία που της δίνουμε το όνομα create patient file.Σαν εισόδους της εργασίας αυτής δίνουμε όνομα ΕΙΣΟΔΟΣ και την συσχετίζουμε με το επιχειρησιακό στοιχείο bed allocate. Επίσης στο πεδίο Λογική εισόδου δίνουμε όνομα Είσοδος .Με αυτή την κίνηση ορίζουμε σαν στοιχεία εισόδου της νέας εργασίας τα στοιχεία

εξόδου της προηγούμενης εργασίας. Στην συνέχεια γίνεται επεξεργασία των στοιχείων αυτών καθώς και μερικές νέες καταχωρήσεις. Αυτό έχει σαν αποτέλεσμα την δημιουργία στην έξοδο ενός νέου επιχειρησιακού στοιχείου που ονομάζεται patient file information το οποίο δημιουργούμε με τον ίδιο τρόπο όπως και τα προηγούμενα και στο οποίο προσθέτουμε τα ίδια χαρακτηριστικά με αυτά της εισόδου της εργασίας συν δυο ακόμα χαρακτηριστικών τα οποία είναι τα εξής:

- patients health history (Ιστορικό ασθενούς)
- other medical treatment (Παλιότερη φαρμακευτική αγωγή ασθενούς)

Στη συνέχεια στην καρτέλα Γνωρίσματα→Έξοδοι της τοπικής εργασίας δίνουμε το όνομα Εξοδος_1 και το συσχετίζουμε με το επιχειρησιακό στοιχείο patient file information καθώς και Εξοδος_2 το οποίο επίσης συσχετίζουμε με το επιχειρησιακό στοιχείο patient file information. Επίσης στο πεδίο Λογική Εξόδου δίνουμε το όνομα Εξοδος_1 και Έξοδος_2 αντίστοιχα. Αυτή η εργασία αποτελείται από δύο εξόδους και όχι μια καθώς τα στοιχεία που αποτελούν το επιχειρησιακό στοιχείο χρησιμοποιούνται σαν εισοδοι σε δυο άλλες εργασίες. Το σημαντικό εδώ είναι να αντιληφθούμε πως αυτή η εργασία δεν έχει σαν έξοδο ή την Εξοδος_1 ή την Εξοδος_2 αλλά και τις 2 ταυτόχρονα. Οπότε αυτό πρέπει να το δηλώσουμε στη λογική εξόδου της εργασίας με το λογικό AND όπως φαίνεται παρακάτω :

Name	Probability (...)	ΕΞΟΔΟΣ_1	ΕΞΟΔΟΣ_2	Criterion
Εξοδος_1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	ΕΞΟΔΟΣ_1 AND ΕΞΟΔΟΣ_2

Εικόνα 21 : Κάρτα εισόδων και εξόδων εργασιών

Ακολούθως αφού δημιουργηθεί ο φάκελος του ασθενούς , ο ασθενής προχωρεί σε εργαστηριακές εξετάσεις (αιματολογικές και άλλων ειδών εξετάσεις) καθώς και σε ιατρικές εξετάσεις από κάποιον γιατρό. Για να υλοποιηθεί αυτή η διαδικασία σε επίπεδο μοντελοποίησης δημιουργούνται δυο διαδικασίες που εκτελούνται και οι δυο αυστηρά ανεξαρτήτου χρόνου περάτωσης της κάθε μιας.

Η πρώτη εργασία ονομάζεται patients clinical examination και αφορά όπως είπαμε την πραγματοποίηση από τον ασθενή εργαστηριακών εξετάσεων και η δεύτερη ονομάζεται patients paraclinical examination και αφορά την πραγματοποίηση εξέτασης από κάποιον

ιατρό. Για την πρώτη εργασία δημιουργούμε ένα επιχειρησιακό στοιχείο που το ονομάζουμε clinical examination exams και αποτελείται από τα εξής ιδιοχαρακτηριστικά:

Name	Type	Minimum	Maximum	Read only
clinical examination exams	String	1	200	False

Εικόνα 22 : Δημιουργία επιχειρησιακού στοιχείου

Αυτό το στοιχείο αποτελεί την έξοδο της εργασίας patients clinical examination και απαρτίζεται όπως βλέπουμε από τα αποτελέσματα των κλινικών εξετάσεων.

Η δεύτερη εργασία που όπως αναφέραμε ονομάζεται patients paraclinical examination δέχεται και αυτή σαν είσοδο το επιχειρησιακό στοιχείο patients file information και βγάζει σαν έξοδο το επιχειρησιακό στοιχείο doctors examination results το οποίο όπως βλέπουμε αποτελείται από ένα πεδίο το doctors examination results και είναι στην ουσία το πόρισμα του γιατρού.

Name	Type	Minimum	Maximum	Read only
doctors examination r...	String	1	200	False

Εικόνα 23 : Δημιουργία επιχειρησιακού στοιχείου

4.2.2.10 Δημιουργώντας ένα resource catalog

Οι κατάλογοι των πόρων(resource catalog) χρησιμοποιούνται για να αποθηκεύσουν τους ορισμούς πόρων των πόρων του project μας .

Για να δημιουργήσουμε ένα resource catalog ακολουθούμε τα επόμενα βήματα:

1. Στο Project –tree κάνουμε δεξί κλικ στο φάκελο “Πόροι “ και επιλέγουμε New→Resource Catalog. Ένα Wizard ανοίγει.
2. Στο πεδίο Name of new resource catalog δίνουμε το όνομα που επιθυμούμε.
3. Πατάμε Finish .Ο Resource catalog δημιουργείται.

Στο συγκεκριμένο project όπως βλέπουμε και συγκεκριμένα στην εργασία (patients paraclinical examination) συμμετέχει και ο ανθρώπινος παράγοντας στη διαδικασία ο οποίος πρέπει να αναπαρασταθεί μέσα στην μοντελοποίηση. Αυτό πραγματοποιείται με τους λεγόμενους ρόλους μέσα στη διαδικασία που αποτελούν πόρους της διαδικασίας. Για

να δημιουργήσουμε ένα ρόλο πηγαίνουμε στο project tree της μοντελοποίησης και στο πεδίο ΠΟΡΟΙ με δεξί κλικ δημιουργούμε ένα ρόλο στον οποίο δίνουμε το όνομα Doctor. Εάν η διαδικασία μας αποτελείται από πολλούς ρόλους (Πόρους) τότε αυτοί ορίζονται μέσα σε Resource Catalogs που δημιουργούνται σύμφωνα με την παραπάνω διαδικασία που αναφέρθηκε. Καθώς όμως στην διαδικασία μας συμμετέχει μόνο ένας πόρος (Doctor) δίνεται η δυνατότητα να μην δημιουργήσουμε ένα Resource Catalog και να προσθέσουμε τον ρόλο κατευθείαν στο πεδίο ΠΟΡΟΙ. Έπειτα αυτό το ρόλο πρέπει να τον συσχετίσουμε σε μια εργασία συγκεκριμένα την patients paraclinical examination. Έτσι στο πεδίο resources προσθέτουμε ένα ρόλο τον οποίο ονομάζουμε DOCTOR και τον συσχετίζουμε με τον πόρο Doctor που δημιουργήσαμε προηγουμένως. Τέλος για να ολοκληρώσουμε τη διαδικασία συσχετίζουμε τον πόρο που δημιουργήσαμε με κάποιο είδος πόρου συγκεκριμένα εδώ με το staff (προσωπικό).

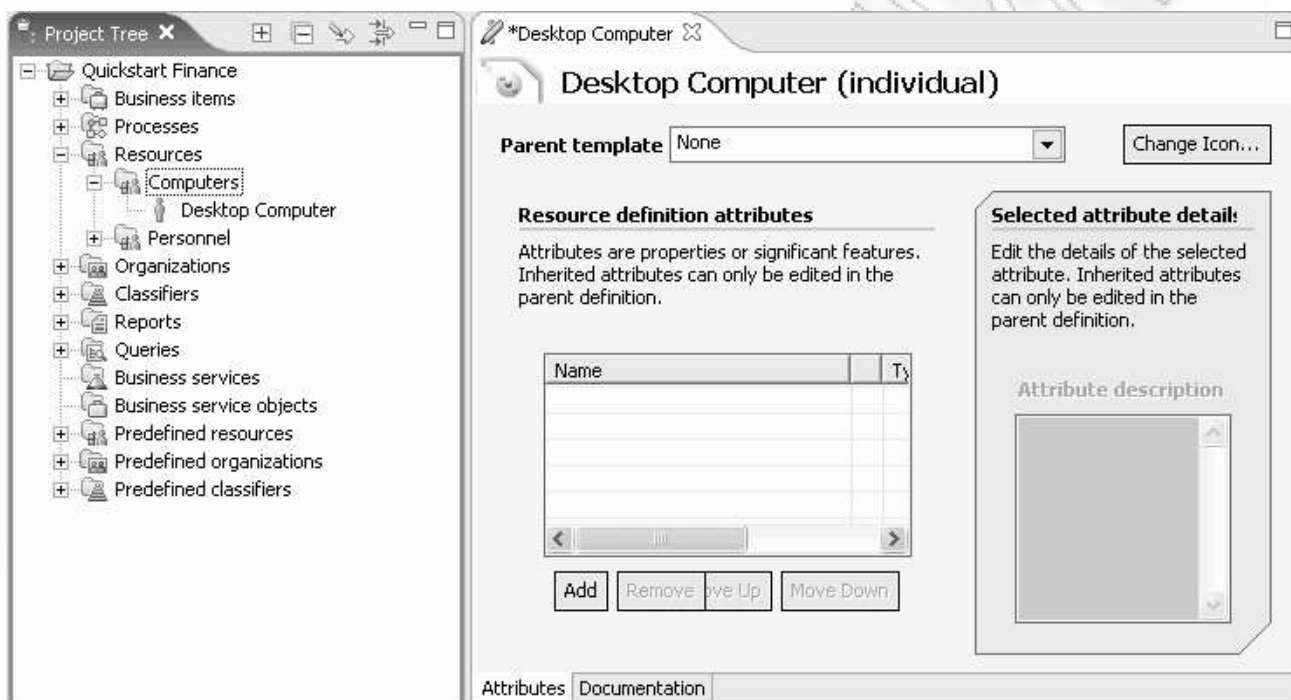
4.2.2.11 Δημιουργώντας ένα resource definition (καθορισμός πόρων)

Οι Resource definitions παρέχουν κοινές ιδιότητες για τους παρόμοιους ή σχετικούς πόρους στα πρότυπα διαδικασίας μας. Μπορούμε να διαμορφώσουμε καθέναν από τους πόρους που χρησιμοποιούνται για να εκτελέσουν έναν στόχο στην επιχείρηση και να τον τοποθετήσουμε στο project tree για χρήση στα διαγράμματα διαδικασίας. Ένας καθορισμός πόρων (Resource definition) είναι ένα σύνολο παρόμοιων χαρακτηριστικών κοινών στους πόρους του ίδιου τύπου. Παραδείγματος χάριν, ένας καθορισμός των πόρων οχημάτων θα μπορούσε να χρησιμοποιηθεί για να δημιουργήσει τους πόρους όπως το φορτηγό παράδοσης και το εκτελεστικό αυτοκίνητο. Στο παράδειγμα μας δεν χρησιμοποιήσαμε Resource definitions καθώς χρησιμοποιούμε μόνο έναν πόρο.

Για να προσθέσουμε ένα resource definition στο project μας ακολουθούμε τα επόμενα βήματα:

1. Στο project –tree κάνουμε δεξί κλικ στο Computers resource catalog και επιλέγουμε New > Resource Definition. Ένα wizard ανοίγει.
2. Δίνουμε κάποιο όνομα στο resource definition.
3. Στο πεδίο assign color αφήνουμε το χρώμα στην default τιμή
4. Στο πεδίο Description of a new resource definition δίνουμε μια περιγραφή του resource definition που δημιουργούμε.

5. Στο πεδίο resource type επιλέγουμε **Individual** εάν ο πόρος μας είναι μετρήσιμο μέγεθος(π.χ κάποιος υπολογιστής) και **bulk** εάν ο πόρος μας είναι μη μετρήσιμο μέγεθος (π.χ νερό)
6. Πατάμε finish. Ο resource definition που δημιουργήσαμε ανοίγει σε έναν Definition editor



Εικόνα 24 : Definition editor

Εφόσον δημιουργήσαμε έναν resource definition πρέπει τ'α'ρα να του προσθέσουμε χαρακτηριστικά.

4.2.2.12 Προσθέτοντας χαρακτηριστικά σε ένα resource definition

Τα χαρακτηριστικά καθορισμού πόρων είναι εκείνα τα χαρακτηριστικά που είναι κοινά για όλους τους πόρους που δημιουργήθηκαν για να χρησιμοποιήσουν αυτούς τους πόρους.

Χρησιμοποιούμε χαρακτηριστικά για να καθορίσουμε ένα στοιχείο στον WebSphere Business Modeler συμπεριλαμβανομένου και τα resource definitions. Οποιοσδήποτε αριθμός πόρων μπορεί να δημιουργηθούν με τη χρήση ενός καθορισμού πόρου και κάθε

ξεχωριστός πόρος μοιράζεται ένα σετ από κοινά χαρακτηριστικά που έχουν καθοριστεί σε ένα καθορισμό πόρων.

Για παράδειγμα ένας καθορισμός πόρων μιας εταιρίας οχημάτων μπορεί να έχει χαρακτηριστικά όπως ο τύπος οχήματος και ο αριθμός άδειας. Κάθε πόρος που δημιουργείται χρησιμοποιώντας τον καθορισμό πόρων της εταιρίας οχημάτων θα κληρονομήσει τα ίδια χαρακτηριστικά.

Ο καθορισμός πόρων παρέχει το βασικό σέτ χαρακτηριστικών για τους πόρους μας ώστε να χρειάζεται να δημιουργήσουμε αυτά τα χαρακτηριστικά μια μόνο φορά αντί να τα δημιουργούμε για κάθε πόρο.

Για να προσθέσουμε χαρακτηριστικά σε ένα resource definition ακολουθούμε τα έξης βήματα :

1. Ανοίγουμε τον definition editor.Οι καθορισμοί πόρων ανοίγουν στον definition editor
2. Πατάμε το κουμπί Attributes
3. Πατάμε Add. Μια νέα ιδιότητα προστίθεται στο πεδίο ιδιοτήτων.
4. Ονοματίζουμε την ιδιότητα στο πεδίο Name
5. Επιλέγουμε τον τύπο της ιδιότητας στο πεδίο Type
6. Διαβεβαιωνόμαστε πως η επιλογή Basic Type είναι επιλεγμένη.
7. Πατάμε save για να αποθηκευτούν οι αλλαγές.

2. Καθώς πρόκειται για πρόσωπο επιλέγουμε το bullet Individual. Εάν επρόκειτο για κάποιον παραδείγματος χάριν υπολογιστή ή κάποια μηχανή θα επιλέγαμε το bullet bulk.

3. Από τη drop-down λίστα καθορισμού των πόρων, επιλέγω προσωπικό.

4. Στο πεδίο όνομα του νέου πόρου βάζουμε το όνομα του νέου πόρου που δημιουργούμε, στη συγκεκριμένη περίπτωση Doctors. Θυμηθείτε ότι το όνομα που εισάγετε πρέπει να είναι μοναδικό μέσα στον κατάλογο των πόρων προσωπικού.

5. Στην περιγραφή του νέου τομέα των πόρων, προσθέτουμε κάποια περιγραφή του πόρου μας.

6. Πατάμε finish. Ο νέος πόρος έχει δημιουργηθεί. Στο συγκεκριμένο παράδειγμα ενώ θα μπορούσαμε να δημιουργήσουμε ένα πόρο αλλά προτιμήσαμε να δημιουργήσουμε ένα ρόλο για την ίδια δουλειά καθώς είναι μια τελείως απλοποιημένη διαδικασία που δεν χρειάζεται να της προσθέσουμε ούτε ιδιοχαρακτηριστικά ούτε χρονοπρογραμματισμό ούτε κάτι άλλο. Παρακάτω αναφέρεται πώς δημιουργείται ο ρόλος doctor.

4.2.2.14 Προσθήκη των attribute values σε έναν πόρο

Οι τιμές ιδιοτήτων (attribute values) παρέχουν τα μοναδικά χαρακτηριστικά ενός συγκεκριμένου πόρου.

Όλοι οι πόροι που χρησιμοποιούνται σε ένα project είναι μεμονωμένα ευπροσδιόριστοι και έχουν τα χαρακτηριστικά που τους διαφοροποιούν από άλλους πόρους, ακόμη και αυτοί που είναι πολύ παρόμοιοι. Παραδείγματος χάριν, κάθε ανώτερος υπάλληλος δανείου έχει ένα πρώτο και τελευταίο όνομα, μια διεύθυνση κατοικίας, και έναν αριθμό υπαλλήλων. Ομοίως, κάθε υπολογιστής γραφείου έχει μια θέση αριθμού και χώρου εργασίας. Για να δώσει τις τιμές σε εκείνες τις ιδιότητες των πόρων, ολοκληρώστε τα ακόλουθα βήματα:

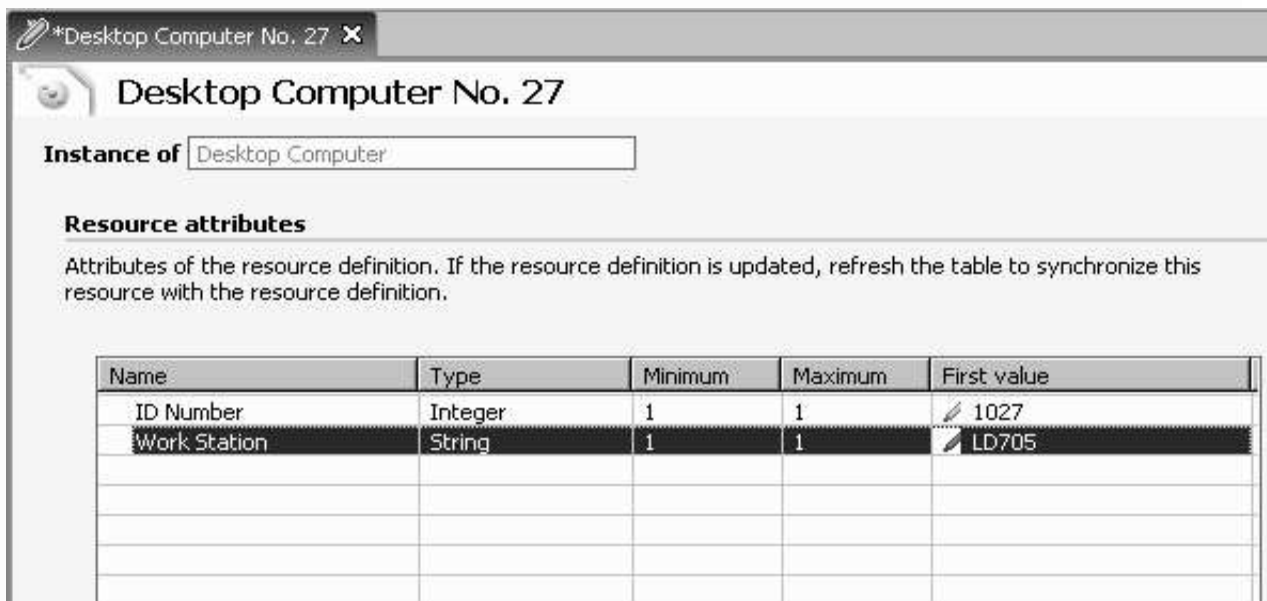
Έστω ότι σε ένα project χρησιμοποιούμε έναν πόρο με όνομα Desktop Computer No.

27. Προσθέτουμε ιδιότητες για το Desktop Computer No. 27

α. Εάν ο Desktop Computer No. 27 πόρος δεν είναι ήδη ανοιχτός στον editor πόρων διπλοκλικάρουμε στο project tree.

β. Κλικάρουμε στο κουμπί attributes στον editor πόρων

γ. Για την κάθε ιδιότητα δίνουμε όνομα και προσδιορίζουμε τι τύπου είναι π.χ ακέραιος, δυαδικός κ.τ.λ.



Εικόνα 26 : Προσθήκη των attribute values

δ. Πατάμε save για να σώσουμε τα ιδιοχαρακτηριστικά.

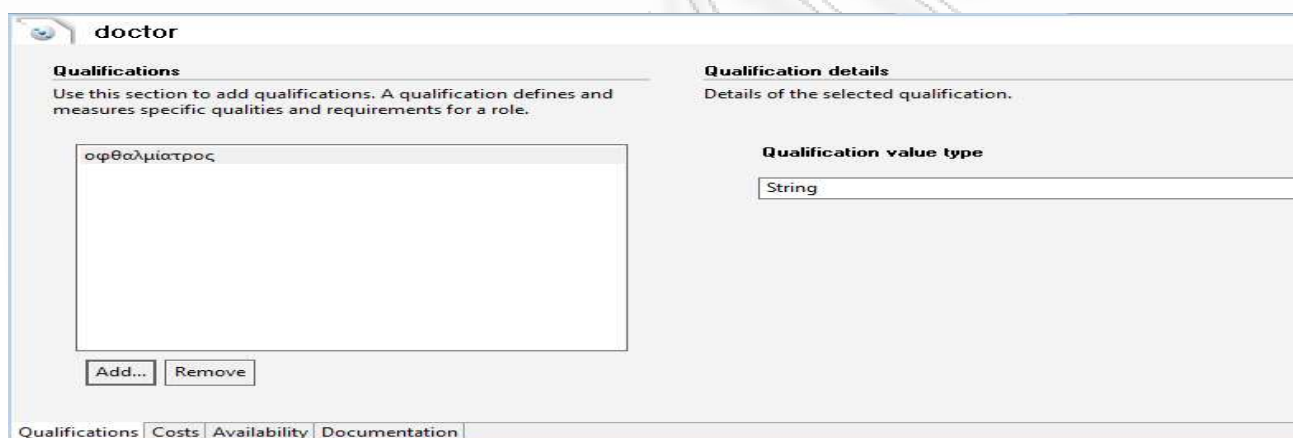
4.2.2.15 Δημιουργία ενός ρόλου

Οι ρόλοι καθορίζουν ένα σύνολο ικανοτήτων που είναι απαραίτητες για την εκτέλεση των στόχων μέσα στις διαδικασίες μας. Οι επιχειρήσεις καθορίζουν χαρακτηριστικά τους ρόλους για να διευκρινίσουν τις ικανότητες που οι πόροι αρμόδιοι για ορισμένους στόχους πρέπει να έχουν. Παραδείγματος χάριν, ένας ρόλος μεταφραστών να απαιτήσει έναν πόρο για να είναι ευφραδής σε τουλάχιστον δύο γλώσσες εάν ένας ιδιαίτερος πόρος υπαλλήλων είναι ευφραδής στα αγγλικά και γαλλικά, σε εκείνο τον πόρο θα μπορούσε να οριστεί ο ρόλος του μεταφραστή. Θα πρέπει να δημιουργήσουμε εκείνο τον ρόλο έτσι ώστε μπορούμε να τον ορίσουμε σε έναν από τους υπάρχοντες πόρους μας που καλύπτει τις απαιτήσεις ρόλου.

Για να δημιουργήσουμε ένα ρόλο ακολουθούμε τα εξής βήματα:

1. Στο project tree κάνουμε δεξί κλικ στο πεδίο Πόροι και έπειτα New→Role . Ένα Wizard ανοίγει .
2. Βεβαιωνόμαστε πως το πεδίο πόροι φωτίζεται στο Wizard.
3. Στο πεδίο Name of new role βάζουμε το όνομα Doctor
4. Αφήνουμε το χρώμα ρόλου όπως έχει.

5. Στο πεδίο Περιγραφή του νέου ρόλου δίνουμε κάποια περιγραφή που εμείς θέλουμε.
6. Πατάμε finish. Ο ρόλος Doctor ανοίγει στον definition editor .
7. Στο πεδίο qualifications πατάμε το κουμπί add. Το Wizard του add qualification ανοίγει. Το qualification μας δίνει τη δυνατότητα να καθορίσουμε ένα χαρακτηριστικό στο ρόλο. Για παράδειγμα εάν θέλουμε ο ρόλος μας γιατρός να είναι συγκριμένης ειδικότητας π.χ οφθαλμίατρος προσθέτουμε το qualification οφθαλμίατρος.
8. Στο πεδίο qualification type επιλέγουμε τον τύπο που θέλουμε π.χ ακέραιος , δυαδικός κ.τ.λ



Εικόνα 27 : definition editor ενός πόρου

9. Πατάμε save για να σώσουμε τα ιδιοχαρακτηριστικά

4.2.2.16 Προσθήκη των δαπανών σε έναν πόρο

Ένα κόστος είναι το χρηματικό ποσό που μια οργάνωση πρέπει να καταβάλει για να χρησιμοποιήσει έναν πόρο σε έναν στόχο ή μια διαδικασία. Οι δαπάνες μπορούν να προστεθούν σε έναν ιδιαίτερο πόρο σε οποιαδήποτε διαδικασία που θέλετε να διαμορφώσετε. Το κόστος είναι συχνά ένα αρχικό need to know παράγοντας σε ένα επιχειρησιακό πρότυπο. Σε αυτό το σεμινάριο, θα υποθέσουμε ότι δεν υπάρχει κανένα κόστος για τον γιατρό .Εάν όμως είχαμε κάποιο κόστος η διαδικασία που θα ακολουθούσαμε για να προσθέσουμε κόστος θα ήταν η εξής:

1. Εάν δεν είναι ήδη ανοιχτό διπλοκλικάρουμε τον πόρο που θέλουμε. Ο πόρος ανοίγει τον πόρων.
2. Κλικάρουμε το tab Costs
3. Κλικάρουμε το κουμπί Add. Ένα παράθυρο ανοίγει.
4. Κλικάρουμε το κουμπί Cost per time unit και έπειτα ok. Ένα κόστος προστίθεται.
5. Στο πεδίο Value βάζουμε όποια τιμή μας αφορά (πχ μισθός Γιατρού)
6. Στο πεδίο currency μπορούμε να επιλέξουμε όποιο νόμισμα θέλουμε.
7. Στο πεδίο Cost Details αφήνουμε την default τιμή που έχει στο ένα. Επιλέγοντας την μια ώρα σημαίνει πως πληρώνεται ο γιατρός με την ώρα
8. Πατάμε save και σώζουμε τα κόστη μας.

4.2.2.17 Δημιουργία ενός χρονοδιαγράμματος

Ένα χρονοδιάγραμμα δείχνει πότε ένας συγκεκριμένος πόρος είναι διαθέσιμος ή πότε ένα συγκεκριμένο κόστος ισχύει. Επιτρέπει σε σας να σχεδιάσετε έναν πόρο ή να καθορίσετε πότε ένα κόστος θα ισχύσει για τους διευκρινισμένους στόχους ή τις διαδικασίες. Στις περισσότερες επιχειρήσεις, οι υπάλληλοι είναι διαθέσιμοι για μια ορισμένη περίοδο ημέρας. Παραδείγματος χάριν, η πλειοψηφία του προσωπικού σε ένα γραφείο λογιστικής να λειτουργήσει την πρωινή βάρδια στις 9:00 π.μ. στις 5:00 μ.μ. Ομοίως, οι δαπάνες θα ισχύσουν συχνά μόνο για ένα ορισμένο χρονικό διάστημα ανάλογα με τη θέση της, οι δαπάνες θέρμανσης του γραφείου λογιστικής μόνο να πρέπει να πληρωθούν από τον Οκτώβριο μέχρι τον Απρίλιο. Για τους σκοπούς αυτού του σεμιναρίου, πρέπει μόνο να δημιουργήσετε ένα χρονοδιάγραμμα. Στο δικό μας έργο για χάρη της απλότητας της μοντελοποίησης δεν χρησιμοποιούμε χρονοδιάγραμμα αν και θα μπορούσαμε για τον πόρο γιατρό αφού ο γιατρός εργάζεται καθημερινά σε βάρδια.

Για να δημιουργήσουμε ένα χρονοδιάγραμμα ακολουθούμε τα επόμενα βήματα :

1. Στο project tree κάνουμε δεξί κλικ στο πεδίο Πόροι και έπειτα New → Timetable. Ένα Wizard ανοίγει.
2. Δίνουμε κάποιο όνομα στο χρονοδιάγραμμα και κάποια περιγραφή
3. Πατάμε finish και ανοίγει στο timetable editor .

4.2.2.18 Προσθήκη των ιδιοτήτων σε ένα χρονοδιάγραμμα

Οι ιδιότητες χρονοδιαγράμματος καθορίζουν το χρονοδιάγραμμα, δείχνοντας τέτοιες παραμέτρους όπως την ημερομηνία που το χρονοδιάγραμμα αρχίζει και ο αριθμός χρόνων που πρέπει να επαναληφθεί. Μπορείτε να καθορίσετε πόσο καιρό θέλετε το χρονοδιάγραμμά σας για να ισχύσει, είτε για μερικές ημέρες είτε εβδομάδες, είτε για έναν απεριόριστο χρόνο. Οι ιδιότητες χρονοδιαγράμματος επιτρέπουν σε σας να καθορίσετε το μήκος του χρονικού διαστήματος, πόσο συχνά επαναλαμβάνεται (παραδείγματος χάριν, καθημερινά ή εβδομαδιαία έκδοση), και πόσο καιρό θα συνεχίσει να επαναλαμβάνεται. Μπορείτε επίσης να θέσετε τα επαναλαμβανόμενα χρονικά διαστήματα για το χρονοδιάγραμμά σας. Ο Recurring time interval κατάλογος χρησιμοποιείται για να καθορίσει μια ιδιαίτερη χρονική περίοδο μέσα στο χρονοδιάγραμμα. Παραδείγματος χάριν, εάν τρέχετε εγκαταστάσεις κατασκευής 24 ώρες την ημέρα, και τα κόστη ηλεκτρικής ενέργειάς σας πρέπει να είναι υψηλότερα κατά τη διάρκεια των ωρών αιχμής μπορείτε να καθορίσετε αυτά τα τμήματα στο χρονοδιάγραμμά σας και έπειτα να χρησιμοποιήσετε το χρονοδιάγραμμα για να προγραμματίσετε τα ενεργειακά θέματα για τους λιγότερο ακριβούς χρόνους.

4.3 Process Diagramms

4.3.1 Δημιουργία των διαγραμμάτων διαδικασίας (process diagramms)

Ένα διάγραμμα διαδικασίας είναι μια παρουσίαση της ροής μιας επιχειρησιακής διαδικασίας, σε πραγματικό χρόνο και αποτελείται από τα μεμονωμένες βήματα ή τις δραστηριότητες που αποτελούν τη διαδικασία. Μπορεί να περιλάβει τους όρους που υπαγορεύουν πότε να εμφανίζονται εκείνες οι δραστηριότητες και οι πόροι που απαιτούνται για την εκτέλεση ή την εφαρμογή τους.

Εφόσον παραπάνω είδαμε πως μπορούμε να δημιουργήσουμε τα επιμέρους στοιχεία μίας επιχειρησιακής διαδικασίας (τοπικές εργασίες, καταλόγους δεδομένων, πόρους κτλ) συνεχίζουμε ποιο κάτω με την ολοκλήρωση της επιχειρησιακής μοντελοποίησης της εισαγωγής ασθενούς σε μονάδα που αναφέραμε στην αρχή του κεφαλαίου.

Έτσι στην συνέχεια της διαδικασίας οι έξοδοι από τις δυο τοπικές εργασίες (Patients Clinical Examination, Patients paraclinical Examination) συγχωνεύονται στην τοπική εργασία access findings to establish a running diagnosis. Αυτό πραγματοποιείται με την διπλή είσοδο στην τοπική εργασία όπου κάθε κλάδος εισόδου της εργασίας αυτής είναι και η έξοδος της κάθε εργασίας. Αυτό πραγματοποιείται κάνοντας δεξί κλικ κάτω στην εργασία

και στη συνέχεια Input and Output → Create a new input. Αυτό δίνει τη δυνατότητα στον χρήστη να προσθέσει παραπάνω από μία εισόδους σε μια τοπική εργασία.

Ακολουθως τα αποτελέσματα που προκύπτουν στο επιχειρησιακό στοιχείο Complete diagnosis θα πρέπει να εξεταστούν ώστε να προκύψει κάποια διάγνωση για τον ασθενή. Θα πρέπει να εξετάσουμε πρωταρχικά εάν είναι ικανά αυτά τα στοιχεία και αρκετά ώστε να έχουμε κάποια αποτελέσματα η αν χρειάζεται η συμβολή κάποιου ποιο ειδικού ιατρού ώστε να προκύψει κάποια θεραπεία. Έτσι σε αυτό το σημείο συνεχίζεται η διαδικασία με την δημιουργία μιας τοπικής εργασίας Έτσι επιλέγουμε από το panel το στοιχείο Τοπική εργασία και το ονομάζουμε Access Findings To Establish A Running Diagnosis. Σαν είσοδο της εργασίας ορίζουμε όπως είπαμε το complete diagnosis στοιχείο. Στο πεδίο λογική εισόδου δίνουμε το όνομα Diagnosis Inputs.

Η τοπική εργασία το πρώτο που κάνει όπως είπαμε είναι να εξετάσει εάν είναι δυνατόν να δοθεί διάγνωση η όχι. Για το σκοπό αυτό χρησιμοποιούμε το στοιχείο απλή απόφαση από το panel στο οποίο δίνουμε το όνομα is it feasible to establish a running diagnosis?. Σαν είσοδο παίρνει τα δεδομένα Complete diagnosis. Σαν έξοδο της απόφασης παίρνουμε το Complete diagnosis. Συνεχίζοντας η εργασία εάν μπορεί να πραγματοποιηθεί διάγνωση οπότε ακολουθείται ο κλάδος Ναι προχωράμε στην εργασία Specify Medical Treatment την οποία δημιουργούμε όπως και τις υπόλοιπες εργασίες που προαναφέρθηκαν. Εάν όμως δεν μπορεί να πραγματοποιηθεί διάγνωση ακολουθείται ο κλάδος Όχι οπότε καταλήγουμε στην τοπική εργασία Consult a Specialist. Σαν έξοδο της εργασίας καταλήγουμε στο επιχειρησιακό στοιχείο exam results of specialist το οποίο δημιουργούμε στην διακλάδωση έργων και στο οποίο ορίζουμε το εξής χαρακτηριστικό exam results of specialist. Επίσης συσχετίζουμε το συγκεκριμένο επιχειρησιακό στοιχείο στην έξοδο της τοπικής εργασίας. Αυτή η έξοδος τελικά γίνεται είσοδος της τοπικής εργασίας που είχαμε δημιουργήσει ποιο πριν και λέγεται Specify Medical Treatment. Αυτό γίνεται με τη λογική πως εφόσον εξεταστεί ο ασθενής από κάποιο ειδικό γιατρό είναι ικανά τα αποτελέσματα ώστε να πραγματοποιηθεί διάγνωση. Έτσι η τοπική εργασία Specify Medical Treatment δέχεται και μια δεύτερη είσοδο την exam results of specialist που προέρχεται από την τοπική εργασία Consult a Specialist. Είναι σημαντικό να τονίσουμε πως για να ξεκινήσει η τοπική εργασία Specify Medical Treatment δεν είναι απαραίτητη η είσοδος και των δυο επιχειρησιακών στοιχείων. Αντίθετα για να ξεκινήσει απαιτείται μόνο μια εκ των δυο

εισόδων. Αυτό το ορίζουμε στο πεδίο Λογική εισόδου με τη επιλογή του διαζευκτικού ανάμεσα στις δυο εισόδους.

Σε αυτή την ενότητα εμφανίζονται τα κριτήρια εισόδου για αυτό το στοιχείο. Θα πρέπει να ικανοποιηθεί τουλάχιστον ένα από τα κριτήρια αυτά για να γίνει έναρξη του στοιχείου.

	Όνομα	Com...	Com...	Κριτήριο	
	Complite Diagnosis	<input checked="" type="checkbox"/>	<input type="checkbox"/>	CompliteDiagnosis	Προσθήκη
Η	Complite Diagnosis2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	CompliteDiagnosis2	Αφαίρεση
					Τροποποίηση

Εάν δεν το αλλάζουμε και το αφήσουμε στην επιλογή ΚΑΙ τότε η εργασία θα περιμένει είσοδο και από τις δυο εισόδους ώστε να εκκινήσει γεγονός που δεν μπορεί να πραγματοποιηθεί καθώς ποιο πριν χρησιμοποιείται στοιχείο διπλής απόφασης. Οπότε η εργασία θα μπλοκάρει.

Τέλος σαν έξοδο της τοπικής εργασίας περνούμε το στοιχείο Specify Medical Treatment το οποίο δημιουργούμε στην διακλάδωση έργων και του ορίζουμε το χαρακτηριστικό MedicalTreatment. Έπειτα αντιστοιχούμε τα δεδομένα με την έξοδο της τοπικής εργασίας. Εφόσον καθοριστεί η φαρμακευτική αγωγή του ασθενούς ολοκληρώνεται η διαδικασία άρα και η μοντελοποίηση του έργου.

Κεφάλαιο 5

5.1 ΕΙΣΑΓΩΓΗ

Σε αυτό το κεφάλαιο παρουσιάζεται ο BPEL και WSDL κώδικας που υλοποιεί την παραπάνω επιχειρησιακή διαδικασία και γίνεται μια προσπάθεια να αναλυθούν τα επιμέρους συστατικά του ώστε να καταλάβει ο αναγνώστης πως αυτά οδηγούν στην ολοκλήρωση της επιχειρησιακής διαδικασίας. Αφού σε πρώτο στάδιο παρατίθεται ο κώδικας της διαδικασίας έπειτα επιλέγεται ένα μέρος της διαδικασίας και γίνεται σχηματική αναπαράσταση του ώστε να μπορέσει ο αναγνώστης μέσω της απεικόνισης να καταλάβει πως τα επιμέρους στοιχεία συμμετέχουν και αλληλεπιδρούν μεταξύ τους ώστε να ολοκληρωθεί η διαδικασία.

5.2 Ο Bpel κώδικας της διαδικασίας

Παρακάτω παρουσιάζεται ο bpeL κώδικας που υλοποιεί την επιχειρησιακή μας διαδικασία στον οποίο φαίνονται όλα τα επιμέρους στοιχεία που την απαρτίζουν όπως τα partner links (συνεργάτες που θα συμμετάσχουν στη διαδικασία), οι μεταβλητές στις οποίες αποθηκεύονται τα μηνύματα τα οποία στέλνονται, τα links μεταξύ των συνεργατών καθώς και οι δραστηριότητες (activities) της διαδικασίας.

```
<?xml version="1.0" encoding="UTF-8"?>
<process xmlns="http://schemas.xmlsoap.org/ws/2003/03/business-process/"
xmlns:ns="http://CE94CEB9CEB5CF81CEB3CEB1CF83CEAFCEB5CF82/CE94CEB9CEB5CF81CEB3CEB1CF83CEAFCEB11/interface"
xmlns:sss="http://www.ibm.com/schemas/workflow/wswf/plugins/staff" xmlns:wpc="http://www.ibm.com/xmlns/prod/websphere/business-process/v5.1/"
name="CE94CEB9CEB5CF81CEB3CEB1CF83CEAFCEB11" suppressJoinFailure="yes"
targetNamespace="http://CE94CEB9CEB5CF81CEB3CEB1CF83CEAFCEB5CF82/CE94CEB9CEB5CF81CEB3CEB1CF83CEAFCEB11"
wpc:displayName=" μοντελοποίηση εισαγωγής ασθενούς" wpc:executionMode="longRunning" wpc:validFrom="2008-10-05T08:09:23">
```

<<ΟΙ ΣΥΝΕΡΓΑΤΕΣ ΠΟΥ ΘΑ ΣΥΜΜΕΤΑΣΧΟΥΝ ΣΤΗΝ ΕΠΙΧΕΙΡΗΣΙΑΚΗ ΔΙΑΔΙΚΑΣΙΑ>>

<partnerLinks>

<partnerLink name="1Partner" partnerLinkType="ns:1PartnerLink" partnerRole="1PService"/>

<partnerLink name="bedallocatePartner" partnerLinkType="ns:bedallocatePartnerLink"
partnerRole="bedallocateService"/>

<partnerLink name="patientsclinicalexaminationPartner"
partnerLinkType="ns:patientsclinicalexaminationPartnerLink"
partnerRole="patientsclinicalexaminationService"/>

<partnerLink name="patientsparaclinicalexaminationPartner"
partnerLinkType="ns:patientsparaclinicalexaminationPartnerLink"
partnerRole="patientsparaclinicalexaminationService"/>

<partnerLink name="createpatientfilePartner" partnerLinkType="ns:createpatientfilePartnerLink"
partnerRole="createpatientfileService"/>

<partnerLink name="accessfindingstoestablisharunningdiagnosisPartner"
partnerLinkType="ns:accessfindingstoestablisharunningdiagnosisPartnerLink"
partnerRole="accessfindingstoestablisharunningdiagnosisService"/>

<partnerLink name="specifymedicaltreatmentPartner"
partnerLinkType="ns:specifymedicaltreatmentPartnerLink"
partnerRole="specifymedicaltreatmentService"/>

<partnerLink name="consultaspecialistPartner"
partnerLinkType="ns:consultaspecialistPartnerLink" partnerRole="consultaspecialistService"/>

</partnerLinks>

<< ΟΙ ΜΕΤΑΒΛΗΤΕΣ ΣΤΙΣ ΟΠΟΙΕΣ ΑΠΟΘΗΚΕΥΟΝΤΑΙ ΤΑ ΜΗΝΥΜΑΤΑ ΠΟΥ ΑΝΤΑΛΛΑΣΣΟΝΤΑΙ ΚΑΤΑ ΤΗΝ ΕΚΤΕΛΕΣΗ ΤΗΣ ΔΙΑΔΙΚΑΣΙΑΣ >>

<variables>

<variable messageType="ns:ΚριτήριοεισόδουMessage" name="ΚριτήριοεισόδουVariable"/>

<variable messageType="ns: bedallocateinputMessage" name="bedallocateinputVariable"/>

<variable messageType="ns: bedallocateoutputMessage" name="bedallocateoutputVariable"/>

**<variable messageType="ns: createpatientfileoutputMessage"
name="createpatientfileoutputVariable"/>**

**<variable messageType="ns: patientsclinicalexaminationinputMessage"
name="patientsclinicalexaminationinputVariable"/>**

**<variable messageType="ns: patientsclinicalexaminationoutputMessage"
name="patientsclinicalexaminationoutputVariable"/>**

**<variable messageType="ns: patientsparaclinicalexaminationinputMessage"
name="patientsparaclinicalexaminationinputVariable"/>**

**<variable messageType="ns: patientsparaclinicalexaminationoutputMessage"
name="patientsparaclinicalexaminationoutputVariable"/>**

**<variable messageType="ns: accessfindingstoestablisharunningdiagnosisinputMessage"
name="accessfindingstoestablisharunningdiagnosisinputVariable"/>**

**<variable messageType="ns: accessfindingstoestablisharunningdiagnosisoutputMessage"
name="accessfindingstoestablisharunningdiagnosisoutputVariable"/>**

**<variable messageType="ns: specifymedicaltreatmentinputMessage"
name="specifymedicaltreatmentinputVariable"/>**

**<variable messageType="ns: specifymedicaltreatmentoutputMessage"
name="specifymedicaltreatmentoutputVariable"/>**

```
<variable messageType="ns: consultaspecialistoutputMessage"
name="consultaspecialistoutputVariable"/>
```

```
</variables>
```

```
<flow wpc:id="1000">
```

```
<links>
```

```
<link name="1_Reply"/>
```

```
<link name="Κρητήρριοεισόδου_to_bedallocate"/>
```

```
<link name="bedallocate_to_createpatientfile"/>
```

```
<link name="createpatientfile_1_to_Assign"/>
```

```
<link name="Assign_to_patientsclinicalexamination"/>
```

```
<link name="createpatientfile_2_to_Assign"/>
```

```
<link name="Assign_to_patientsparaclinicalexamination"/>
```

```
<link name="patientsclinicalexamination_to_Assign"/>
```

```
<link name="Assign_to_accessfindingstoestablisharunningdiagnosis1"/>
```

```
<link name="patientsparaclinicalexamination_to_Assign"/>
```

```
<link name="Assign_to_accessfindingstoestablisharunningdiagnosis2"/>
```

```
<link name="accessfindingstoestablisharunningdiagnosis_to_isitfeasibletoestablisharunningdiagnosis"/>
```

```
<link name="isitfeasibletoestablisharunningdiagnosis_to_Assign"/>
```

```
<link name="Assign_to_specifymedicaltreatment"/>
```

```
<link name="isitfeasibletoestablisharunningdiagnosis_to_consultaspecialist"/>
```

```
<link name="consultaspecialist_to_Assign"/>
```

```
<link name="Assign_to_specifymedicaltreatment2"/>
```

```
</links>
```

```
<<Η ΔΟΜΗ ΤΗΣ ΔΙΑΔΙΚΑΣΙΑΣ>>
```

```
<<Η ΔΙΑΔΙΚΑΣΙΑ ΛΑΜΒΑΝΕΙ ΣΑΝ ΑΙΤΗΣΗ ΤΗΝ ΕΙΣΟΔΟ ΤΟΥ ΧΡΗΣΤΗ ΣΤΗ ΔΙΑΔΙΚΑΣΙΑ>>
```

```
<receive createInstance="yes" name="send"
```

```
operation="send1"
```

```
partnerLink="1Partner"
```

```
portType="ns:1PT"
```



```
variable="1Variable"  
wpc:displayName=" 1 Receive" wpc:id="1001">  
  <source linkName=" Κρητήριοεισόδου_to_bedallocate "/>  
</receive>
```

```
<<ΣΤΗ ΣΥΝΕΧΕΙΑ ΕΝΤΟΠΙΖΕΙ ΚΡΕΒΑΤΙ ΓΙΑ ΤΟΝ ΑΣΘΕΝΗ>>  
<invoke inputVariable=" ΚρητήριοεισόδουVariable "  
  name="bedallocate"  
  operation="sendbedallocate"  
  outputVariable="bedallocateinputVariable"  
  partnerLink="bedallocatePartner"  
  portType="ns:bedallocatePT"  
  wpc:displayName="bed allocate" wpc:id="1002">  
  <target linkName=" Κρητήριοεισόδου_to_bedallocate"/>  
  <source linkName=" bedallocate_to_createpatientfile"/>  
</invoke>
```

```
<<ΕΠΕΙΤΑ Η ΔΙΑΔΙΚΑΣΙΑ ΔΗΜΙΟΥΡΓΕΙ ΦΑΚΕΛΟ ΓΙΑ ΤΟΝ ΑΣΘΕΝΗ>>  
<invoke inputVariable=" bedallocateoutputVariable"  
  name=" createpatientfile"  
  operation="sendcreatepatientfile"  
  outputVariable=" createpatientfileoutputVariable"  
  partnerLink="createpatientfilePartner"  
  portType="ns:createpatientfilePT"  
  wpc:displayName="create patient file" wpc:id="1003">  
  <target linkName="bedallocate_to_createpatientfile"/>  
  <source linkName=" createpatientfile_2_to_Assign "/>  
  <source linkName=" createpatientfile_1_to_Assign"/>  
</invoke>
```

```
<<Η ΔΙΑΔΙΚΑΣΙΑ ΕΤΟΙΜΑΖΕΙ ΤΟ ΜΗΝΥΜΑ ΠΟΥ ΘΑ ΑΠΟΣΤΑΛΕΙ ΓΙΑ ΚΛΙΝΙΚΕΣ ΕΞΕΤΑΣΕΙΣ>>  
<assign wpc:id="1004">  
  <target linkName="createpatientfile_1_to_Assign"/>  
  <source linkName="Assign_to_patientsclinicalexamination"/>  
  <copy>  
    <from part="1Part" variable="createpatientfileoutputVariable"/>  
    <to part="Part3" variable="patientsclinicalexaminationinputVariable"/>  
  </copy>
```

</assign>

<<Η ΔΙΑΔΙΚΑΣΙΑ ΕΤΟΙΜΑΖΕΙ ΤΟ ΜΗΝΥΜΑ ΠΟΥ ΘΑ ΑΠΟΣΤΑΛΕΙ ΓΙΑ ΠΑΡΑΚΛΙΝΙΚΕΣ ΕΞΕΤΑΣΕΙΣ>>

<assign wpc:id="1005">

<target linkName="createpatientfile_2_to_Assign"/>

<source linkName="Assign_to_patientsparaclinicalexamination"/>

<copy>

<from part="2Part" variable="createpatientfileoutputVariable"/>

<to part="Part3" variable="patientsparaclinicalexaminationinputVariable"/>

</copy>

</assign>

<<Η ΔΙΑΔΙΚΑΣΙΑ ΕΤΟΙΜΑΖΕΙ ΤΟ ΜΗΝΥΜΑ ΠΟΥ ΘΑ ΑΠΟΣΤΑΛΕΙ ΓΙΑ ΔΙΑΓΝΩΣΗ ΤΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ>>

<assign wpc:id="1006">

<target linkName="patientsclinicalexamination_to_Assign"/>

<source linkName="Assign_to_accessfindingstoestablisharunningdiagnosis1"/>

<copy>

<from part="Part3" variable="patientsclinicalexaminationoutputVariable"/>

<to part="Part2" variable="accessfindingstoestablisharunningdiagnosisinputVariable"/>

</copy>

</assign>

<< ΕΔΩ ΠΡΑΓΜΑΤΟΠΟΙΕΙΤΑΙ Η ΚΛΙΝΙΚΗ ΕΞΕΤΑΣΗ ΤΟΥ ΑΣΘΕΝΟΥΣ >>

<invoke inputVariable="patientsclinicalexaminationinputVariable"

name="patientsclinicalexamination"

operation="sendpatientsclinicalexamination"

outputVariable="patientsclinicalexaminationoutputVariable"

partnerLink="patientsclinicalexaminationPartner"

portType="ns:patientsclinicalexaminationPT"

wpc:displayName="patients clinical examination" wpc:id="1007">

<target linkName="Assign_to_patientsclinicalexamination"/>

<source linkName="patientsclinicalexamination_to_Assign"/>

</invoke>

<<Η ΔΙΑΔΙΚΑΣΙΑ ΕΤΟΙΜΑΖΕΙ ΤΟ ΜΗΝΥΜΑ ΠΟΥ ΘΑ ΑΠΟΣΤΑΛΕΙ ΓΙΑ ΔΙΑΓΝΩΣΗ ΤΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ>>

<assign wpc:id="1008">

<target linkName="patientsparaclinicalexamination_to_Assign"/>

<source linkName="Assign_to_accessfindingstoestablisharunningdiagnosis2"/>

```

<copy>
  <from part="Part4" variable="patientsparaclinicalexaminationoutputVariable"/>
  <to part="2Part"

variable="accessfindingstoestablisharunningdiagnosisinputVariable"/>
</copy>

```

```

</assign>

```

<< ΕΔΩ ΠΡΑΓΜΑΤΟΠΟΙΕΙΤΑΙ Η ΠΑΡΑΚΛΙΝΙΚΗ ΕΞΕΤΑΣΗ ΤΟΥ ΑΣΘΕΝΟΥΣ-ΑΠΑΙΤΕΙΤΑΙ Η ΣΥΜΜΕΤΟΧΗ ΤΟΥ ΙΑΤΡΟΥ >>

```

<invoke inputVariable="patientsparaclinicalexaminationinputVariable"      name="patientsparaclinicalexamination"
operation="sendpatientsparaclinicalexamination"
outputVariable="patientsparaclinicalexaminationoutputVariable"
partnerLink="null"
portType="ns:patientsparaclinicalexaminationPT"
wpc:displayName="patients paraclinical examination" wpc:id="1009">
  <wpc:staff>
    <wpc:potentialOwner>
      <sss:verb>
        <sss:name>Role Members</sss:name>
        <sss:id>Role Members</sss:id>
        <sss:parameter id="RoleName">doctor</sss:parameter>
        <sss:parameter id="IncludeNestedRoles">>false</sss:parameter>
      </sss:verb>
    </wpc:potentialOwner>
  </wpc:staff>
  <target linkName="Assign_to_patientsparaclinicalexamination"/>
  <source linkName="patientsparaclinicalexamination_to_Assign"/>
</invoke>

```

<<ΑΞΙΟΛΟΓΟΥΝΤΑΙ ΤΑ ΑΠΟΤΕΛΕΣΜΑΤΑ ΤΩΝ ΚΛΙΝΙΚΩΝ ΚΑΙ ΠΑΡΑΚΛΙΝΙΚΩΝ ΕΞΕΤΑΣΕΩΝ>>

```

<invoke inputVariable="accessfindingstoestablisharunningdiagnosisinputVariable"
name="accessfindingstoestablisharunningdiagnosis"
operation="sendaccessfindingstoestablisharunningdiagnosis"
outputVariable="accessfindingstoestablisharunningdiagnosisoutputVariable"
partnerLink="accessfindingstoestablisharunningdiagnosisPartner"
portType="ns:accessfindingstoestablisharunningdiagnosisPT"
wpc:displayName="access findings to establish a running diagnosis" wpc:id="1010">

```

```

    <wpc:joinCondition>
    <wpc:javaCode><![CDATA[return
    ((getLinkStatus("Assign_to_accessfindingstoestablisharunningdiagnosis1")) &&
    (getLinkStatus("Assign_to_accessfindingstoestablisharunningdiagnosis2")));

    ]]></wpc:javaCode>
    </wpc:joinCondition>
    <target linkName="Assign_to_accessfindingstoestablisharunningdiagnosis1"/>
    <target linkName="Assign_to_accessfindingstoestablisharunningdiagnosis2"/>
    <source
    linkName="accessfindingstoestablisharunningdiagnosis,_to_isitfeasibletoestablisharunningdiagnosis"/>
</invoke>

```

<<ΑΠΟΦΑΣΙΖΕΤΑΙ ΑΝ ΕΙΝΑΙ ΔΥΝΑΤΟΝ ΝΑ ΠΡΟΧΩΡΗΣΕΙ Ο ΑΣΘΕΝΗΣ ΣΕ ΦΑΡΜΑΚΕΥΤΙΚΗ ΑΓΩΓΗ>>

<empty name="isitfeasibletoestablisharunningdiagnosis"

wpc:displayName="is it feasible to establish a
running diagnosis ?" wpc:id="1011">

<wpc:description> ΕΑΝ ΝΑΙ ΤΟΤΕ ΠΡΟΧΩΡΑΜΕ ΣΕ ΚΑΘΟΡΙΣΜΟ ΦΑΡΜΑΚΕΥΤΙΚΗΣ
ΑΓΩΓΗΣ .ΕΑΝ ΟΧΙ ΤΟΤΕ ΖΗΤΕΙΤΑΙ Η ΣΥΜΒΟΥΛΗ ΚΑΠΟΙΟΥ ΕΙΔΙΚΟΥ ΓΙΑΤΡΟΥ
</wpc:description>

<target linkName="accessfindingstoestablisharunningdiagnosis,_to_isitfeasibletoestablisharunningdiagnosis"/>

<source linkName="isitfeasibletoestablisharunningdiagnosis,_to_consultaspecialist">

<wpc:transitionCondition>

<wpc:javaCode><![CDATA[return (((true) != (true))) && (!(((true) == (true))))];

]]></wpc:javaCode>

</wpc:transitionCondition>

</source>

<source linkName="isitfeasibletoestablisharunningdiagnosis,_to_Assign">

<wpc:transitionCondition>

<wpc:javaCode><![CDATA[return ((true) == (true));

]]></wpc:javaCode>

</wpc:transitionCondition>

</source>

</empty>

<<Η ΔΙΑΔΙΚΑΣΙΑ ΕΤΟΙΜΑΖΕΙ ΤΟ ΜΗΝΥΜΑ ΠΟΥ ΘΑ ΣΤΕΙΛΕΙ ΣΤΟ ΣΤΑΔΙΟ ΚΑΘΟΡΙΣΜΟΥ ΤΗΣ ΦΑΡΜΑΚΕΥΤΙΚΗΣ ΑΓΩΓΗΣ>>

```
<assign wpc:id="1012">
  <target linkName="isitfeasibletoestablisharunningdiagnosis_to_Assign"/>
  <source linkName="Assign_to_specifymedicaltreatment"/>
  <copy>
    <from part="Part"
      variable="accessfindingstoestablisharunningdiagnosisoutputVariable"/>
    <to part="Part3" variable="specifymedicaltreatmentinputVariable"/>
  </copy>
</assign>
```

<<ΑΥΤΟ ΤΟ WEB SERVICE ΚΑΘΟΡΙΖΕΙ ΤΗ ΦΑΡΜΑΚΕΥΤΙΚΗ ΑΓΩΓΗ ΠΟΥ ΘΑ ΑΚΟΛΟΥΘΗΣΕΙ Ο ΑΣΘΕΝΗΣ>>

```
<invoke
  inputValue="specifymedicaltreatmentinputVariable"
  name="specifymedicaltreatment"
  operation="sendspecifymedicaltreatment"
  outputVariable="specifymedicaltreatmentoutputVariable"
  partnerLink="specifymedicaltreatmentPartner"
  portType="ns:specifymedicaltreatmentPT"
  wpc:displayName="specify medical treatment" wpc:id="1013">
  <wpc:joinCondition>
    <wpc:javaCode><![CDATA[return ((getLinkStatus("Assign_to_specifymedicaltreatment2")) &&
      (getLinkStatus("Assign_to_specifymedicaltreatment")));
    ]]></wpc:javaCode>
  </wpc:joinCondition>
  <target linkName="Assign_to_specifymedicaltreatment2"/>
  <target linkName="Assign_to_specifymedicaltreatment"/>
  <source linkName="Reply"/>
</invoke>
```

<<Η ΔΙΑΔΙΚΑΣΙΑ ΕΤΟΙΜΑΖΕΙ ΤΟ ΜΗΝΥΜΑ ΠΟΥ ΘΑ ΣΤΑΛΕΙ ΣΤΟ ΣΤΑΔΙΟ ΚΑΘΟΡΙΣΜΟΥ ΤΗΣ ΦΑΡΜΑΚΕΥΤΙΚΗΣ ΑΓΩΓΗΣ>>

```
<assign wpc:id="1014">
  <target linkName="consultspecialist_to_Assign"/>
  <source linkName="Assign_to_specifymedicaltreatment2"/>
```

```

    <copy>
    <from part="Part3" variable="consultaspecialistoutputVariable"/>
    <to part="Part2" variable="specifymedicaltreatmentinputVariable"/>
    </copy>
</assign>

<<ΑΥΤΟ ΤΟ WEB SERVICE ΕΙΝΑΙ ΥΠΕΥΘΥΝΟ ΓΙΑ ΤΗΝ ΠΡΑΓΜΑΤΟΠΟΙΗΣΗ ΕΞΕΤΑΣΗΣ ΑΠΟ ΕΙΔΙΚΟ ΣΕ
ΠΕΡΙΠΤΩΣΗ ΠΟΥ ΤΑ ΑΠΟΤΕΛΕΣΜΑΤΑ ΔΕΝ ΕΙΝΑΙ ΑΡΚΕΤΑ ΩΣΤΕ ΝΑ ΠΡΟΧΩΡΗΣΟΥΜΕ ΣΕ ΔΙΑΓΝΩΣΗ>>
<invoke
  inputValue="accessfindingstoestablisharunningdiagnosisoutputVariable" name="consultaspecialist"
  operation="sendconsultaspecialist"
  outputVariable="consultaspecialistVariable"
  partnerLink="consultaspecialistPartner"
  portType="ns:consultaspecialistPT"
  wpc:displayName="consult a specialist" wpc:id="1015">
  <target linkName="isitfeasibletoestablisharunningdiagnosis_to_consultaspecialist"/>
  <source linkName="consultaspecialist_to_Assign"/>
</invoke>

<reply
  name="sendReply"
  operation="send"
  partnerLink="1Partner"
  portType="ns:1PT"
  variable="1Variable"
  wpc:displayName=" 1 Reply" wpc:id="1016">
  <target linkName="1_Reply"/>
</reply>

</flow>
</process>

```

5.3 Ο WSDL ορισμός της διαδικασίας

Στο παρακάτω WSDL έγγραφο περιλαμβάνονται όλοι οι WSDL ορισμοί που είναι απαραίτητοι στην επιχειρησιακή διαδικασία. Το συγκεκριμένο WSDL ακολουθεί τη σύνταξη

και τη δομή ενός πλήρους Wsdl εγγράφου με όλα τα επιμέρους στοιχεία του όπως πχ port types και μηνύματα.

Στην αρχή του εγγράφου δηλώνονται τα partner link types τα οποία αντιπροσωπεύουν τις διάφορες αλληλεπιδράσεις της BPEL διαδικασίας με τις οντότητες που συμμετέχουν, όπου στη συγκεκριμένη περίπτωση είναι τα web services και ο χρήστης. Επομένως θα ορίσουμε τόσα partnerLinkTypes όσες είναι και οι οντότητες αυτές. Επιπλέον, τα partnerLinkTypes έχουν έναν ή δύο ρόλους ανάλογα με το αν έχουμε σύγχρονη ή ασύγχρονη επικοινωνία αντίστοιχα, ενώ για κάθε ρόλο πρέπει να καθορίσουμε το portType που χρησιμοποιεί. Στο παράδειγμα αυτό όπως αναφέραμε πριν ασύγχρονη επικοινωνία δεν έχουμε οπότε όλα τα partner link types έχουν ένα ρόλο.

Έπειτα ακολουθεί ο ορισμός των μηνυμάτων που χρησιμοποιούνται για την επικοινωνία της διαδικασίας με άλλες οντότητες. Η δήλωση κάθε μηνύματος γίνεται με το tag <message> όπως φαίνεται και στο αντίστοιχο τμήμα του WSDL εγγράφου που ακολουθεί. Τέλος ορίζουμε τα portTypes τόσο της ίδιας της διαδικασίας όσο και των web services που συμμετέχουν. Ορίζουμε ένα portType για κάθε web service που έχουμε.

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions
xmlns:CE95CF80CEB9CF87CEB5CEB9CF81CEB7CF83CEB9CEB1CEBACEACCF8
3CF84CEBFCEB9CF87CEB5CEAFCEB1="http://CE95CF80CEB9CF87CEB5CEB9C
F81CEB7CF83CEB9CEB1CEBACEACCF83CF84CEBFCEB9CF87CEB5CEAFCEB1 "
xmlns:plnk="http://schemas.xmlsoap.org/ws/2003/05/partner-link/"
xmlns:tns="http://CE94CEB9CEB5CF81CEB3CEB1CF83CEAFCEB5CF82/CE94CEB
9CEB5CF81CEB3CEB1CF83CEAFCEB11/interface"
xmlns:wsa="http://schemas.xmlsoap.org/ws/2003/03/addressing"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://CE94CEB9CEB5CF81CEB3CEB1CF83CEAFCEB5CF82/CE
94CEB9CEB5CF81CEB3CEB1CF83CEAFCEB11/interface">
<wsdl:types>
<xsd:schema
xmlns:CE95CF80CEB9CF87CEB5CEB9CF81CEB7CF83CEB9CEB1CEBACEACCF83CF84CEBFCE
B9CF87CEB5CEAFCEB1="http://CE95CF80CEB9CF87CEB5CEB9CF81CEB7CF83CEB9CEB1CEB
ACEACCF83CF84CEBFCEB9CF87CEB5CEAFCEB1"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
```

```

<xsd:import
  namespace="http://CE95CF80CEB9CF87CEB5CEB9CF81CEB7CF83CEB9CEB1CEBACEACCF83C
  F84CEBFCEB9CF87CEB5CEAFCEB1"
  schemaLocation="../../CE95CF80CEB9CF87CEB5CEB9CF81CEB7CF83CEB9CEB1CEBACEACCF83
  CF84CEBFCEB9CF87CEB5CEAFCEB1/CE95CF80CEB9CF87CEB5CEB9CF81CEB7CF83CEB9CEB
  1CEBACEACCF83CF84CEBFCEB9CF87CEB5CEAFCEB1.xsd"/>
</xsd:schema>
</wsdl:types>
<PARTNER LINK TYPES>
<PARTNER LINK TYPE ΓΙΑ ΤΗΝ ΑΛΛΗΛΕΠΙΔΡΑΣΗ ΜΕ ΤΟ WEB SERVICE RECEIVE>
<plnk:partnerLinkType name="1PartnerLink">
  <plnk:role name="1Service">
    <plnk:portType name="tns:1PT"/>
  </plnk:role>
</plnk:partnerLinkType>

<PARTNER LINK TYPE ΓΙΑ ΤΗΝ ΑΛΛΗΛΕΠΙΔΡΑΣΗ ΜΕ ΤΟ WEB SERVICE BED ALLOCATE>
<plnk:partnerLinkType name="bedallocatePartnerLink">
  <plnk:role name="bedallocateService">
    <plnk:portType name="tns:bedallocatePT"/>
  </plnk:role>
</plnk:partnerLinkType>

<PARTNER LINK TYPE ΓΙΑ ΤΗΝ ΑΛΛΗΛΕΠΙΔΡΑΣΗ ΜΕ ΤΟ WEB SERVICE CREATE PATIENTS FILE>
<plnk:partnerLinkType name="createpatientfilePartnerLink">
  <plnk:role name="createpatientfileService">
    <plnk:portType name="tns:createpatientfilePT"/>
  </plnk:role>
</plnk:partnerLinkType>

<PARTNER LINK TYPE ΓΙΑ ΤΗΝ ΑΛΛΗΛΕΠΙΔΡΑΣΗ ΜΕ ΤΟ WEB SERVICE PATIENTS CLINICAL
EXAMINATION>
<plnk:partnerLinkType name="patientsclinicalexaminationPartnerLink">
  <plnk:role name="patientsclinicalexaminationService">
    <plnk:portType name="tns:patientsclinicalexaminationPT"/>
  </plnk:role>
</plnk:partnerLinkType>

<PARTNER LINK TYPE ΓΙΑ ΤΗΝ ΑΛΛΗΛΕΠΙΔΡΑΣΗ ΜΕ ΤΟ WEB SERVICE PATIENTS PARACLINICAL

```


EXAMINATION>

```
<plnk:partnerLinkType name="patientsparaclinicalexaminationPartnerLink">  
  <plnk:role name="patientsparaclinicalexaminationService">  
    <plnk:portType name="tns:patientsparaclinicalexaminationPT"/>  
  </plnk:role>  
</plnk:partnerLinkType>
```

<PARTNER LINK TYPE ΓΙΑ ΤΗΝ ΑΛΛΗΛΕΠΙΔΡΑΣΗ ΜΕ ΤΟ WEB SERVICE ACCESS FINDINGS TO ESTABLISH A RUNNING DIAGNOSIS>

```
<plnk:partnerLinkType name="accessfindingstoestablisharunningdiagnosisPartnerLink">  
  <plnk:role name="accessfindingstoestablisharunningdiagnosisService">  
    <plnk:portType name="tns:accessfindingstoestablisharunningdiagnosisPT"/>  
  </plnk:role>  
</plnk:partnerLinkType>
```

<PARTNER LINK TYPE ΓΙΑ ΤΗΝ ΑΛΛΗΛΕΠΙΔΡΑΣΗ ΜΕ ΤΟ WEB SERVICE SPECIFY MEDICAL TREATMENT>

```
<plnk:partnerLinkType name="specifymedicaltreatmentPartnerLink">  
  <plnk:role name="specifymedicaltreatmentService">  
    <plnk:portType name="tns:specifymedicaltreatmentPT"/>  
  </plnk:role>  
</plnk:partnerLinkType>
```

<PARTNER LINK TYPE ΓΙΑ ΤΗΝ ΑΛΛΗΛΕΠΙΔΡΑΣΗ ΜΕ ΤΟ WEB SERVICE CONSULT A SPECIALIST>

```
<plnk:partnerLinkType name="consultaspecialistPartnerLink">  
  <plnk:role name="consultaspecialistService">  
    <plnk:portType name="tns:consultaspecialistPT"/>  
  </plnk:role>  
</plnk:partnerLinkType>
```

<ΤΑ ΜΗΝΥΜΑΤΑ ΕΠΙΚΟΙΝΩΝΙΑΣ ΤΩΝ WEB SERVICES>

<ΤΟ ΜΗΝΥΜΑ ΠΟΥ ΠΕΡΙΕΧΕΙ ΤΑ ΣΤΟΙΧΕΙΑ ΕΙΣΟΔΟΥ ΤΟΥ ΧΡΗΣΤΗ >

```
<wsdl:message name="Message">  
  <wsdl:part name="Part"  
    type="CE95CF80CEB9CF87CEB5CEB9CF84CEBFCEB9CF87CEB5CEAFCEB1:Patientsdatas"/>
```

```
</wsdl:message>
  <wsdl:message name="Message">
    <wsdl:part name="Part" type="xsd:string"/>
  </wsdl:message>
```

<ΤΟ ΜΗΝΥΜΑ ΠΟΥ ΠΕΡΙΕΧΕΙ ΤΟ ΜΗΝΥΜΑ ΓΙΑ ΤΟΝ ΕΝΤΟΠΙΣΜΟ ΚΡΕΒΑΤΙΟΥ>

```
<wsdl:message name="Message2">
  <wsdl:part name="Part2"
    type="CE95CF80CBACEACCF83CF84CEBFCEB9CF87CEB5CEAFCEB1:Bedallocates"/>
</wsdl:message>
```

<ΜΗΝΥΜΑ ΠΟΥ ΔΗΜΙΟΥΡΓΕΙΤΑΙ ΜΕΤΑ ΤΗΝ ΔΗΜΙΟΥΡΓΙΑ ΤΟΥ ΦΑΚΕΛΟΥ ΤΟΥ ΑΣΘΕΝΟΥΣ ΚΑΙ ΠΗΓΑΙΝΕΙ ΣΤΗ ΔΙΑΔΙΚΑΣΙΑ ΚΛΙΝΙΚΗΣ ΕΞΕΤΑΣΗΣ >

```
<wsdl:message name="Message3">
  <wsdl:part name="1Part"
    type="CE95CF80CEEACCF83CF84CEBFCEB9CF87CEB5CEAFCEB1:Patientsfileinformations"/>
  <wsdl:part name="2Part"
    type="CE95CF80CBACEACCF83CF84CEBFB9CF87CEB5CEAFCEB1:Patientsfileinformations"/>
</wsdl:message>
```

<ΜΗΝΥΜΑ ΠΟΥ ΔΗΜΙΟΥΡΓΕΙΤΑΙ ΜΕΤΑ ΤΗΝ ΔΗΜΙΟΥΡΓΙΑ ΤΟΥ ΦΑΚΕΛΟΥ ΤΟΥ ΑΣΘΕΝΟΥΣ ΚΑΙ ΠΗΓΑΙΝΕΙ ΣΤΗ ΔΙΑΔΙΚΑΣΙΑ ΠΑΡΑΚΛΙΝΙΚΗΣ ΕΞΕΤΑΣΗΣ >

```
<wsdl:message name="Message3">
  <wsdl:part name="Part3"
    type="CE95CF80CEB9CCF83CF84CEBFCEB9CF87CEB5CEAFCEB1:Patientsfileinformations"/>
</wsdl:message>
```

<ΤΑ ΜΗΝΥΜΑΤΑ ΠΟΥ ΔΗΜΙΟΥΡΓΟΥΝΤΑΙ ΑΠΟ ΤΗΝ ΚΛΙΝΙΚΗ ΚΑΙ ΠΑΡΑΚΛΙΝΙΚΗ ΕΞΕΤΑΣΗ ΤΟΥ ΑΣΘΕΝΟΥΣ >

```
<wsdl:message name="Message4">
  <wsdl:part name="Part3"
    type="CE95CF80CEB9CF87CF83CF84CEE9CF87CEB5CEAFCEB1:clinicalexaminationexams"/>
</wsdl:message>
```

```
<wsdl:message name="Message5">
  <wsdl:part name="Part2">
```

```
type="CE95CB1CEBACEAF83CF84CEBFCEB9CF87CEB5CEAFCEB1:clinicalexaminationexams"/>
<wsdl:part name="2Part"
type="CE953CEB9CACCF83CF84CEBFCEB9CF87CEB5CEAFCEB1:doctorexaminationresults"/>
</wsdl:message>
```

```
<wsdl:message name="Message5">
<wsdl:part name="Part4"
type="CE95CFEB9CEACCF83CF84CEBFCEB9CF87CEB5CEAFCEB1:doctorexaminationresults"/>
</wsdl:message>
```

<ΤΑ ΜΗΝΥΜΑΤΑ ΠΟΥ ΠΑΡΑΓΟΝΤΑΙ ΜΕΤΑ ΤΗΝ ΕΠΕΞΕΡΓΑΣΙΑ ΤΩΝ ΔΕΔΟΜΕΝΩΝ ΤΩΝ ΕΞΕΤΑΣΕΩΝ ΤΟΥ ΑΣΘΕΝΟΥΣ >

```
<wsdl:message name="Message6">
<wsdl:part name="Part"
type="CE95CFCEBACEACCF83CF84CEBFCEB9CF87CEB5CEAFCEB1:Completediagnosiss"/>
</wsdl:message>
```

```
<wsdl:message name="Message6">
<wsdl:part name="Part2"
type="CE95CF80CEEBAECECF83CF84CEBFCEB9CF87CEB5CEAFCEB1:examresultsofspecialist"/>
<wsdl:part name="Part3"
type="CE95CFCEB9CF87CEB5F83CF84CEBFCEB9CF87CEB5CEAFCEB1:Completediagnosiss"/>
</wsdl:message>
```

<ΤΟ ΜΗΝΥΜΑ ΠΟΥ ΠΑΡΑΓΕΤΑΙ ΓΙΑ ΤΟΝ ΚΑΘΟΡΙΣΜΟ ΤΗΣ ΦΑΡΜΑΚΕΥΤΙΚΗΣ ΑΓΩΓΗΣ ΤΟΥ ΑΣΘΕΝΟΥΣ>

```
<wsdl:message name="Message7">
<wsdl:part name="Part2"
type="CE95CF80CEE1CEBACF83CF84CEBFCEB9CF87CEB5CEAFCEB1:medicaltreatment"/>
</wsdl:message>
```

<ΤΟ ΜΗΝΥΜΑ ΜΕ ΤΗΝ ΠΛΗΡΟΦΟΡΙΑ ΠΟΥ ΠΑΡΑΓΕΤΑΙ ΜΕΤΑ ΤΗΝ ΕΞΕΤΑΣΗ ΑΠΟ ΚΑΠΟΙΟ ΕΙΔΙΚΟ>

```
<wsdl:message name="Message8">
<wsdl:part name="Part3"
type="CE95CF80CEBACEACC3CF84CEBFCEB9CF87CEB5CEAFCEB1:examresultsofspecialist"/>
</wsdl:message>
```

< TA PORT TYPES ΠΟΥ ΠΑΡΕΧΟΝΤΑΙ ΑΠΟ ΤΑ WEB SERVICES>

<PORT TYPE ΕΙΣΟΔΟΥ>

```
<wsdl:portType name="1PT">  
  <wsdl:operation name="send1">  
    <wsdl:input message="tns:Message" name="Message"/>  
    <wsdl:output message="tns:Message" name="Message"/>  
  </wsdl:operation>  
</wsdl:portType>
```

< BED ALLOCATE PORT TYPE>

```
<wsdl:portType name="bedallocatePT">  
  <wsdl:operation name="sendbedallocate">  
    <wsdl:input message="tns:Message" name="Message"/>  
    <wsdl:output message="tns:Message2" name="Message2"/>  
  </wsdl:operation>  
</wsdl:portType>
```

< CREATE PATIENT FILE PORT TYPE>

```
<wsdl:portType name="createpatientfilePT">  
  <wsdl:operation name="sendcreatepatientfile">  
    <wsdl:input message="tns:Message2" name="Message2"/>  
    <wsdl:output message="tns:Message3" name="Message3"/>  
  </wsdl:operation>  
</wsdl:portType>
```

< PATIENTS CLINICAL EXAMINATION PORT TYPE >

```
<wsdl:portType name="patientsclinicalexaminationPT">  
  <wsdl:operation name="sendpatientsclinicalexamination">  
    <wsdl:input message="tns:Message3" name="Message3"/>  
    <wsdl:output message="tns:Message4" name="Message4"/>  
  </wsdl:operation>  
</wsdl:portType>
```

<PATIENTS PARA CLINICAL EXAMINATION PORT TYPE>

```
<wsdl:portType name="patientsparaclinicalexaminationPT">  
  <wsdl:operation name="sendpatientsparaclinicalexamination">
```

```

    <wsdl:input message="tns:Message3" name="Message4"/>
    <wsdl:output message="tns:Message5" name="Message5"/>
  </wsdl:operation>
</wsdl:portType>

<ACCESS FINDINGS TO ESTABLISH A RUNNING DIAGNOSIS PORT TYPE >
<wsdl:portType name="accessfindingsstoestablisharunningdiagnosisPT">
  <wsdl:operation name="sendaccessfindingsstoestablisharunningdiagnosis">
    <wsdl:input message="tns:Message5" name="Message5"/>
    <wsdl:output message="tns:Message6" name="Message6"/>
  </wsdl:operation>
</wsdl:portType>

<SPECIFY MEDICAL TREATMENT PORT TYPE>
<wsdl:portType name="specifymedicaltreatmentPT">
  <wsdl:operation name="sendspecifymedicaltreatment">
    <wsdl:input message="tns:Message6" name="Message6"/>
    <wsdl:output message="tns:Message7" name="Message7"/>
  </wsdl:operation>
</wsdl:portType>

<CONSULT A SPECIALIST PORT TYPE>
<wsdl:portType name="consultaspecialistPT">
  <wsdl:operation name="sendconsultaspecialist">
    <wsdl:input message="tns:Message6" name="Message7"/>
    <wsdl:output message="tns:Message8" name="Message8"/>
  </wsdl:operation>
</wsdl:portType>

</wsdl:definitions>

```

5.4 Σχηματική αναπαράσταση

Έχοντας παρουσιάσει σε προηγούμενο κεφάλαιο τον τρόπο με τον οποίο δημιουργούμε την επιχειρησιακή διαδικασία που μας ενδιαφέρει μέσω του εργαλείου Websphere

business modeler και έχοντας δει σε αυτό το κεφάλαιο τον κώδικα που την υλοποιεί, παραθέτουμε στη συνέχεια ένα κομμάτι της διαδικασίας αυτής ώστε να εξηγήσουμε τον τρόπο που λειτουργεί. Συγκεκριμένα επιλέξαμε το κομμάτι **Bed allocate → Create patients file**. Έχοντας υπόψη τον κώδικα που υλοποιεί το συγκεκριμένο κομμάτι καταλήξαμε στην εικόνα 28. Συγκεκριμένα στην εικόνα αυτή βλέπουμε πως οι partners απεικονίζονται σαν τετράγωνα και πως κάθε partner αντιστοιχεί σε μια υπηρεσία της διαδικασίας. Έτσι για κάθε τοπική εργασία υπάρχει και ένας partner. Εδώ βλέπουμε δύο partners τον **Bed Allocate Partner** και τον **Create Patients File Partner**.



Οι δύο αυτοί partners συνδέονται μεταξύ τους με “φυσική” σύνδεση τα λεγόμενα **links**. Τα **links** συμβολίζουν τη σύνδεση μεταξύ δύο υπηρεσιών και μπορεί να είναι αμφίδρομα ή μονόδρομα ανάλογα την περίπτωση. Στη συγκεκριμένη περίπτωση τα **links** είναι μονόδρομα και με κατεύθυνση από τον **Bed allocate partner** προς τον **Create patients file partner** και ονομάζεται **bedallocate_to_createpatientfile link**.



Σχήμα 2 : *bedallocate_to_createpatientfile link*

Στη συνέχεια βλέπουμε τις **μεταβλητές** των μηνυμάτων που ανταλλάσσονται στην διαδικασία. Επιλέξαμε να συμβολίσουμε τις **μεταβλητές** αυτές ως γραμματοκιβώτια στα οποία έρχονται τα αντίστοιχα μηνύματα της διαδικασίας.



Σχήμα 3 : μεταβλητή μηνύματος

Επίσης παρατηρούμε πως συμμετέχουν τα διάφορα **activities** στην διαδικασία και συγκεκριμένα βλέπουμε την activity **Invoke bedallocate**. Μια activity (δραστηριότητα) χρησιμοποιείται από μια διαδικασία για να κάνει κλήσεις σε Web services. Σε συνδυασμό με το port type και τους partners η invoke καθορίζει τα συστατικά των διαδικασιών εισόδου και εξόδου που καλούνται. Πιο συγκεκριμένα η invoke καθορίζει τι θα δεχθεί η bed allocate σαν είσοδο και τι θα βγάλει σαν έξοδο.



Σχήμα 4 : invoke activity

Όλα τα παραπάνω συστατικά που αναφέρθηκαν απαρτίζουν τα συστατικά του bpel εγγράφου μιας επιχειρησιακής διαδικασίας. Στην συνέχεια θα δούμε τα συστατικά που απαρτίζουν ένα wdsi έγγραφο και πως αυτά συμμετέχουν στην διαδικασία.

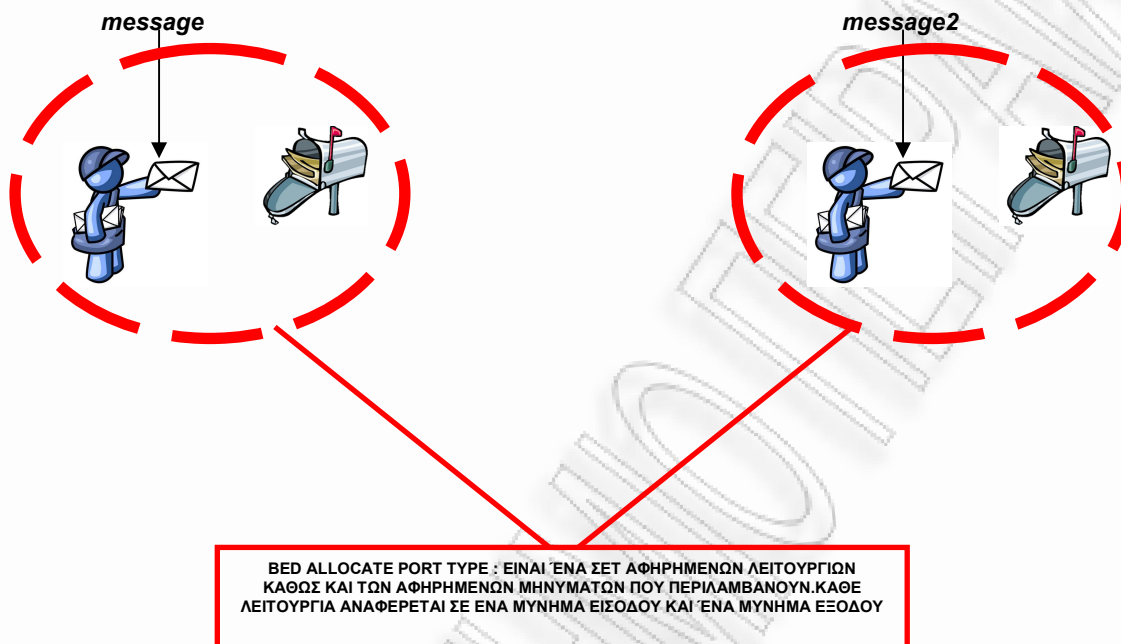
Πρώτα απ όλα βλέπουμε τα μηνύματα που ανταλλάσσονται στη διαδικασία. Παρατηρούμε πως ένας partner δέχεται στην είσοδο του και συγκεκριμένα στην μεταβλητή μηνυμάτων του ένα μήνυμα και βγάζει στην έξοδο ένα μήνυμα απόκρισης.



Σχήμα 5 : Μήνυμα

Τέλος παρατηρούμε τα port types της συγκεκριμένης διαδικασίας. Ποιο συγκεκριμένα βλέπουμε το Bed Allocate Port Type. Ένα port type είναι ένα σετ αφηρημένων λειτουργιών καθώς και των αφηρημένων μηνυμάτων που περιλαμβάνουν. Κάθε λειτουργία αναφέρεται

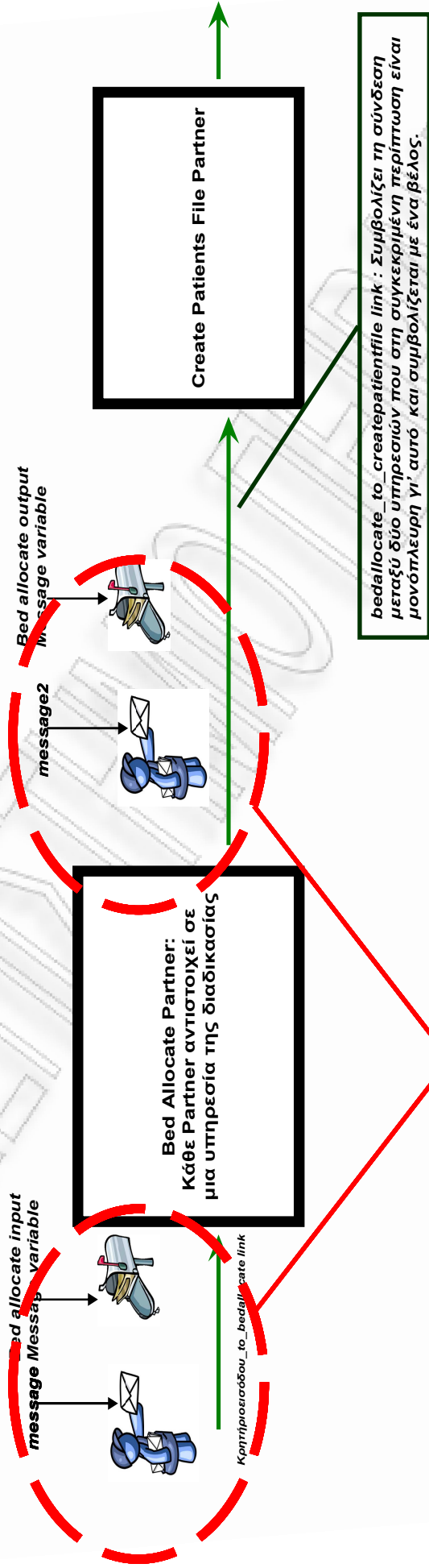
σε ένα μήνυμα εισόδου και ένα μήνυμα εξόδου. Έτσι εδώ το συγκεκριμένο port type αποτελείται από τα μηνύματα εισόδου και εξόδου του bed allocate και τις μεταβλητές τους.



Σχήμα 6 : Port types

Απεικόνιση Bpel και Wsdl στοιχείων σε μία διαδικασία

Invoke bedallocate:
 Χρησιμοποιείται από μία διαδικασία για να κάνει κλήσεις σε Web services. Σε συνδυασμό με το port type και τους partners η invoke καθορίζει τα στοιχεία των διαδικασιών εισόδου και εξόδου που καλούνται



BED ALLOCATE PORT TYPE : ΕΙΝΑΙ ΈΝΑ ΣΕΤ ΑΦΗΡΜΕΝΩΝ ΛΕΙΤΟΥΡΓΙΩΝ ΚΑΘΩΣ ΚΑΙ ΤΩΝ ΑΦΗΡΜΕΝΩΝ ΜΗΝΥΜΑΤΩΝ ΠΟΥ ΠΕΡΙΛΑΜΒΑΝΟΥΝ.ΚΑΘΕ ΛΕΙΤΟΥΡΓΙΑ ΑΝΑΦΕΡΕΤΑΙ ΣΕ ΈΝΑ ΜΥΝΗΜΑ ΕΙΣΟΔΟΥ ΚΑΙ ΈΝΑ ΜΥΝΗΜΑ ΕΞΟΔΟΥ

Εικόνα 28 : Εκτέλεση μέρους επιχειρησιακής διαδικασίας

Στην εικόνα 28 παρουσιάζεται η εκτέλεση ενός τμήματος της επιχειρησιακής διαδικασίας με τη συμμετοχή όλων των παραπάνω στοιχείων. Ποιο συγκεκριμένα η διαδικασία εκτελείται ως εξής : Το μήνυμα φτάνει στην είσοδο του κάθε partner και εναποτίθεται στην αντίστοιχη μεταβλητή του. Έπειτα ο κάθε partner παράγει το αντίστοιχο μήνυμα εξόδου το οποίο τοποθετείται στην αντίστοιχη μεταβλητή μηνύματος της εξόδου ώστε αυτό να προχωρήσει στον επόμενο partner. Η συγκεκριμένη τοπική εργασία η bed allocate είναι σχετικά μια απλή εργασία με μια είσοδο και μία έξοδο αλλά γενικότερα δεν ισχύει κάτι τέτοιο στις επιχειρησιακές διαδικασίες συνήθως. Για το λόγο αυτό υπάρχουν τα διάφορα activities που λένε στον κάθε partner τι θα δεχτεί σαν είσοδο και ποια θα είναι η έξοδος απόκρισης. Στο συγκεκριμένο παράδειγμα αυτό κάνει η **invoke bed allocate**. Έτσι ολοκληρώνεται η διαδικασία.

Σύνοψη και συμπεράσματα

Στόχος της συγκεκριμένης εργασίας σε πρώτο επίπεδο ήταν να κατανοήσουμε τις βασικές έννοιες των web services καθώς και να μάθουμε ορισμένα πράγματα σχετικά με τις διαδικτυακές υπηρεσίες όπως η αρχιτεκτονική τους καθώς και τα πρότυπα που τις υλοποιούν. Για το λόγο αυτό μελετήθηκαν τα πρωτόκολλα SOAP, η WSDL, το UDDI και η BPEL.

Στη συνέχεια γίνεται μια λεπτομερής περιγραφή της Business Process Execution Language για Web Service (BPEL4WS) και παρουσιάζονται οι βασικές έννοιες της, οι δραστηριότητες που εκτελούνται σε αυτή καθώς και οι μεταβλητές που συμμετέχουν.

Έπειτα παρουσιάστηκε ένα παράδειγμα επιχειρησιακής μοντελοποίησης με bpeI και έγινε προσπάθεια να κατασκευαστεί το συγκεκριμένο παράδειγμα με το εργαλείο Websphere business modeler έτσι ώστε να παρουσιαστούν οι δυνατότητες του συγκεκριμένου εργαλείου της IBM σε σχέση με αντίστοιχα εργαλεία που κυκλοφορούν στην αγορά. Στο παράδειγμα αυτό γίνεται εκτενής αναφορά της διαδικασίας εκτέλεσης που ακολουθείται καθώς και λεπτομερής περιγραφή υλοποίησής του με το Websphere.

Συμπερασματικά θα μπορούσαμε να καταλήξουμε πως πρόκειται για ένα ολοκληρωμένο πακέτο για ανάπτυξη web services που μας δίνει τη δυνατότητα να μοντελοποιήσουμε, να σχεδιάσουμε, και να αναλύσουμε και να αναθεωρήσουμε τις επιχειρησιακές διαδικασίες.

Ο καθορισμός και η μοντελοποίηση μιας επιχειρησιακής διαδικασίας είναι ένας σημαντικός παράγοντας στην βελτιστοποίηση της επιχειρησιακής προετοιμασίας. Μια επιχειρησιακή μοντελοποίηση είναι μια ποικιλία αλληλεπιδράσεων που προκύπτουν μεταξύ των συμμετεχόντων ενός οργανισμού και είναι ιδιαίτερα πολύπλοκες. Χωρίς μια καλή δομή και ένα καλό σύστημα διαχείρισης αυτές οι διαδικασίες μπορεί να κωλυσιεργήσουν την επιχείρηση. Μια καλοδομημένη επιχειρησιακή μοντελοποίηση μας βοηθά να εντοπίσουμε και να μειώσουμε αυτά τα κόστη. Ο Websphere business modeler μας δίνει αυτή τη δυνατότητα και μας βοηθά επίσης να δημιουργήσουμε ρεαλιστικά μοντέλα. Επίσης μας δίνει τη δυνατότητα να προσομοιώνουμε και να αναλύουμε σενάρια υπό συγκεκριμένες συνθήκες που θέλουμε να μελετήσουμε ώστε να καταλήξουμε σε συμπεράσματα που μας ενδιαφέρουν.

Ένα βασικό μειονέκτημα του Websphere business modeler όπως και όλων των συναφών του εργαλείων που κυκλοφορούν στην αγορά είναι η αδυναμία του να μοντελοποιηθεί σε ικανοποιητικό βαθμό ο ανθρώπινος παράγοντας. Στην συγκεκριμένη περίπτωση που εξετάσαμε ως ανθρώπινο παράγοντα αναφέρουμε τον πόρο Doctor που επεμβαίνει στην πορεία της διαδικασίας και είναι υποχρεωτική η παρουσία του στην ολοκλήρωση της. Γενικότερα πρόκειται για ένα σημαντικό μειονέκτημα που ούτε αυτό το εργαλείο κατάφερε να παρακάμψει καθώς ο ανθρώπινος παράγοντας είναι πολύ σημαντικό κομμάτι των επιχειρησιακών διαδικασιών. Μάλλον όμως αυτή η αδυναμία είναι περισσότερο μια αδυναμία της γλώσσας brpel και όχι του συγκεκριμένου εργαλείου.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] <http://www.eclipse.org/>
- [2] UDDI.org UDDI: Universal Description, Discovery and Integration of web services.2002. <http://www.uddi.org/>
- [3] <http://www.webservices.org/index.php/ws/content/view/full/1534>
- [4] <http://www.webservices.org>
- [5] <http://www.ibm.com/developerworks/library/specification/ws-bpel/>
- [6] <http://www.w3.org/TR/wsdl>
- [7] <http://www.ibm.com/developerworks/websphere>
- [8] <http://www.ibm.com/developerworks/websphere/zones/businessintegration/newto/>
- [9] <http://www.ibm.com/developerworks/websphere/services/>
- [10] <http://www.netbeans.org/kb/55/websvc-jax-ws.html>
- [11] <http://msdn.microsoft.com/en-us/library/aa479358.aspx>
- [12] <http://www-01.ibm.com/software/integration/wbimodeler/library/61x.html#Tutorials>
- [13] <http://publib.boulder.ibm.com/infocenter/dmndhelp/v6rxmx/index.jsp>
- [14] <http://www.dit.hua.gr/~mara/e-services.htm>
- [15] <http://bpel.xml.org/>
- [16] BPEL4WS Business Processes with WebSphere Business Integration

Understanding Modeling, Migrating

- [17] ibm.com/redbooks
- [18] Best Practices for Using WebSphere Business Modeler and Monitor
- [19] Patterns: Building Serial and Parallel Processes for IBM WebSphere Process Server V6
- [20] Technical Overview of WebSphere Process Server and WebSphere Integration Developer
- [21] Patterns: SOA Foundation Service Creation Scenario
- [22] Business Process Management: Modeling through Monitoring Using WebSphere V6 Products
- [23] Εκτέλεση Ροών Συνδυασμένων Υπηρεσιών σε BPEL4WS διπλωματική εργασία του ΠΑΝΑΓΙΩΤΗ Δ. ΖΩΓΡΑΦΟΥ
- [24] Web Services architecture overview: The next stage of evolution for e-business, <http://www-106.ibm.com/developerworks/library/w-ovr/?dwzone=ws>
- [25] Simple Object Access Protocol (SOAP) 1.1, W3C Note 08, May 2000, <http://www.w3.org/TR/SOAP/>.
- [26] M. Kloppmann, D. Koenig, F. Leymann, G. Pfau, A. Rickayzen, C. von Riegen, P. Schmidt, and I. Trickovic, WS-BPEL Extension for People (BPEL4-People), IBM Corporation and SAP AG (2005), <http://www.ibm.com/developerworks/webservices/library/specification/ws-bpel4people/>.
- [27] S. A. White: Business Process Modeling Notation. BPMN 1.0. Business Process Modeling Initiative, 2004. <http://www.bpmn.org>
- [28] A. Arkin: Business Process Modeling Language (BPML), Working Draft

0.4,BPMI, March 2001. <http://www.bpmi.org/>

[29] S. Weerawana, F. Curbera: Business Process with BPEL4WS. IBM white paper (2002).

[30] Tidwell, D. (2000) 'Web Services . The Web.s Next Revolution', <http://www-106.ibm.com/developerworks/webservices>

[31] UDDI Consortium: UDDI Executive White Paper, Nov 2001. http://uddi.org/pubs/UDDI-Executive_White_Paper.pdf.

[32] <http://www.daml.org/services/owl-s/>

[33] <http://ifr.sap.com/bpel4ws/>

[34] <http://www.siebel.com/bpel>

[35] <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnbiz2k2/html/bpel1-1.asp>

[36] <http://dev2dev.bea.com/technologies/webservices/BPEL4WS.jsp>