

# Jump Onto The Bus

A Guided Tour To The WS-\* Landscape

Keynote at ICSOC 2003  
International Conference on Service Oriented Computing  
( Trento, Italy, December 16 - 18, 2003 )

Prof. Dr. Frank Leymann  
IBM Distinguished Engineer  
Member, IBM Academy Of Technology

IBM Software Group  
Schoenicherstr. 220  
71032 Boeblingen  
Germany

Phone +49-7031-16 3998  
Fax +49-7031-16 4890  
Mobile +49-172-731 5858  
e-mail Ley1@de.ibm.com

2

## Copyright Notice:

Most of the material of this presentation is copyrighted by IBM or by others. Please don't use any of the material of this presentation without checking with the author mentioned on the front page. Thanks!

© IBM

Frank Leymann

3

## Agenda

The Landscape  
Virtual Components  
Virtual Environments  
Application Structure  
Aggregations  
Summary & Conclusion

© IBM

Frank Leymann

4

## Agenda

**The Landscape**  
Virtual Components  
Virtual Environments  
Application Structure  
Aggregations  
Summary & Conclusion

© IBM

Frank Leymann

# ...expect more!



## 6 The Ordering and Guiding Principles

- All standard efforts should foster...
  1. an environment which is
    - a. heterogeneous,
    - b. distributed,
    - c. interoperable,
    - d. loosely coupled,
    - e. multi-vendor
  2. secure, reliable, recoverable interactions
  3. a single uniform usage model for "components"
  4. business criteria to discover components needed

**"The Bus"!**

## 7 Agenda

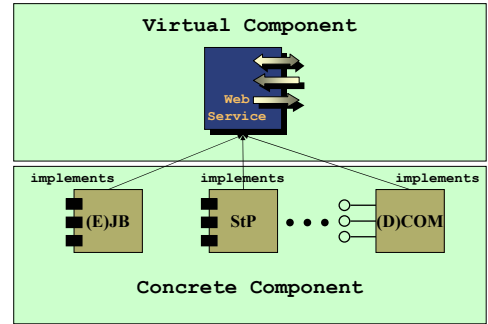
- The Landscape
- Virtual Components**
- Virtual Environments
- Application Structure
- Aggregations
- Summary & Conclusion

## 8 What Are Web Services At All?

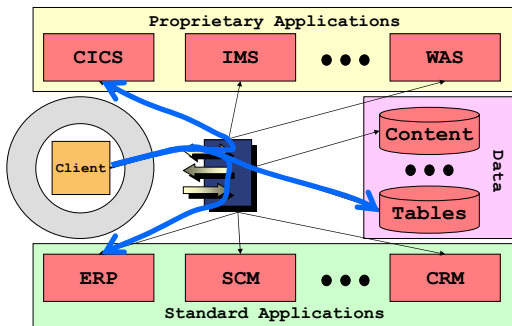
## 9 Web Service Technology: The Key Thing!

1. Web service technology provides a "virtual component model" for using components (read: message processor)
2. When exploiting a Web service the supporting container hides „middleware idiosyncrasies“ like the underlying component model of the actual target component used, the invocation protocol etc. as far as possible.
3. Web service technology does *not* provide a new component model for implementing components.

## 10 Virtualizing Components



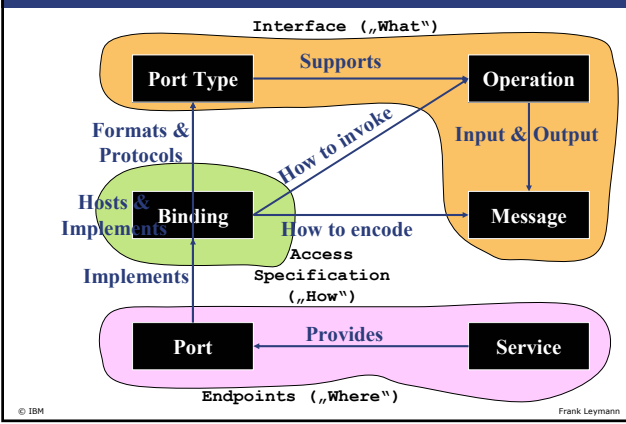
## 11 Again: Web Services Usages



## 12

**WSDL:  
Bindings Are Key!**

13 **Ingredients Of WSDL**



14 **Port Type: Example**

```
<?xml version="1.0"?>
<definitions name="StockQuote" ... />

  <portType name="StockQuotePortType">
    <operation name="GetLastTradePrice">
      <input message="tns:GetLastTradePriceRequest"/>
      <output message="tns:GetLastTradePriceResponse"/>
    </operation>
  </portType>

</definitions>
```

15 **Port & Binding: Example - SOAP/HTTP**

```
<binding name="StockQuoteSoapBinding"
  type="tns:StockQuotePortType">
  <soap:binding style="document"
    transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="GetLastTradePrice">
    <input>
      <soap:body use="encoded" namespace="http://leymann.com/stockquote"
        encodingStyle="http://schemas.xmlsoap.org/soap/encoding"/>
    </input>
    <output>
      ...
    </output>
  </operation>
</binding>

<service name="StockQuoteService">
  <port name="StockQuotePort"
    binding="tns:StockQuoteSoapBinding">
    <soap:address location="http://leymann.com/stockquote"/>
  </port>
</service>
```

16 **Port & Binding: Example - SOAP/JMS**

```
<binding name="StockQuoteSoapJMSBinding"
  type="tns:StockQuotePortType">
  <soap:binding style="rpc"
    transport="http://schemas.xmlsoap.org/soap/jms"/>
  (... )
</binding>

<service name="StockQuoteService">
  <port name="StockQuotePort"
    binding="tns:StockQuoteSoapJMSBinding">
    <jms:address destinationType="queue"
      jndiConnectionFactoryName="myQCF"
      jndiDestinationName="myQ"
      initialContextFactory="com.ibm.NamingFactory"
      jndiProviderURL="iiop://something:900"/>
  </port>
</service>
```

17 Port & Binding: Example - Java/EJB

```
<binding name="StockQuoteEJBBinding" type="tns:StockQuotePortType">
  <ejb:binding/>
  <operation name="GetLastTradePrice">
    <ejb:operationmapping method="getStockPrice"
      partOrdering="tickerSymbol time"
      interface="remote" />
    <input name="GetLastTradePriceRequest"/>
    <output name="GetLastTradePriceResponse"/>
    <fault name="FaultMessageSymbolNotFound"/>
    <fault name="FaultMessageInvalidTime"/>
  </operation>
</binding>

<service name="StockQuoteEJBService">
  <port name="StockQuoteEJBPort"
    binding="tns:StockQuoteEJBBinding">
    <ejb:address class="com.ibm.example.EJBStockQuoteService"
      jndiName="myapp/EJBStockQuoteService" />
  </port>
</service>
```

18 Port & Binding: Example - CICS

```
<binding name="StockQuoteCICSBinding" type="tns:StockQuotePortType">
  <cics:binding/>
  <operation name="GetLastTradePrice">
    <cics:operation functionName="GETLTP"/>
    <input name="Request" ... </input>
    <output name="Response" ... </output>
  </operation>
</binding>

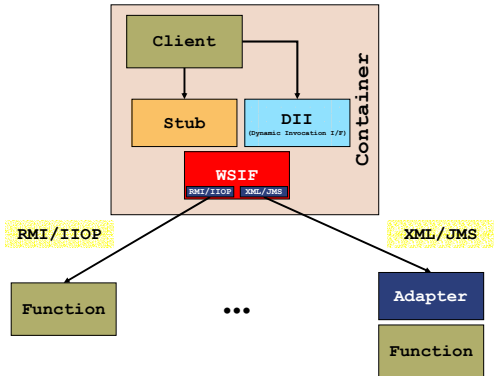
<service name="StockQuoteCICSService">
  <port name="StockQuoteCICSForT"
    binding="tns:StockQuoteCICSBinding">
    <cics:address connectionURL="FL.com" serverName="cics21"/>
  </port>
</service>
```

Defining structure of COMMAREA

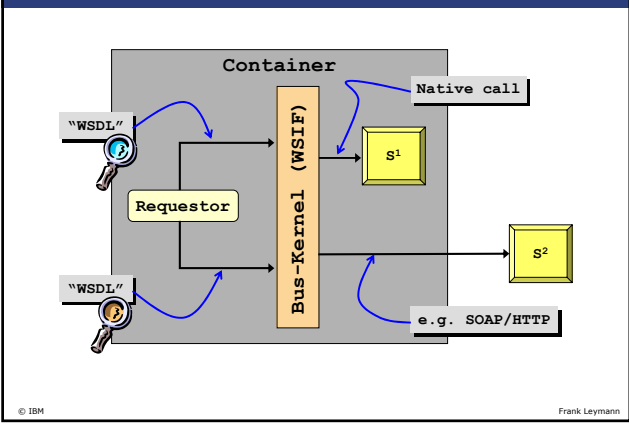
19 Invoking Web Services

# Invoking Web Services

20 WSIF: Using All Kinds of Bindings In J2EE



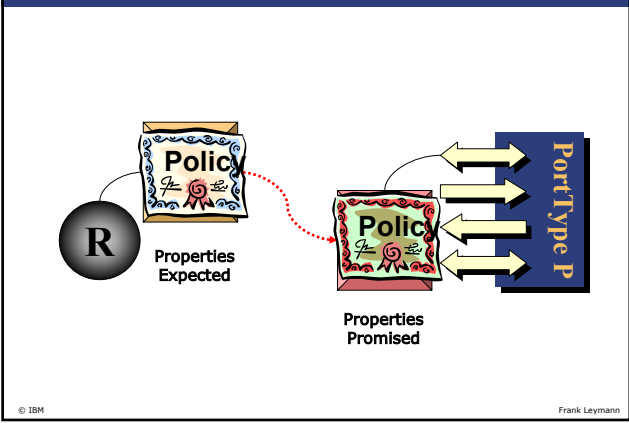
21 Invoking Services Within A Container



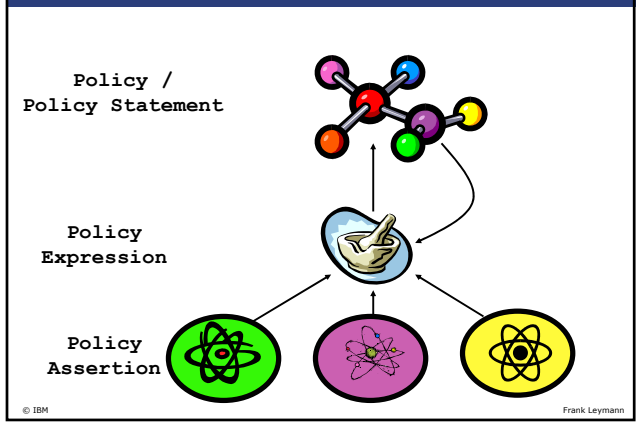
22

Policies

23 Quality Of Services & Policies



24 Abstract Policy Model



25 Policy: Example

```

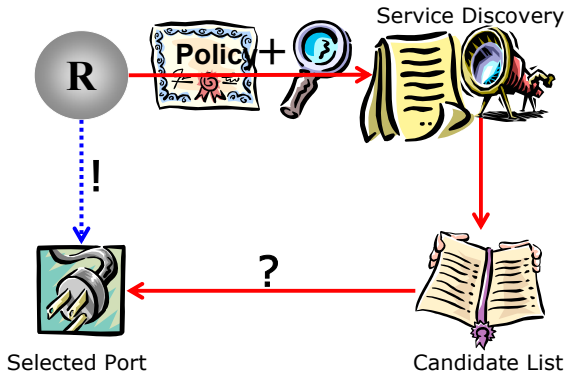
<wsp:Policy Name="Payments" TargetNamespace="http://fl.com/pol"
  xmlns:my="http://fl.com/pol" ...>
  <wsp:ExactlyOne>

  <my:onlinePayment>
    <wsp:ExactlyOne wsu:Id="opts">
      <my:creditCard wsp:Usage="wsp:Required" />
      <my:check wsp:Usage="wsp:Required" />
      <my:transfer wsp:Usage="wsp:Required" />
    </wsp:ExactlyOne >
    <my:perUse Usage="wsp:Required" />
  </my:onlinePayment>

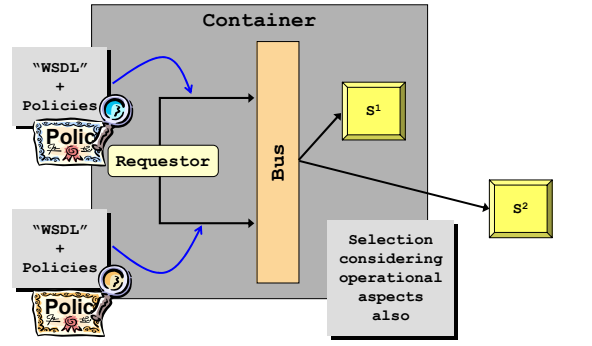
  <my:subscriptionPayment>
    <wsp:PolicyReference URI="#opts"/>
    <wsp:monthly Usage="wsp:Required" />
  </my:subscriptionPayment>

  </wsp:ExactlyOne>
</wsp:Policy>
  
```

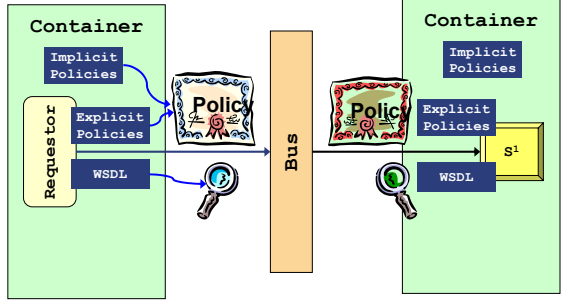
26 Matching Endpoints Based On Policies



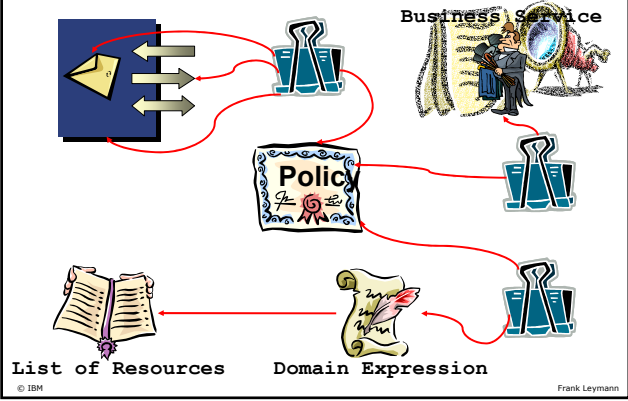
27 Invoking Services Within A Container



28 Matchmaking By Service Bus: Full Picture



29 Attaching Policies



30 Policy Attachment: Example

```
<wsp:PolicyAttachment>  
  <wsp:AppliesTo>  
    <wsp:EndpointReference>  
      <wsp:Address URI="http://www.fabrikam123.com/acct"/>  
    </wsp:EndpointReference>  
  </wsp:AppliesTo>  
  <wsp:PolicyReference  
    Ref="http://www.fabrikam123.com/acct-policy.xml"/>  
</wsp:PolicyAttachment>
```

Everything located at <http://www.fabrikam123.com/acct> (e.g. html-, wsdl-, ... files, ports, ...) is associated with the specified policy

31 Endpoint References: XML Representation

```
<wsa:EndpointReference>  
  <wsa:Address>xs:anyURI</wsa:Address>  
  <wsa:ReferenceProperties> ... </wsa:ReferenceProperties?>  
  <wsa:PortType>xs:QName</wsa:PortType?>  
  <wsa:ServiceName PortName="xs:NCName"?>  
    xs:QName  
  </wsa:ServiceName?>  
  <wsp:Policy/*>  
</wsa:EndpointReference>
```

Service Bus



33 **Dynamic Selection Of Services**

I think I need some help to do my business more efficiently! Who is providing the following functions?

I offer the functions you need! Here is how you can bind to them!

WSDL

© IBM Frank Leymann

34 **Be Flexible!**

I want another business partner that offers the same functions... But one that is cheaper, from Europe, and not just available via e-mail!

{WSDL, Policy}

© IBM Frank Leymann

35 **Don't Care! Virtualizing Services**

I want ... But I don't care about a particular provider: Chose the one who is at this point in time „the best“ for me!

Service Bus

- Environmental Properties
- Business Properties

Virtual Service

© IBM Frank Leymann

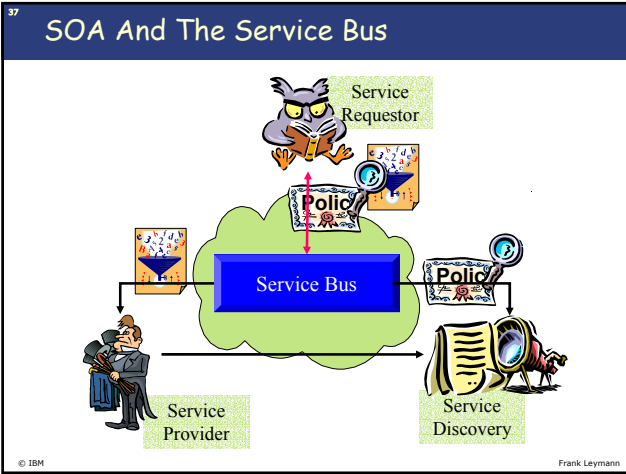
36 **Service Bus: Virtualizing Services**

I need a service of a particular type, but I don't care about the provider of the service: Anyone will do!

Service Bus

{WSDL, Policies, Data}

© IBM Frank Leymann

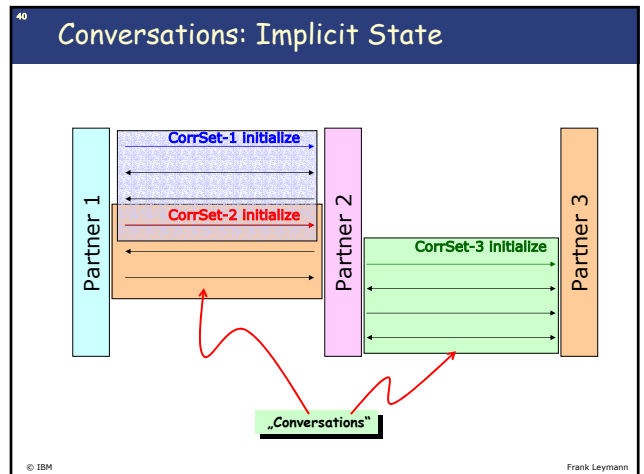


38

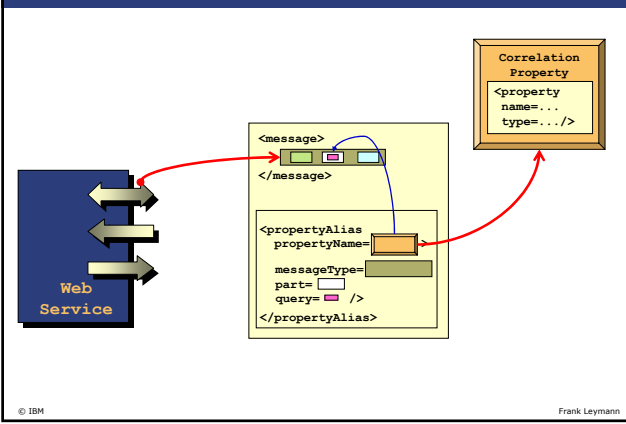
## Web Services And State

© IBM Frank Leymann

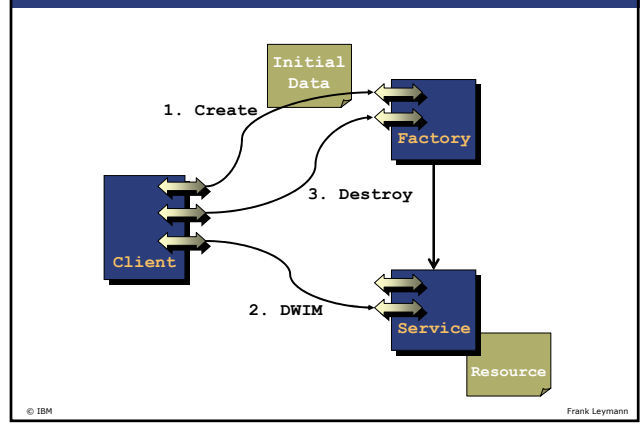
- 39 State And Web Services
- Web service can be stateful
    - Based on ability to support conversations, i.e. long-running interactions like business processes
    - Based on being associated with a resource, having "identity"
  - Implicit state management, e.g. BPEL
    - Correlation properties embedded in application messages exchanged
    - New set of correlation properties implicitly creates new "conversation" (= process instance in case of BPEL)
  - Explicit state management, e.g. OSGI
    - Resources have observable properties and identity
    - Lifecycle functions to manage each resource
      - Especially Softstate management ("leasing")
- © IBM Frank Leymann



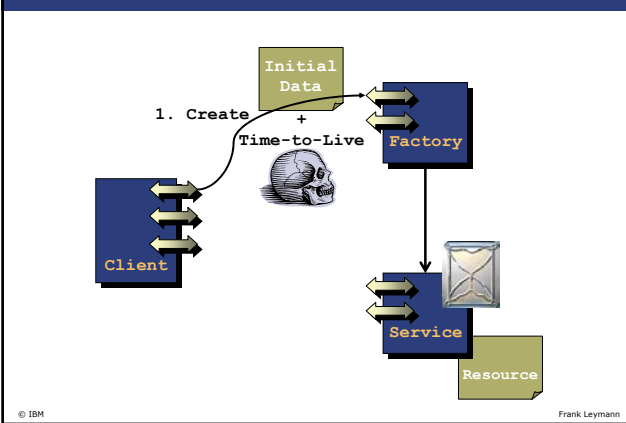
41 Correlation Properties: The Mechanics



42 Factories And Explicit Lifecycle

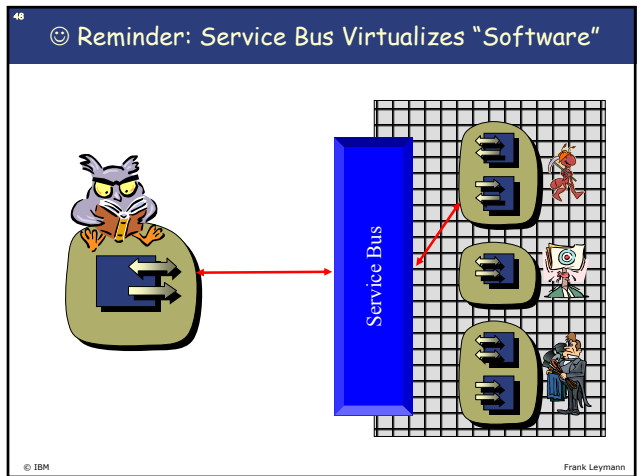
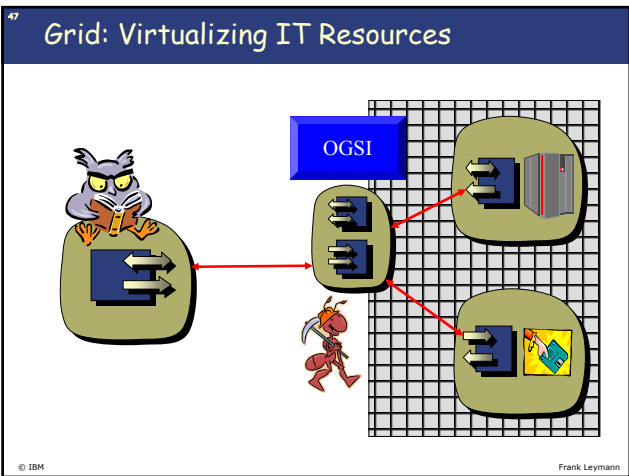
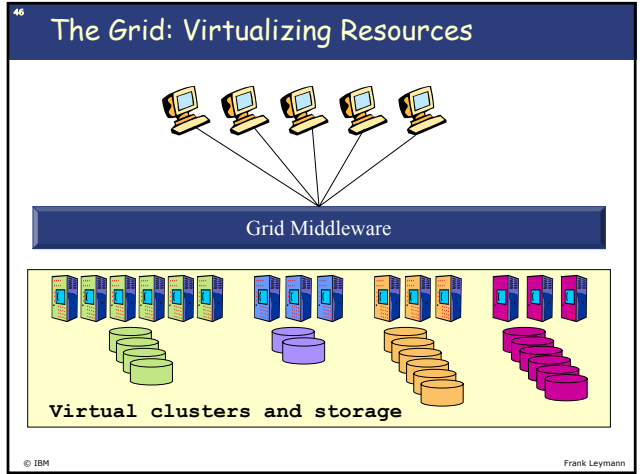
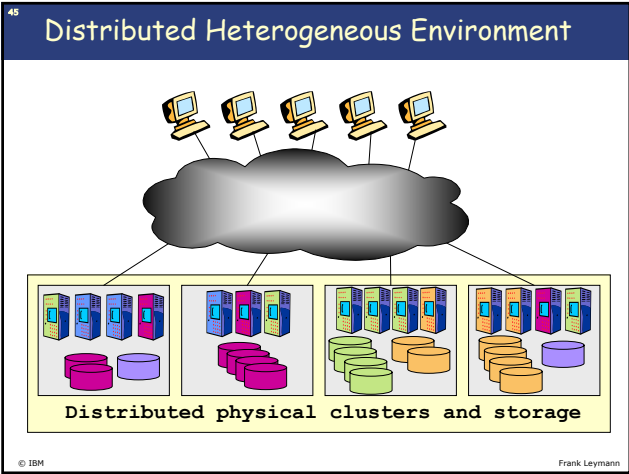


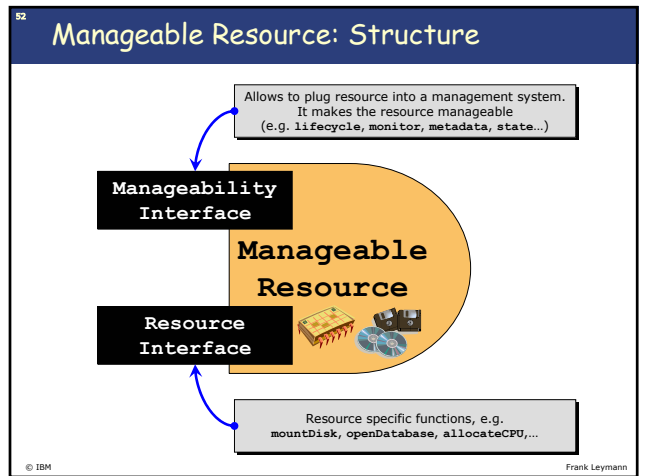
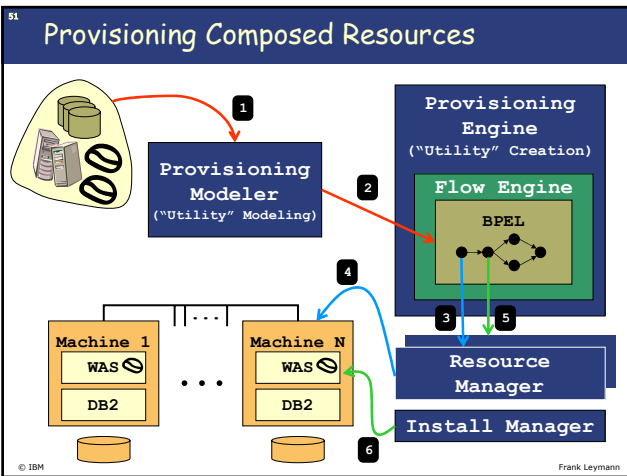
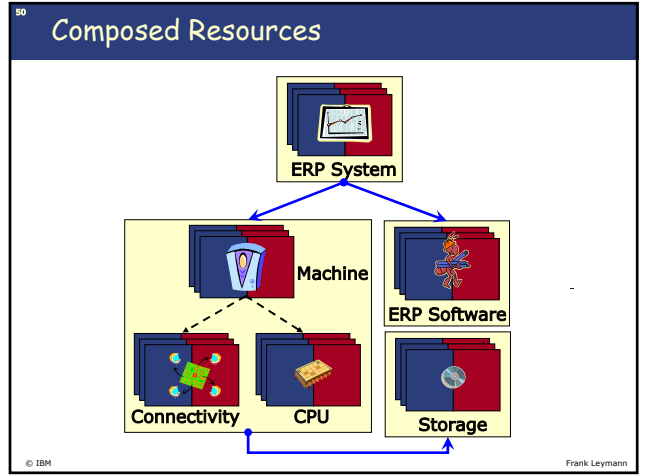
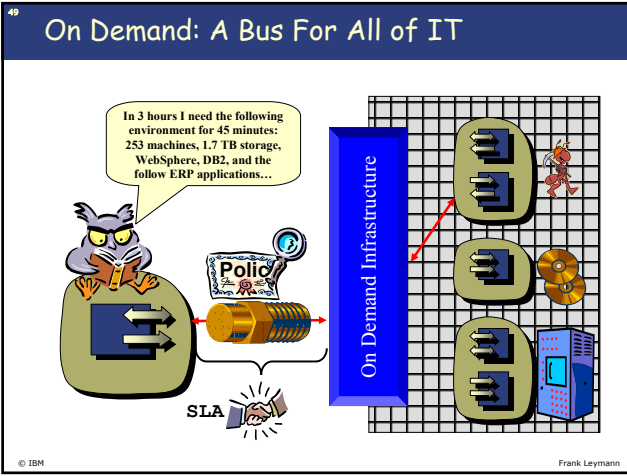
43 Softstate Management

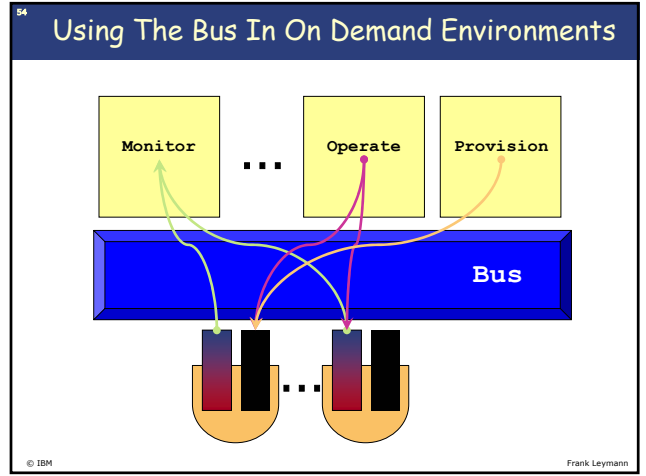
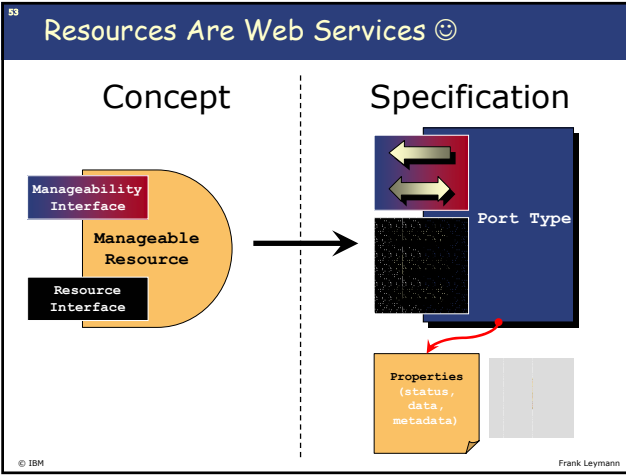


44 Agenda

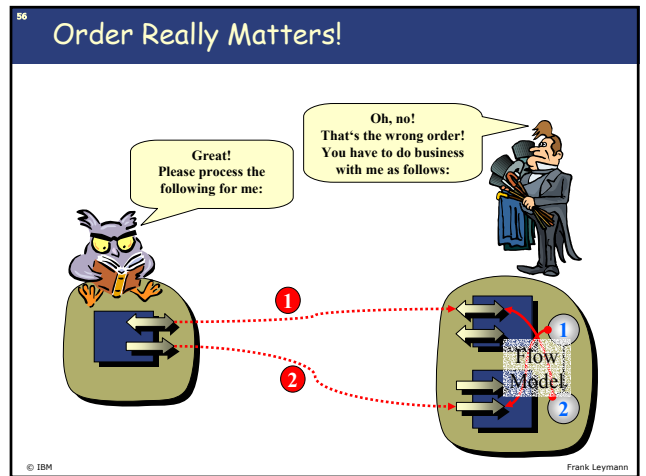
- The Landscape
- Virtual Components
- Virtual Environments**
- Application Structure
- Aggregations
- Summary & Conclusion

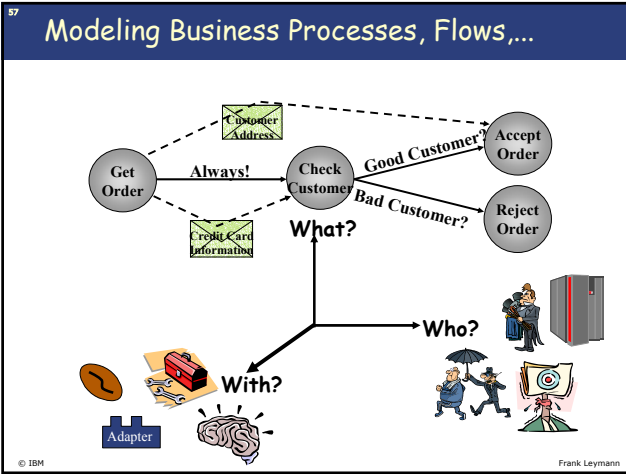




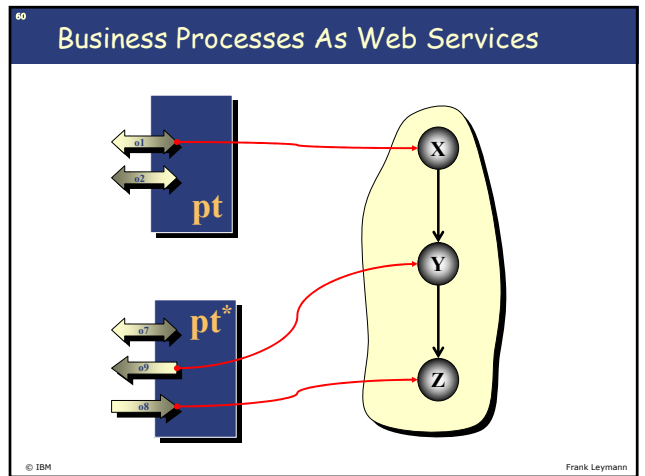
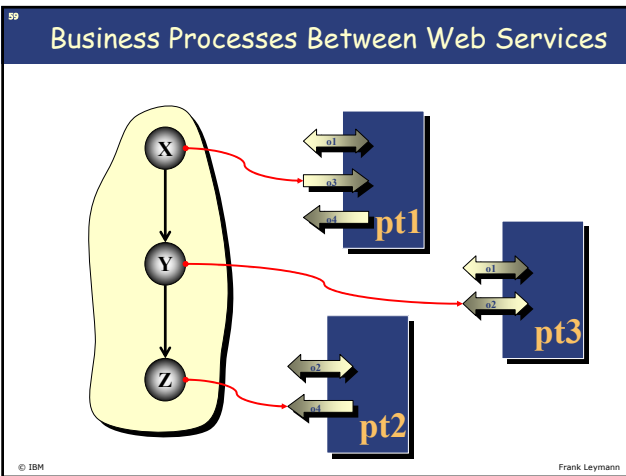


- 55 Agenda
- The Landscape
  - Virtual Components
  - Virtual Environments
  - Application Structure**
  - Aggregations
  - Summary & Conclusion
- © IBM Frank Leymann

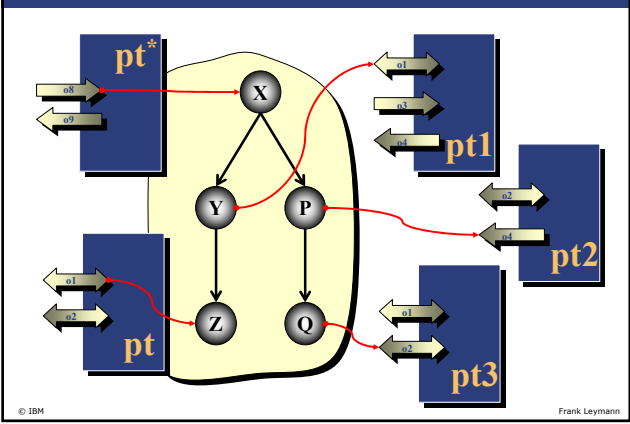




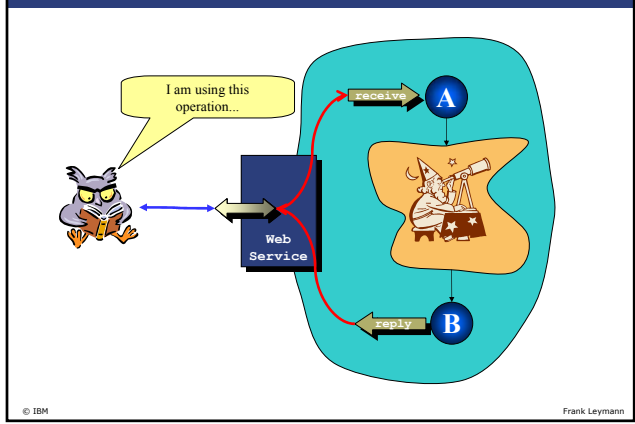
- 58 What Is BPEL
- A language to specify behavior of business processes
    - Between Web services...
    - ...and as Web services
  - Same language to define ...
    - Executable processes
      - Can be performed at all compliant environments
      - Interoperability of heterogeneous environments
    - Abstract processes
      - Specify constraints of message exchange
  - BPEL = Microsoft XLANG + IBM WSFL
    - Combined calculus camp & Petri Net camp
    - Submitted to and standardized within OASIS
- © IBM Frank Leymann



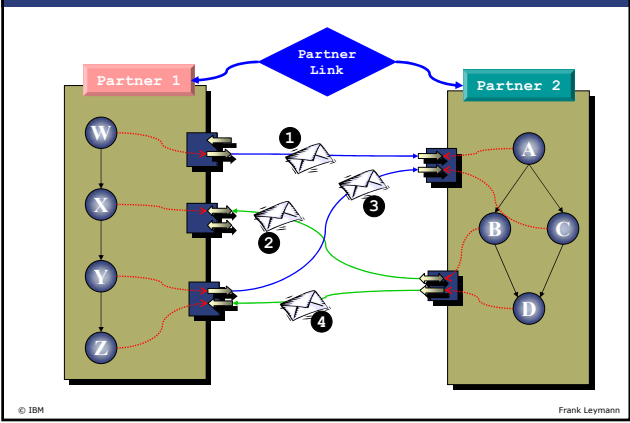
61 Choreography: Aggregating Web Services



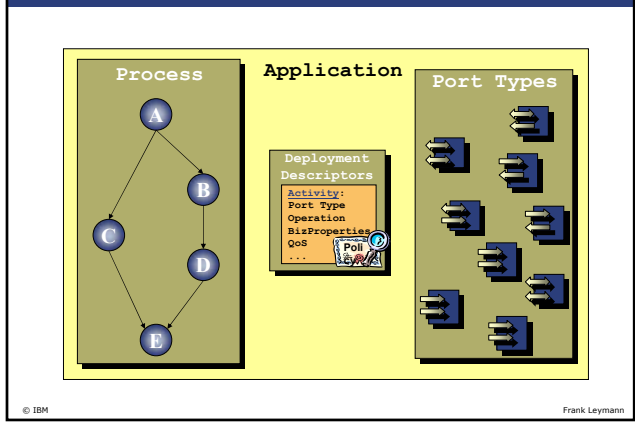
62 Twist: Implementing Web Services



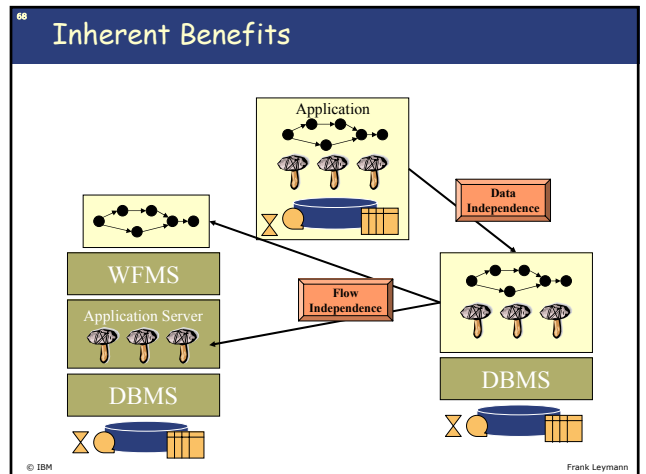
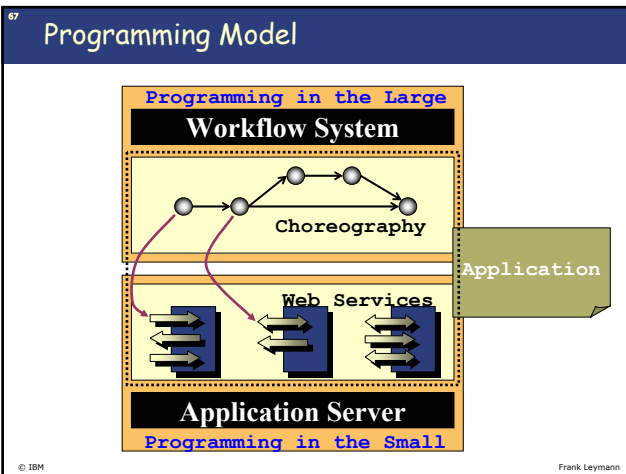
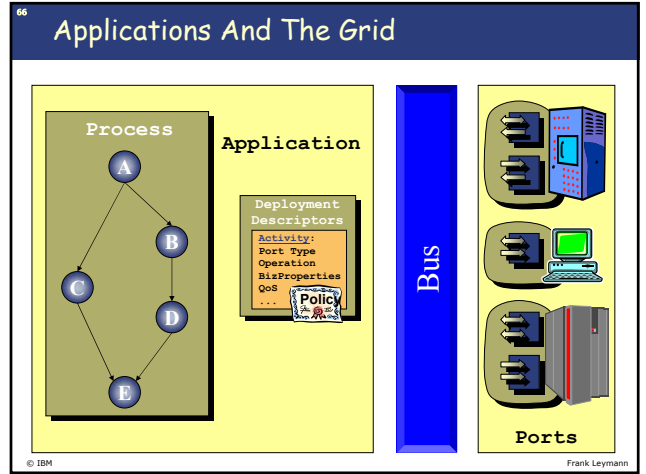
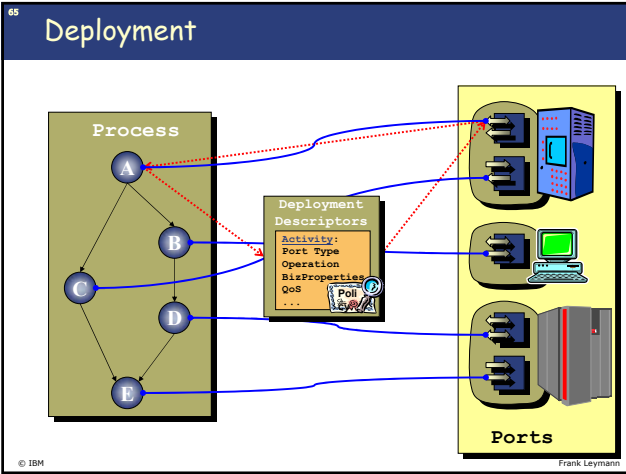
63 Twist: Business Protocols



64 SOA Application Structure







69

## Agenda

- The Landscape
- Virtual Components
- Virtual Environments
- Application Structure
- Aggregations**
- Summary & Conclusion

© IBM Frank Leymann

- 70
- ## Why Aggregation?
1. Value increase:  
Aggregation often has higher value than individual functions
  2. Usability:  
Aggregation often is more convenient to deal with than individual functions
  3. Semantics increase:  
Aggregation mechanism often add semantics to set of individual functions
- Different application areas require different (but composable) mechanisms of aggregation
- © IBM Frank Leymann

71

## Classifying Service Aggregations

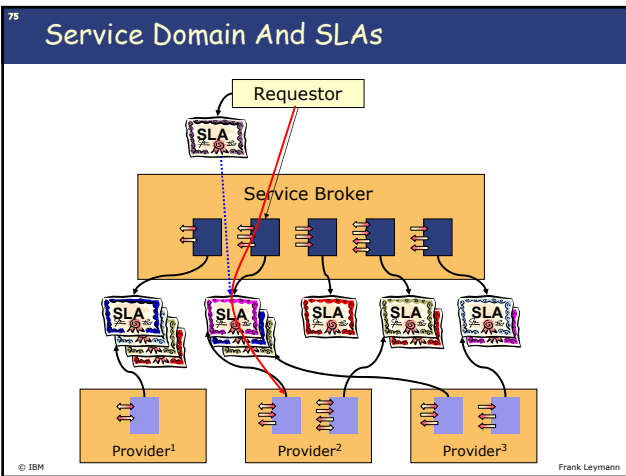
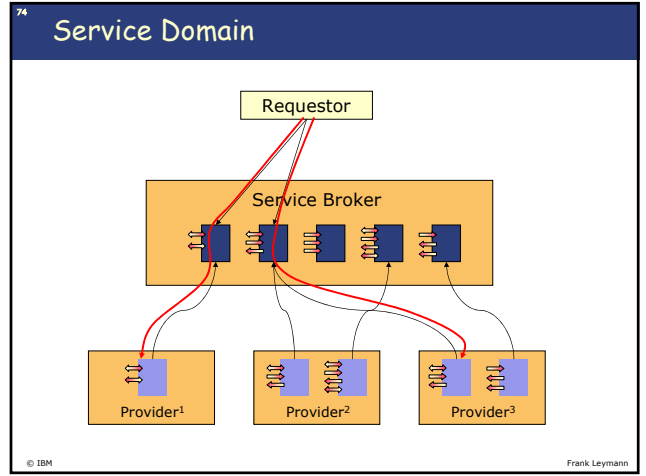
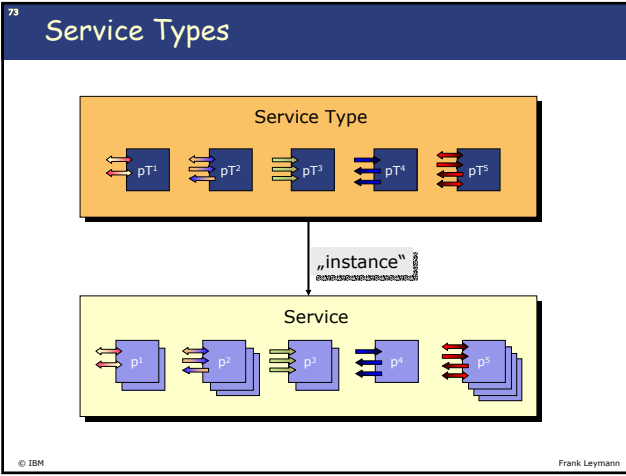
What? How?	Types	Instances
Constrained	Choreography MEP Global Models	Agreement Service Broker
Unconstrained	Inheritance Service Types	Collections

© IBM Frank Leymann

72

# Service Broker

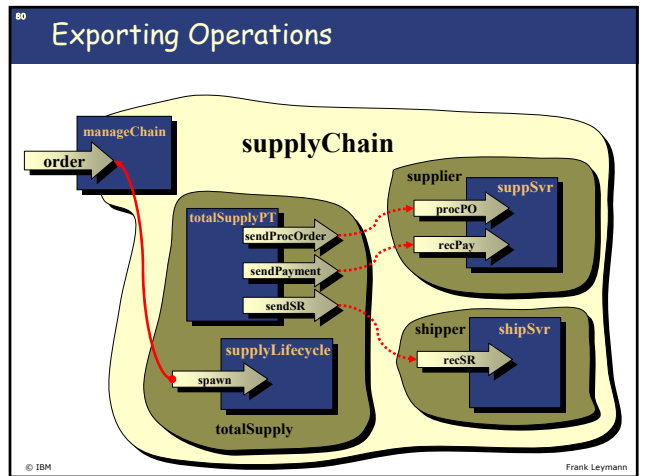
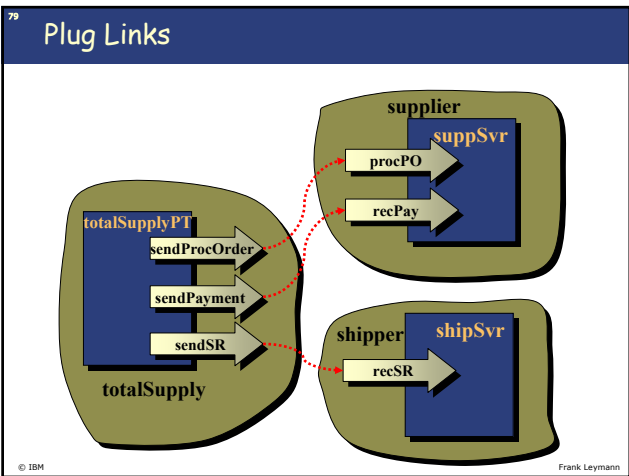
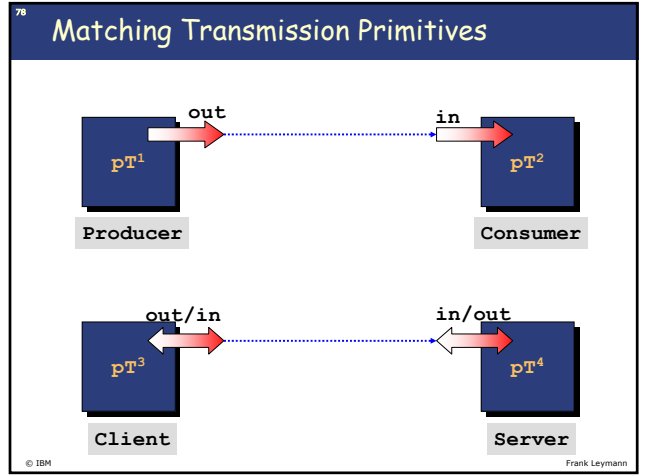
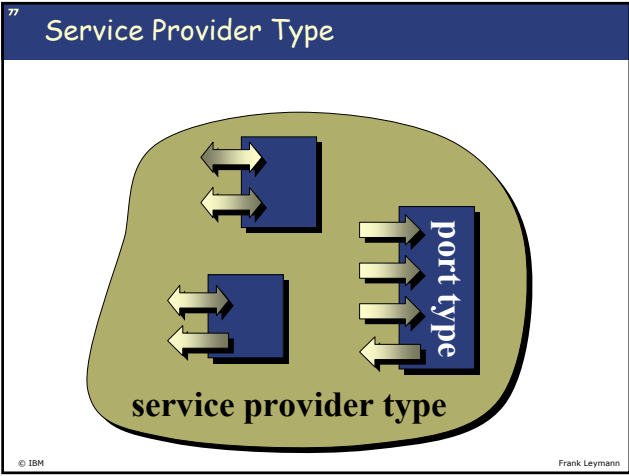
© IBM Frank Leymann



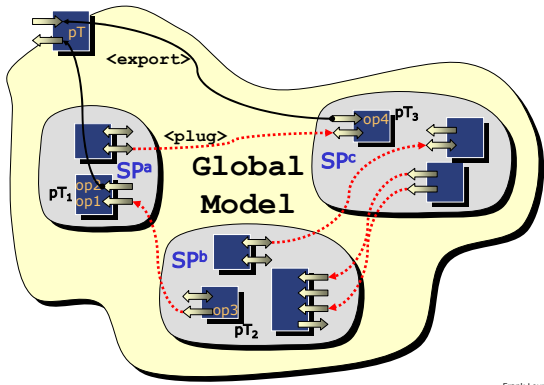
76

**Global Models:  
Recursive Composition**

© IBM Frank Leymann



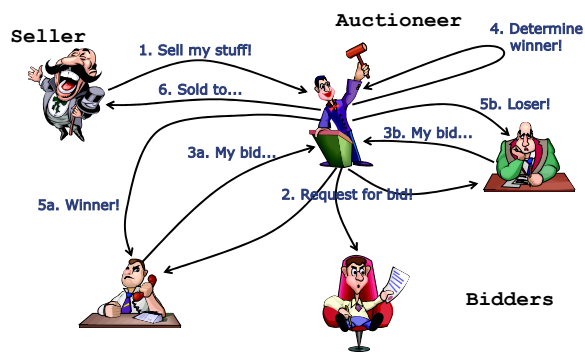
81 Global Models: Aggregation Without Sequencing



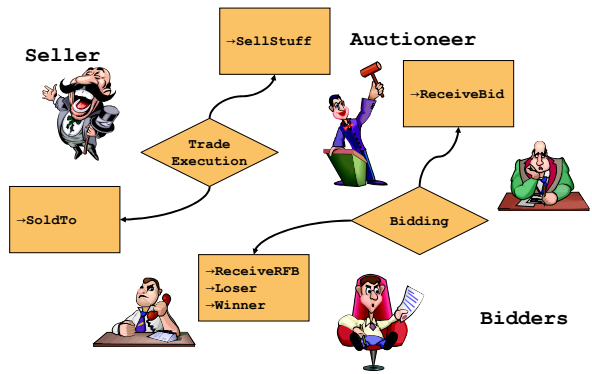
82 Coordination: Out of The Galaxy

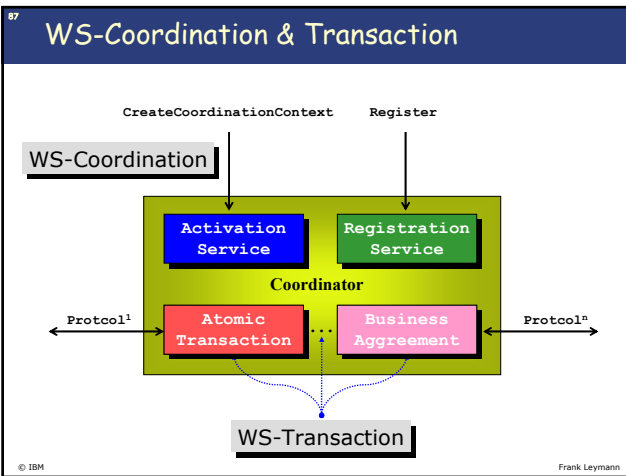
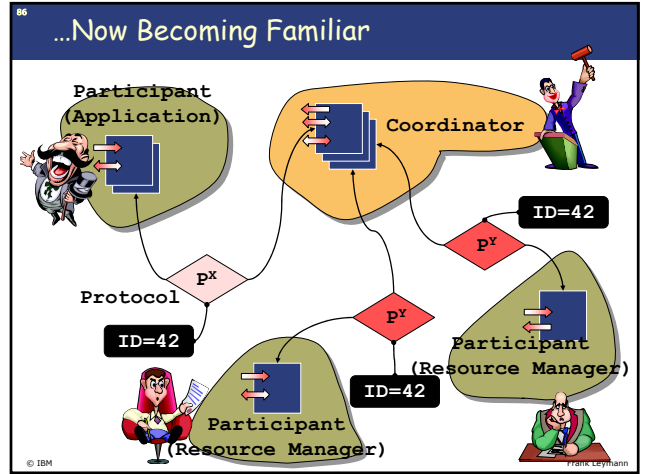
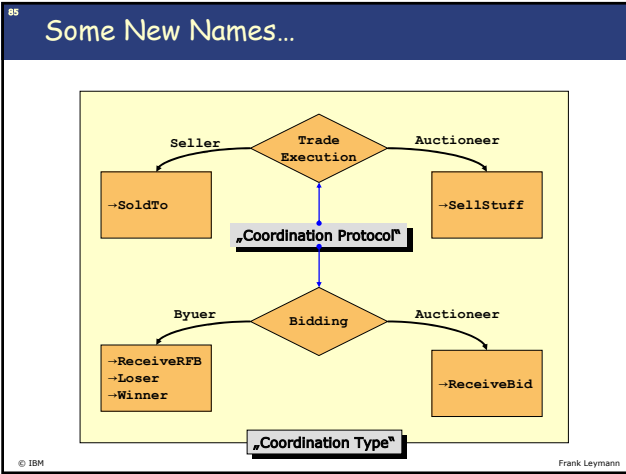
Coordination:  
Out of The Galaxy

83 Agreement In A Distributed Activity

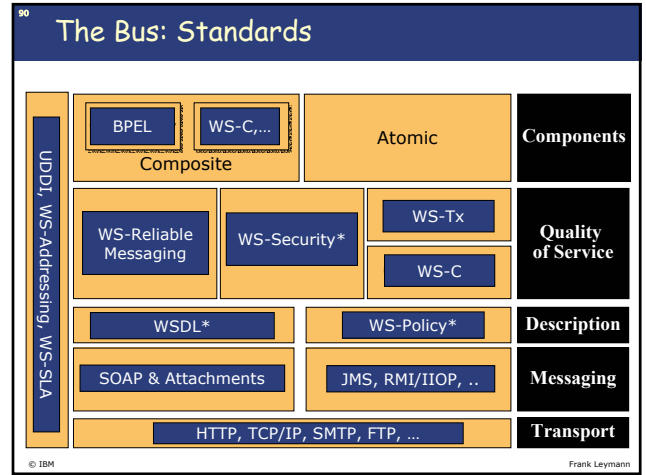
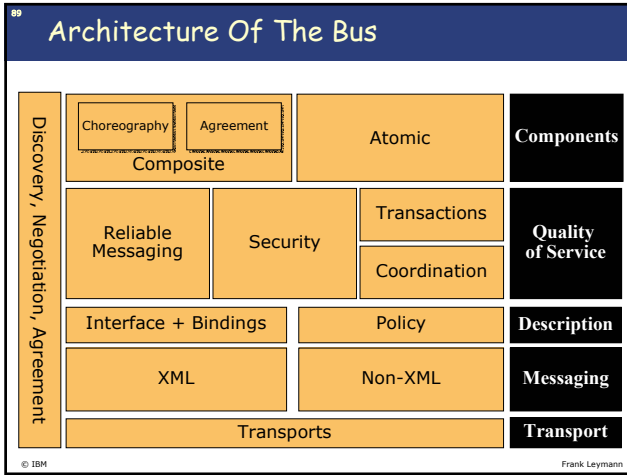


84 ...And The Aggregated Web Services









- 88 Agenda
- The Landscape
  - Virtual Components
  - Virtual Environments
  - Application Structure
  - Aggregations
  - Summary & Conclusion**
- © IBM Frank Leymann



- 91 ...Won't It Take Many Years To Be Ready?!?!
- These standards are build modularized and composable!
  - Combinations of few standards already solve real-world problems
    - Remote access across heterogeneous platforms, integration of application functions,...
  - This allows a phased roll-out and usage
- © IBM Frank Leymann

- 92 ...And This Is All Web-Ready?!?!
- I didn't say anything about the Web!
  - Web services offer profound benefits even within a single domain (i.e. enterprise, division, location, department,...)
    - Heterogeneity (platforms), interoperability (messages,...), integration (functions, business processes,...),...
  - Many problems discussed are less complex or go away in such settings
    - Security, dynamic discovery, payment,...
- © IBM Frank Leymann

## Why Should It Work Out This Time?!?!

- I am not a fortune teller...    
- Unanimous agreement in the industry
  - Vendors, analysts,...
  - Broad commitment to make it work (well, I know... 😊😊)
- WS-I (Web Services Interoperability)
  - Consortium to ensure interoperability
    - Specification of so-called *profiles* ("regulation" of how and what to use, "exegesis" of standards)
    - Tools (test, analysis, monitor)
    - Sample implementation

## Conclusion

- Huge momentum in industry on Web services
  - Standards, interoperability, implementation
- Web services are the base for the new evolving distributed computing platform ("The Bus")
  - Virtualizing components
  - Heterogeneous "ab ovo"
- Application model is centered around...
  - SOA – Policy driven dynamics
  - Choreography – Two-level programming
- Although middleware begins to roll-out, tremendous amount of research in this space must be done

# Thank You!