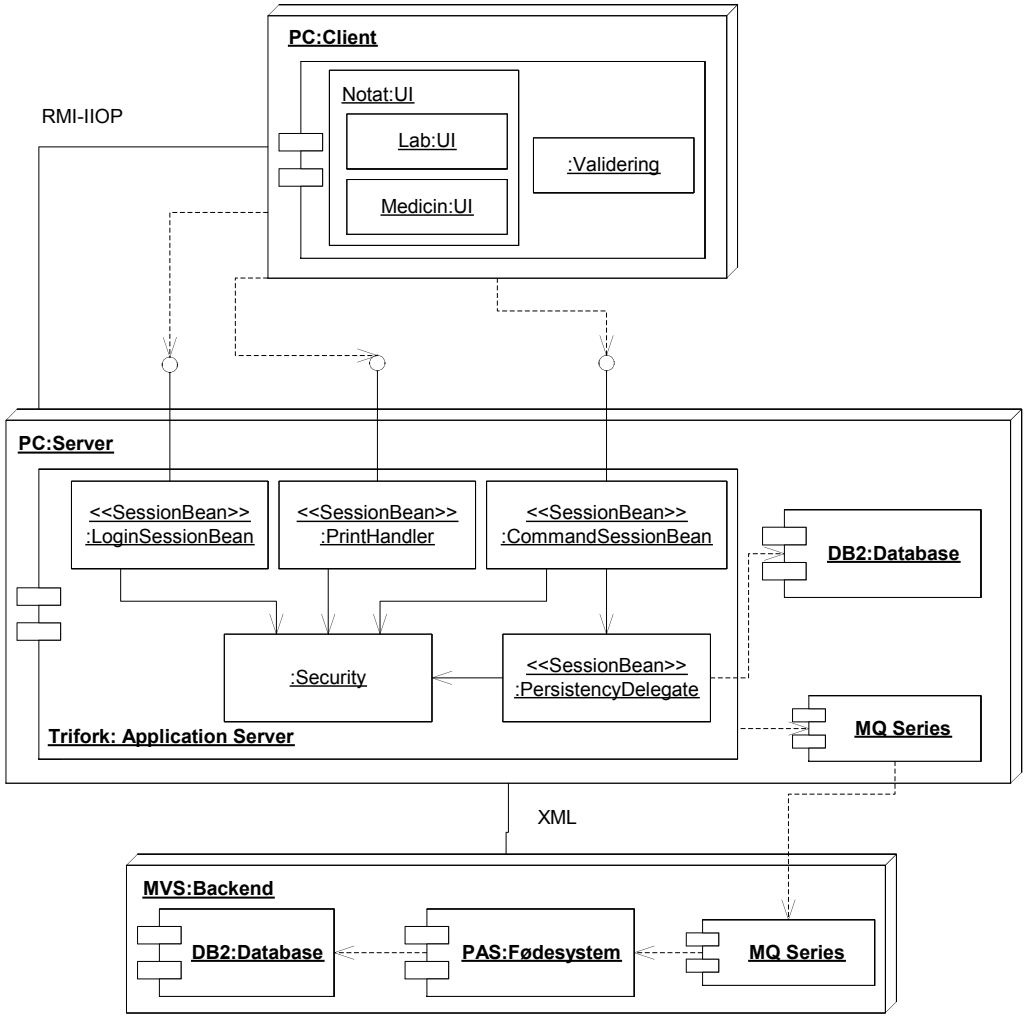
Service-Oriented Architecture (SOA)

Literature

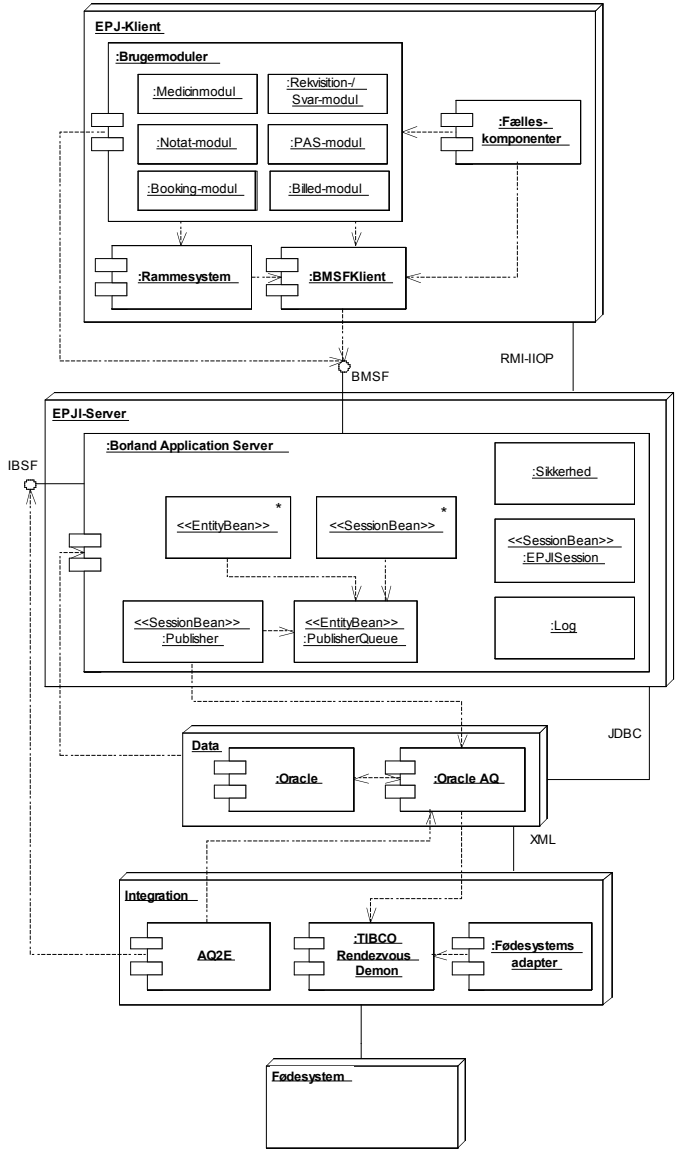
Main

- [MacKenzie et al., 2006]
  - MacKenzie, C. M., Laskey, K., McCabe, F., Brown, P. F., and Metz, R. (2006). *Reference Model for Service Oriented Architecture 1.0*. Technical Report Committee Specification 1, 2 August 2006, OASIS. Accessed 2008-03-01
- [Curbera et al., 2002]
  - Curbera, F., Duftler, M., Khalaf, R., Nagy, W., Mukhi, N., and Weerawarana, S. (2002). Unraveling the web services web. An introduction to SOAP, WSDL, and UDDI. *IEEE Internet Computing*, 6(2):86–92

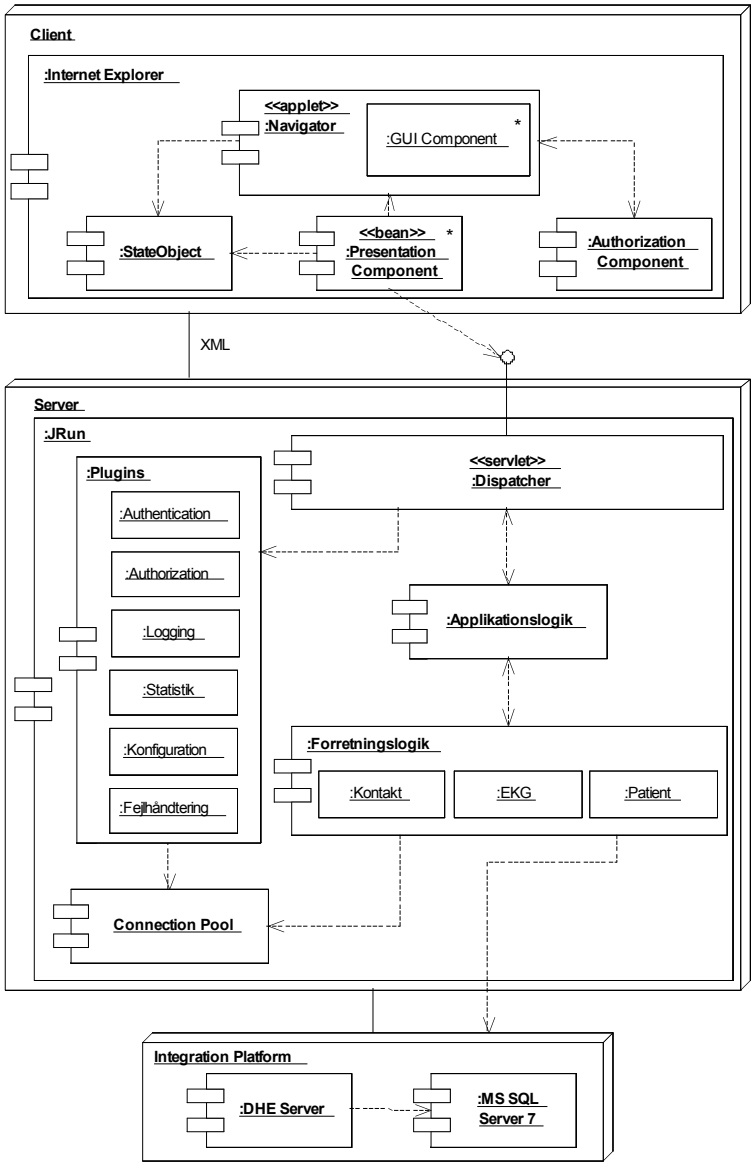
Motivation: EHR Ribe Amt



Motivation: EHR Aarhus Amt



Motivation: EHR H:S Amager



Motivation

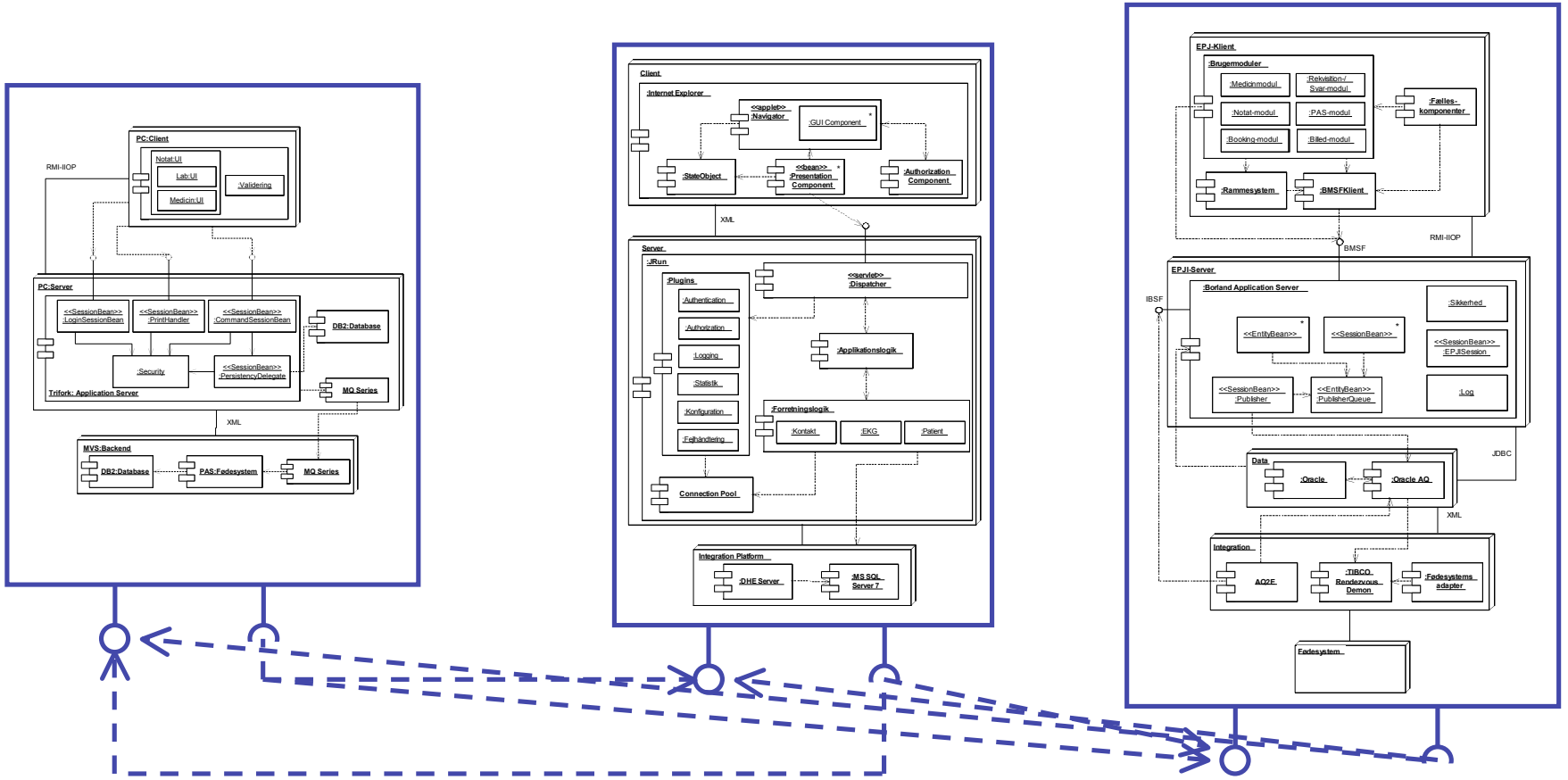
AARHUS UNIVERSITET

Ribe Amt, Aarhus Amt, H:S Amager EHR systems

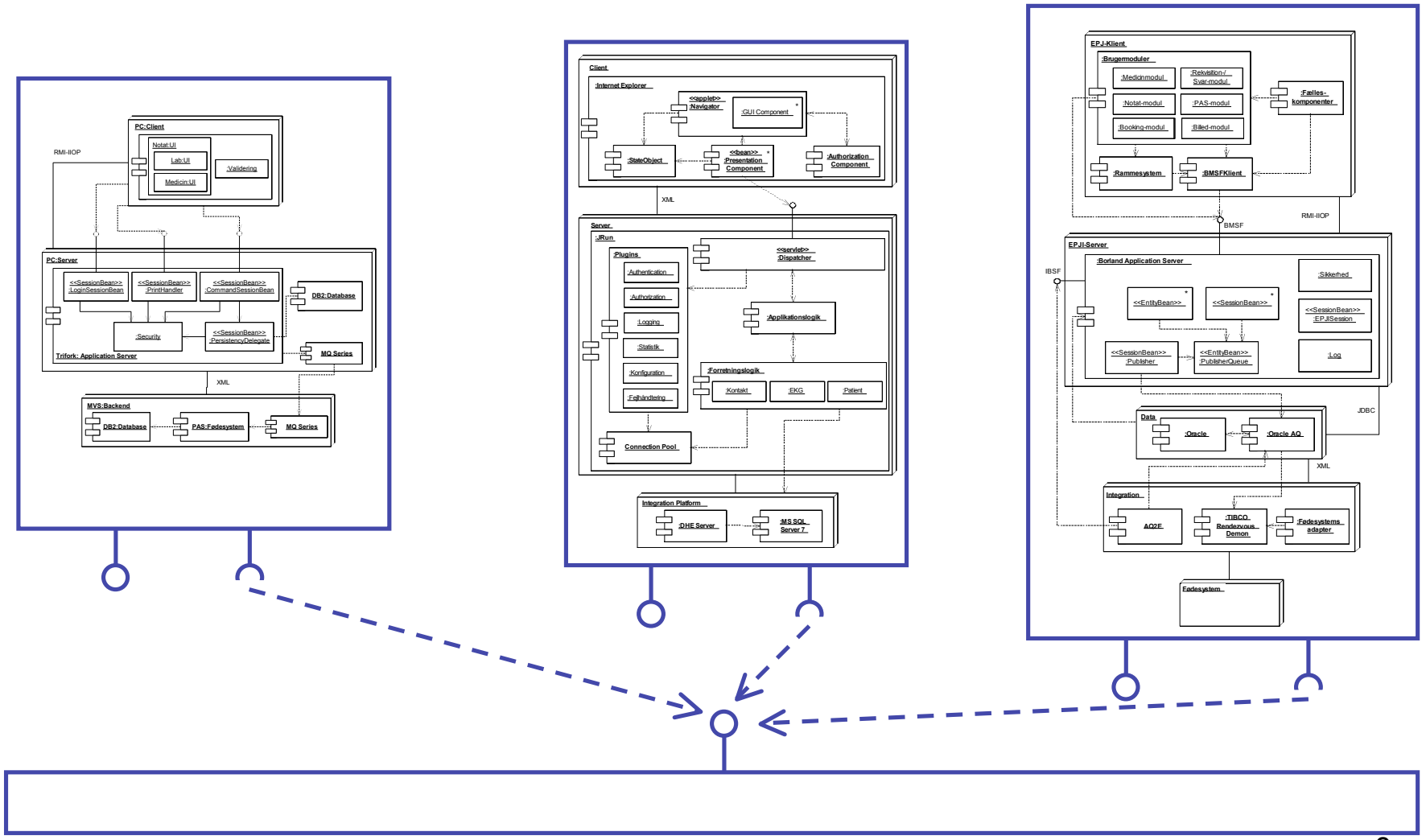
All fine systems, but use of commonalities?

- Interoperability?
- Reuse?

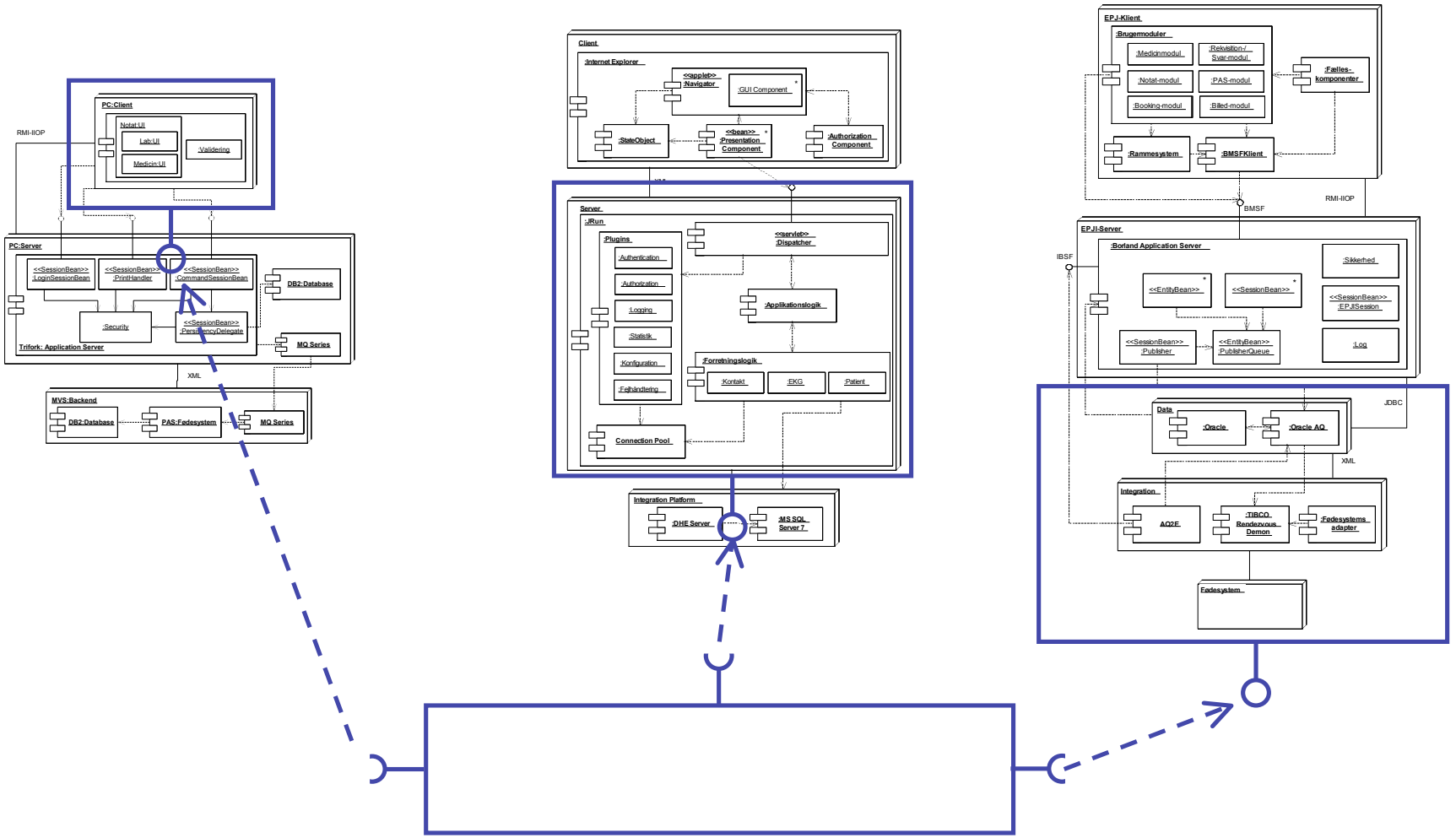
Motivation: SOA 1



Motivation: SOA 2



Motivation: SOA 3



A Definition

Service Oriented Architecture

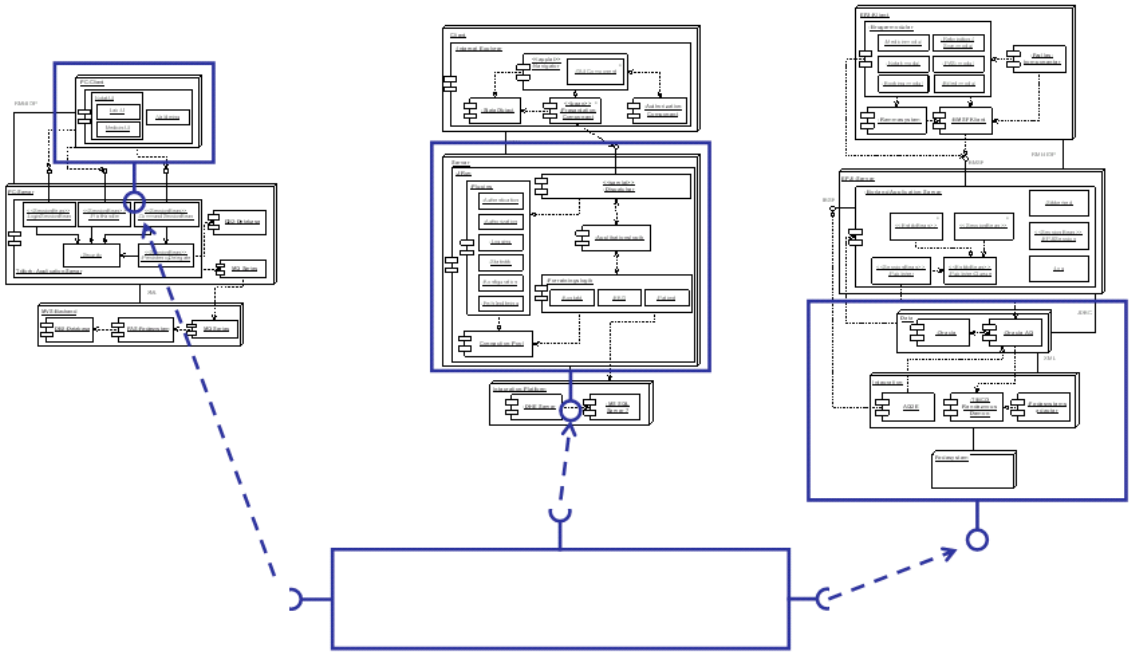
- is a paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains
- provides a uniform means to offer, discover, interact with and use capabilities to produce desired effects consistent with measurable preconditions and expectations

[MacKenzie et al., 2006]

Exercise!

1. What kind of quality tradeoffs would you expect from an architecture like this?

2. Which types of technologies would/could you use to realize an architecture like this?



More Motivation…

”We keep on reinventing the wheel. That is the way we as vendors make our money”

– Beat Schwegler, software architect at Microsoft (paraphrased after Computerworld.dk report from JAOO 2006)

”I hope this is a bad joke – or that you get fired!”

– Bert Vogels, CTO, Amazon

“Importance”

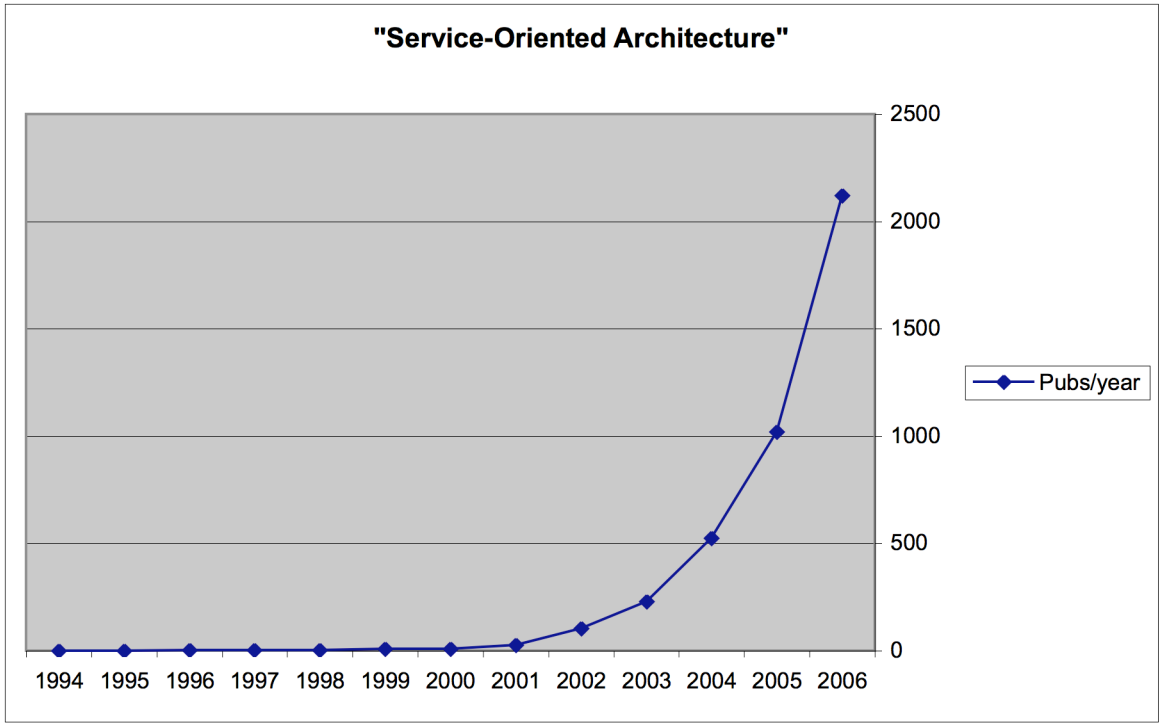
AARHUS UNIVERSITET

Google says…

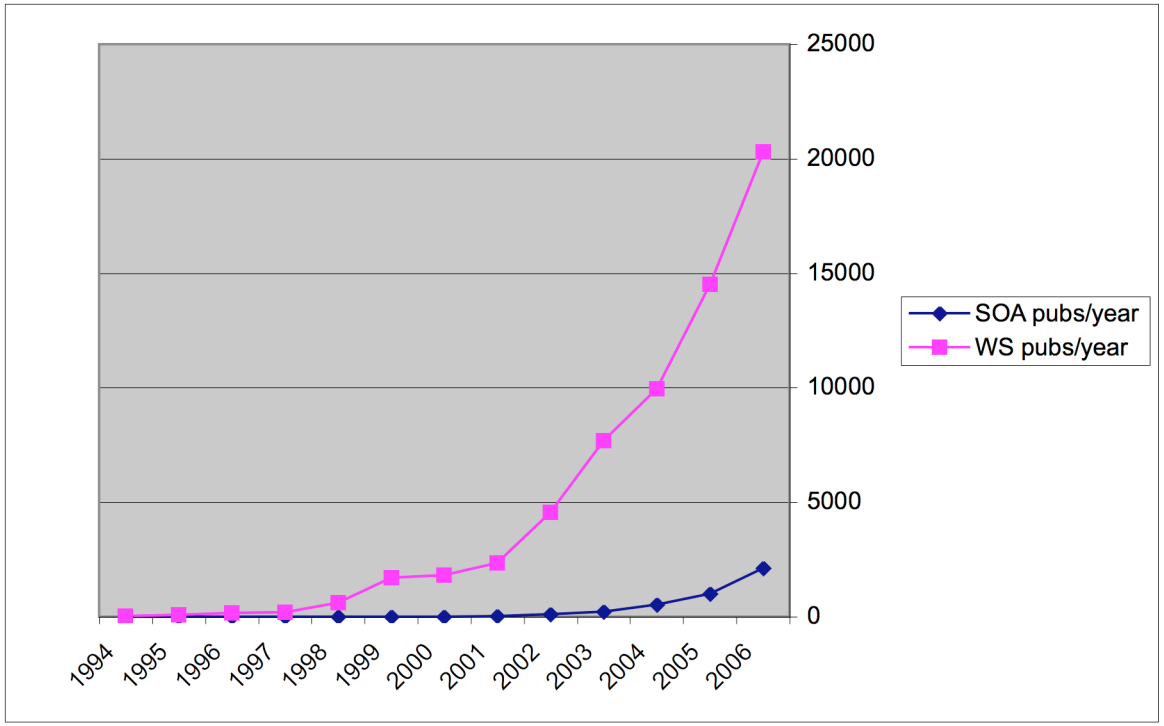
- ”Service-Oriented Architecture”
  - 2007: Ca. 501.000 hits
  - 2008: Ca. 5.220.000 hits
- ”Web Services”
  - 2007: Ca. 24.100.000 hits
  - 2008: Ca. 59.600.600 hits

Google Scholar…

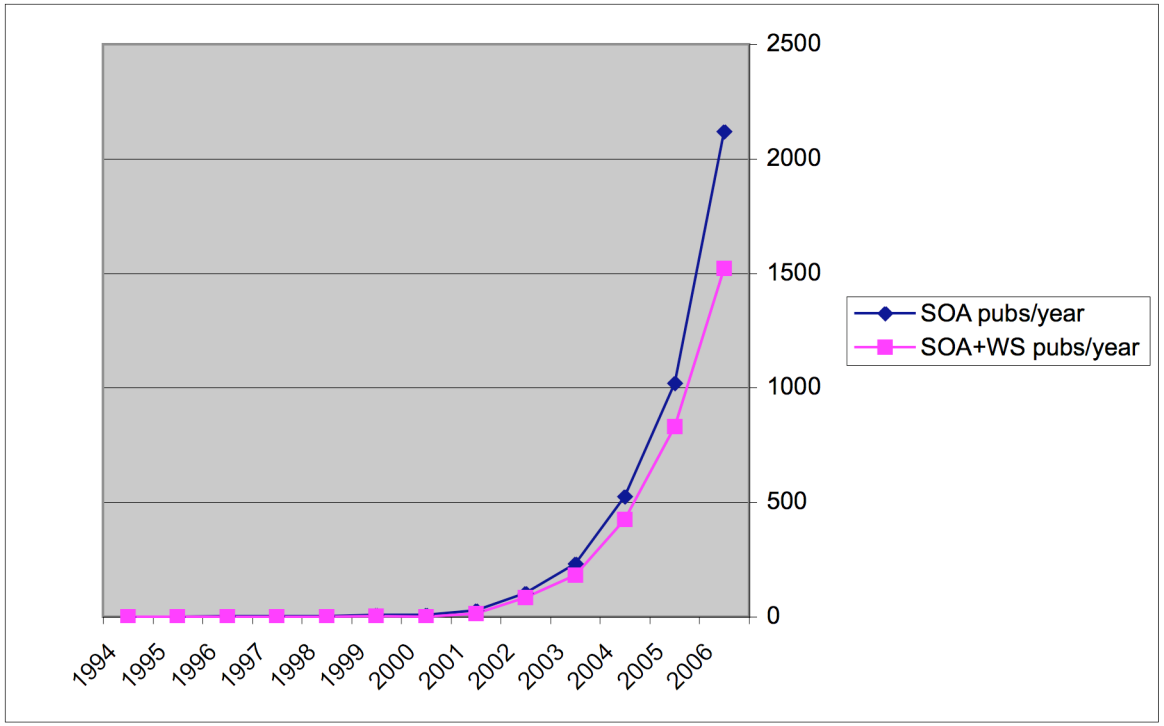
”Service Oriented Architecture”



Compared to ”Web Services”



”SOA is not WS”…


2500
2000
1500
1000
500
0
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
SOA pubs/year
SOA+WS pubs/year

[MacKenzie et al., 2006]

OASIS

Reference Model for Service Oriented Architecture 1.0

Committee Specification 1, 2 August 2006

Document identifier:
soa-rm-cs

Location:
http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=soa-rm

Editors:
C. Matthew MacKenzie, Adobe Systems Incorporated, mattm@adobe.com
Ken Laskey, MITRE Corporation, klaskey@mitre.org
Francis McCabe, Fujitsu Laboratories of America Limited, frankmccabe@mac.com
Peter F Brown, peter@justbrown.net
Rebekah Metz, Booz Allen Hamilton, metz_rebekah@bah.com

Abstract:
This Reference Model for Service Oriented Architecture is an abstract framework for understanding significant entities and relationships between them within a service-oriented environment, and for the development of consistent standards or specifications supporting that environment. It is based on unifying concepts of SOA and may be used by architects developing specific service oriented architectures or in training and explaining SOA.

A reference model is not directly tied to any standards, technologies or other concrete implementation details. It does seek to provide a common semantics that can be used unambiguously across and between different implementations. The relationship between the Reference Model and particular architectures, technologies and other aspects of SOA is illustrated in Figure 1.

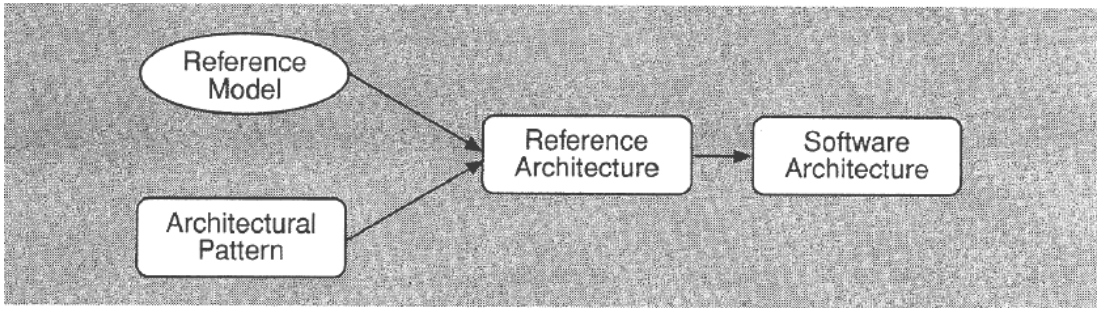
While service-orientation may be a popular concept found in a broad variety of applications, this reference model focuses on the field of software architecture. The concepts and relationships described may apply to other "service" environments; however, this specification makes no attempt to completely account for use outside of the software domain.

Status:
This document is updated periodically on no particular schedule. Send comments to the editor(s).
Committee members should send comments on this specification to the soa-rm@lists.oasis-open.org list. Others should visit the SOA-RM TC home page at http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=soa-rm, and record comments using the web form available there.
For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the SOA-RM TC web page at: http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=soa-rm
The errata page for this specification is at: http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=soa-rm.

Towards the Point…

Presents a *reference model* for Service-Oriented Architecture

- Reconcile different definitions of ‘SOA’
- Provide minimal set of unifying concepts



We will discuss this

- in relation to web services
- in relation to the topics of this course

Vs Web Services

AARHUS UNIVERSITET

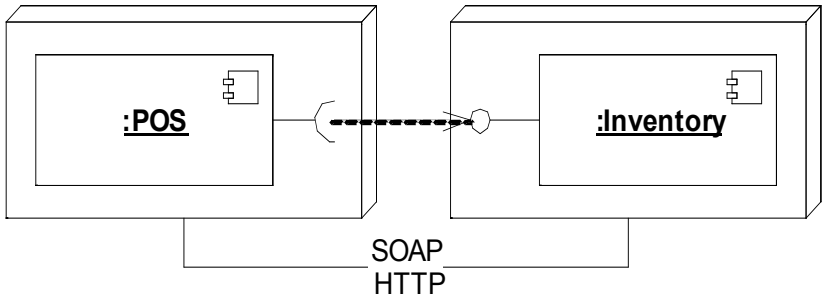
A software system designed to support interoperable machine-to-machine interaction over a network

- It has an interface described in a machine-processable format (specifically *WSDL*)
- Other systems interact with the Web service in a manner prescribed by its description using *SOAP* messages
- These are typically conveyed using *HTTP with an XML serialization* in conjunction with other Web-related standards

[W3C, 2004]

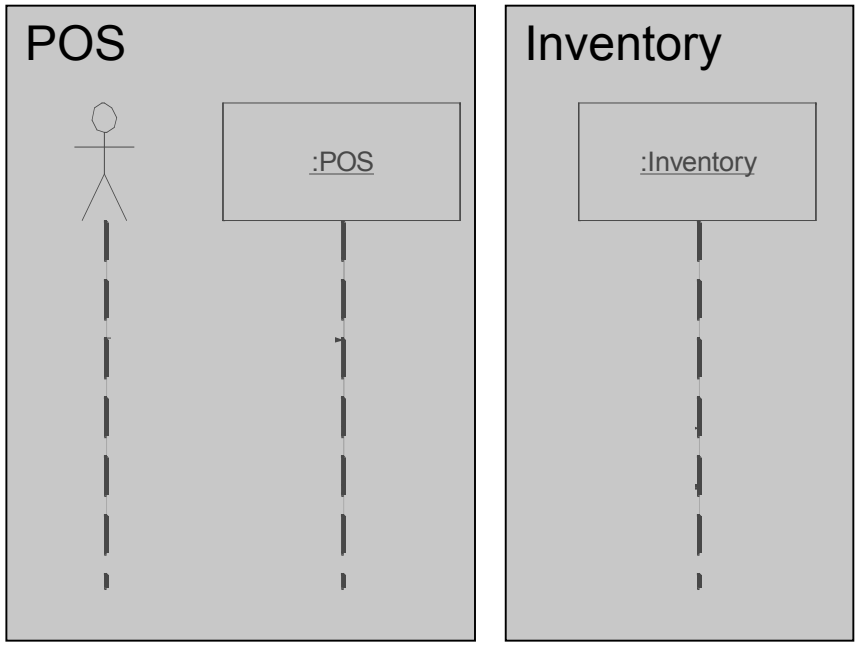
(+ UDDI for registration and discovery)

POS Example – Deployment



(Code will be made available on the course website)

POS Example – Protocol



Key Elements

SOAP

- Communication protocol for web services
- Typically over HTTP or SMTP

WSDL (Web Service Description Language)

- An XML-based language for describing network services
- An Interface Description Language for Web services

UDDI (Universal Description, Discovery, and Integration)

- Provides a registration and lookup mechanism for clients and servers to find each other
- Uses SOAP for communication

WSDL Example


```
<?xml version="1.0" encoding="UTF-8"?>
<definitions …>

    <types>
      <xs:schema targetNamespace="http://pos.com">
        <xs:complexType name="Request">
          <xs:sequence>
            <xs:element name="Command"/>
            <xs:element name="Value"/>
          </xs:sequence>
        </xs:complexType>
      </xs:schema>
    </types>

    <message name="executeUpdateRequest">
      <part name="parameter type="pos:Request"/>
    </message>
    <message name="executeUpdateResponse">
      <part name="parameter" type="xs:string"/>
    </message>

    <portType name="InventoryPort">
      <operation name="executeUpdate">
        <input message="pos:executeUpdateRequest"/>
        <output message="pos:executeUpdateResponse"/>
      </operation>
    </portType>
```

Abstract description

```
<binding name="InventoryBinding" type="pos:InventoryPort">
  <soap:binding style="rpc"
     transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="executeUpdate">
    <soap:operation
     soapAction="http://pos.com/inventory/executeUpdate"
     style="rpc"/>
      <input/>
        <soap:body use="literal"/>
      </input>
      <output>
        <soap:body use="literal"/>
      </output>
    </operation>
</binding>

<service name="InventoryService">
  <port name="InventoryPort" binding="pos:InventoryBinding">
    <soap:address location="http://pos.com/..."/>
  </port>
</service>

</definitions>
```

Concrete binding

Should really be handled by UDDI…

SOAP Request Example

AARHUS UNIVERSITET

```
POST /axis/services/InventoryPort HTTP/1.0
Content-Type: text/xml; charset=utf-8
Accept: application/soap+xml, application/dime, multipart/related,
text/*
User-Agent: Axis/1.3
Host: 127.0.0.1:9999
Cache-Control: no-cache
Pragma: no-cache
SOAPAction: "http://pos.com/inventory/executeUpdate"
Content-Length: 459

<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchemainstance">
  <soapenv:Header/>
  <soapenv:Body>
    <executeUpdate xmlns="">
      <parameter xsi:type="ns1:Request" xmlns:ns1="http://pos.com">
        <Command xsi:type="xsd:string">Buy</Command>
        <Value xsi:type="xsd:string">SAIP</Value>
      </parameter>
    </executeUpdate>
  </soapenv:Body>
</soapenv:Envelope>
```

SOAP Response Example

AARHUS UNIVERSITET

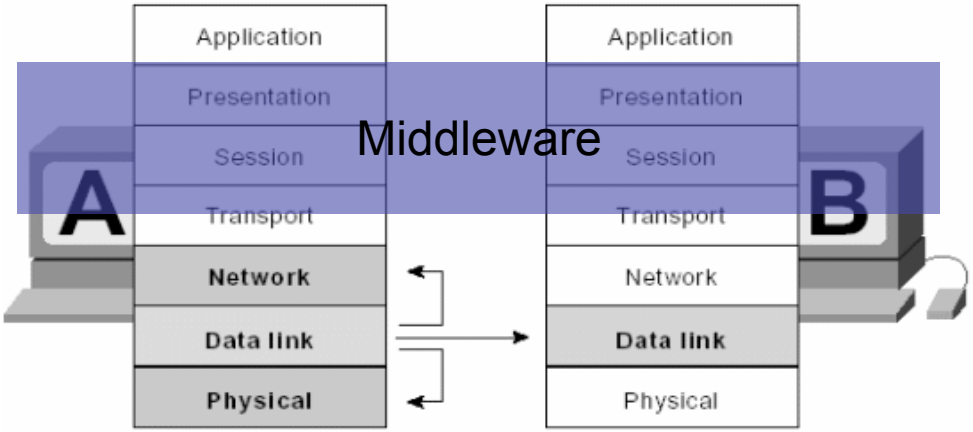
```
HTTP/1.1 200 OK
Server: Jetty/5.1.10 (Windows XP/5.1 x86 java/1.5.0_07
Content-Type: text/xml; charset=utf-8
Connection: close

<?xml version="1.0" encoding="utf-8"?>
<soapenv:Envelope
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <executeUpdateResponse xmlns="">
      <parameter>Buy SAIP</parameter>
    </executeUpdateResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

SOA and Middleware

A definition

– *In a distributed computing system,* middleware *is defined as the software layer that lies between the operating system and the applications on each site of the system [ObjectWeb, 2003]*



Characteristics of Middleware

Hiding distribution

Hiding heterogeneity of hardware, communication, operating systems

Providing uniform, standard, high-level interfaces for developers

Supplying a set of common services

[ObjectWeb, 2003]

”Set of Common Services”

AARHUS UNIVERSITET

CORBA services

- Collection Service
- Persistent State Service
- Concurrency Service
- Property Service
- Enhanced View of Time
- Query Service
- Event Service
- Relationship Service
- Externalization Service
- Security Service
- Naming Service
- Time Service
- Licensing Service
- Trading Object Service
- Life Cycle Service
- Transaction Service

http://www.omg.org/cgi-bin/doc?formal/98-10-01

Web services specifications

- WS-Addressing
- WS-Attachments
- WS-BusinessActivity
- WS-Coordination
- WS-Discovery
- WS-Enumeration
- WS-Eventing
- WS-Federation
- WS-Inspection
- WS-Manageability
- WS-MetadataExchange
- WS-Notification
- WS-PolicyFramework
- WS-Provisioning
- WS-ReliableMessaging
- WS-Resource
- WS-Security
- WS-Topics
- WS-Transactions
- WS-Transfer

http://msdn.microsoft.com/Webservices/understanding/specs/

http://www.ibm.com/developerworks/webservices/standards/

”Set of Common Services”

Business Domain Specific extensions	Various	**Business Domain**
Distributed Management	WSDM, WS-Manageability	**Management**
Provisioning	WS-Provisioning	
Security	WS-Security	**Security**
Security Policy	WS-SecurityPolicy	
Secure Conversation	WS-SecureConversation	
Trusted Message	WS-Trust	
Federated Identity	WS-Federation	
Portal and Presentation	WSRP	**Portal and Presentation**
Asynchronous Services	ASAP	**Transactions and Business Process**
Transaction	WS-Transactions, WS-Coordination, WS-CAF	
Orchestration	BPEL4WS, WS-CDL	
Events and Notification	WS-Eventing, WS-Notification	**Messaging**
Multiple message Sessions	WS-Enumeration, WS-Transfer	
Routing/Addressing	WS-Addressing, WS-MessageDelivery	
Reliable Messaging	WS-ReliableMessaging, WS-Reliability	
Message Packaging	SOAP, MTOM	
Publication and Discovery	UDDI, WSIL	**Metadata**
Policy	WS-Policy, WS-PolicyAssertions	
Base Service and Message Description	WSDL	
Metadata Retrieval	WS-MetadataExchange	

Status of the Common Services

Mainstream	Early Adoption	Experimentation	Specification
SOAP	WS-Security	ASAP	WS-Addressing
WSDL	WS-RP	BPEL	WS-CAF
UDDI	WS-Reliability	WS-Coordination	WS-Choreography
	SOAP MTOM	WS-Policy	WSDM
			WS-Eventing
			WS-Federation
			WS-IL
			WS-Provisioning
			WS-ReliableMessaging
			WS-Resource Framework

Approved Standards

Proposals

A Definition

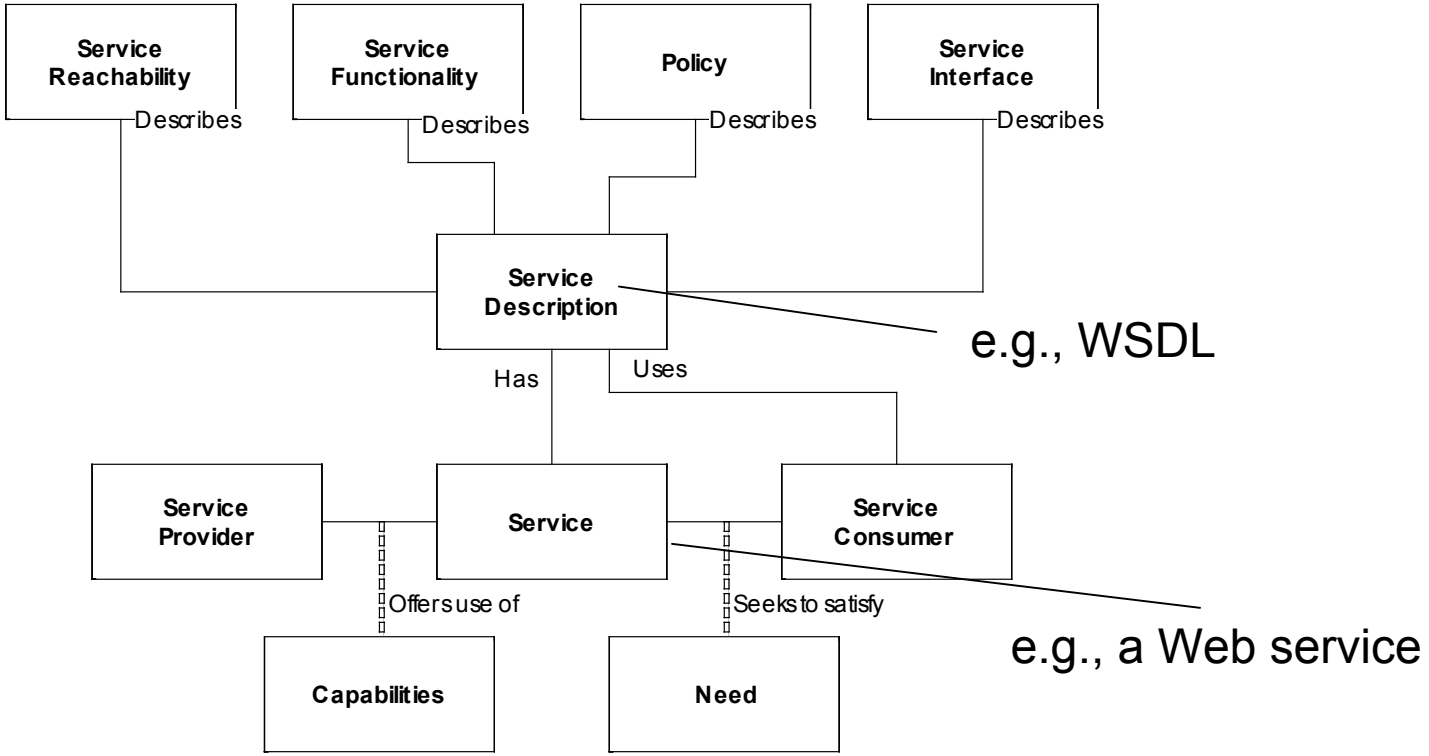
Service Oriented Architecture

- is a paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains
- provides a uniform means to offer, discover, interact with and use capabilities to produce desired effects consistent with measurable preconditions and expectations

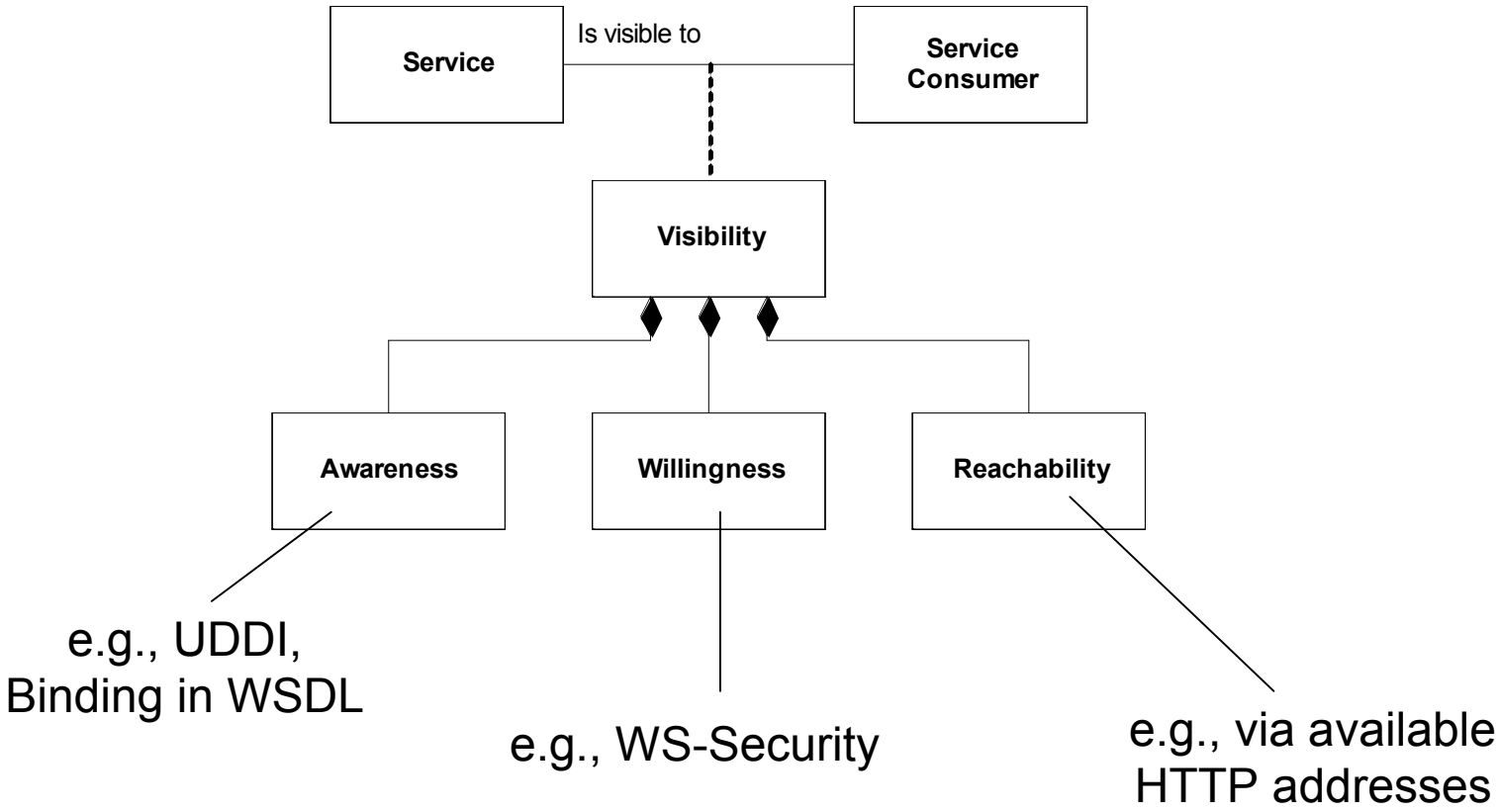
[MacKenzie et al., 2006]

An SOA Ontology…

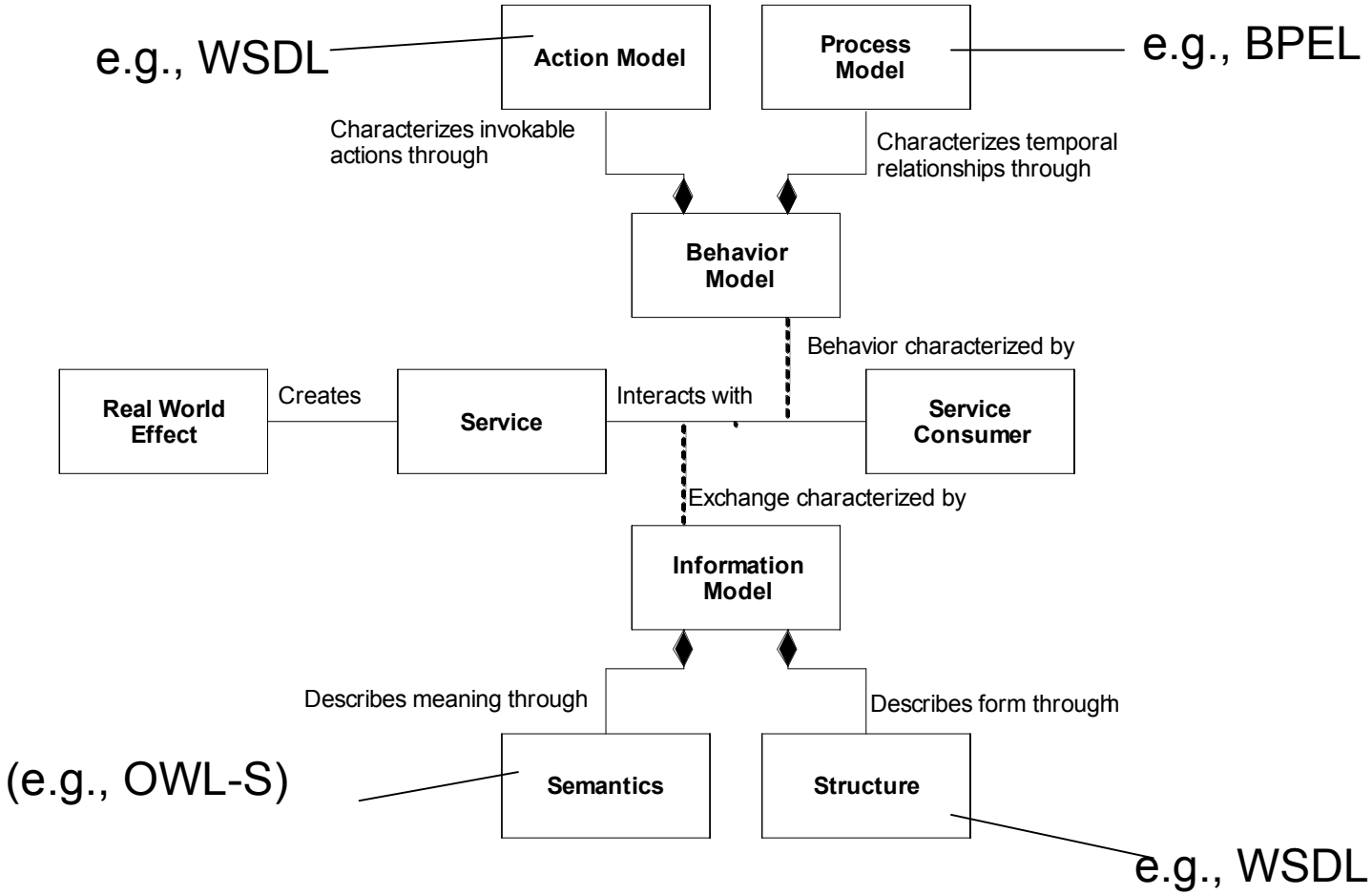
Adapted from [MacKenzie et al., 2006] topic maps



Dynamics: Visibility



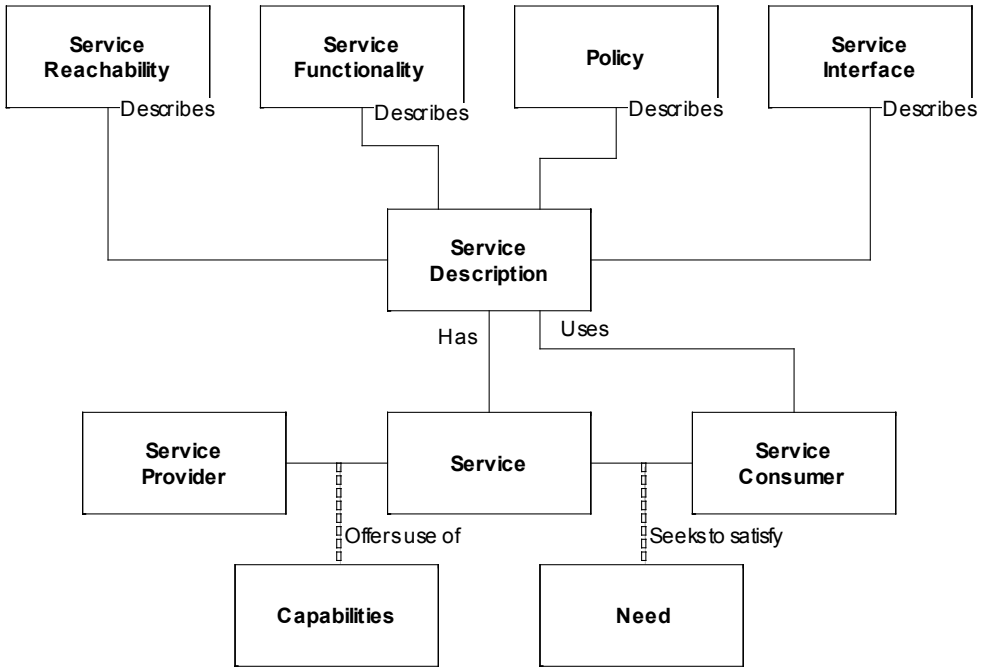
Dynamics: Interaction



Definition

What is the software architecture of an SOA?

The software architecture of a program or computing system is the structure or *structures* of the system, which comprise *software elements*, the *externally visible properties* of those elements, and the *relationships* among them [Bass et al., 2003]



Quality Attributes

Claims of SOA (according to [MacKenzie et al., 2006]

– ”*reuse*, *growth*, and *interoperability*”

Versus [Bass et al., 2003] quality attributes?

– System quality attributes
  - Performance
  - Availability
  - Modifiability
  - Security
  - Testability
  - Usability
– Business quality attributes
– Architectural quality attributes

Designing Services

AARHUS UNIVERSITET

Service guidelines

- Self-contained
  - Service should have “coarse-grained” functionality
- Context-independence
  - Service should not depend on other services
    - Aka “loose coupling”
- “Stateless”
  - Service should not retain client state between invocations

Versus [Bass et al., 2003] tactics?

Summary

Service-Oriented Architecture is *very* hyped currently

There is a degree of technical substance

- But most technical substance is well-known in enterprise systems
- Using it in new areas is challenging and interesting

There is a high degree organizational/business substance