



Επιχειρησιακές Διαδικασίες και Υπηρεσιοστρεφής Αρχιτεκτονική

Σχεδιασμός και Ολοκλήρωση (Integration) Επιχειρησιακών Διαδικασιών

▶ Βασικές Απαιτήσεις

- ▶ **Διαμοιρασμός:** πολλοί χρήστες χρησιμοποιούν τους ίδιους πόρους για τον ίδιο σκοπό
- ▶ **Επαναχρησιμοποίηση:** πολλοί χρήστες χρησιμοποιούν τους ίδιους πόρους για διαφορετικούς σκοπούς

Διαμοιρασμός και Επαναχρησιμοποίηση

- ▶ Αρχιτεκτονικές επιλογές για την επίτευξή τους
 - ▶ Μεμονωμένες εφαρμογές (ERP*, CRM**, ...)
 - ▶ Πολλαπλές εφαρμογές με κοινόχρηστη βάση δεδομένων
 - ▶ Κοινόχρηστα συστατικά λογισμικού (components), κοινά APIs
 - ▶ Επικοινωνία εφαρμογών μέσω μηνυμάτων
 - ▶ **Υπηρεσιοστρεφείς Αρχιτεκτονικές**

* Enterprise Resource Planning – Συστήματα Ενδοεπιχειρησιακού Σχεδιασμού

** Customer Relationship Management – Συστήματα Διαχείρισης Πελατειακών Σχέσεων

Σύστημα Ενδοεπιχειρησιακού Σχεδιασμού (ERP)

Χαρακτηριστικά

- ▶ Λειτουργεί σε πραγματικό χρόνο (ή κοντά σε πραγματικό χρόνο)
- ▶ Υπάρχει μια κοινή βάση δεδομένων που υποστηρίζει όλες τις εφαρμογές.
- ▶ Παρόμοια εμφάνιση και αίσθηση σε κάθε ενότητα.
- ▶ Διευκολύνει τη ροή πληροφοριών μεταξύ όλων των επιχειρησιακών λειτουργιών και τις συνδέσεις προς τα έξω με τα ενδιαφερόμενα μέρη.
- ▶ Εύκολη εγκατάσταση

Σύστημα Διαχείρισης Πελατών (CRM)

- ▶ Πελατοκεντρικό εργαλείο
- ▶ Ο όρος CRM εμφανίσθηκε στις αρχές της δεκαετίας του '90
- ▶ Στρατηγική για τη διαχείριση των αλληλεπιδράσεων μιας επιχείρησης με πελάτες, και προοπτικές πωλήσεων.
- ▶ Περιγράφει τις πρακτικές και το λογισμικό μέσω του οποίου μία επιχείρηση κατανοεί και εξυπηρετεί καλύτερα τις ανάγκες των πελατών της.
- ▶ Περιλαμβάνει χρήση τεχνολογίας για να
 - ▶ οργανώσει,
 - ▶ αυτοματοποιήσει και
 - ▶ συγχρονίσει διεργασίες – κυρίως πωλήσεις, μάρκετινγκ, εξυπηρέτηση πελατών, αποθήκες και τεχνική υποστήριξη.

Υπηρεσιοστρέφεια

- Μια πρόσφατη τάση στην ανάπτυξη κατακευματισμένων εφαρμογών και επιχειρησιακών διαδικασιών
- Χρησιμοποιεί την έννοια της **υπηρεσίας** ως δομικό στοιχείο μιας εφαρμογής/επιχειρησιακής διαδικασίας
- Βασίζεται στη χρήση κοινά αποδεκτών προτύπων στο διαδίκτυο (HTTP, XML)
- Υπόσχεται **διαλειτουργικότητα** στο επίπεδο της εφαρμογής μεταξύ επιχειρήσεων και οργανισμών

Τί είναι μια υπηρεσία;

- ▶ Μινιμαλιστικός ορισμός [Vogels, 2003]
 - ▶ Μια *υπηρεσία* είναι λογισμικό που **επεξεργάζεται έγγραφα XML**, τα οποία λαμβάνει μέσω κάποιου συνδυασμού πρωτοκόλλων επιπέδου επικοινωνίας και επιπέδου εφαρμογής
 - ▶ Η σύνταξη των εγγράφων XML που ανταλλάσσονται από μια υπηρεσία ορίζεται στο **έγγραφο περιγραφής** της υπηρεσίας, το οποίο είναι προσβάσιμο από τον χρήστη
 - ▶ Ένας χρήστης αποκτά πρόσβαση σε μια υπηρεσία μέσω μιας **διεύθυνσης**, η οποία συνδέει το πρωτόκολλο επικοινωνίας με το πρωτόκολλο εφαρμογής και τα συσχετίζει με μια δικτυακή διεύθυνση

Συνοπτικός και Περιεκτικός ορισμός

“Οι υπηρεσίες ιστού είναι αυτόνομες, αρθρωτές (modular) εφαρμογές, που μπορούν να δημοσιευθούν, να εντοπισθούν και να κληθούν μέσω του Διαδικτύου, με τη χρήση κοινώς αποδεκτών προτύπων”

...όπως XML, HTTP, WSDL, UDDI και SOAP

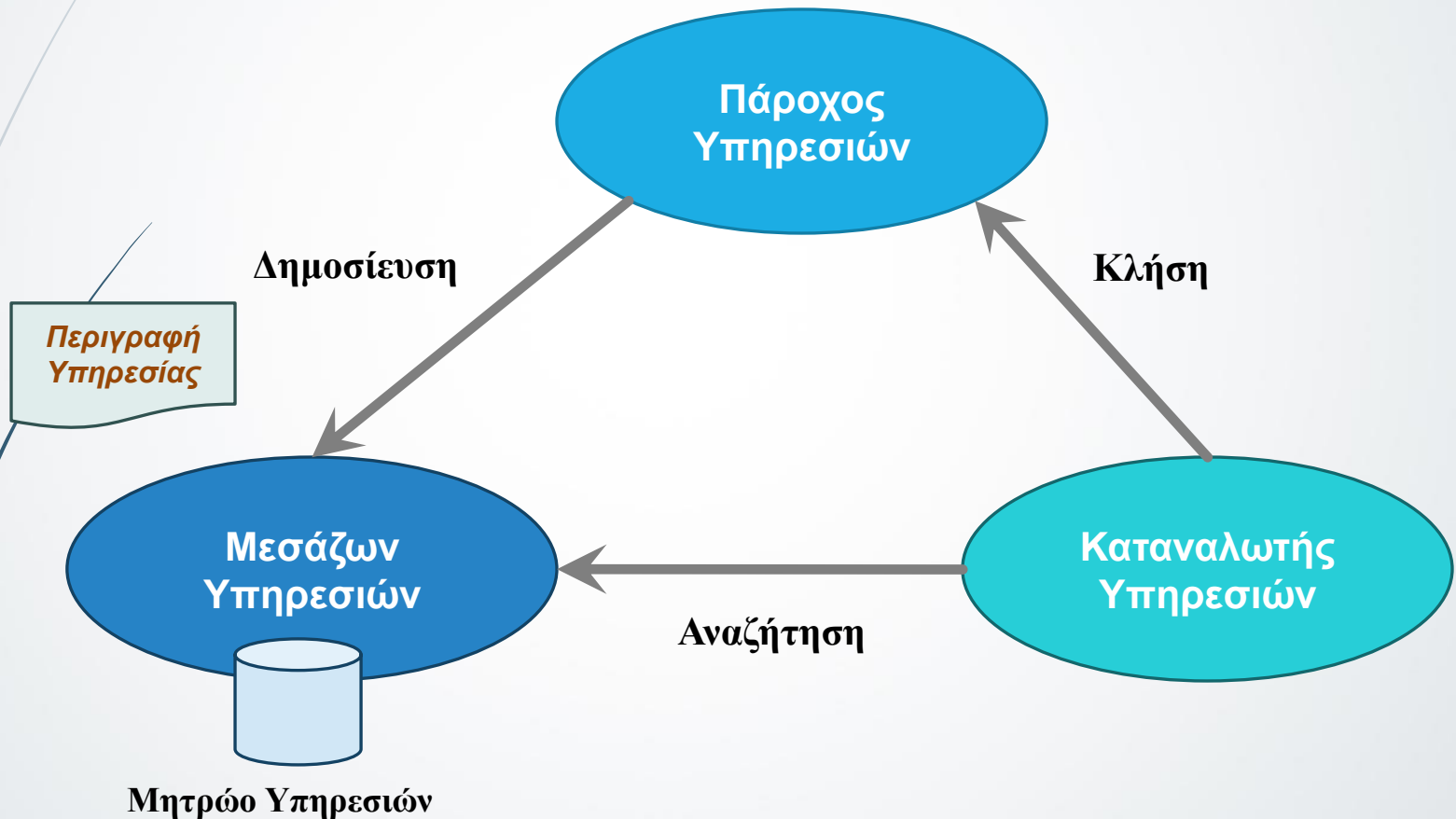
Παραδείγματα Υπηρεσιών

- ▶ **Υπηρεσίες γενικής χρήσης**
 - ▶ Ανάκτηση ημερήσιων κρουσμάτων κορονοϊού
 - ▶ Λήψη μετεωρολογικής πρόγνωσης
 - ▶ Μετατροπή νομισματικών μονάδων
 - ▶ Μετάφραση κειμένου
 - ▶ Google search

- ▶ **Υπηρεσίες ειδικής χρήσης**
 - ▶ Υποβολή αίτησης για λήψη καταναλωτικού δανείου
 - ▶ Λήψη ιατρικού ιστορικού ασθενούς
 - ▶ Έλεγχος διαθεσιμότητας ασθενοφόρου
 - ▶ Κράτηση αεροπορικού εισιτηρίου

Υπηρεσιοστρεφές Μοντέλο

Βασικοί ρόλοι και λειτουργίες σε μια υπηρεσιοστρεφή αρχιτεκτονική



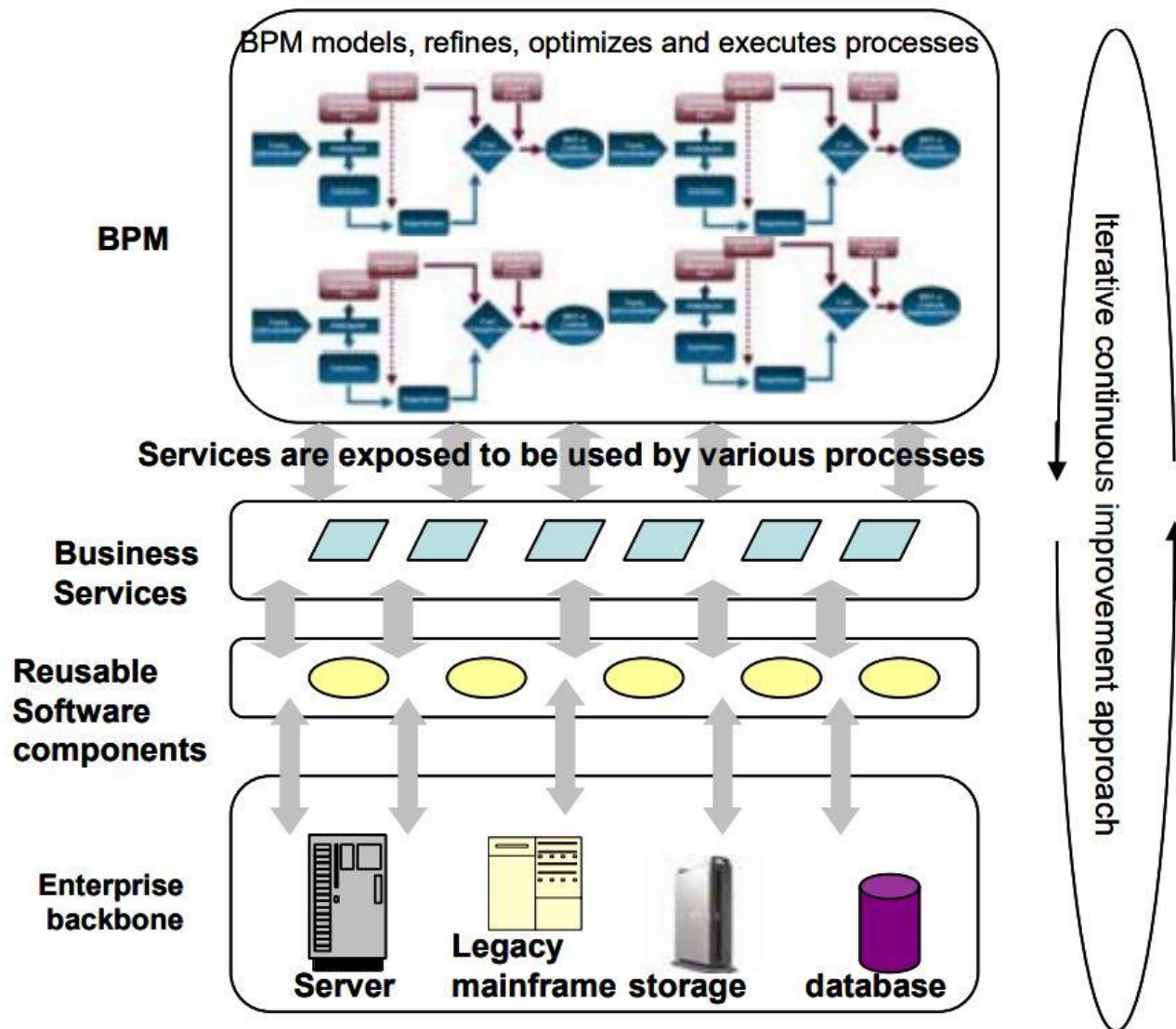
Πλεονέκτημα Υπηρεσιών Ιστού

- ▶ Αυτό που κυρίως διαφοροποιεί τις Υπηρεσίες Ιστού από αντίστοιχες τεχνολογίες είναι η **χρήση κοινά αποδεκτών προτύπων**
 - ▶ Χρήση **HTTP** για τη μεταφορά δεδομένων
 - ▶ firewall-friendly
 - ▶ Χρήση **XML** για την περιγραφή δεδομένων
 - ▶ γενικού σκοπού
 - ▶ πιο κατανοητή (βέβαια, τίθεται το θέμα της απόδοσης)
 - ▶ Χρήση της έννοιας του **URL** για την ταυτοποίηση των υπηρεσιών ιστού
 - ▶ ευρέως διαδεδομένη

BPM vs. SOA

BPM	SOA
Business-driven	IT-driven
Top-down process approach	Bottom-up architectural approach
Reuses process model	Reuses service implementation
Project-oriented	Enterprise infrastructure-oriented
Success measured by business metrics and Key Performance Indicators (KPIs)	Success measured by architectural metrics, logical consistency, ease of integration, and cost of change.

Ενοποιημένη Αρχιτεκτονική για BPM και SOA



Υπηρεσιοστρέφεια στις Επιχειρησιακές Διαδικασίες - 1

- ▶ Σχεδιασμός και ανάπτυξη επιχειρησιακών διαδικασιών ως **συνθέσεις** υπηρεσιών
 - ▶ Προσδιορισμός *ρόλων (roles)*
 - ▶ Αναγνώριση *βημάτων (tasks)* και απόδοσή τους στους ρόλους
 - ▶ Προσδιορισμός *συμμετεχόντων (participants)* και ανάθεση *ρόλων*
 - ▶ Υλοποίηση βημάτων από διαθέσιμες, **διαμοιραζόμενες υπηρεσίες** των συμμετεχόντων

Υπηρεσιοστρέφεια στις Επιχειρησιακές Διαδικασίες - 2

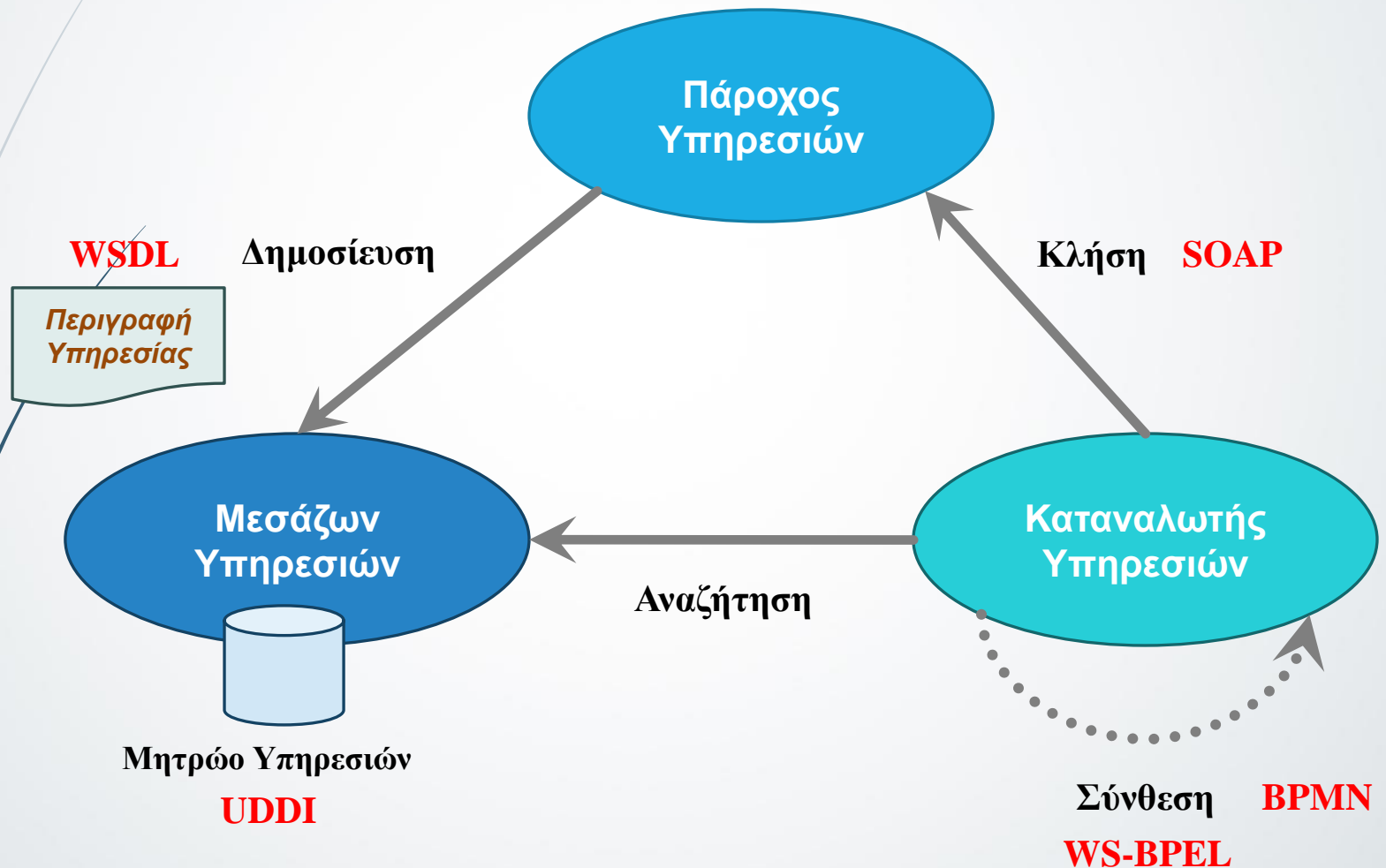
- ▶ Πλεονεκτήματα
 - ▶ Εστίαση στο επιχειρησιακό επίπεδο παρά στο τεχνολογικό
 - ▶ Αρθρωτός, ευέλικτος σχεδιασμός
 - ▶ Εστίαση στη σχεδίαση σε **υψηλό επίπεδο αφαίρεσης**, μακριά από τεχνικές λεπτομέρειες χαμηλού επιπέδου
 - ▶ **Διαλειτουργικότητα** μεταξύ διαφορετικών τεχνολογικών υποδομών
 - ▶ Ευκολότερη συντήρηση και διαχείριση μέσω **επαναχρησιμοποίησης**

Υπηρεσιοστρεφείς Τεχνολογίες

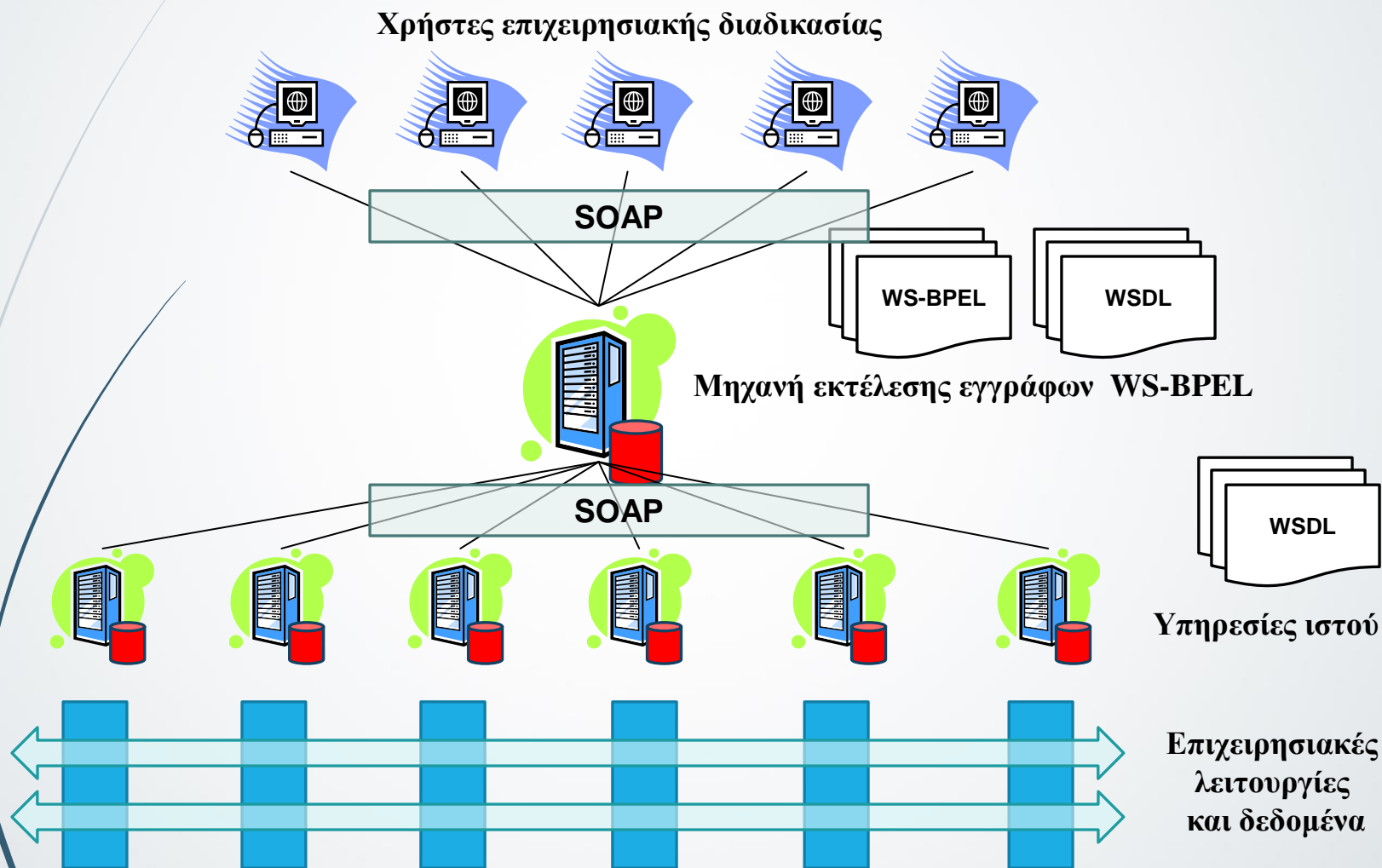
- ▶ Ανάπτυξη υπηρεσιών
 - ▶ Υπηρεσίες ιστού (Web services)
 - ▶ Περιγραφή: **WSDL** (Web Services Description Language)
 - ▶ Δημοσίευση & αναζήτηση: **UDDI** (Universal Description, Discovery and Integration)
 - ▶ Κλήση: **SOAP** (Simple Object Access Protocol)
 - ▶ Σύνθεση υπηρεσιών
 - ▶ Ενορχήστρωση υπηρεσιών ιστού
 - ▶ Σχεδιασμός: **BPMN** (Business Process Modeling Notation)
 - ▶ Εκτέλεση: **WS-BPEL** (Web Services Business Process Execution Language)

Υπηρεσιοστρεφές Μοντέλο

Τεχνολογίες και πρόσθετες λειτουργίες υπηρεσιοστρεφών αρχιτεκτονικών



Ανάπτυξη Υπηρεσιοστρεφών Επιχειρησιακών Διαδικασιών



Κύκλος Ζωής Υπηρεσιών Ιστού

**Basic
Layer**

Create

Describe

Publish

Discover

Invoke

Unpublish

**Value-
Added
Layer**

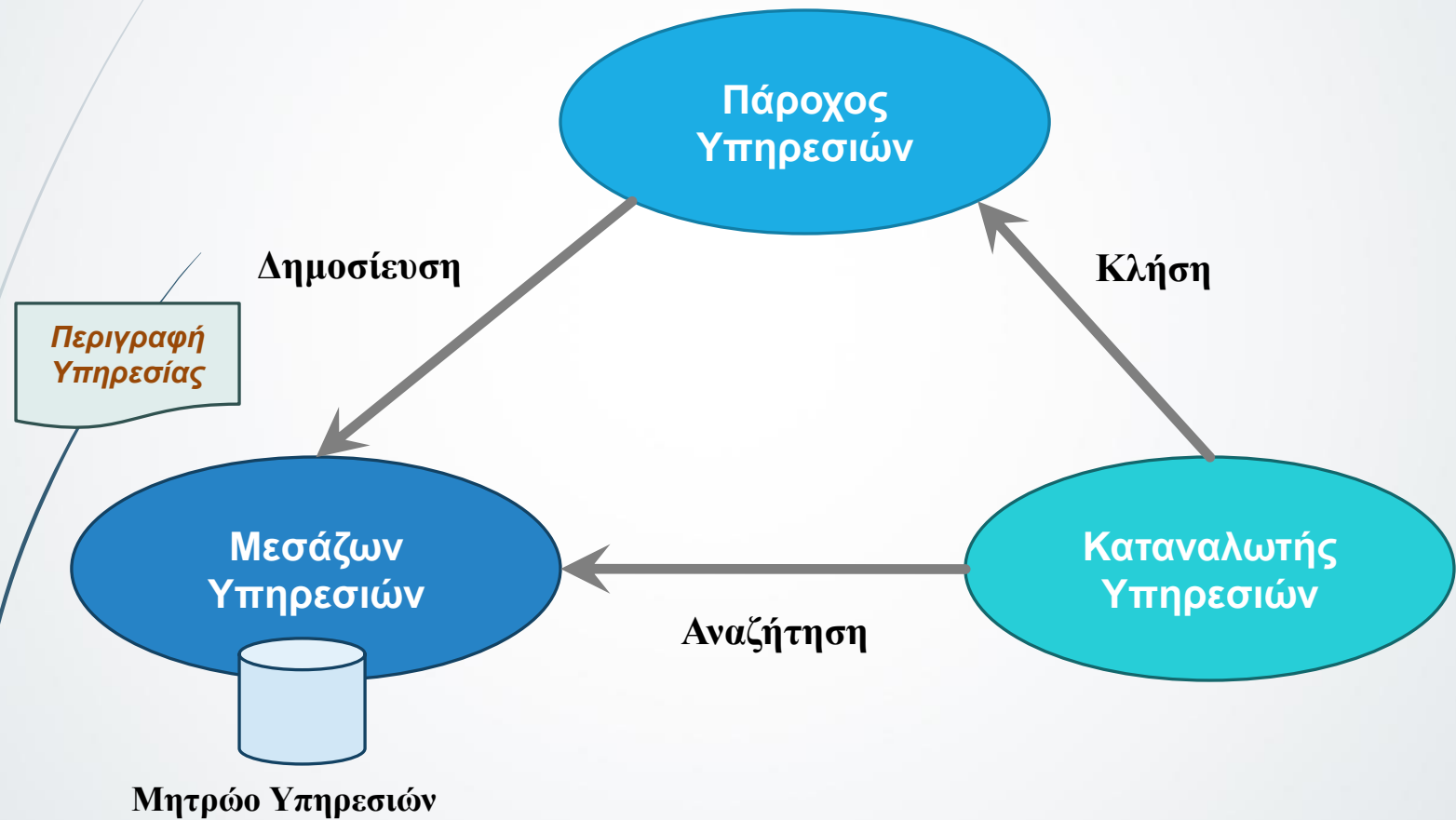
**Composition
Transaction
Reliability**

**Security
Handling
Monitoring**

**Brokering
Billing
Contracting**

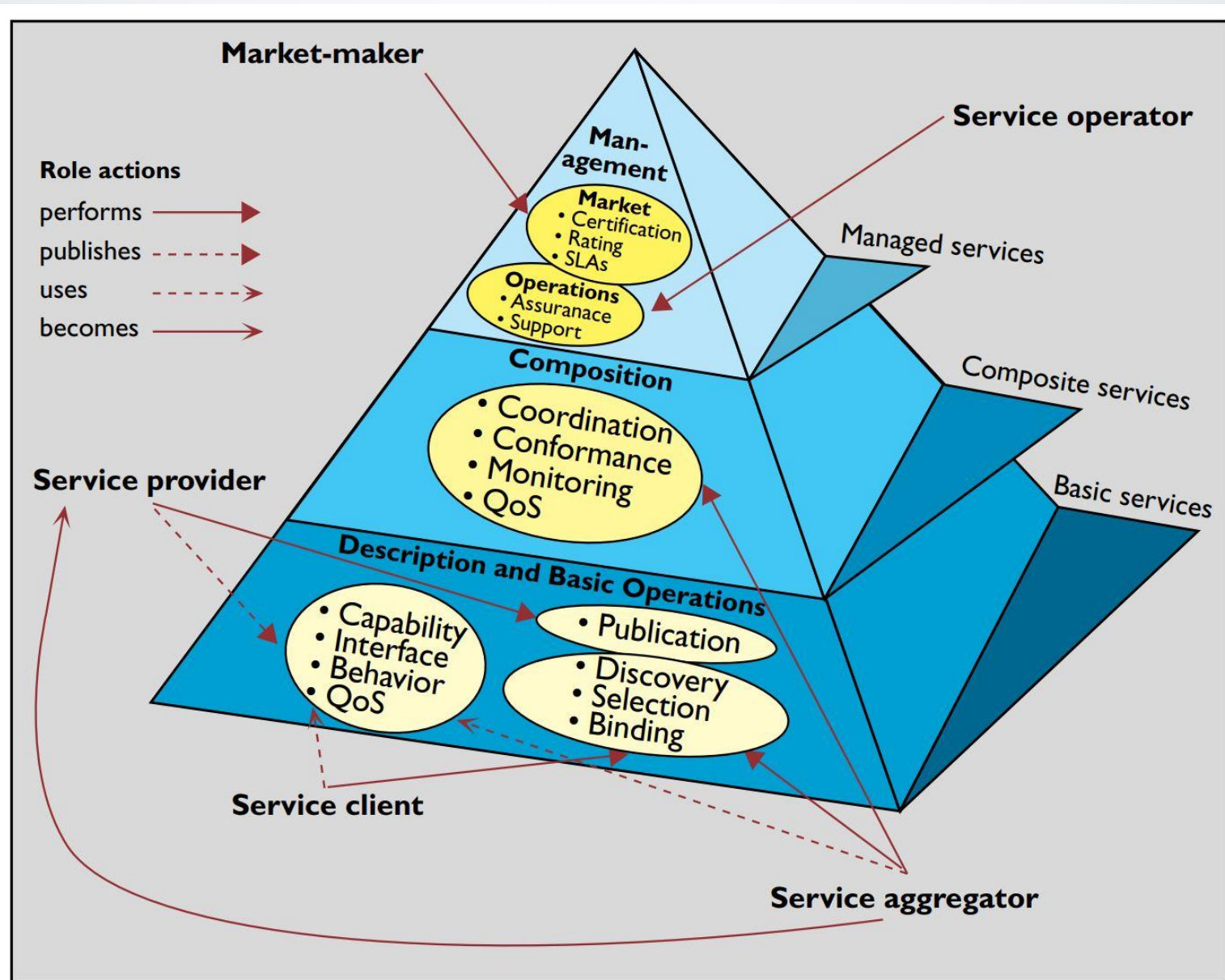
Υπηρεσιοστρεφές Μοντέλο

Βασικοί ρόλοι και λειτουργίες σε μια υπηρεσιοστρεφή αρχιτεκτονική



Εκτεταμένη Υπηρεσιοστρεφής Αρχιτεκτονική

Επίπεδα Υπηρεσιών, Λειτουργικότητα, Ρόλοι





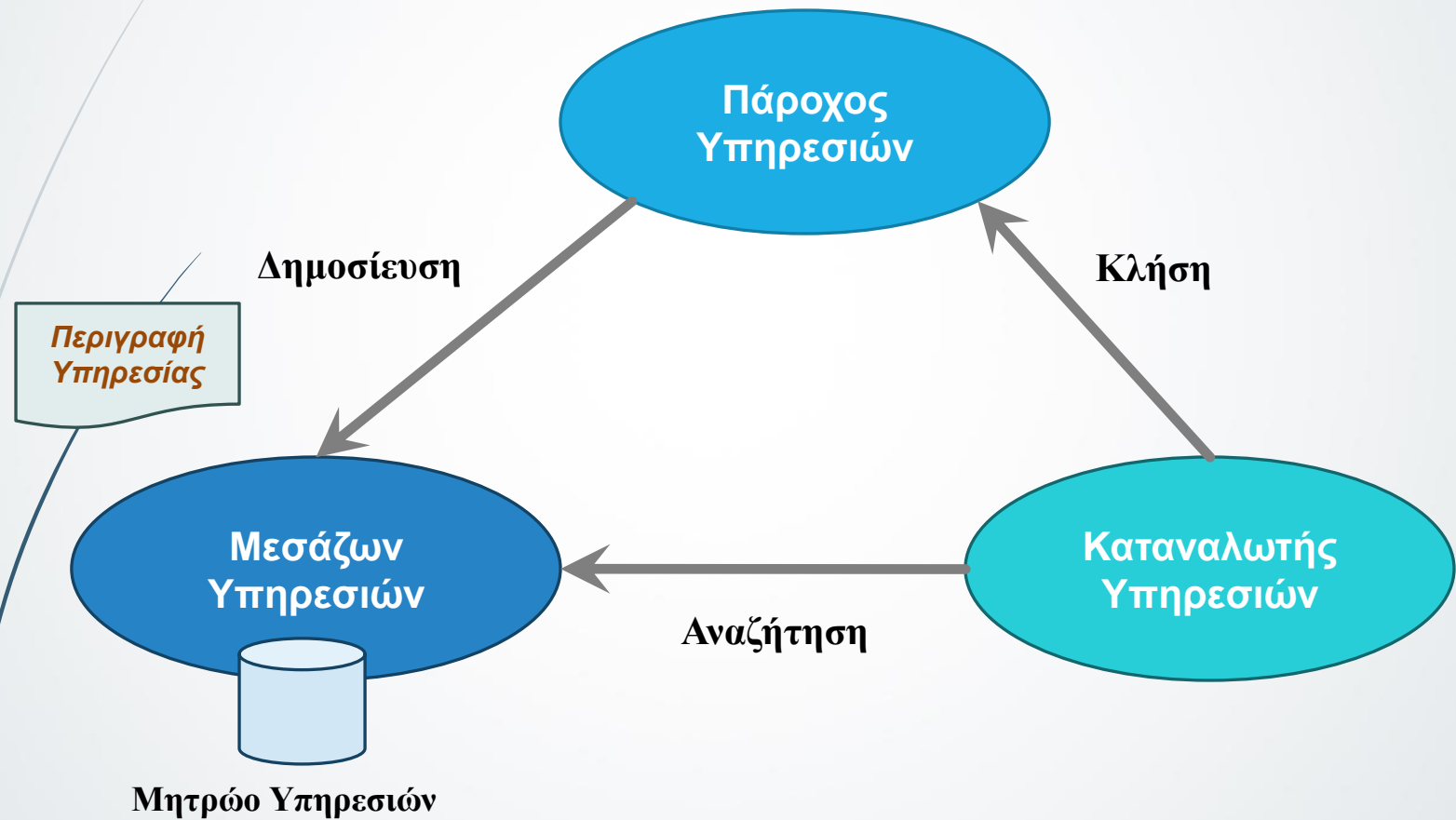
ΠΡΩΤΟΚΟΛΛΑ ΥΠΗΡΕΣΙΩΝ ΙΣΤΟΥ

Γιατί χρειαζόμαστε πρωτόκολλα;

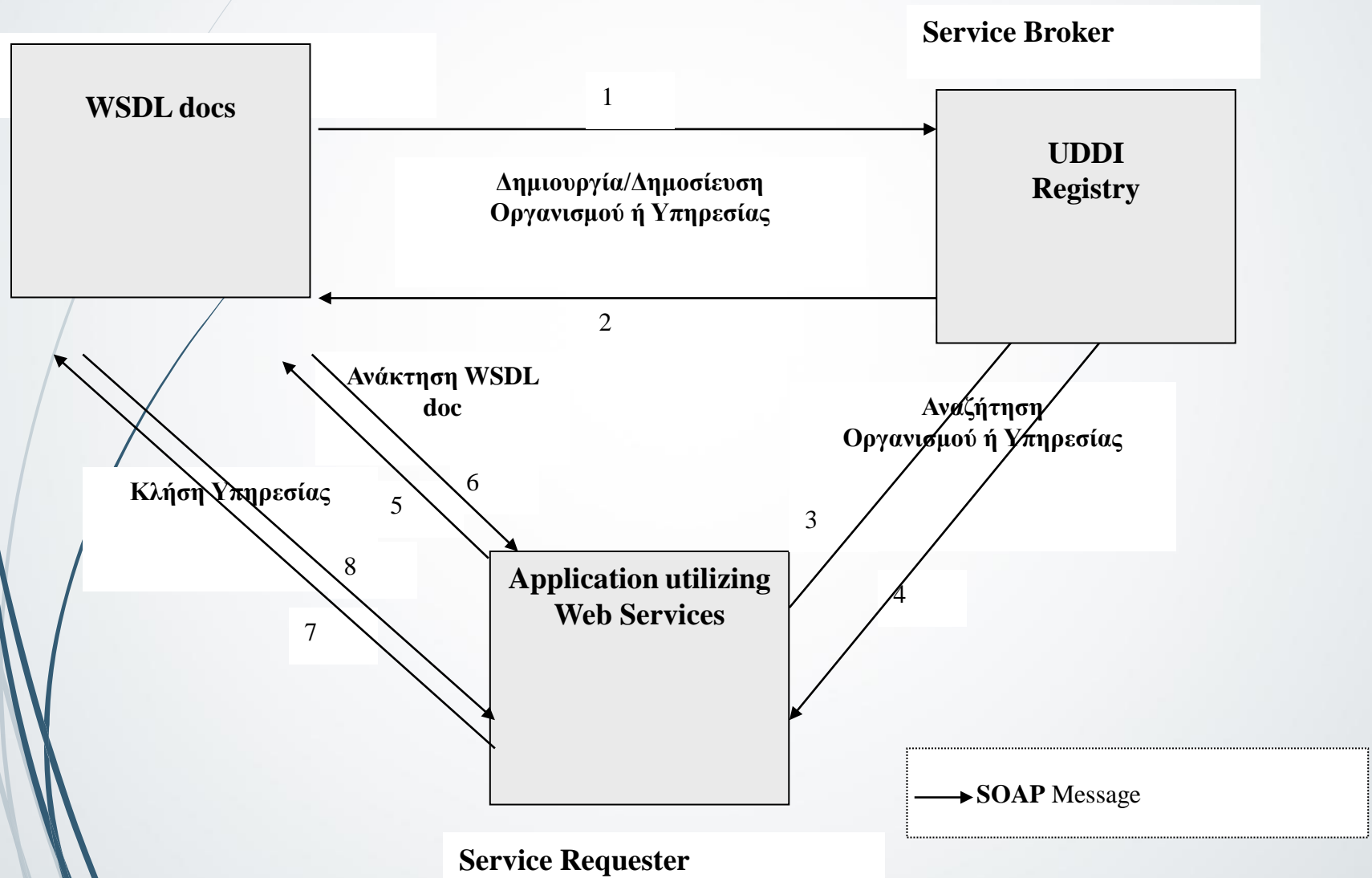
- ▶ Για να ξέρουμε
 - ▶ τι μεταδίδεται «πάνω στο καλώδιο» και
 - ▶ τι περιγράφει αυτό που μεταδίδεται «πάνω στο καλώδιο»
- ▶ Για να βρίσκουμε τις περιγραφές υπηρεσιών
- ▶ Για να μπορούμε να χρησιμοποιούμε τις περιγραφές των υπηρεσιών για
 - ▶ να τις καλέσουμε (invoke) και
 - ▶ να τις συνθέσουμε (compose)

Υπηρεσιοστρεφές Μοντέλο

Βασικοί ρόλοι και λειτουργίες σε μια υπηρεσιοστρεφή αρχιτεκτονική



WSDL, SOAP, UDDI: Αλληλεπιδράσεις



Πρωτόκολλα για Υπηρεσίες Ιστού

- WSDL - Web Service Description Language
- SOAP - Simple Object Access Protocol
- UDDI – Universal Description, Discovery, and Integration



WSDL

Web Services Description Language

Τι είναι η WSDL (1/3)

- ▶ Πρωτόκολλο περιγραφής Υπηρεσιών Ιστού
- ▶ Οι περιγραφές είναι XML documents που περιγράφουν τα interfaces των υπηρεσιών.
Συγκεκριμένα περιγράφουν
 - ▶ Τι κάνει η υπηρεσία, δηλ. τις **λειτουργίες** που προσφέρει
 - ▶ τη **θέση** που είναι διαθέσιμη η υπηρεσία
 - ▶ τους **μηχανισμούς πρόσβασης** - πως μπορεί να κληθεί

Τι είναι η WSDL (2/3)

- ▶ Το έγγραφο WSDL
 - ▶ Ορίζει μια υπηρεσία σαν ένα σύνολο από network end points ή ports. Ο αφαιρετικός ορισμός των endpoints και την μηνυμάτων διαχωρίζεται από το concrete network deployment ή τα data format bindings.
 - ▶ περιλαμβάνει μια συλλογή από διάφορους τρόπους αλληλεπίδρασης (interaction paradigms), με τη δυνατότητα να συνδυάζονται λειτουργίες ή ομάδες λειτουργιών σε ένα interface
- ▶ Ο διαχωρισμός των interfaces, bindings and protocols δημιουργούν την ανάγκη για σπονδυλωτές προδιαγραφές

Τι είναι η WSDL (3/3)

- ▶ Μπορεί να έχουμε διαφορετικές υπηρεσίες που να υλοποιούν το ίδιο *interface* αλλά
 - ▶ να προσφέρουν την υπηρεσία σε διαφορετικές διευθύνσεις και
 - ▶ να αλληλεπιδρούν με διαφορετικά πρωτόκολλα
- ▶ Η επικοινωνία μεταξύ πελάτη και υπηρεσίας είναι ασύγχρονη

Δομή ενός εγγράφου WSDL

- ▶ **Abstract** περιγραφή -> περιγράφει τα interfaces αντίστοιχα με τα παραδοσιακά middleware IDL
 - ▶ Περιέχει
 - ▶ μηνύματα, που είναι αφαιρετικές περιγραφές των δεδομένων που ανταλλάσσονται και
 - ▶ Port types, που είναι αφαιρετικές συλλογές μηνυμάτων
- ▶ **Concrete** περιγραφή -> protocol bindings και άλλες πληροφορίες
 - ▶ Ένα reusable port binding είναι η concrete περιγραφή των προδιαγραφών του πρωτοκόλλου και των προδιαγραφών των data formats για ένα συγκεκριμένο τύπο port
 - ▶ Ένα port ορίζεται με τη σύνδεση μιας network address μ' ένα reusable binding
 - ▶ Ένα service ορίζεται ως μια συλλογή από ports

Δομή ενός εγγράφου WSDL

Abstract περιγραφή

➤ **Abstract** περιγραφή

➤ Ορισμούς των **port types**

- Κάθε port type είναι μια λογική ομαδοποίηση λειτουργιών
- Κάθε λειτουργία περιγράφει απλή ανταλλαγή μηνυμάτων
 - Ένα μήνυμα είναι μια μονάδα επικοινωνίας με μια υπηρεσία που αναπαριστά τα δεδομένα που ανταλλάσσονται σε μια μοναδική λογική συναλλαγή (single logical transaction)

Δομή ενός εγγράφου WSDL

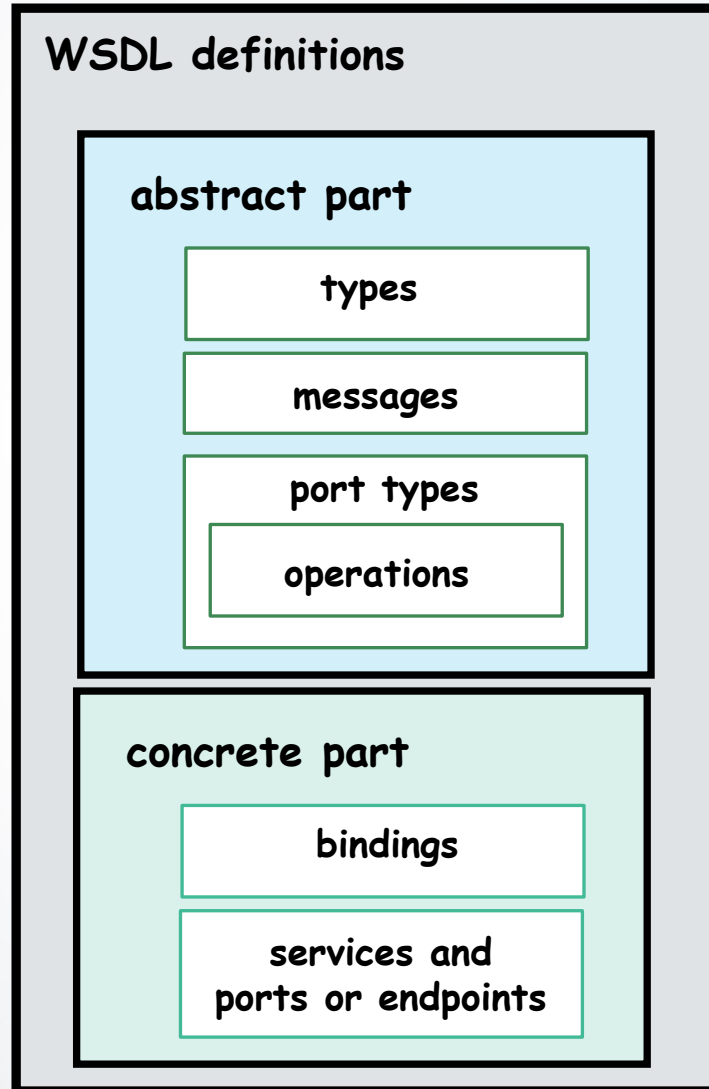
Abstract περιγραφή – Port Types

- ▶ Τα **port types** είναι το πιο σημαντικό στοιχείο στο WSDL. Περιγράφει
 - ▶ ένα web service,
 - ▶ τις λειτουργίες που προσφέρει και
 - ▶ τα μηνύματα που ανταλλάσσονται.
- ▶ Μπορεί να συγκριθεί με μια function library (ή μια κλάση ή ένα module) σε μια γλώσσα προγραμματισμού ή με τα interfaces στις παραδοσιακές middleware IDLs

Δομή ενός εγγράφου WSDL

Σχηματικά

34



Δομή ενός εγγράφου WSDL

Βήματα ορισμού της abstract περιγραφής (1/2)

- 1. Ορισμός του συστήματος τύπων δεδομένων** που χρησιμοποιείται για την περιγραφή των μηνυμάτων (βασίζεται στο XML Schema) ώστε και οι δύο πλευρές να μεταφράζουν σωστά τα δεδομένα που ανταλλάσσονται.
 - ▶ Σημειώστε ότι αυτό είναι διαφορετικό από τα IDLs σε conventional middleware, όπου το σύστημα τύπων επιβάλλεται από την πλατφόρμα (π.χ. CORBA ή J2EE)
 - ▶ Τα σχήματα XML έχουν built-in βασικούς τύπους και μπορούν οι χρήστες να ορίζουν πιο σύνθετους τύπους. Επομένως, το πρώτο βήμα στον ορισμό ενός WSDL interface είναι να οριστούν όλοι οι τύποι δεδομένων που θα ανταλλάσσονται ως μέρος των μηνυμάτων μεταξύ των εφαρμογών
- 2. Ορισμός μηνυμάτων** που χρησιμοποιούν αυτούς τους τύπους δεδομένων
 - ▶ Κάθε μήνυμα αποτελείται από μέρη που αφορούν στους τύπους δεδομένων εισόδου και εξόδου

Δομή ενός εγγράφου WSDL

Βήματα ορισμού της abstract περιγραφής (2/2)

3. Ορισμός λειτουργιών ή ορισμός αλληλεπιδράσεων (transmission primitives)

- ▶ 4 βασικές λειτουργίες: one-way, request-response, notification, solicit-response
- ▶ Σύγχρονες αλληλεπιδράσεις -> request-response, solicit-response
- ▶ Ασύγχρονες αλληλεπιδράσεις -> one-way, notification

<http://www.roguewave.com/portals/0/products/hydraexpress/docs/4.3.0/html/rwsfexpwsfabricationug/9-2.html>

4. Ομαδοποίηση λειτουργιών σε port types

- ▶ Τα ports μπορούν να επεκτείνουν άλλα ports

WSDL Παράδειγμα

```
<?xml version="1.0"?>
<definitions name="Procurement"
  targetNamespace=http://example.com/procurement/definitions xmlns:tns="http://example.com/procurement/definitions"
  xmlns:xs=http://www.w3.org/2001/XMLSchema xmlns:soap=http://schemas.xmlsoap.org/wsdl/soap/
  xmlns="http://schemas.xmlsoap.org/wsdl/" >
```

```
<message name="OrderMsg">
  <part name="productName" type="xs:string"/>
  <part name="quantity" type="xs:integer"/>
</message>
```

abstract part

messages

```
<portType name="procurementPortType">
  <operation name="orderGoods">
    <input message = "OrderMsg"/>
  </operation>
</portType>
```

operation and
port type

```
<binding name="ProcurementSoapBinding" type="tns:procurementPortType">
  <soap:binding style="document"
    transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="orderGoods">
    <soap:operation soapAction="http://example.com/orderGoods"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
  </operation>
</binding>
```

concrete
part

binding

```
<service name="ProcurementService">
  <port name="ProcurementPort" binding="tns:ProcurementSoapBinding">
    <soap:address location="http://example.com/procurement"/>
  </port>
</service>
```

port and
service

```
</definitions>
```

Δομή ενός εγγράφου WSDL

Γιατί χρειαζόμαστε την concrete περιγραφή (1/2)

- ▶ Οι προηγούμενοι ορισμοί είναι abstract διότι
 - ▶ δεν περιέχουν κάποιο concrete binding ή encoding ή ορισμό της υπηρεσίας που υλοποιεί κάποια port types
- ▶ Για τον ορισμό μιας πραγματικής υπηρεσίας χρειάζεται να οριστούν
 - ▶ τα συγκεκριμένα port types που υλοποιεί
 - ▶ τα transport bindings που τα υλοποιούν και
 - ▶ οι διευθύνσεις αυτών των υλοποιήσεων

Δομή ενός εγγράφου WSDL

Γιατί χρειαζόμαστε την concrete περιγραφή (2/2)

- ▶ Το ίδιο port type μπορεί
 - ▶ να υλοποιηθεί με πολλά bindings
 - ▶ να συνδυαστεί με άλλες υλοποιήσεις port types και
 - ▶ να σχηματίσει διαφορετικές υπηρεσίες
- ▶ Τα μηνύματα είναι abstract γιατί λείπει
 - ▶ πληροφορία για το XML encoding που έχουν και
 - ▶ το protocol binding.
- ▶ Ο ίδιος τύπος δεδομένων μπορεί να γίνει encoded σ' ένα μήνυμα με διαφορετικούς κανόνες και το ίδιο μήνυμα μπορεί να ανταλλαγεί με διαφορετικά protocol bindings

Δομή ενός εγγράφου WSDL

Concrete περιγραφή

Αποτελείται από 3 μέρη

- ▶ **Interface Bindings:** περιγράφουν τη σύνδεση (binding) της διεπαφής (interface) με το πρωτόκολλο μεταφοράς (transport protocol)
- ▶ **Ports:** ορίζουν το endpoint ή την διεύθυνση δικτύου (network address) μίας σύνδεσης
- ▶ **Services:** ορίζουν την υπηρεσία, ως συλλογή όλων των συνδέσεων (bindings) μιας διεπαφής

Δομή ενός εγγράφου WSDL

Concrete περιγραφή - *Interface Bindings*

► *Interface Bindings*

- περιγράφουν τη σύνδεση (binding) της διεπαφής (interface) με το πρωτόκολλο μεταφοράς (transport protocol) για όλα τα μηνύματα που περιγράφονται σ' ένα port type. Για παράδειγμα, μπορεί να ορίζουν
 - RPC-style operation (οπότε τα input/output είναι οι παράμετροι του procedure call) ή document-style operation (όπου τα input/output μηνύματα μεταφέρουν documents)
 - Πως μεταφέρονται τα μηνύματα: με SOAP protocol και HTTP transport bindings (ή SMTP)
 - Κανόνες κωδικοποίησης (Encoding rules) e.g. Κωδικοποίηση κατά γράμμα (literal encoding) για document style interactions, SOAP encoding για RPC-style interactions

Δομή ενός εγγράφου WSDL

Concrete περιγραφή - Ports or Endpoints

► *Ports ή Endpoints*

Συνδυάζουν την πληροφορία που υπάρχει στο Interface Bindings με μια διεύθυνση δικτύου (με ένα URI) όπου μπορεί κάποιος να έχει πρόσβαση στο port type

- Αυτό δεν χρειάζεται στα παραδοσιακά middleware λόγω της κεντροποιημένης υποδομής που διαχειρίζεται τις διευθύνσεις

Δομή ενός εγγράφου WSDL

Concrete περιγραφή - Services

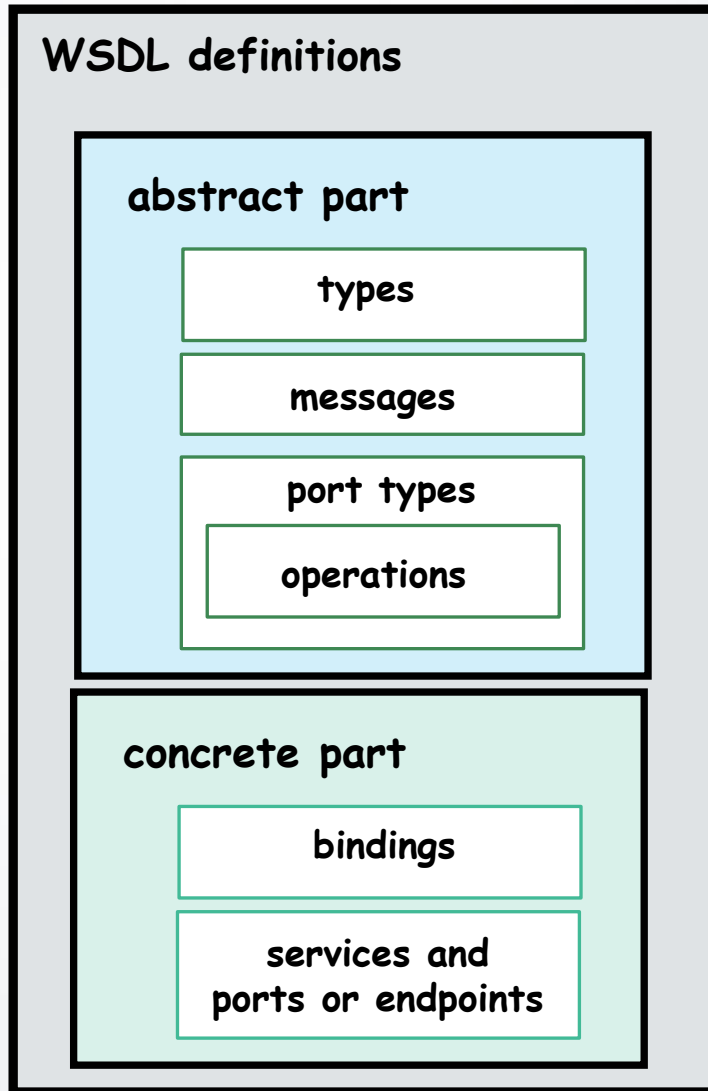
Services

- Λογικές ομαδοποιήσεις των ports
- Θεωρητικά ένα Web service μπορεί να είναι διαθέσιμο σε διαφορετικά URIs, και μπορεί να συνδυάζει τελείως διαφορετικά είδη ports.
- Στην πράξη, συνήθως ένα έγγραφο WSDL ομαδοποιεί σχετικά ports που είναι διαθέσιμα στην ίδια διεύθυνση
- Μια άλλη ομαδοποίηση είναι εκείνη όπου τα διάφορα ports αναπαριστούν διαφορετικά bindings των ίδιων port types. Αυτό επιτρέπει κάποιον να έχει πρόσβαση στην ίδια λειτουργικότητα με διαφορετικά transport protocols και interaction styles

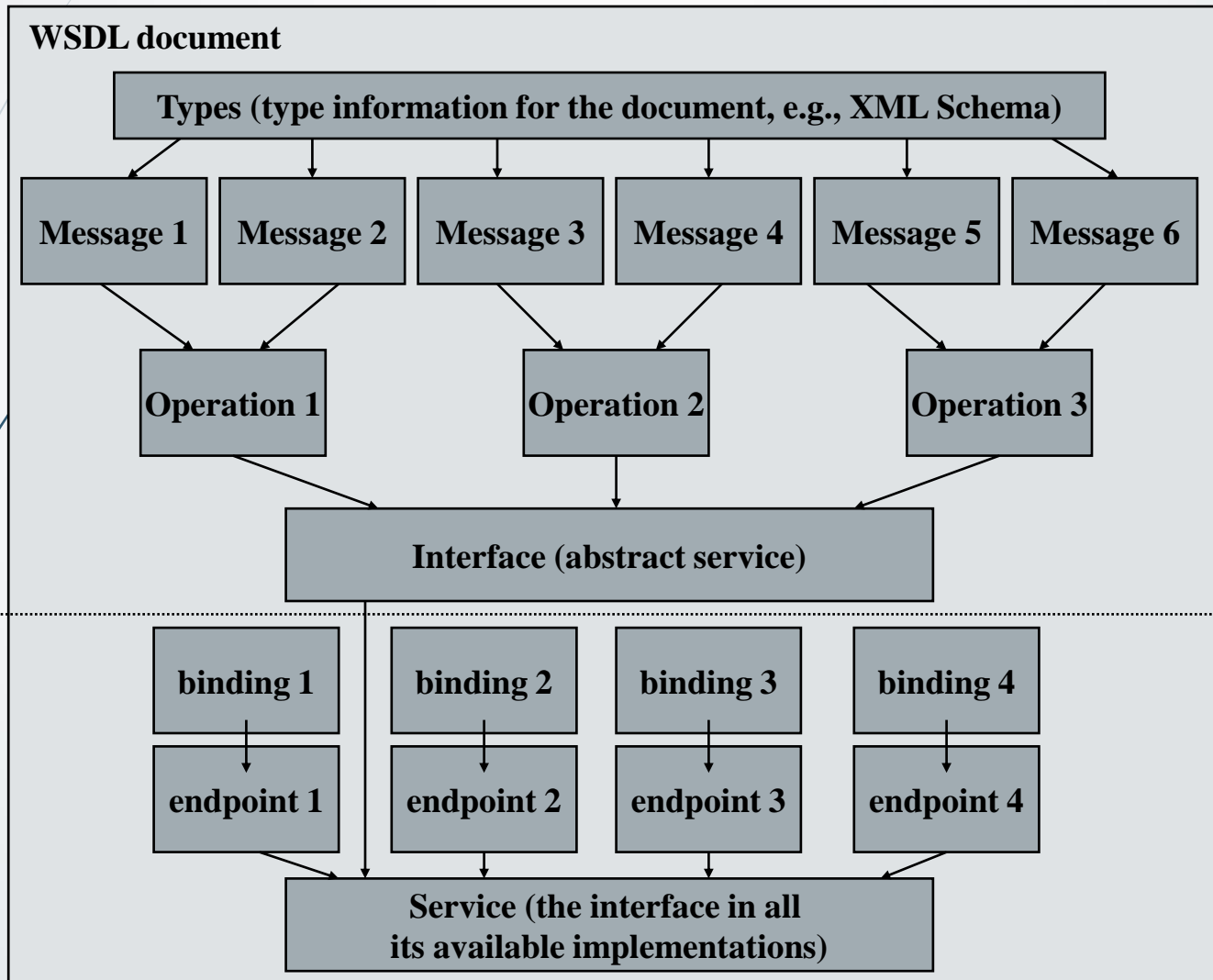
Δομή ενός εγγράφου WSDL

Σχηματικά

44



Στοιχεία ενός εγγράφου WSDL



Abstract description of the service

Concrete description of the service

WSDL Παράδειγμα

```
<?xml version="1.0"?>
<definitions name="Procurement"
  targetNamespace=http://example.com/procurement/definitions xmlns:tns="http://example.com/procurement/definitions"
  xmlns:xs=http://www.w3.org/2001/XMLSchema xmlns:soap=http://schemas.xmlsoap.org/wsdl/soap/
  xmlns="http://schemas.xmlsoap.org/wsdl/" >
```

```
<message name="OrderMsg">
  <part name="productName" type="xs:string"/>
  <part name="quantity" type="xs:integer"/>
</message>
```

abstract part

messages

```
<portType name="procurementPortType">
  <operation name="orderGoods">
    <input message = "OrderMsg"/>
  </operation>
</portType>
```

operation and
port type

```
<binding name="ProcurementSoapBinding" type="tns:procurementPortType">
  <soap:binding style="document"
    transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="orderGoods">
    <soap:operation soapAction="http://example.com/orderGoods"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
  </operation>
</binding>
```

concrete
part

binding

```
<service name="ProcurementService">
  <port name="ProcurementPort" binding="tns:ProcurementSoapBinding">
    <soap:address location="http://example.com/procurement"/>
  </port>
</service>
```

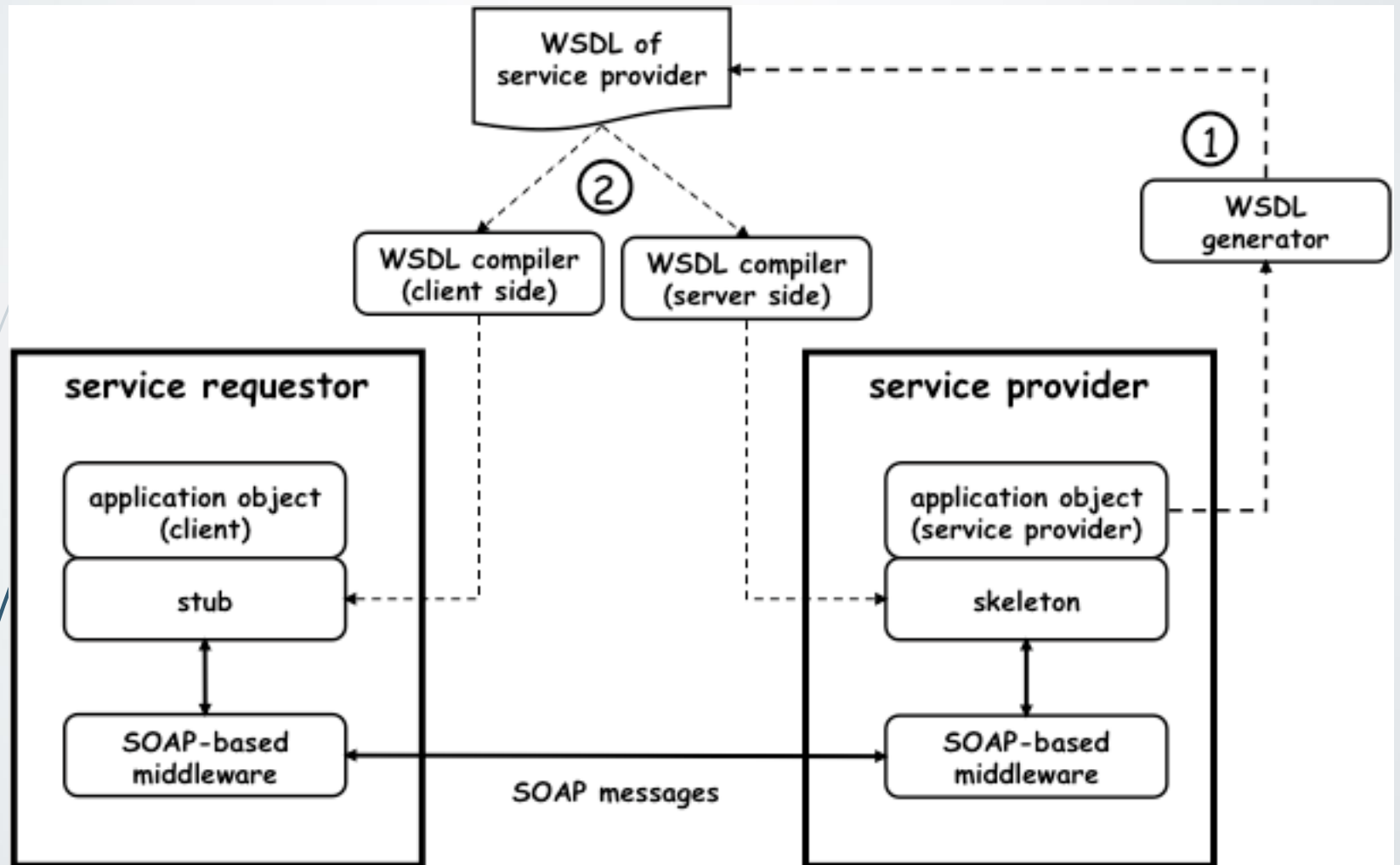
port and
service

```
</definitions>
```

Χρήση ενός εγγράφου WSDL

- ▶ Ως συμβόλαιο που περιγράφει
 - ▶ Πως μπορείς να αλληλεπιδράσεις με την υπηρεσία
 - ▶ Τι δεδομένα χρειάζεται να στείλεις και τι να περιμένεις
 - ▶ Ποιές λειτουργίες εκτελούνται
 - ▶ Το format και τα πρωτόκολλα κλήσης της υπηρεσίας
- ▶ Επιτρέπει σε εργαλεία να παράγουν client και server stubs και τις πληροφορίες που χρειάζονται για την κλήση της υπηρεσίας
- ▶ Επιτρέπει τη δημιουργία σημασιολογίας από τις διάφορες εταιρείες
- ▶ Επιτρέπει δημοσίευση των περιγραφών και επιτρέπει αναζήτηση (σε συνδυασμό με το UDDI) και binding με τις κατάλληλες υπηρεσίες

Χρήση της WSDL



Είναι αρκετή η WSDL;

- ▶ Η WSDL καλύπτει την περιγραφή των:
 - ▶ Μηνυμάτων και των τύπων τους
 - ▶ Υποστηριζόμενων λειτουργιών και των προτύπων συναλλαγών για κάθε λειτουργία
 - ▶ Τελικών URLs και των πρωτοκόλλων που πρέπει να χρησιμοποιηθούν
- Αλλά δεν μπορεί να καλύψει θέματα όπως:
 - Σημασιολογική περιγραφή των υπηρεσιών, λειτουργιών και μηνυμάτων, π.χ. Τι πραγματικά κάνει μια λειτουργία; Σε ποια πραγματική έννοια αντιστοιχεί ένα μήνυμα;
 - Περιγραφή των πιθανών 'συνολικών' συναλλαγών που μπορούμε να έχουμε με μια υπηρεσία, π.χ. με ποια σειρά πρέπει να πραγματοποιηθούν οι κλήσεις των προσφερόμενων λειτουργιών μιας υπηρεσίας, στο πλαίσιο κάποιου σεναρίου
 - Περιγραφή των πολιτικών και των κανόνων που διέπουν τη λειτουργία μιας υπηρεσίας

WSDL On line Tutorial

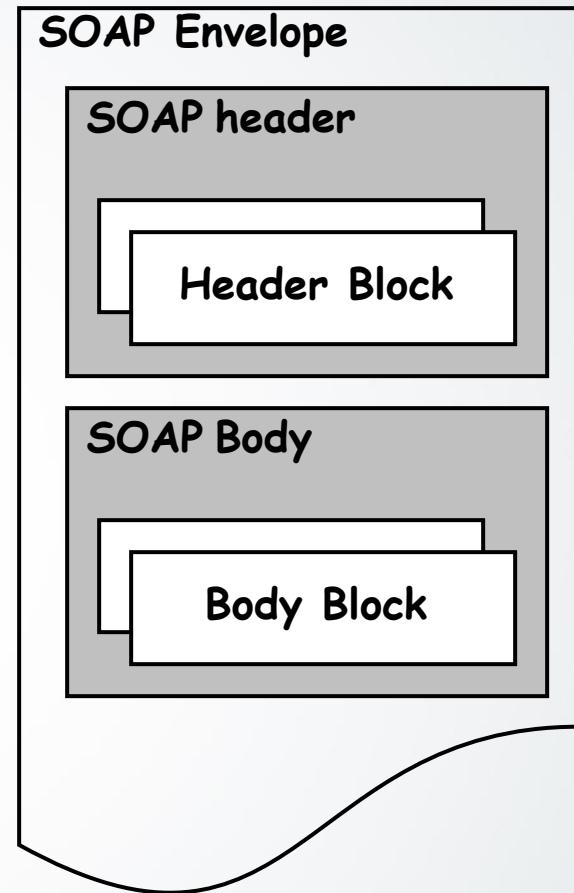
➔ <https://www.tutorialspoint.com/wsdl/>



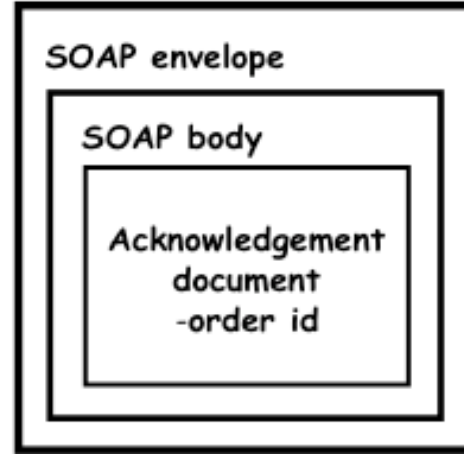
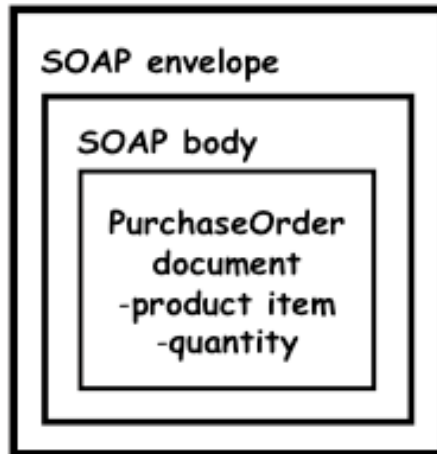
SOAP - Simple Object Access Protocol

SOAP μηνύματα

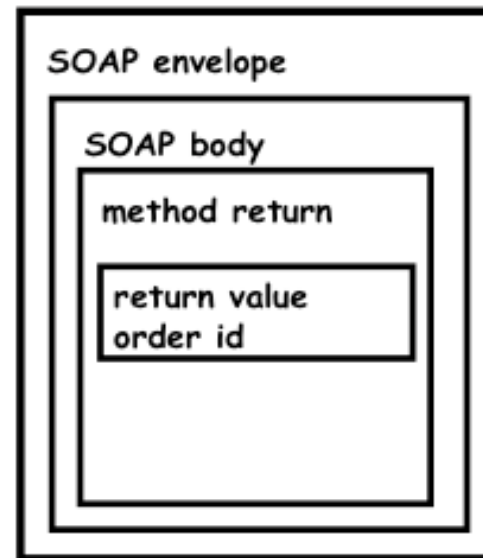
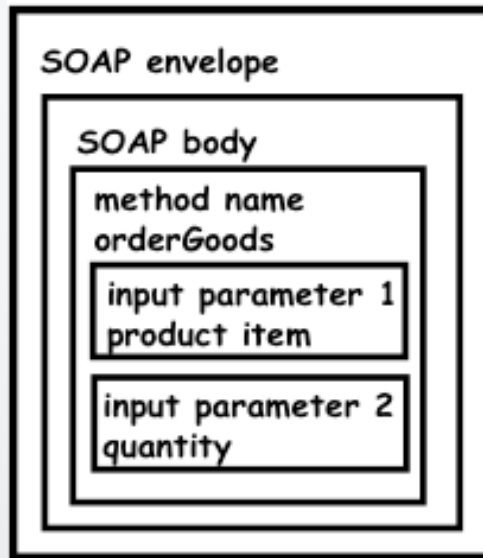
- ▶ Έχουν την μορφή φακέλων στους οποίους οι εφαρμογές εναποθέτουν τα δεδομένα που ανταλλάσσονται
- ▶ Ένα μήνυμα έχει 2 μέρη:
 - ▶ header: το οποίο μπορεί να χωριστεί σε blocks
 - ▶ body: το οποίο μπορεί να χωριστεί σε blocks
- ▶ Το SOAP δεν καθορίζει τι πρέπει να κάνει κάποιος με το header και το body, παρά μόνο ορίζει ότι το header δεν είναι υποχρεωτικό να υπάρχει ενώ το body είναι
- ▶ Η χρήση του header και του body, ωστόσο είναι υπονοούμενη. Το body είναι για τα δεδομένα της εφαρμογής. Το header είναι για δεδομένα της υποδομής



SOAP – Τύποι μηνυμάτων



(a) Document-style interaction



(b) RPC-style interaction

Κωδικοποιήσεις μηνυμάτων

```
<ProductItem>  
  <name>...</name>  
  <type>...</type>  
  <make>...</make>  
</ProductItem>
```

```
<ProductItem  
  name="..."  
  type="..."  
  make="..."/>
```

```
<ProductItem name="...">  
  <type>...</type>  
  <make>...</make>  
</ProductItem>
```

SOAP παράδειγμα

```
<SOAP-ENV:Envelope  
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"  
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding"/>
```

```
<SOAP-ENV:Header>  
  <t:Transaction  
    xmlns:t="some-URI"  
    SOAP-ENV:mustUnderstand="1">  
    5  
  </t:Transaction>  
</SOAP-ENV:Header>
```

```
<SOAP-ENV:Body>  
  <m:GetLastTradePrice xmlns:m="Some-URI">  
    <symbol>DEF</symbol>  
  </m:GetLastTradePrice>  
</SOAP-ENV:Body>
```

```
</SOAP-ENV:Envelope>
```

SOAP – Παράδειγμα κλήσης

```
POST /StockQuote HTTP/1.1
```

```
Host: www.stockquoteserver.com
```

```
Content-Type: text/xml; charset="utf-8"
```

```
Content-Length: nnnn
```

```
SOAPAction: "http://example.com/stockquote.xsd"
```

```
<SOAP-ENV:Envelope
```

```
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
```

```
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
```

```
  <SOAP-ENV:Body>
```

```
    <m:GetLastTradePrice xmlns:m="http://example.com/stockquote.xsd">
```

```
      <symbol>MS</symbol>
```

```
    </m:GetLastTradePrice>
```

```
  </SOAP-ENV:Body>
```

```
</SOAP-ENV:Envelope>
```


SOAP – Παράδειγμα απόκρισης

HTTP/1.1 200 OK

Content-Type: text/xml; charset="utf-8"

Content-Length: nnnn

<SOAP-ENV:Envelope

xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"

SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />

<SOAP-ENV:Body>

<m:GetLastTradePriceResponse xmlns:m="http://example.com/stockquote.xsd">

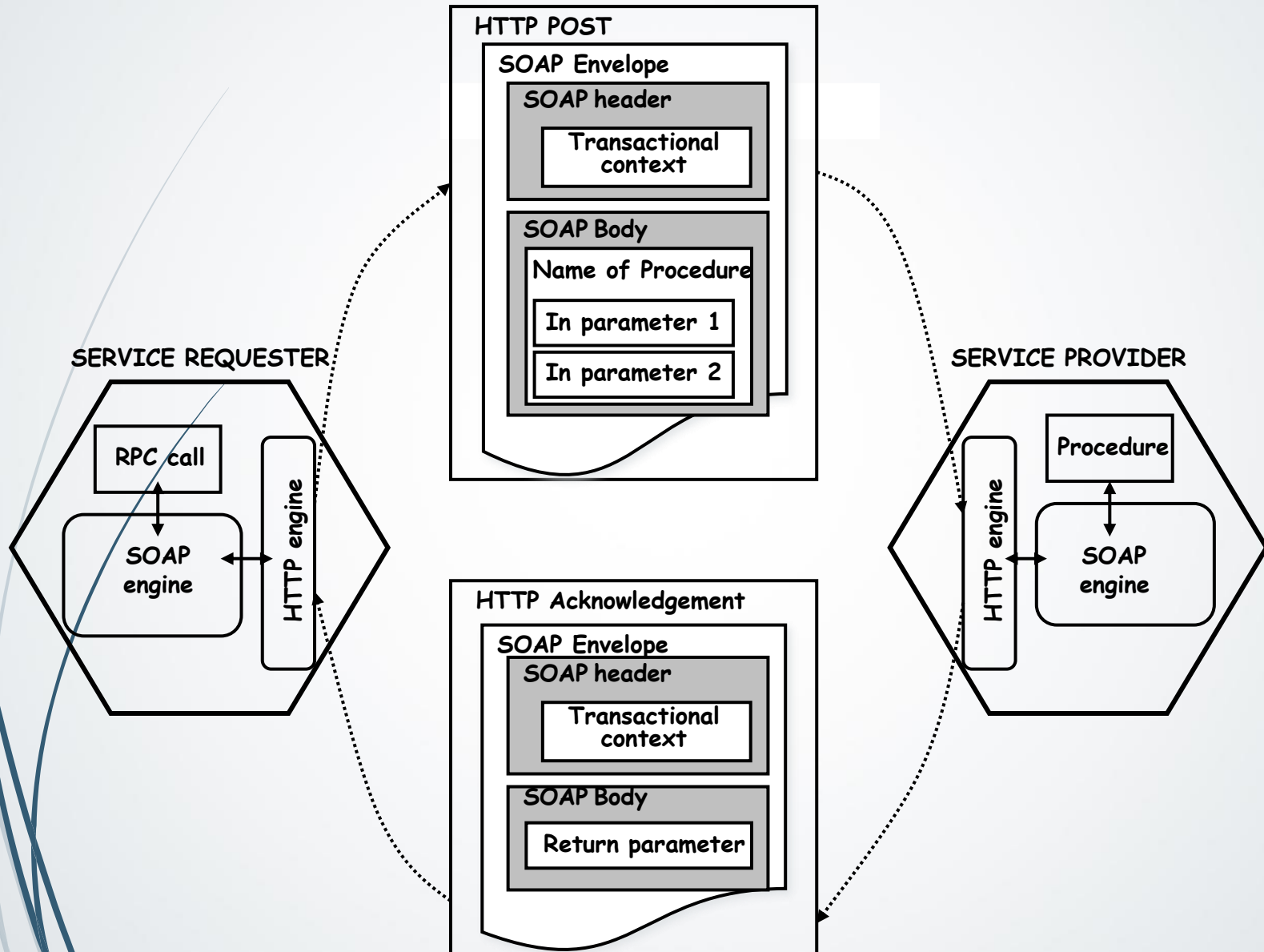
<Price>143</Price>


</m:GetLastTradePriceResponse>

</SOAP-ENV:Body>

</SOAP-ENV:Envelope>

Χρήση SOAP εντός του υπηρεσιοστρεφούς μοντέλου





UDDI – Universal Description, Discovery, and Integration

Τι είναι το UDDI?

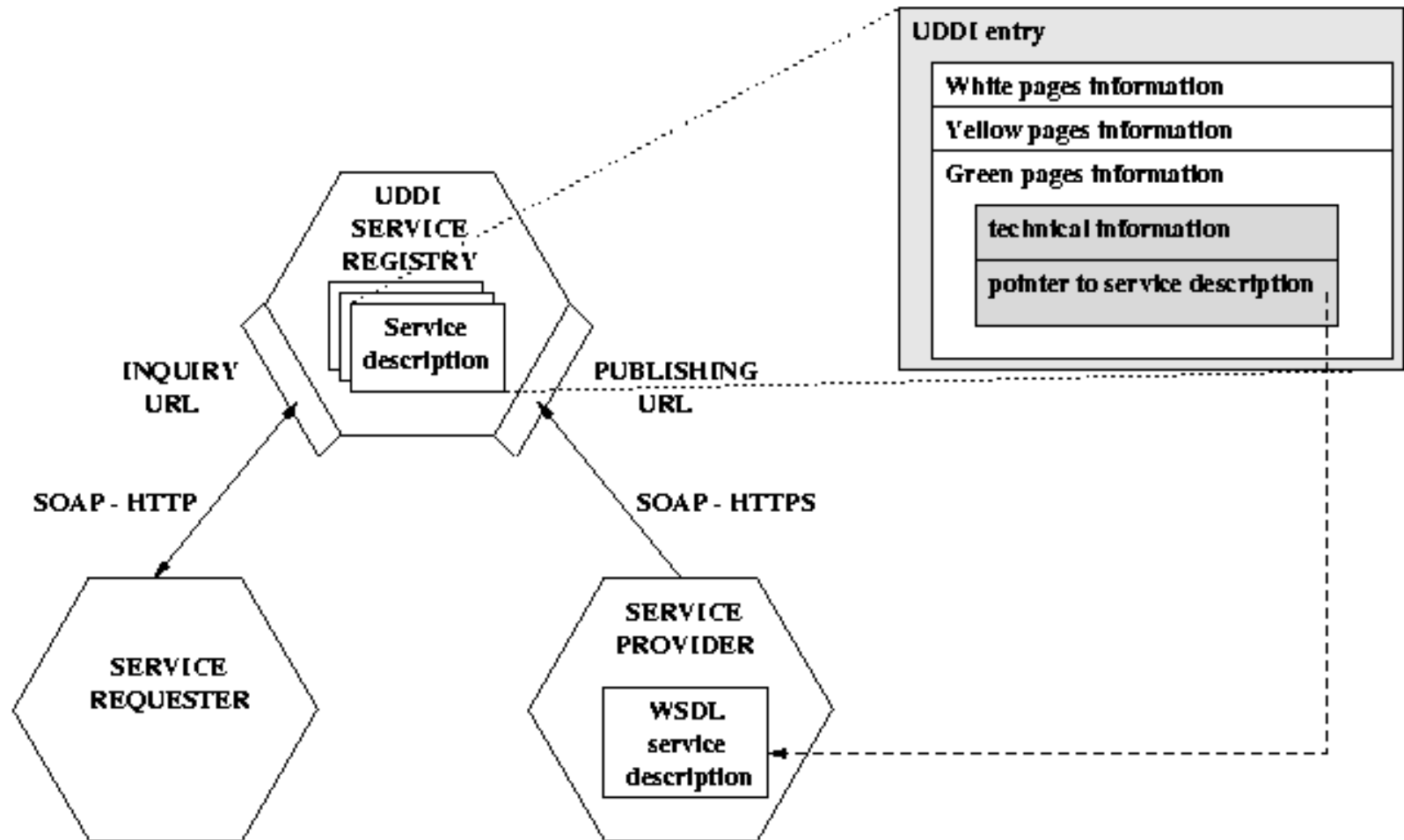
Universal Description, Discovery and Integration

- ▶ Μια υπηρεσία καταλόγου όπου οι εταιρείες μπορούν να καταγράψουν τα web services που παρέχουν και οι πελάτες μπορούν να βρίσκουν πληροφορίες
- ▶ Δημιουργήθηκε το 2000, με το όραμα να δημιουργήσει μια δημόσια ή ιδιωτική δυναμική υποδομή μεσάζοντα μεταξύ πελατών και παρόχων υπηρεσιών ιστού.
- ▶ Δεν υιοθετήθηκε ευρέως. Η IBM, Microsoft, and SAP ανακοίνωσαν ότι κλείνουν τα public UDDI nodes in January 2006. Η ομάδα του OASIS που δημιουργούσε τα specifications για το UDDI αποφάσισε ότι ολοκλήρωσε τη δουλειά το 2007, και η ομάδα αυτή πλέον δεν υπάρχει.
- ▶ Σήμερα βρίσκουμε UDDI συστήματα μέσα σε εταιρείες

Είδος πληροφορίας στο UDDI

- ▶ **white pages information:** πληροφορίες σχετικά με το ποιος παρέχει την υπηρεσία (όνομα, διεύθυνση, contact person, κτλ.)
- ▶ **yellow pages information:** τι τύπου υπηρεσίες προσφέρονται και μία λίστα με τις υπηρεσίες που προσφέρονται
- ▶ **green pages information:** τεχνικές πληροφορίες σχετικά με την χρήση του κάθε ενός service που προσφέρεται, συμπεριλαμβανομένου και δείκτες σε WSDL περιγραφές των υπηρεσιών (οι οποίες δεν αποθηκεύονται στο UDDI registry)

UDDI και WSDL



Βιβλιογραφία

- Werner Vogels, “**Web Services are not Distributed Objects**”, *IEEE Internet Computing*, Nov-Dec 2003.
- M. P. Papazoglou and D. Georgakopoulos, “**Service Oriented Computing**”, *Communications of the ACM*, vol. 46, no, 10, Oct. 2003, pp. 25–28
- A. Tsalgatidou, T. Pilioura, “**An Overview of Standards and Related Technology in Web Services**”. *International Journal of Distributed and Parallel Databases, Special Issue on E-Services*, 12(2), Sep 2002, 135-162.
- Graham et al., ***Building Web Services with Java: Making Sense of XML SOAP, WSDL and UDDI***, Sams Publishing 2001.
- G. Alonso, F. Casati, H. Kuno, V. Machiraju, ***Web Services Concepts, Architectures and Applications***, Springer-Verlag 2004

Χρήσιμοι Ιστότοποι

- ▶ SOAP - <http://www.w3.org/TR/soap12-part0/>
- ▶ WSDL 2.0 - <http://www.w3.org/TR/wsdl20-primer>
- ▶ WSDL 1.1 - <http://www.w3.org/TR/wsdl>
- ▶ UDDI - <http://uddi.xml.org/>

- ▶ Πληροφορίες σχετικά με Υπηρεσίες Ιστού
 - ▶ <http://www.webservicesarchitect.com/>
 - ▶ <http://www.w3schools.com/webservices/default.asp>