

Tutorial at VLDB 2001
(Rome, Italy, September 11-14, 2001)

Managing Business Processes Via Workflow Technology



Frank Leymann
IBM Distinguished Engineer
Member, IBM Academy Of Technology


IBM Software Group
Schoenaicherstr. 220
71032 Boeblingen
Germany

Phone +49-7031-16 3998
Fax +49-7031-16 4890
Cellular +49-172-731 5858
e-mail ley1@de.ibm.com

Copyright Notice:

All of the material is copyrighted.


With a few exceptions, the copyright
is help by IBM or Prentice Hall, Inc.



Agenda

1. History
2. Workflow Basics
3. Some Plumbing
4. WFMS Architecture
5. Transactional Workflows
6. Application Structures
7. Web Services

© IBM Frank Leymann



Agenda

- 1. History**
2. Workflow Basics
3. Some Plumbing
4. WFMS Architecture
5. Transactional Workflows
6. Application Structures
7. Web Services

© IBM Frank Leymann

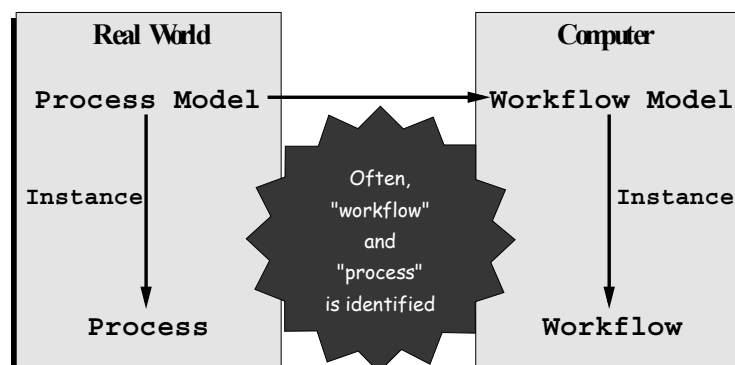
Why Do Care About Workflow Technology?

- Companies use computers to support their business, most frequently
 - The way to do business is prescribed via a business process, very often
 - Applications support business processes and have to ensure compliance with business processes
- => Application = Business Process + Business Functions
- Changes in how to perform business must be reflected as soon as possible in applications
 - A workflow is a business process in execution (an instance of a process model) in a computing environment

© IBM

Frank Leymann

Processes And Workflows




Not all parts of a process are run
in a computing environment -
some processes are not run on a computer at all!

© IBM

Frank Leymann

"The Business You Are In Determines What Your Business Processes Are!"

- ✕ Manufacturing
 - ✓ Assembly lines of cars, PCs, cloths,...
- ✕ Insurance
 - ✓ Handling of claims, policies,...
- ✕ Finance
 - ✓ Stock brokering, settlement, clearing,...
- ✕ Banking
 - ✓ Loans, savings, current accounts,...
- ✕ Database administration
 - ✓ Backup & recovery, reorganization, tuning,...
- ✕ Software development
 - ✓ Waterfall model, spiral model,...
- ✕ Telecommunications, administration, government, data warehousing...



There is nothing like a "typical business process"!!!!

© IBM

Frank Leymann

People Workflow Evolution: 1st Generation

- ✕ Electronic document and folder routing (late 80s)
 - ✓ Document = image, folder,...
 - ✓ Routing through enterprise's organizational structure
 - ✓ User associated electronic basket is key
 - Container for documents a certain user has to work on to contribute to a case
 - ✓ Potential flow of documents prescribed in advance
 - Routing conditions in terms of document content or document properties
 - Actual routing based on actual content or properties of subject document
- ✕ In "paper factories" (administration, insurance, banking,...) work mainly equates to processing documents, thus the term

workflow

has been used for routing documents between people

© IBM

Frank Leymann

People Workflow Evolution: 2nd Generation

- ✗ Functions performed by users in 1st generation WFMS are mainly retrieval, browsing, editing, archiving,...
- ✗ But cases represented by documents were recognized to be only part of larger business processes
 - ✓ Not only performance of document management functions required but also usage of other functions provided by application systems supporting the operation of an enterprise
- ✗ WFMS extensions needed to invoke any kind of executable
- ✗ In-/Out-Basket grew towards worklists
 - ✓ Launch-pad for executables
 - ✓ Workitem management
 - Prioritization, duration management, life-cycle,...

© IBM

Frank Leymann

People Workflow Evolution: 2nd Generation (cont.)

- ✗ Launching executables requires parameter passing
- ✗ Thus, data flow features complemented available control flows
- ✗ In turn, control flows can now be expressed in terms of these new parameters ("business rules")
- ✗ Data flow is used for integrating applications with long temporal delays between their initiations
 - ✓ Parameters managed by data flow must be persistent
 - ✓ Data flow must be allowed to be different from control flow
 - Data produced by application A might be used by application B to be started after a couple of intermediate applications run

© IBM

Frank Leymann

People Workflow Evolution: 2nd Generation (cont.)

- ✗ Being able to support large spectrum of business processes in computing environments made WFMS of strong interest for Business Process Reengineering (BPR) projects - early 90s
- ✗ Goal of BPR is to speedup business processes and reduce their costs. Resulting requirements:
 - ✓ Parallelism in workflows (-> speedup)
 - ✓ Deadline processing (-> speedup)
 - ✓ Monitor actual workflow status (-> speedup)
 - ✓ Auditing of significant events, i.e. processing history (-> cost reduction)
 - ✓ Maintain execution history for analysis (-> cost reduction)
 - ✓ Process activities without human intervention (-> speedup + cost reduction)
 - So-called automatic activities
 - Consequence: (parts of) business processes can be automated ("macro-scripts")

© IBM

Frank Leymann

People Workflow Evolution: 3rd Generation

- ✗ Workflow-based applications become state-of-the-art (mid 90s)
 - ✓ Strict separation of business process logic and business functions
 - Business processes implemented via workflow system
 - Business functions implemented "traditionally" (TP-monitor, ORB,...)
- ✗ Enterprises become dependent on WFMS
 - ✓ Similar to TP-Monitors and DBMS before
 - ✓ The term **production workflow** has been coined to indicate that WFMS is driving operational aspects of an enterprise
- ✗ Consequences:
 - ✓ WFMS had to provide quality of services known before from "production systems" like DBMS and TPM
 - High/continuous availability
 - Scalability
 - Robustness

© IBM

Frank Leymann

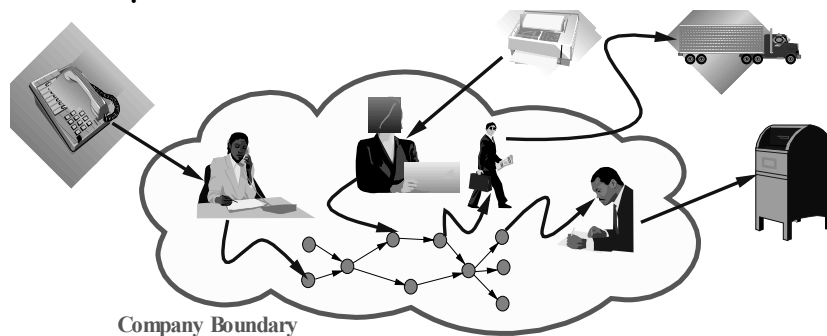
People Workflow Evolution: Latest Moves

- ✕ **Application integration is of fundamental importance**
 - ✓ Integrate diversity of application functions
 - legacy applications, newly written applications (e.g. component based),...
 - new invocation paradigms (e.g. message queuing, pubsub)
 - workflows as granules to be integrated
 - automatic workflows
- ✕ **Organizational integration becomes more and more important**
 - ✓ Workflow expand across business units of enterprise ("intra-enterprise")
 - ✓ Workflows across enterprises become key for B2B ("inter-enterprise")
 - Creation and enactment of workflows in virtual enterprises
 - Stimulated by mergers and acquisitions, outsourcing, supply chains, e-market places,...
 - New technologies like Web Services, UDDI, SOAP, ... stimulate this
- ✕ **Workflows understood as business oriented "logical units of work"**
 - ✓ Advanced transaction management functions required
 - ✓ Forward recovery of workflows as well as workflow-based applications
 - ✓ Backward recovery (global transactions and compensation-based recovery)

© IBM

Frank Leymann

Encapsulated Workflows



- ✕ Company's personnel "translate" requests/responses with the outside into actions performed within workflows
- ✕ Inquiries about status usually via phone calls
 - ✓ Call center agents receive requested information
 - ✓ Limited service to customers & suppliers (e.g. restricted service hours,...)

© IBM

Frank Leymann

Opening Up Workflows

- ✗ Customers invoke company's applications to perform certain steps of the business process
 - ✓ E.g. place an order, inquire status,...
 - ✓ Company's applications must get a browser-based front-end for that purpose ("web-up")
- ✗ Workflow activities may directly communicate with the outside
 - ✓ Send e-mail, faxes, messages,...
- ✗ Workflow activities may trigger actions in another company
 - ✓ Simple invocation of program or start of another workflow ("subprocess" from invokers point-of-view)
 - ✓ Such "business-to-business" scenarios are the base for realizing sophisticated "supply chains"

© IBM Frank Leymann

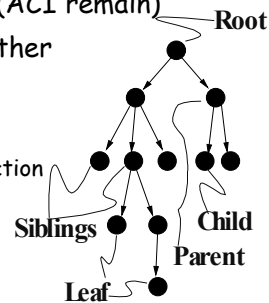
Transactional Workflow Evolution

- ✗ Success of TP Monitors and concept of (classical) transactions have been overwhelming
- ✗ Hidden assumption behind classical transactions:
 - ✓ Short duration (fractions of a second to a few seconds)
- ✗ Technical underpinnings based on this assumption
 - ✓ 2-phase-locking, log based recovery,...
- ✗ Early 80s started to extend transaction technology towards longer durations
 - ✓ Technical underpinnings have to be adopted
- ✗ Most famous "transaction models"
 - ✓ Nested transactions (closed & open)
 - ✓ Sagas
 - ✓ Multilevel transactions

© IBM Frank Leymann

Transactional Workflow Evolution: Nested Transactions

- ✗ Structure transaction into a tree of subtransactions
- ✗ Allow intra-transaction parallelism to speedup processing:
siblings may run concurrently
- ✗ Overall nested transaction has ACID properties
- ✗ Durability of subtransactions are given up (ACI remain)
- ✗ Overall nested transaction isolated from other
nested transactions ("closed")
- ✗ Result
 - ✓ Possible speedup of a single closed nested transaction
 - ✓ Moderate throughput increase of environment



© IBM

Frank Leymann

Transactional Workflow Evolution: Open Nested Transactions

- ✗ Open nested transactions give up isolation and to a
certain degree atomicity
- ✗ Subtransactions commit their changes to the outside as
soon as they commit
- ✗ Consequence:
Recovery via restoring before-images does not work any
more
- ✗ Already performed subtransactions of an aborting root
must be undone by running application specific logic
("compensation action")

© IBM

Frank Leymann

Transactional Workflow Evolution: Sagas

- ✗ Open nested transactions assumed that compensation actions are scheduled manually
- ✗ Sagas require to specify compensation actions in advance and run them automatically on abort

Definition:

A Saga is a sequence $[(T_1, C_1), \dots, (T_n, C_n)]$ having the following properties:

1. T_1, \dots, T_n and C_1, \dots, C_n are two sets of transactions, such that C_i is the compensation function for T_i ,
2. $[(T_1, C_1), \dots, (T_n, C_n)]$ is executed as one of the following sequences:
 - i. $[T_1, \dots, T_n]$, if all T_i committed, or
 - ii. $[T_1, \dots, T_i, C_{i-1}, \dots, C_1]$ if T_i aborts and T_1, \dots, T_{i-1} committed before.

© IBM

Frank Leymann

Transactional Workflow Evolution: Structures

- ✗ Structures of transactions have been extended from sequences and trees to directed acyclic graphs
 - ✓ Dependencies between transactions are described
- ✗ Backward recovery based on ACID semantics as well as compensation has been folded in
 - ✓ E.g. "ConTracts"
- ✗ Late 80s, early 90s:
The term "**transactional workflow**" has been coined for prescribing control flow dependencies between transactions and their joint backward recovery

© IBM

Frank Leymann

Transactional Features of Production Workflows: Merging People Workflow & Transactional Workflow


- ✗ Production workflow have the following characteristics:
 - ✓ Many executables invoked are
 - classical transactions
 - run automatic (i.e. launched as soon as detected to be performed)
 - run unattended (i.e. no interactions with human beings)
- ✗ Thus, today's workflow systems impose directed graph structures on set of transactions as discussed for "transactional workflows"
- ✗ It is only natural that users now require "transactional workflow features" within production workflow systems

© IBM

Frank Leymann

Transactional Features of Production WF (cont.)

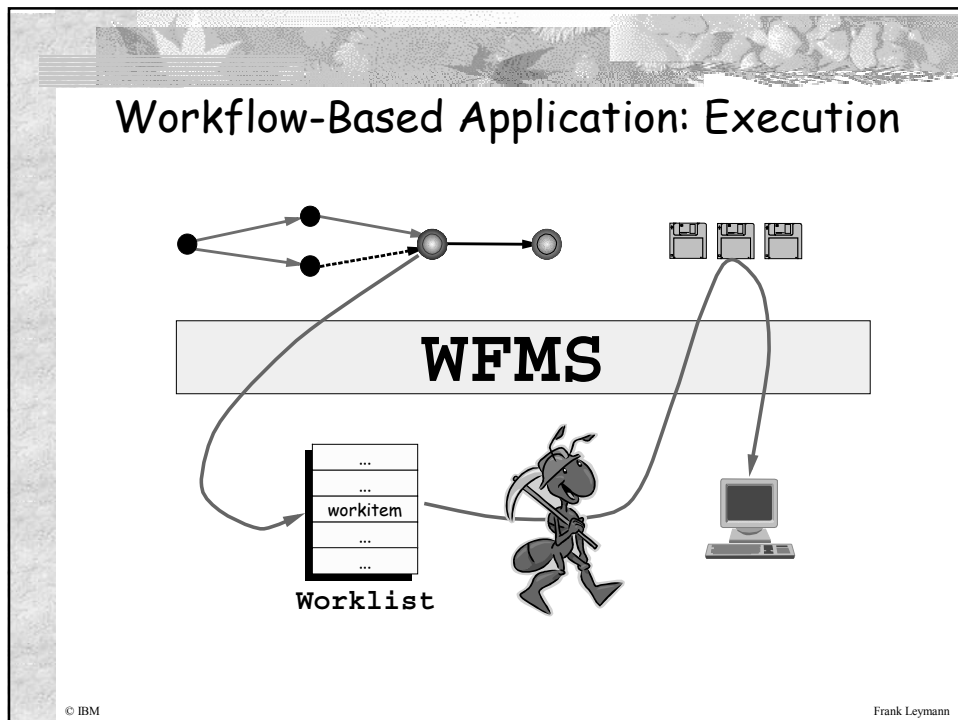
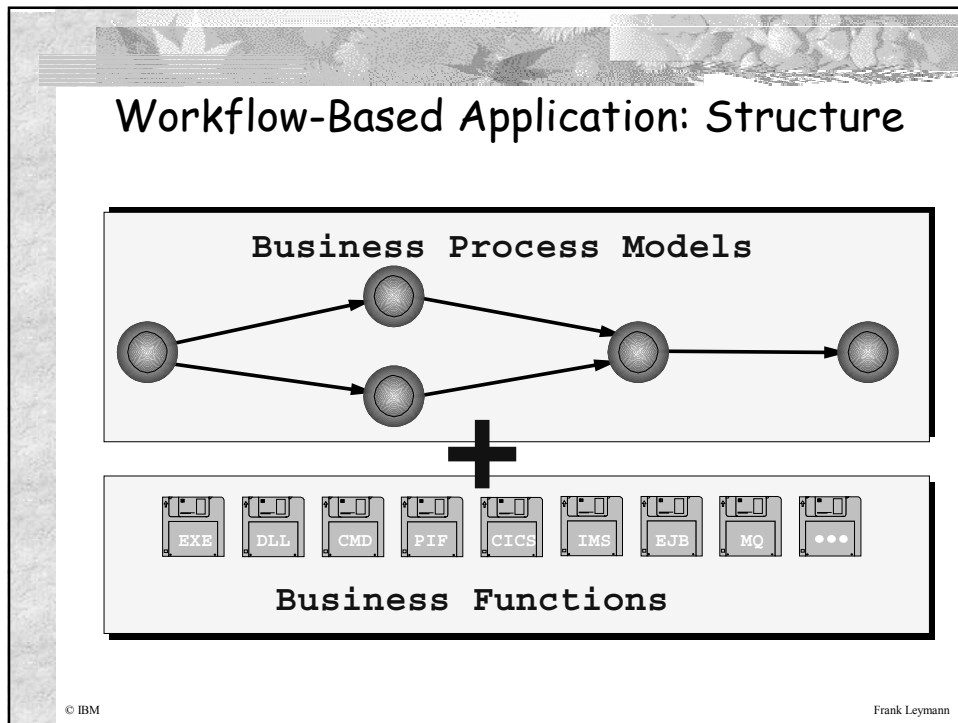
- ✗ Production workflows invoke a lot of non-transactional programs too (i.e. programs that cannot be simply undone)
- ✗ Thus, supporting compensation based recovery in production workflow systems is only natural
- ✗ Especially, a "unit of work" must allow to include
 - ✓ transactional as well as non-transactional programs
 - ✓ long running programs
 - ✓ programs that demand human interactions
- ✗ Ability to involve people in recovery:
 - ✓ In exceptional situations people can be notified as part of recovery processing
 - ✓ Human beings might "repair" the exceptional situation allowing to continue processing



Not focus of transactional workflow area

© IBM

Frank Leymann



WF-Based Apps: The Role Of Business Processes

- ✗ Very important to understand: **Product = Process**
from an internal company point of view in many industries
 - ✓ E.g. finance (settlement, credit,...), insurance (policy, claim,...),...
- ✗ Consequence: Time to create/modify business processes equates time to market for new/modified products
- ✗ Thus: Competitiveness of company depends on this time
- ✗ Business process represents rules of procedure
 - ✓ Often optimized wrt time & costs
- ✗ Thus: Process participants must precisely follow specifications
- ✗ Workflow-based application
 - ✓ flexibility: Creation and modification of business functions independent from specification of business processes
 - ✓ enforcement: Workitems scheduled exactly as defined by process model

© IBM

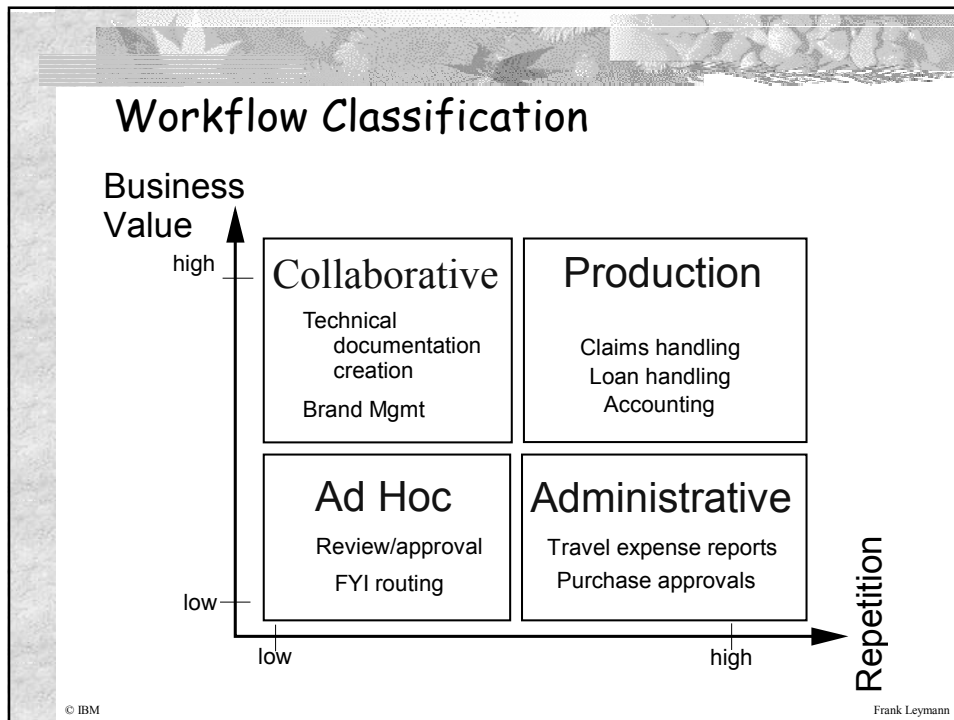
Frank Leymann

WF-Based Apps: Industry Acceptance

- ✗ Large companies adopted this paradigm in the early 90s
 - ✓ Built their own workflow systems at that time
 - No real production workflow system was available
 - ✓ Benefits: Time to market for new/modified products
- ✗ Standard application vendors adopted this paradigm mid 90s
 - ✓ Most vendors built their own workflow system because no system dominated the market
 - ✓ Benefits: Customization and internationalization
- ✗ Standardization started mid 90s
 - ✓ Workflow Management Coalition (WfMC) since 95
 - The standard consortium for workflow standards since 99
 - ✓ OMG's Workflow Management Facility = Objectification of WfMC I/Fs
 - ✓ Since 2000: ebXML, BPMI.org, OMG process modeling,...
- ✗ Vendors role out production workflow systems since late 90s
 - IBM MQSeries Workflow, HP ChangEngine, MS Orchesterator,...

© IBM

Frank Leymann



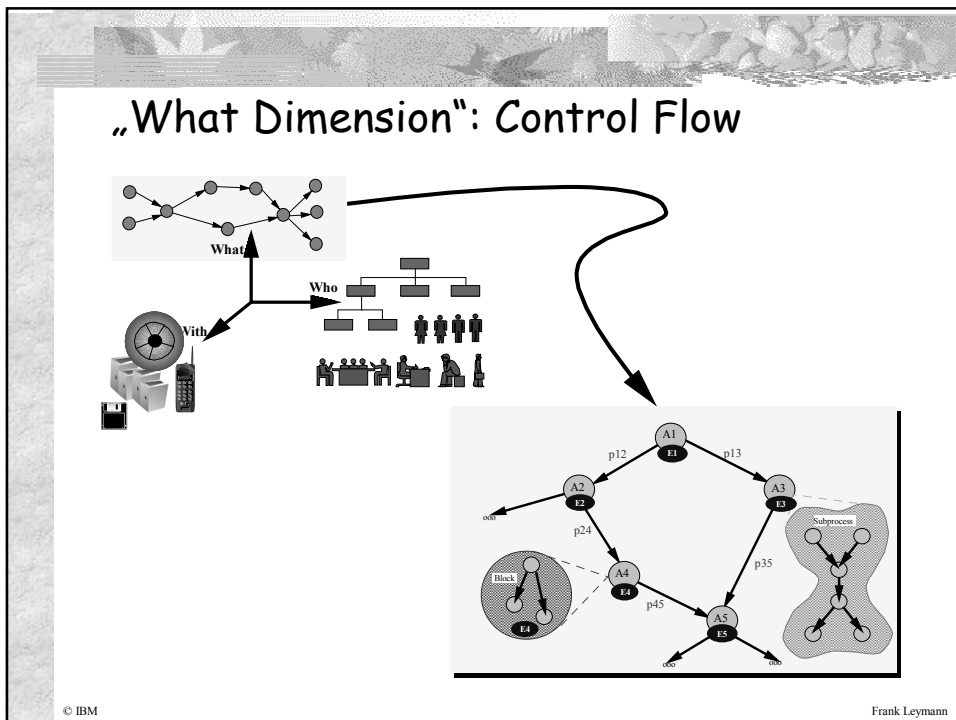
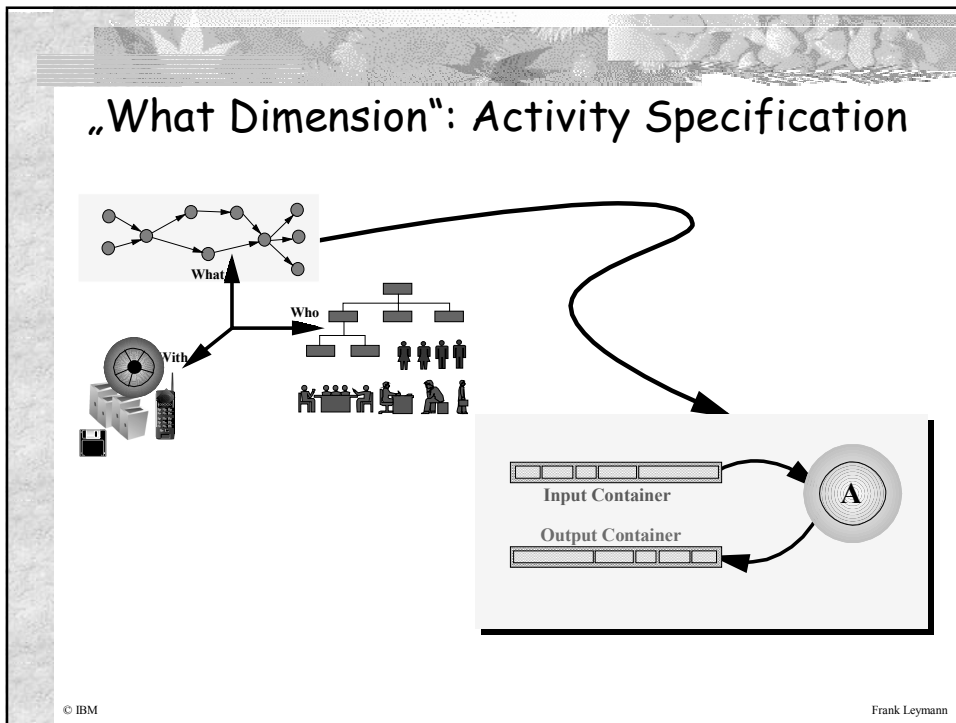
- ## Agenda
1. History
 - 2. Workflow Basics**
 3. Some Plumbing
 4. WFMS Architecture
 5. Transactional Workflows
 6. Application Structures
 7. Web Services
- © IBM Frank Leymann

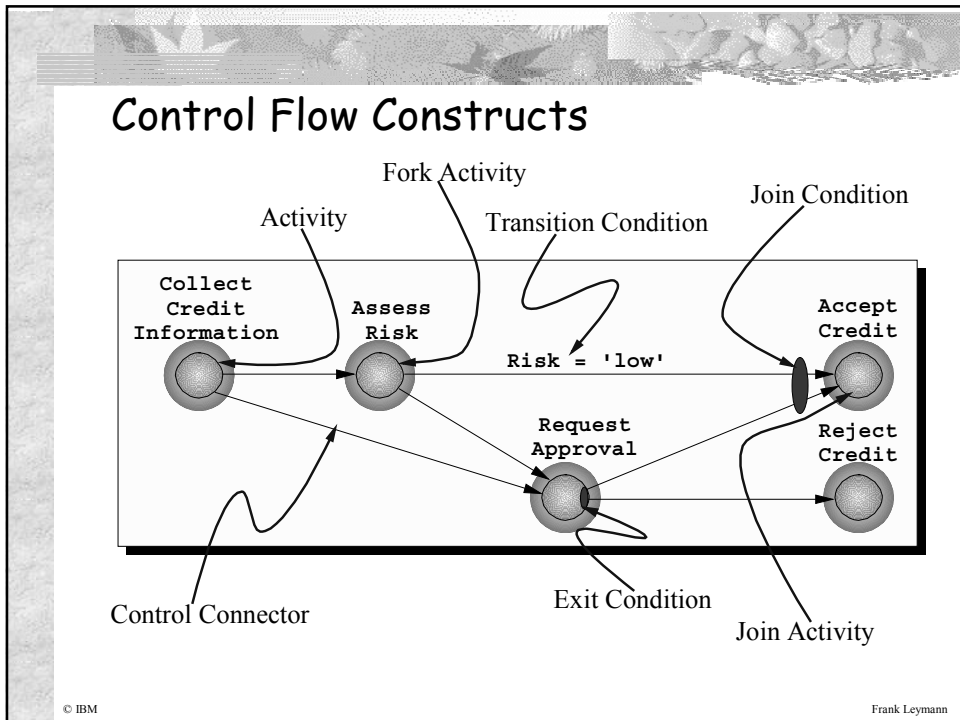
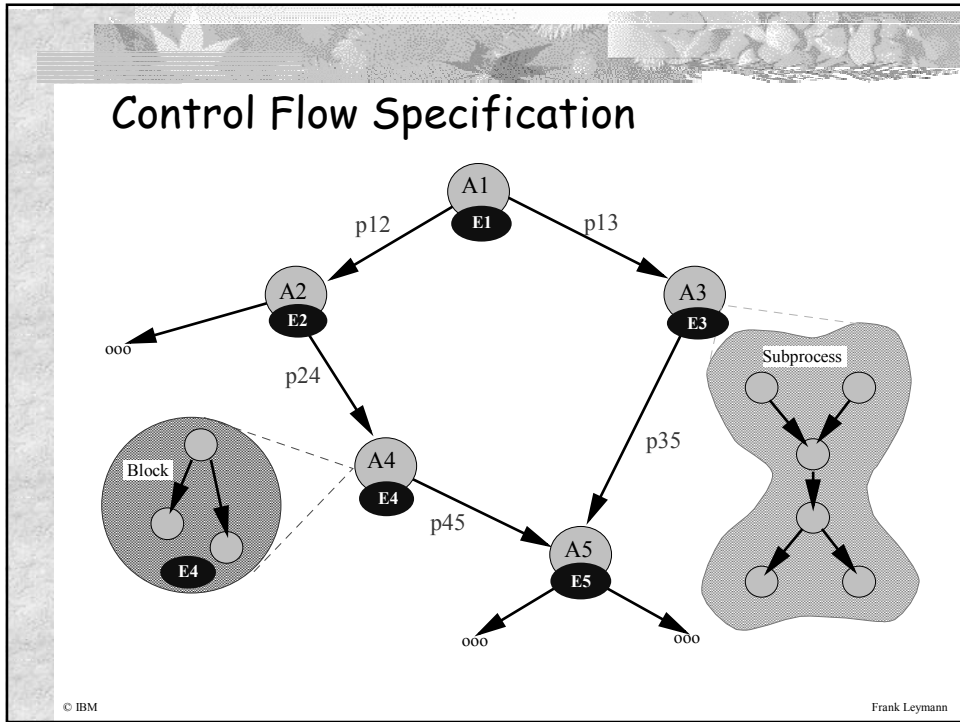
What Are Workflows?

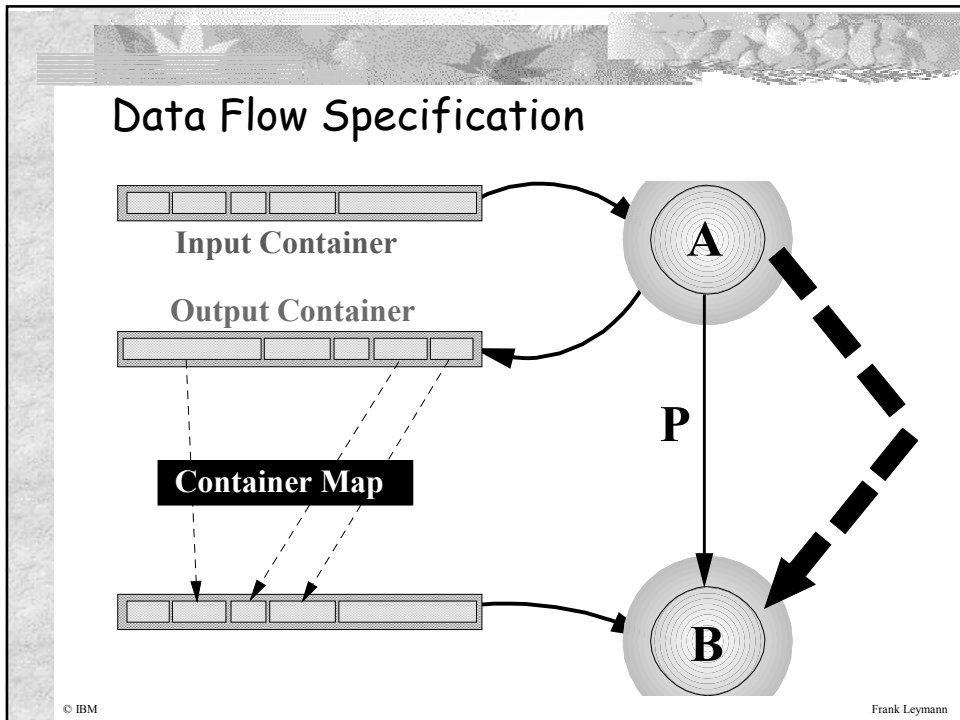
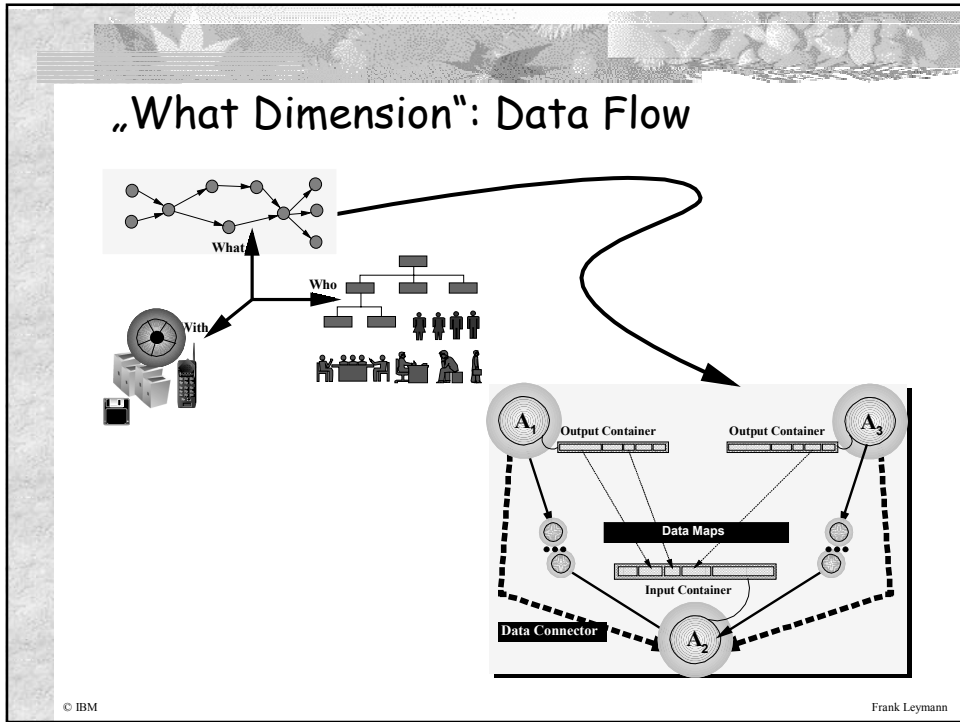
© IBM Frank Leymann

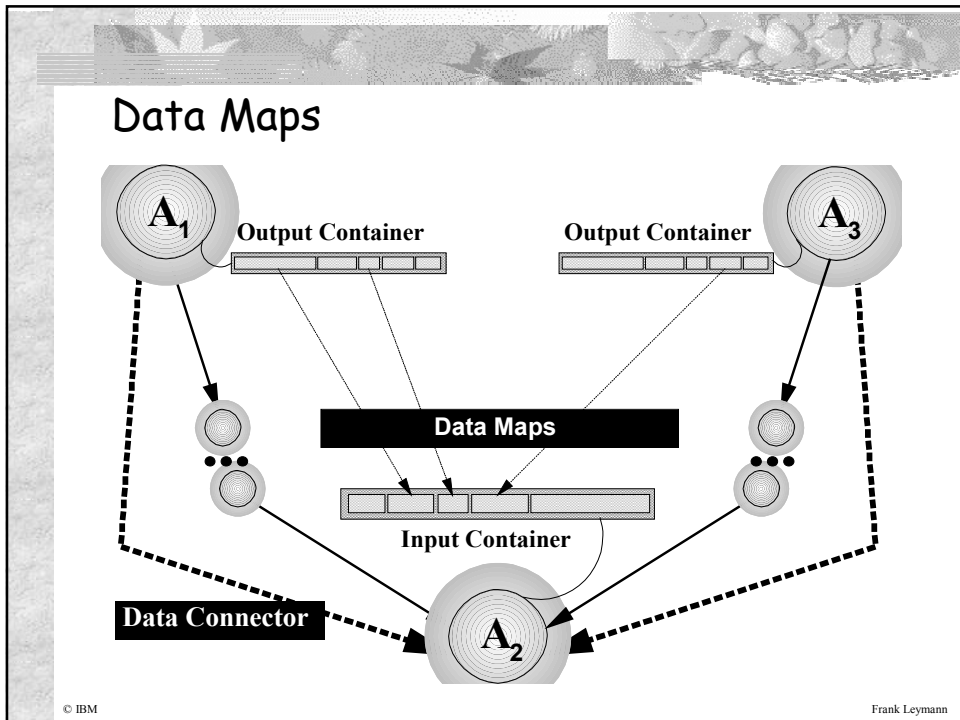
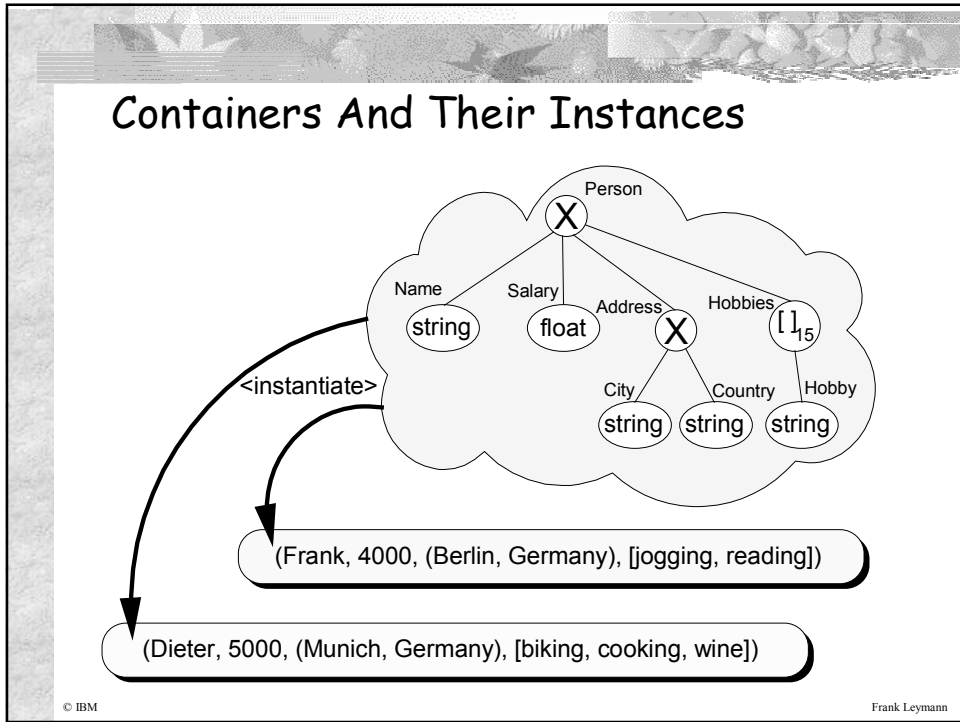
Workflows: N-Dimensional „Things“

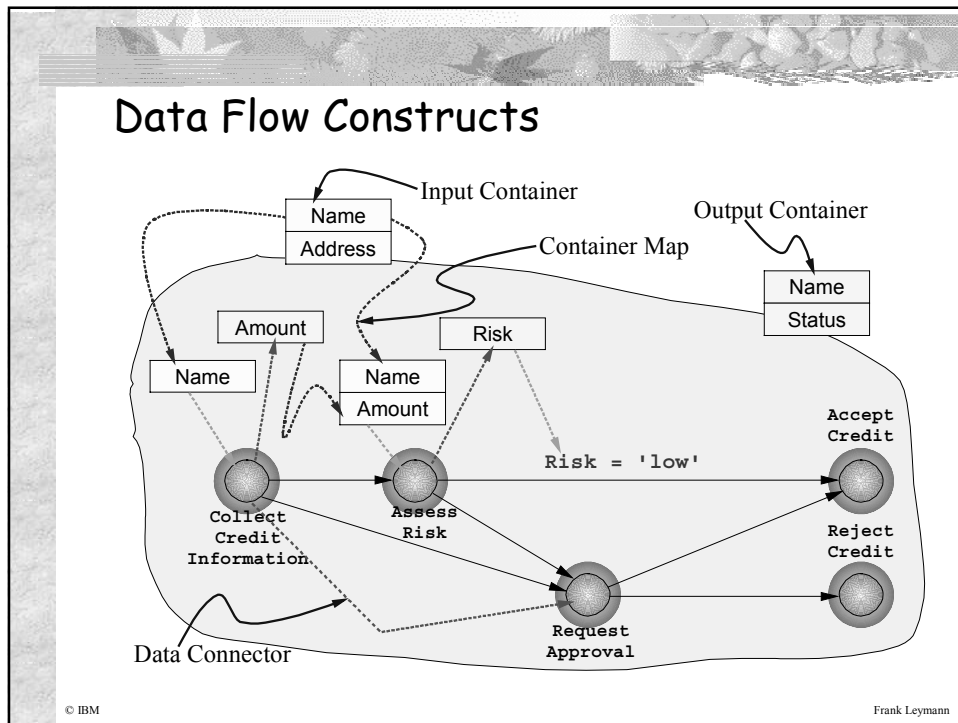
© IBM Frank Leymann



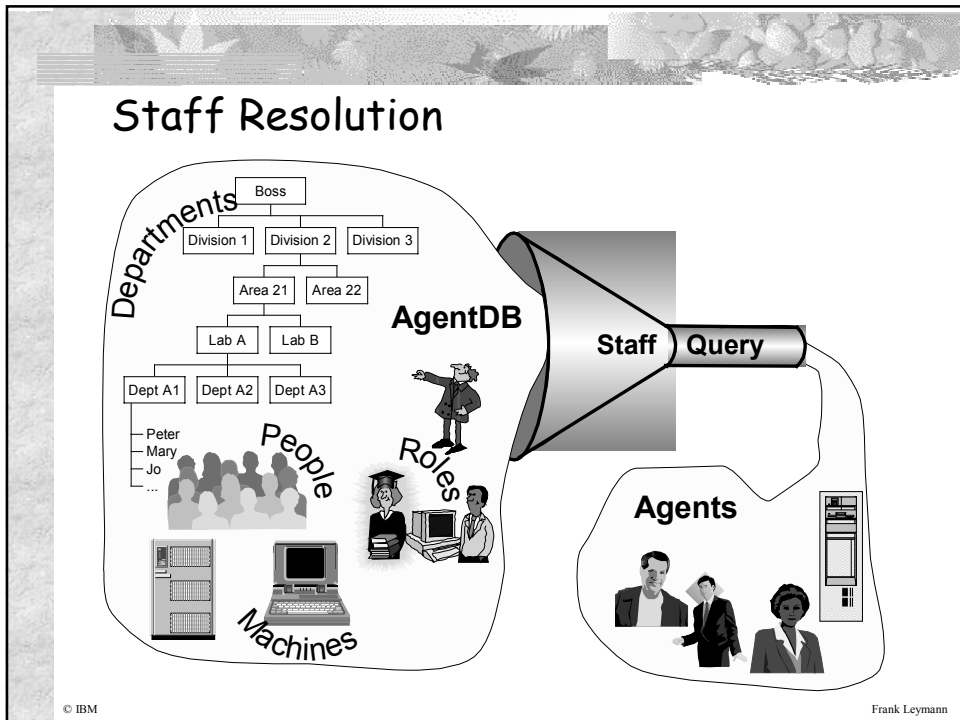
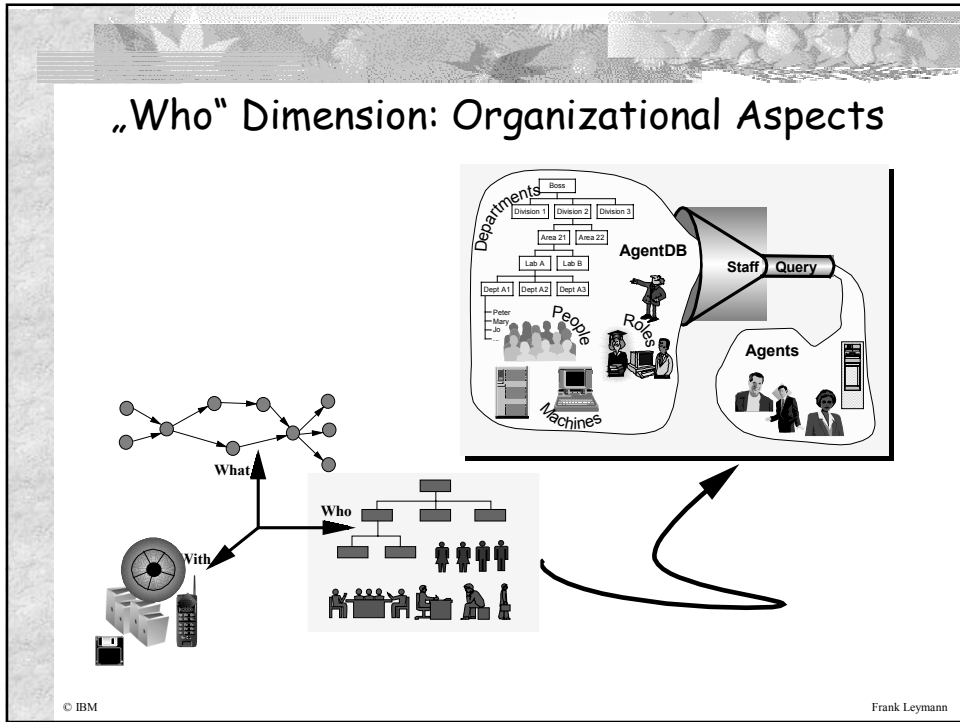


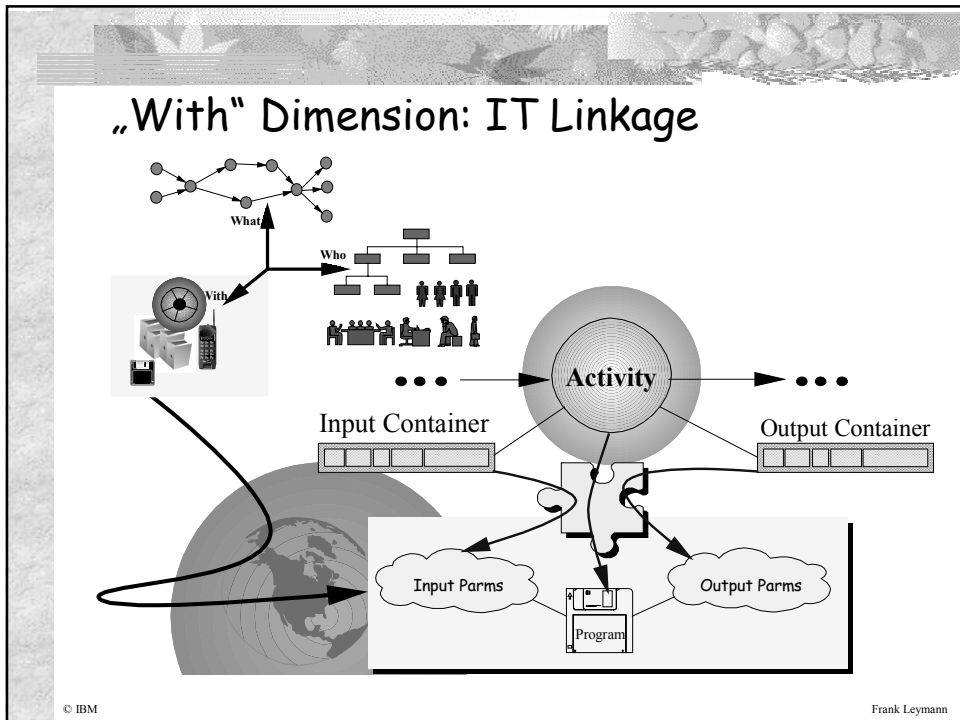
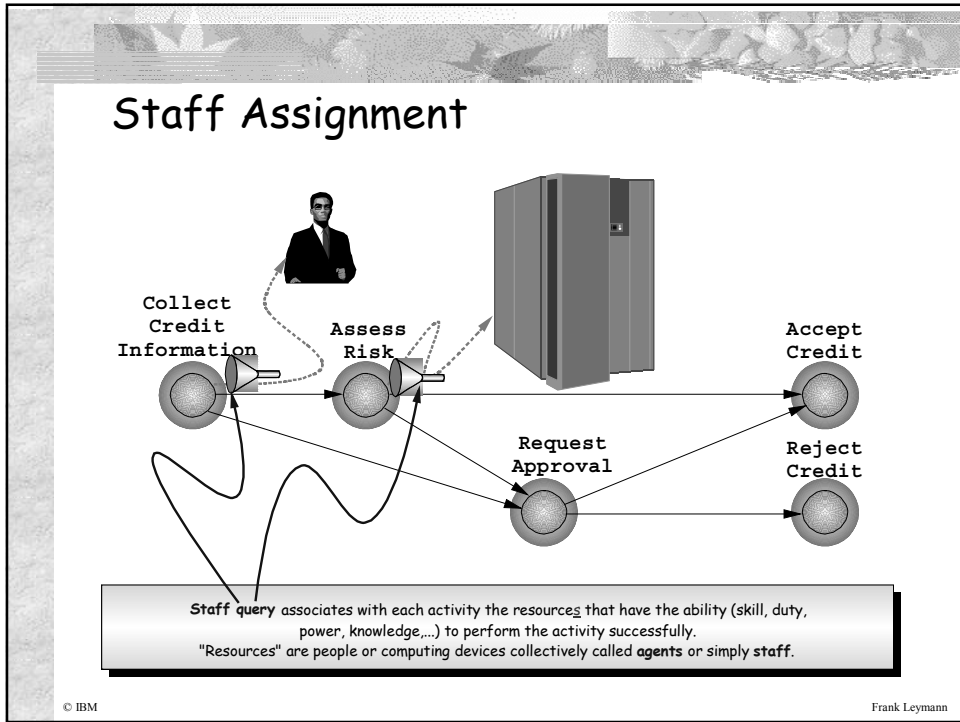


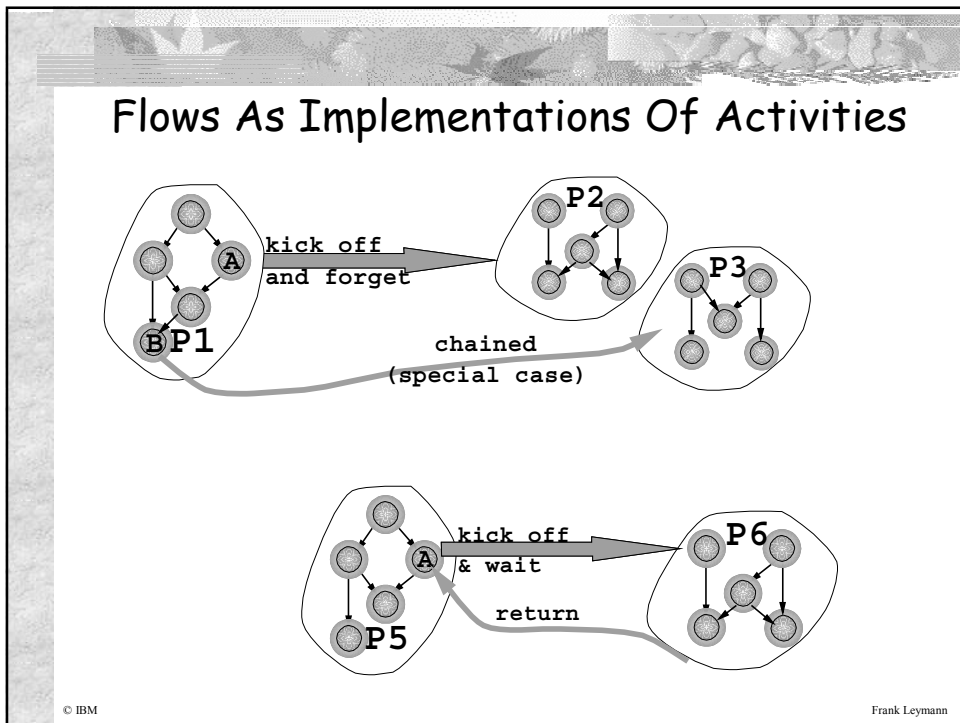




- ### Other Approaches To Flow Modelling
- Graph-/PTN-based modeling
 - ...realized in a bunch of products
 - ...standardized by WfMC (see later)
 - Many other possible approaches
 - State-Transition Diagrams
 - Calculus-based
 - ...realized on some products
 - ...standard proposals
 - ...
- © IBM Frank Leymann

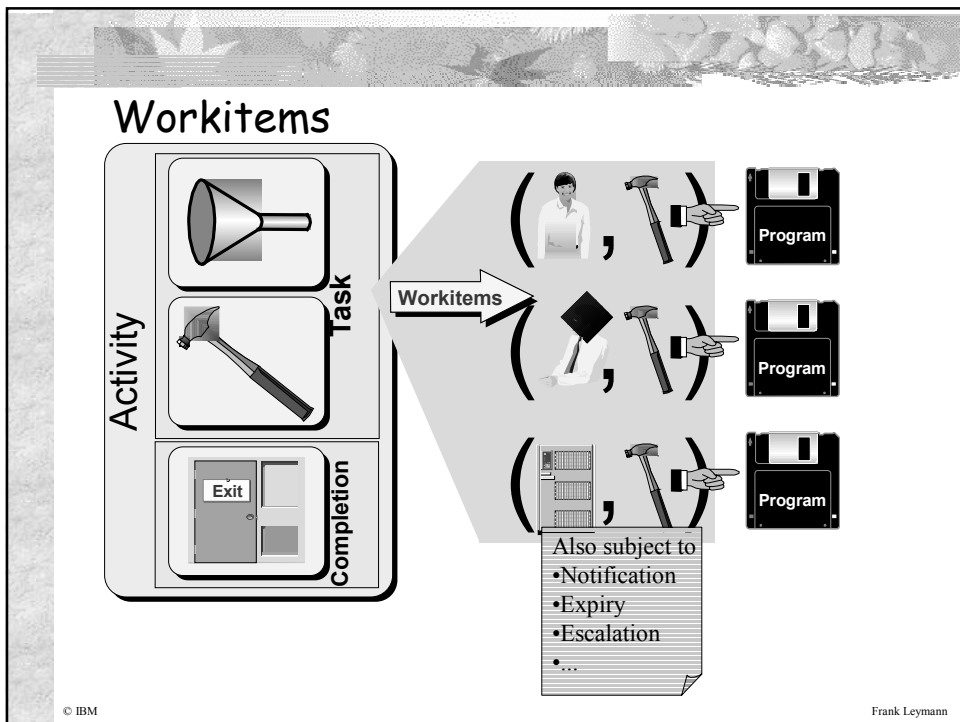
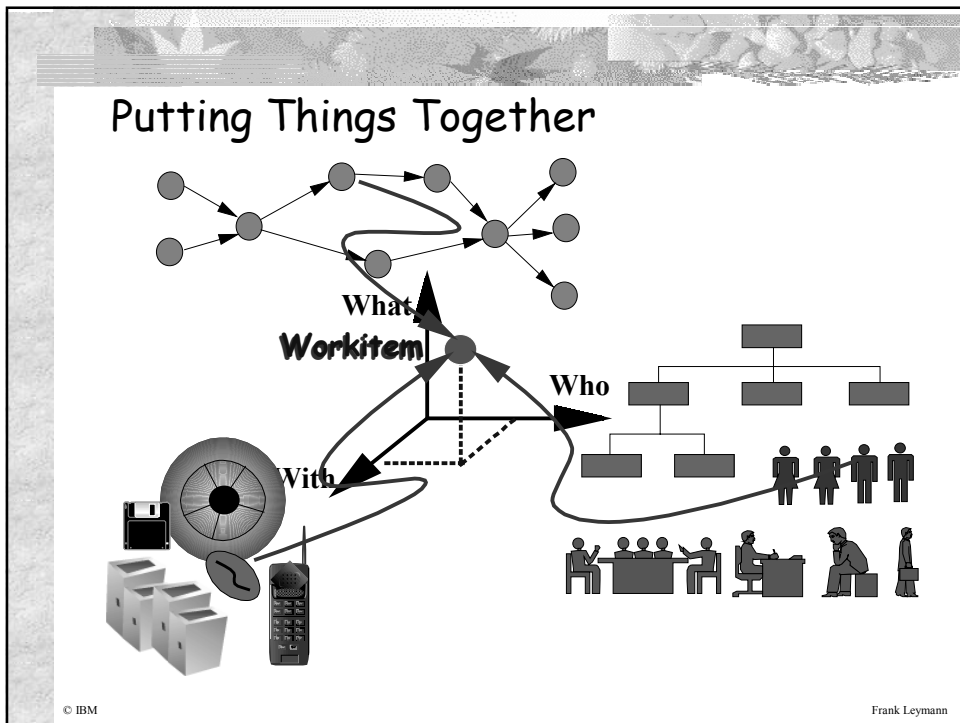






How To Work With Workflows?

© IBM Frank Leymann



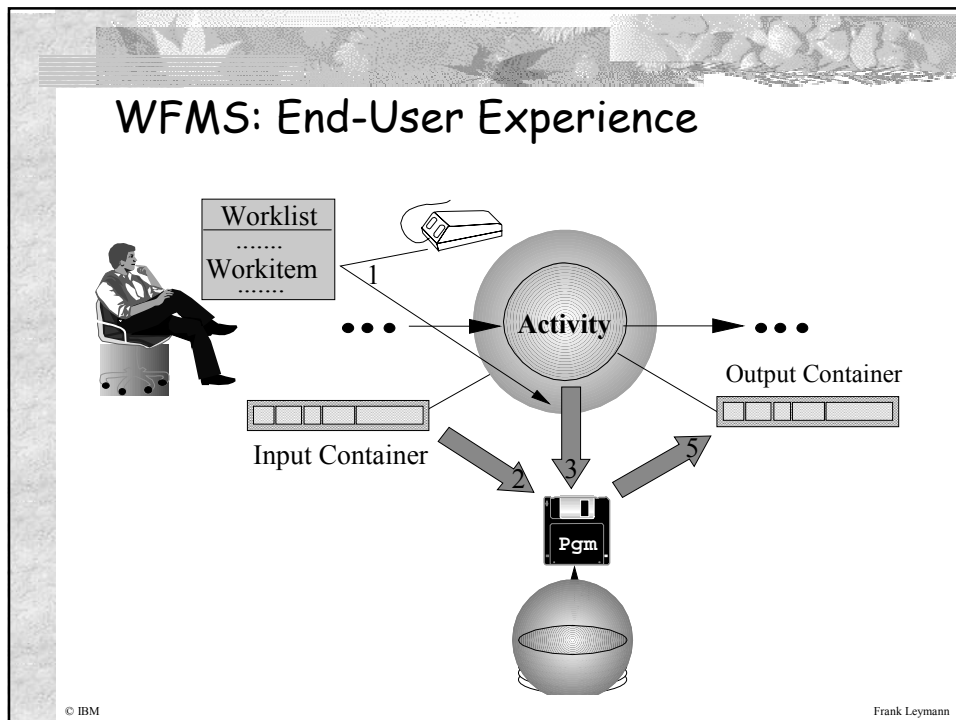
Worklists

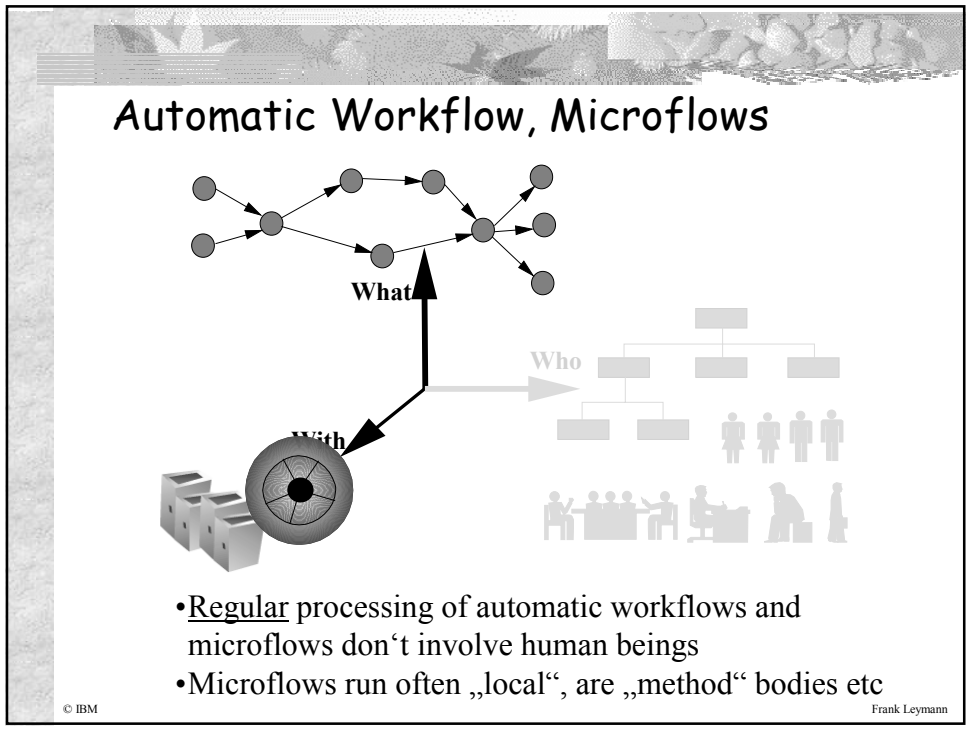
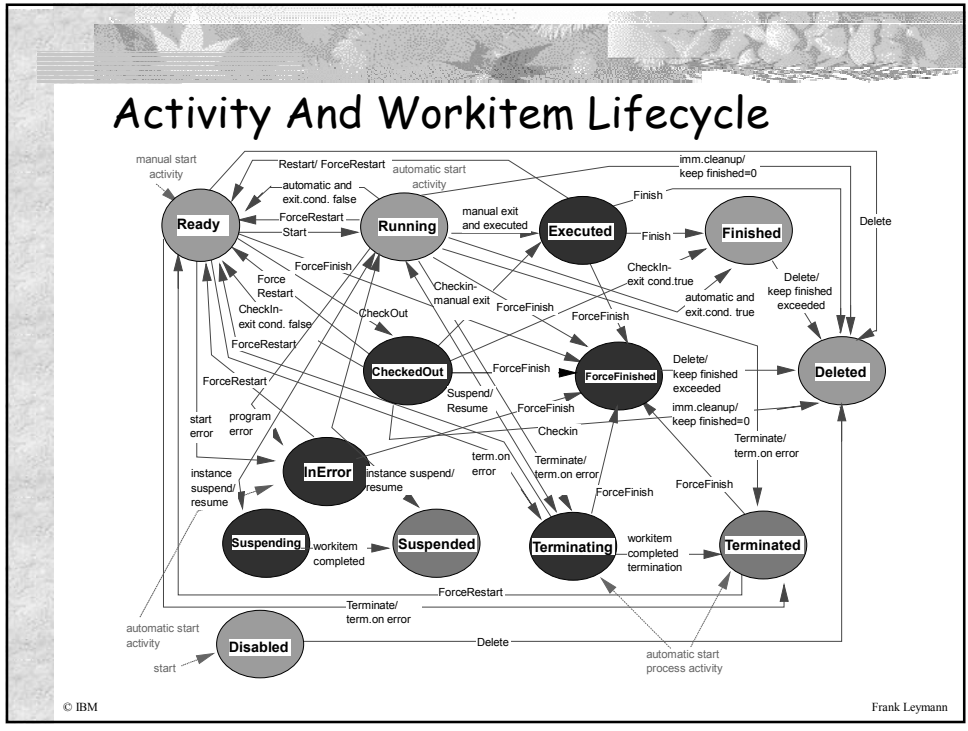
... a launchpad for business functions

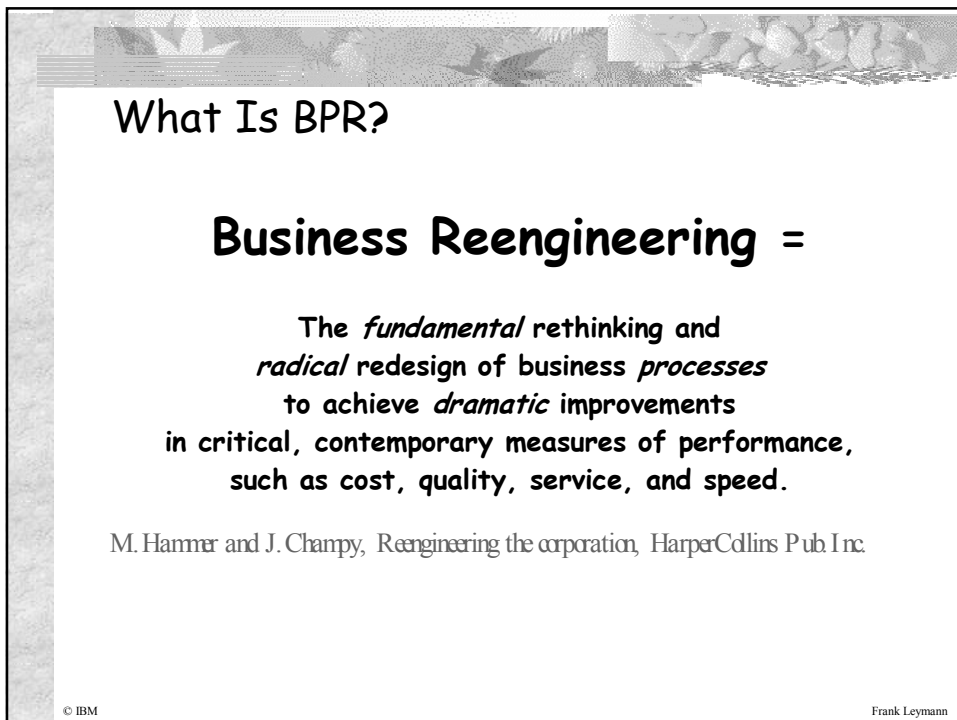
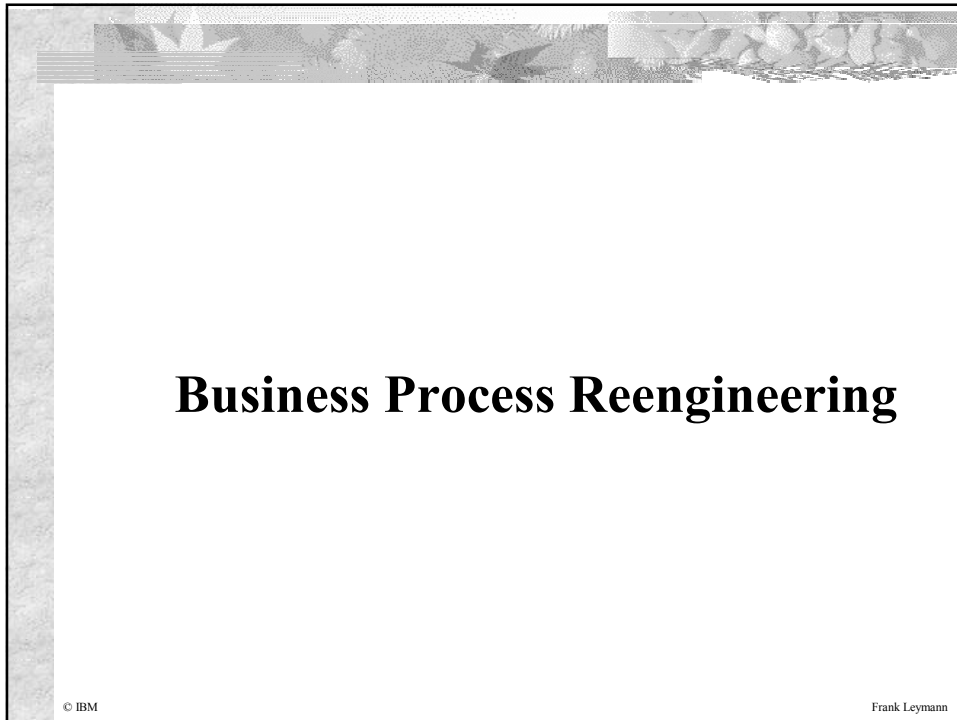
- ♣ Filtered list of workitems per agent
- ♣ Automatic prioritization of work
- ♣ Associates tools to pieces of work
- ♣ Automatic data provision
- ♣ ...

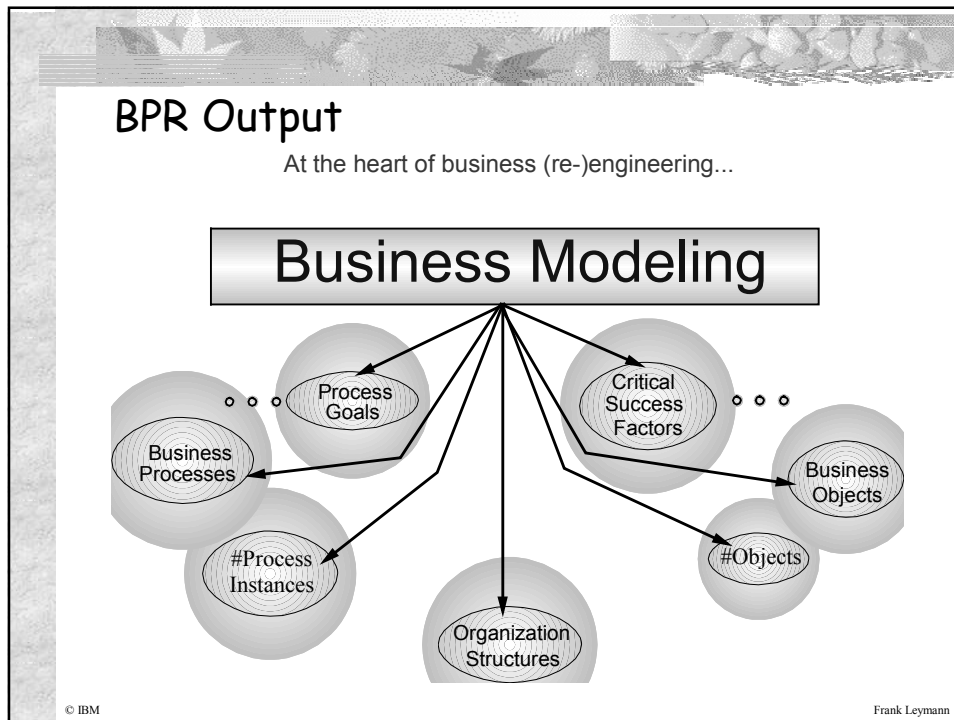
User gains focus on business aspects of work instead of computer aspects

© IBM Frank Leymann









- ## Optimizing Business Processes
- ✗ **Dynamic analysis...**
 - ✓ takes into account quantitative aspects
 - number of processes per time unit, probabilities that certain paths are taken,...
 - ✓ produces quantitative aspects
 - resources consumed to perform certain activities, to carry out business process,...
 - ✗ **Simulation generates information about...**
 - ✓ human resources needed to execute business process
 - impact on hiring strategy
 - ✓ skills needed to handle business process
 - impact on skill planning
 - ✓ time and cost for performing business process
 - indicator for outsourcing
 - ✗ **Used to compare and select from alternative models of a given business process the "optimal" one**
 - optimal in terms of metrics like cost, duration,...
- © IBM Frank Leymann

Purpose Of Simulation

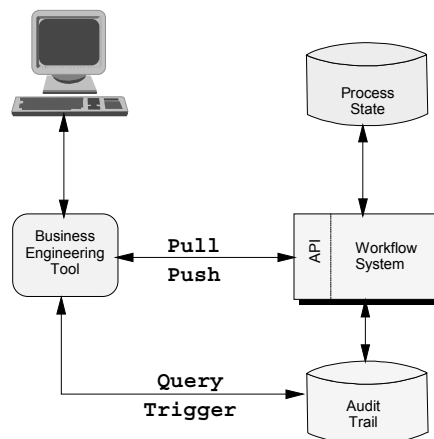
Verify capability of organization to support expected workload

- ✗ Performed based on metrical information ("instrumentation")
- ✗ Instrumentation requires to specify
 - ✓ Number of processes started per time intervall, i.e. distribution patterns of starts - for example:
 - constant: same number for each time intervall
 - exponential: smaller numbers more frequent than large numbers
 - uniform: numbers random within lower and upper bound
 - customer defined: 57 between 9AM and 11AM, 341 between 11AM and noon,...
 - ✓ Probability of transition conditions (likelihood of different branches taken)
 - ✓ Probability of activation-, join- and exit conditions (likelihood of repetitions)
 - ✓ Average duration of activities (work time, idle time,...), i.e. their distribution patterns
 - ✓ Processing power of resources, availability (based on calendar, shifts,...)

© IBM

Frank Leymann

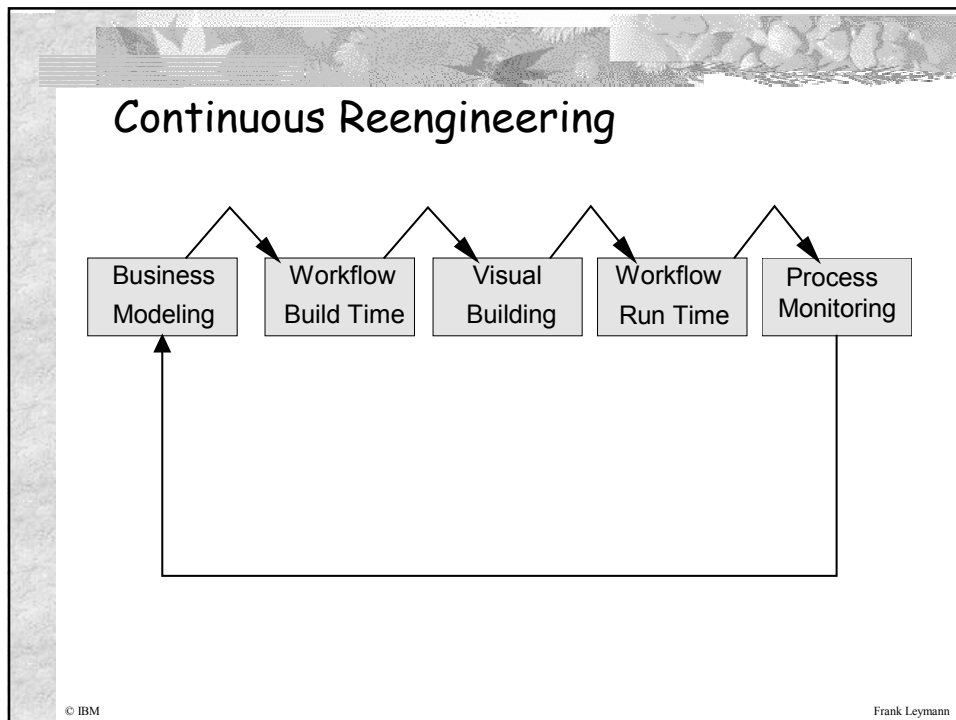
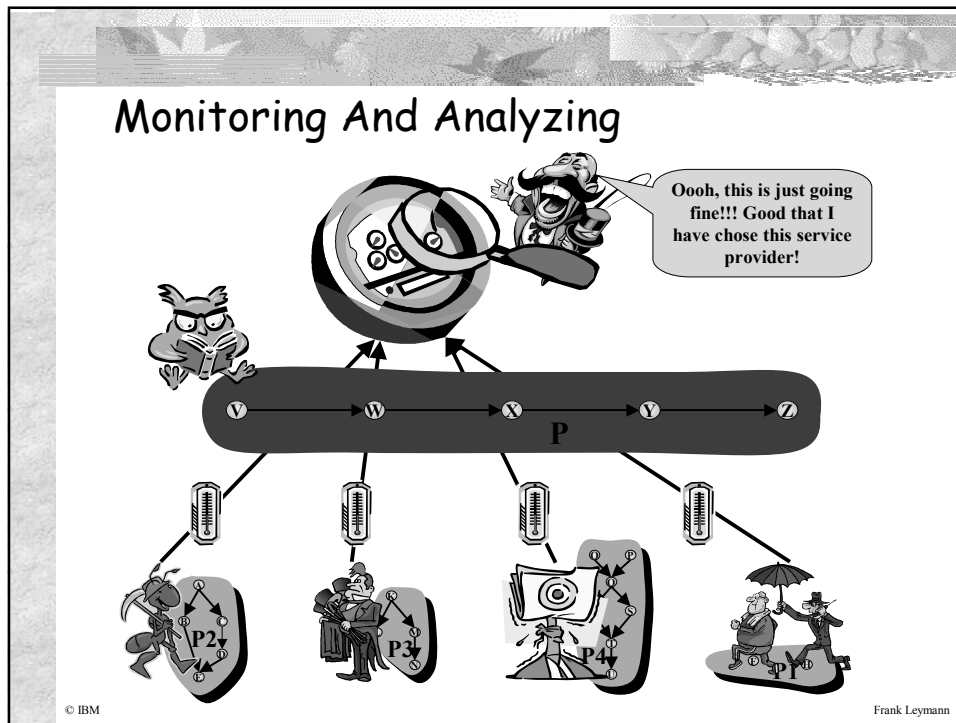
Monitoring And Auditing



- ▶ Business modeling based on assumptions about cardinalities, duration, etc.
- ▶ Based on these assumption process characteristics are derived (costs,...) which trigger optimizations
- ▶ Thus, incorrect assumptions result in non-optimal process models
- ▶ WFMS allows to access actual state (monitoring) as well as history (auditing) of each workflow
- ▶ Analyzing audit trail ("vanilla" SQL, OLAP, mining) derives "real data" for optimizing process models (re-engineering)
- ▶ Monitoring (manually or automatically) individual processes or instances of the same model allows to detect out-of-line situations and to react accordingly (re-assignment of work,...)

© IBM

Frank Leymann



Agenda

1. History
2. Workflow Basics
- 3. Some Plumbing**
4. WFMS Architecture
5. Transactional Workflows
6. Application Structures
7. Web Services

© IBM

Frank Leymann

Messages

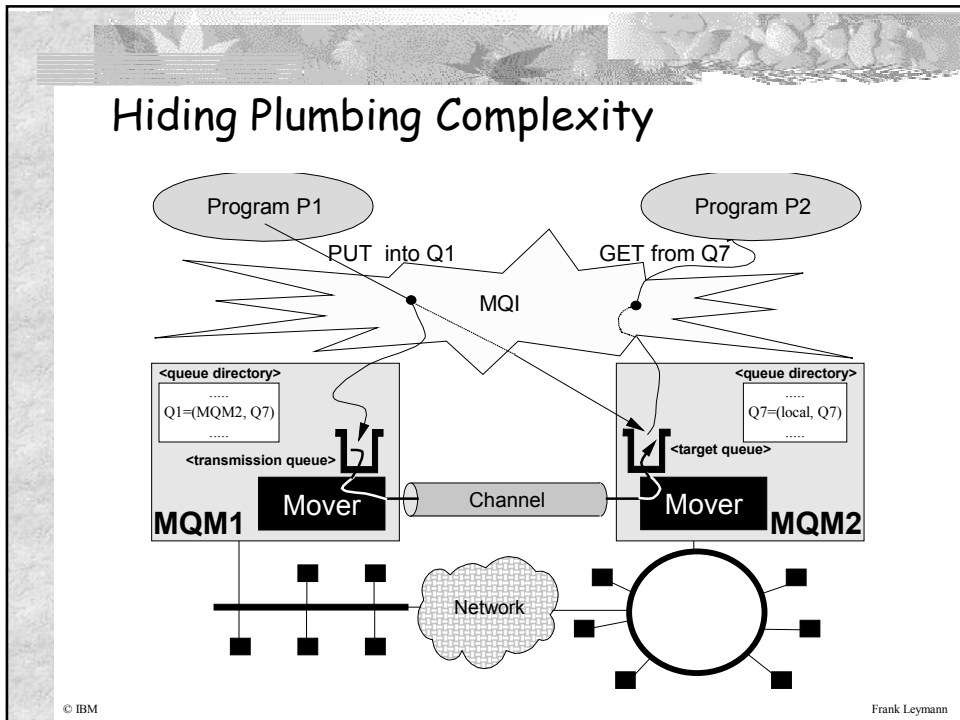
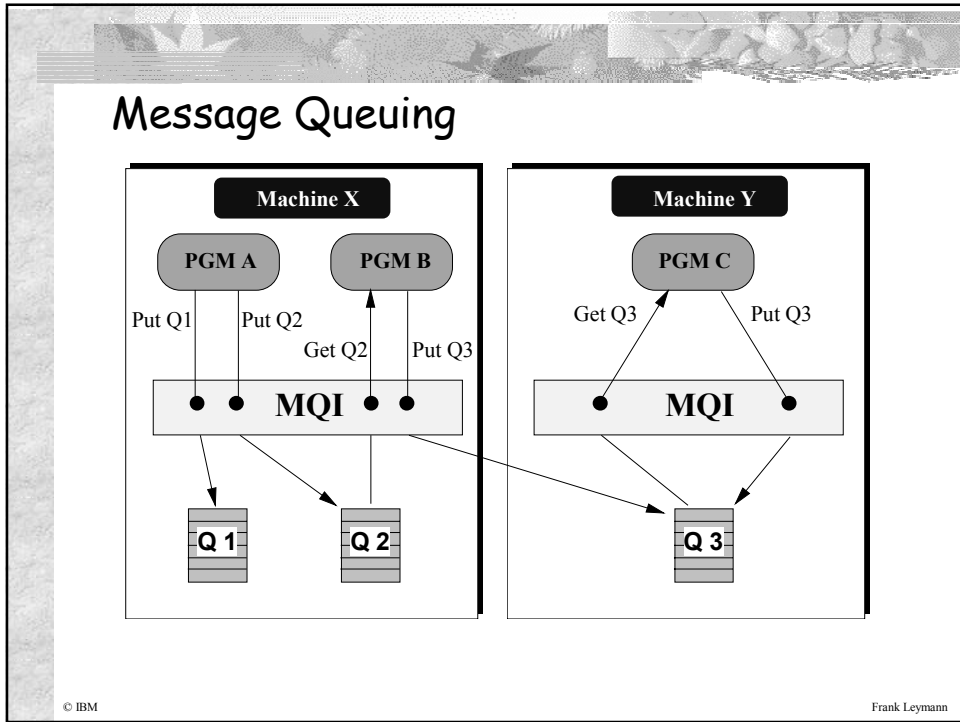


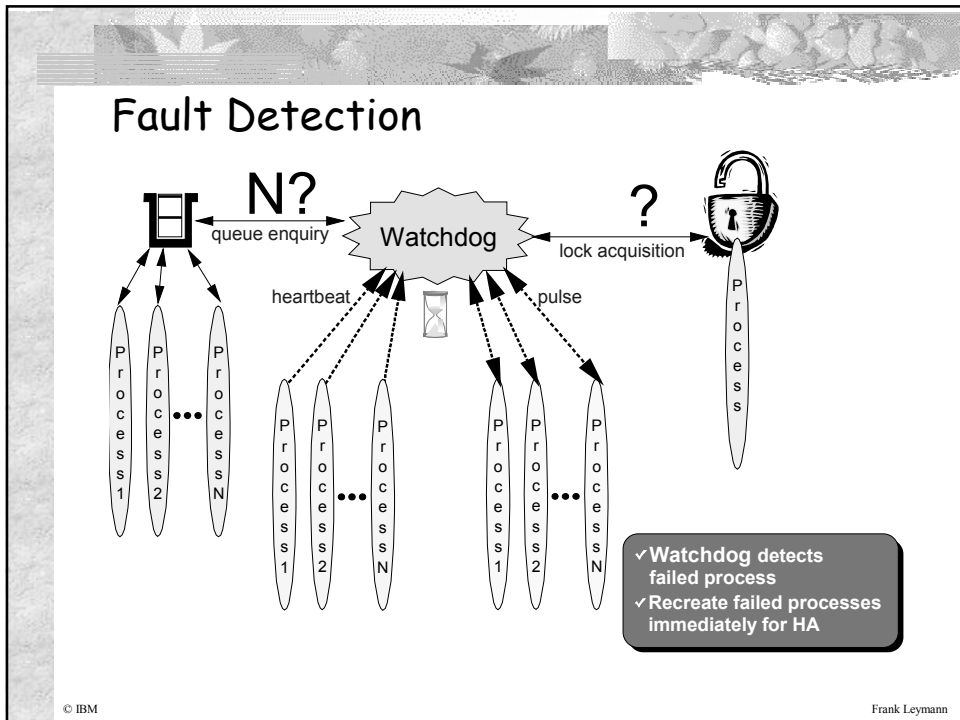
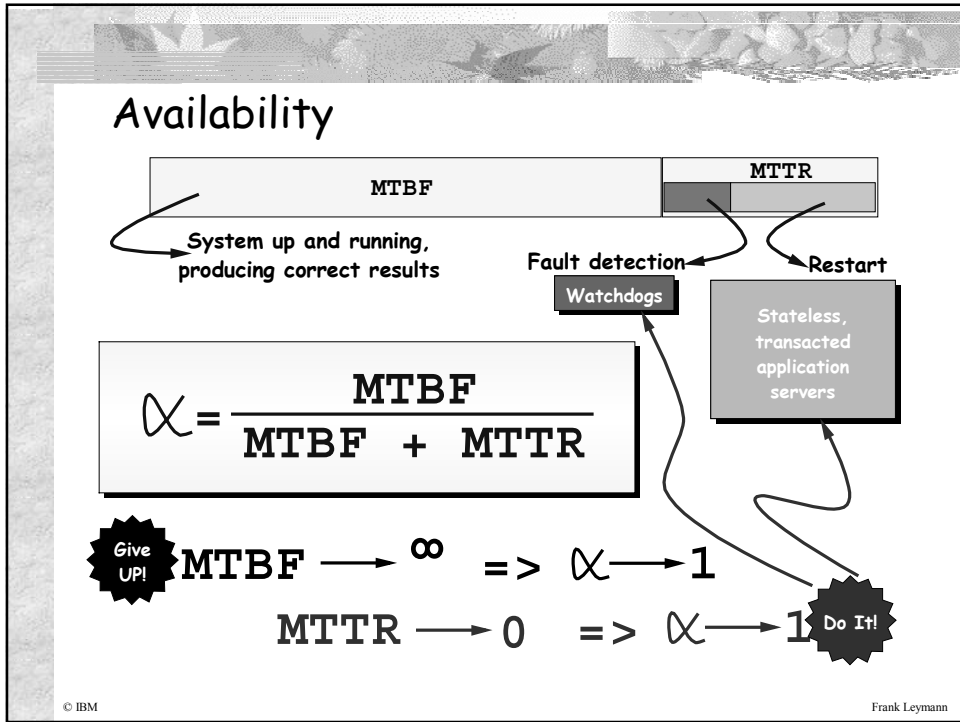
↪ Reply-to queue, message lifetime,...

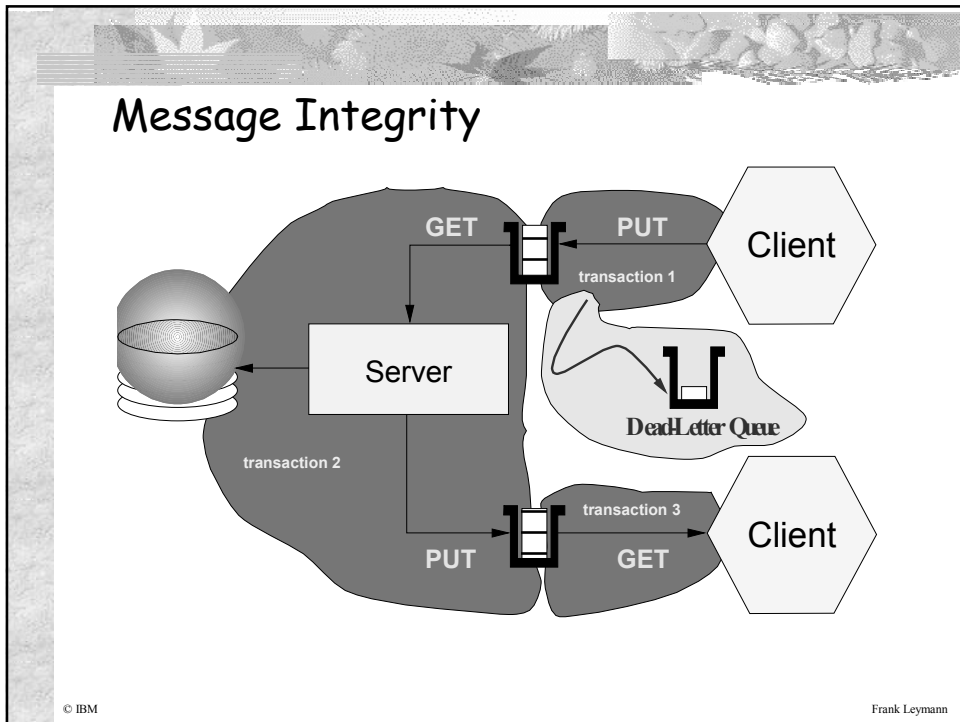
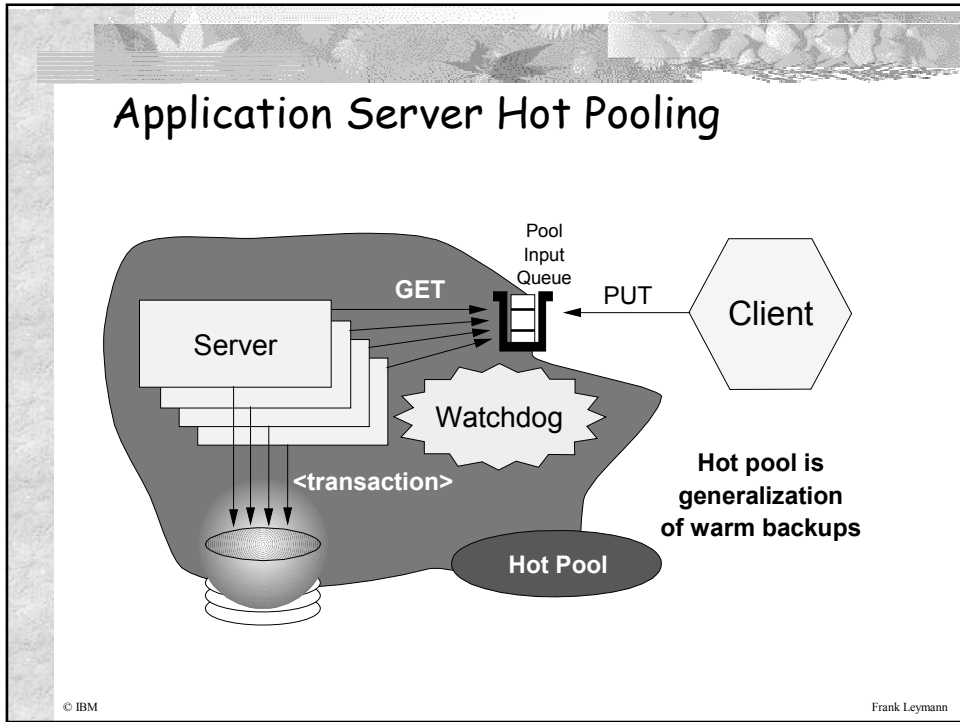
- ✗ Request component may be omitted [or minor aspect] (e.g. in information brokering)
- ✗ Data is often a significant amount of information e.g. a complete business object (opposed to "simple parameters" in method invocations)
- ✗ Emphasis is reliable data/information exchange as well as exactly once invocation

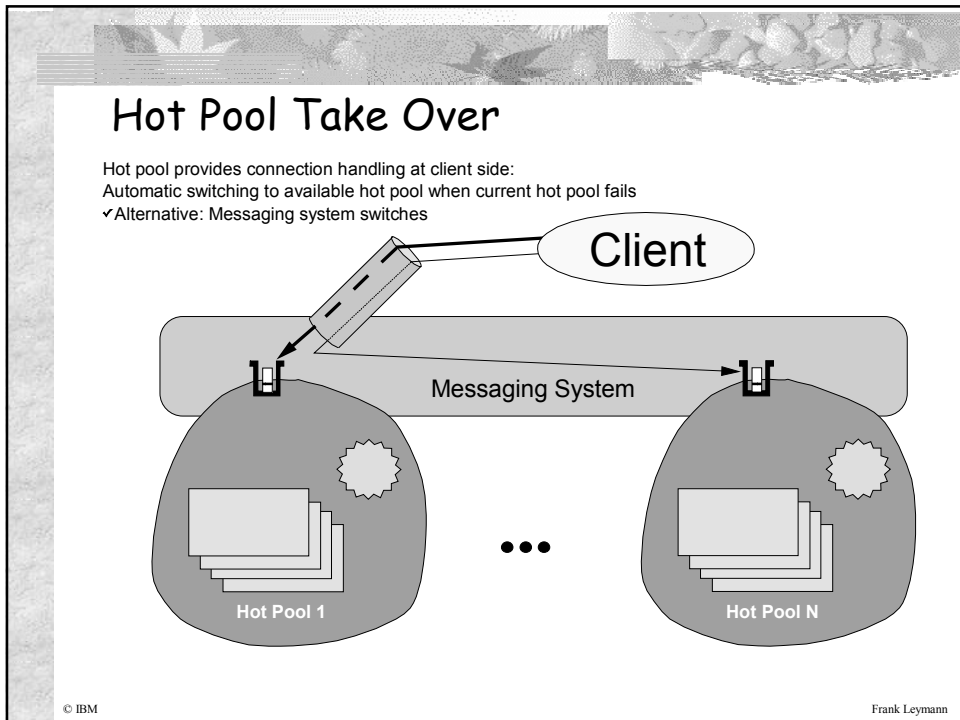
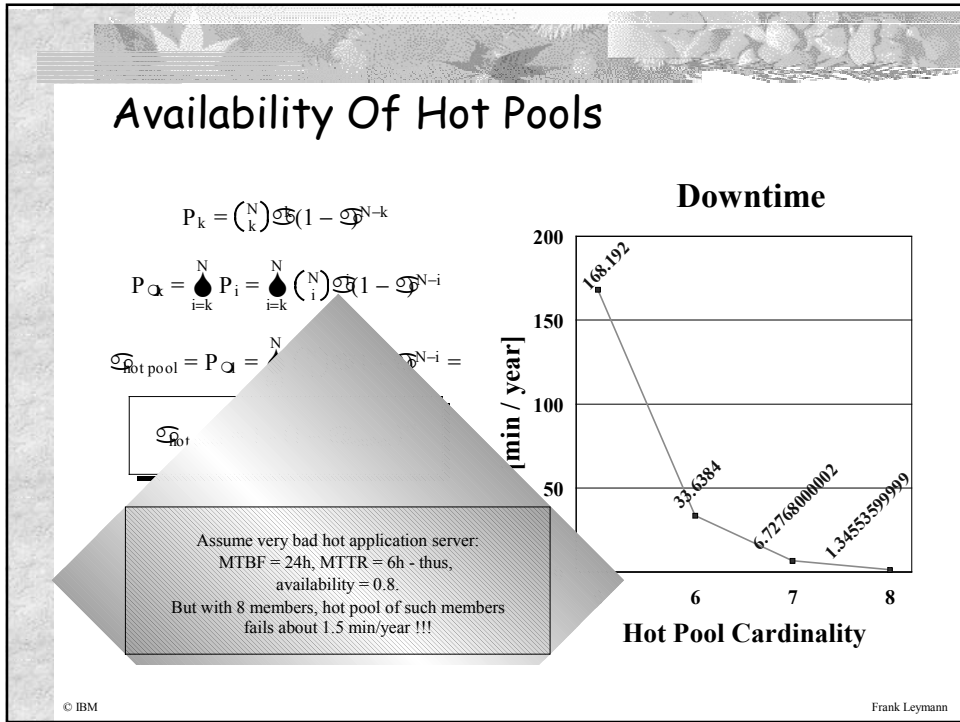
© IBM

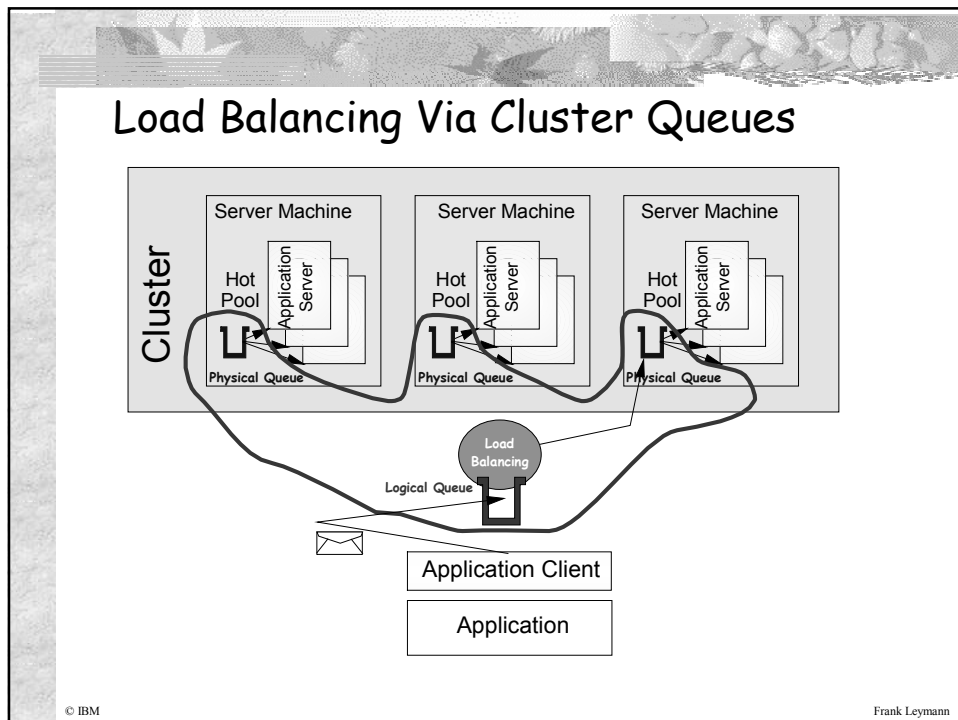
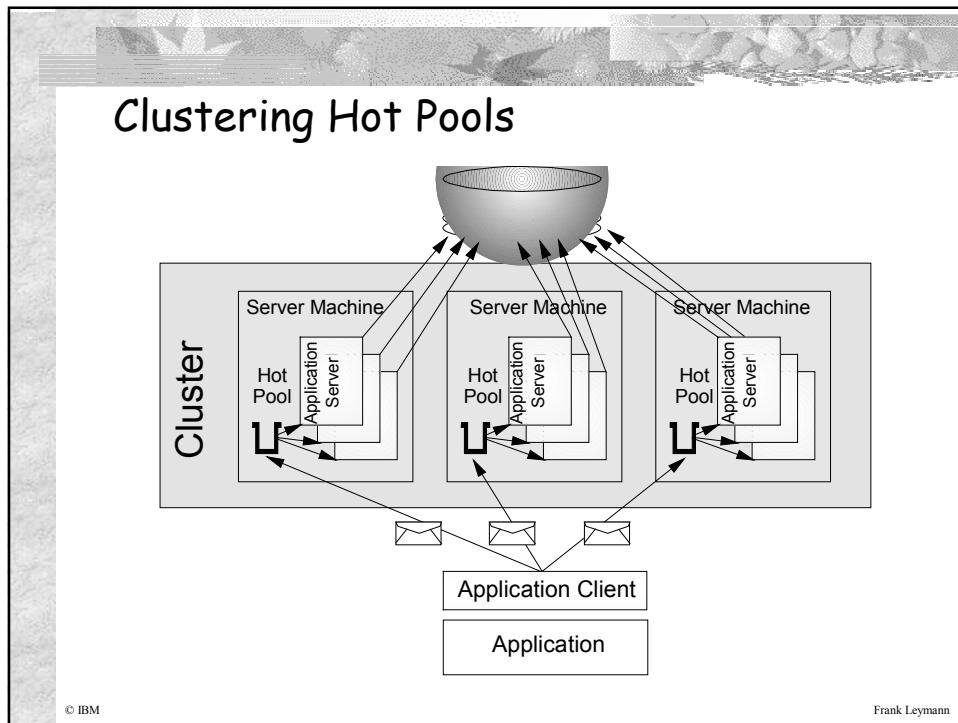
Frank Leymann

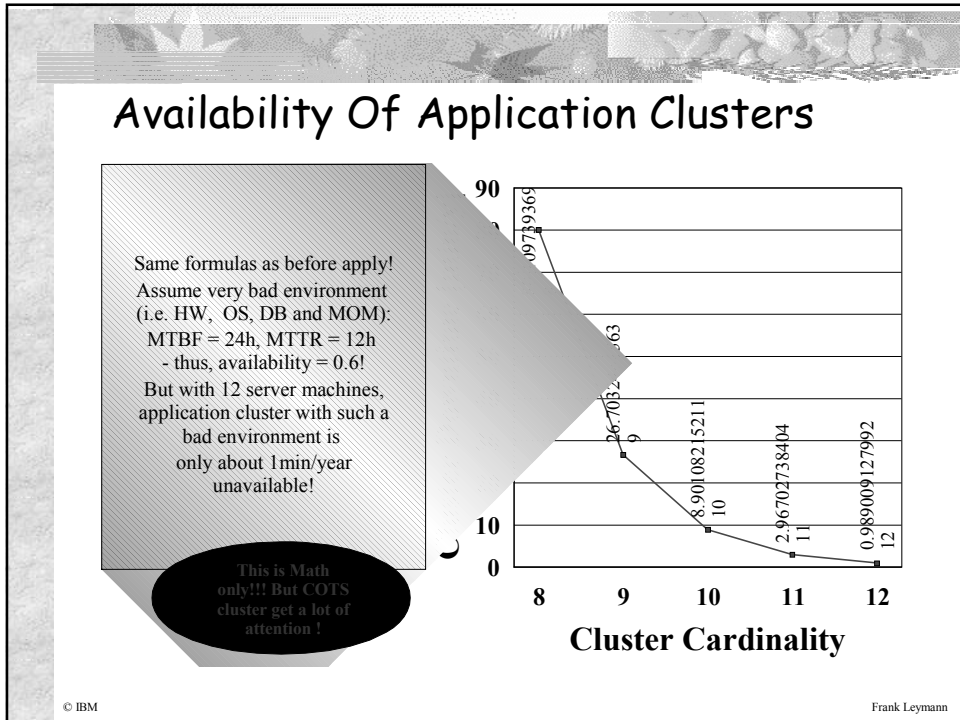












- ### Summary: HA In Application Clusters
- Servers are stateless
 - Servers use transactions to process client calls
 - C/S communication is based on recoverable queues
 - Ensured message integrity
 - Server hot pooling
 - Automatic recreation of hot pool members
 - Take-over between cluster machines
- Plus platform
 specifics:
 (HACMP, ARM,
 Paralles Sysplex,...)
- © IBM Frank Leymann

Summary: Scale In Application Clusters

- Hot pool allows scaling on each machine
(increase cardinality of hot pool until CPU bound is reached)
- Application Cluster allows scaling via hot plug-in of machines
(attach additional machines hosting server [hot pool]
until DBMS server becomes bound [CPU, I/O])
- Domain (see below) allows scaling via attaching system groups
(for organizational reasons or DBMS bounds)
- Using messaging as software bus allows scaling by
distributing app components on different machines
(dedication of CPUs for particular logical tasks)



Plus platform specifics:
(WLM, Enclaves,...)

© IBM

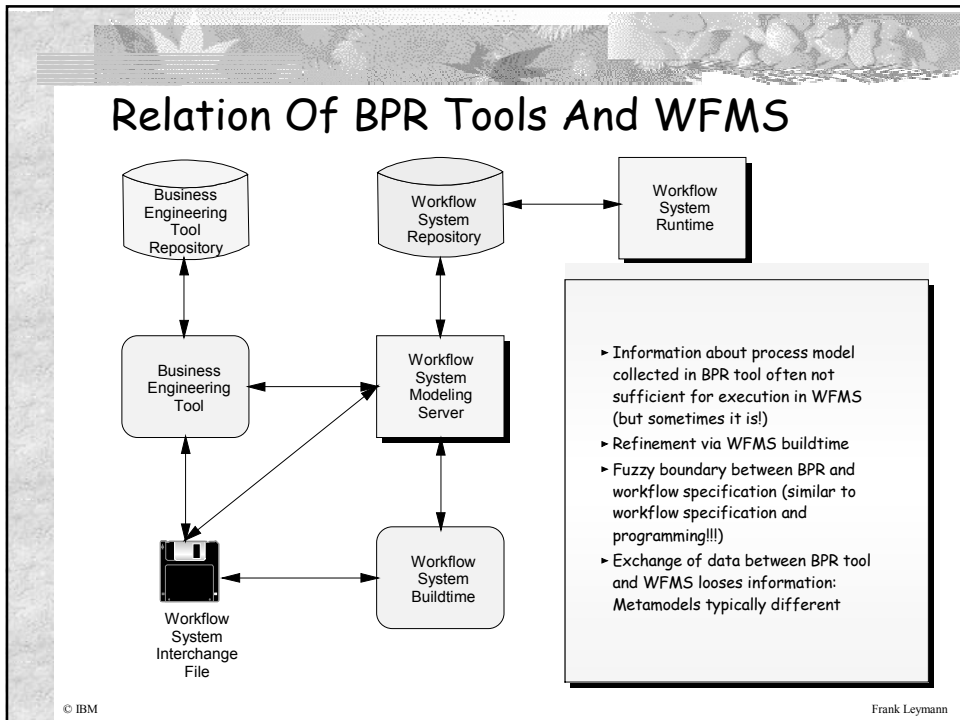
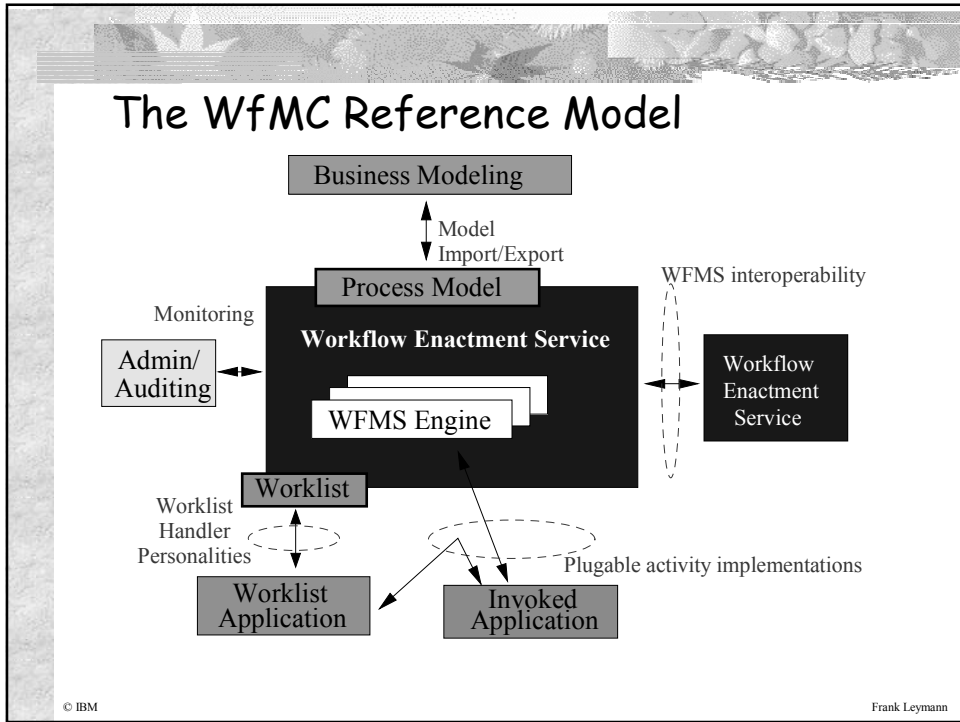
Frank Leymann

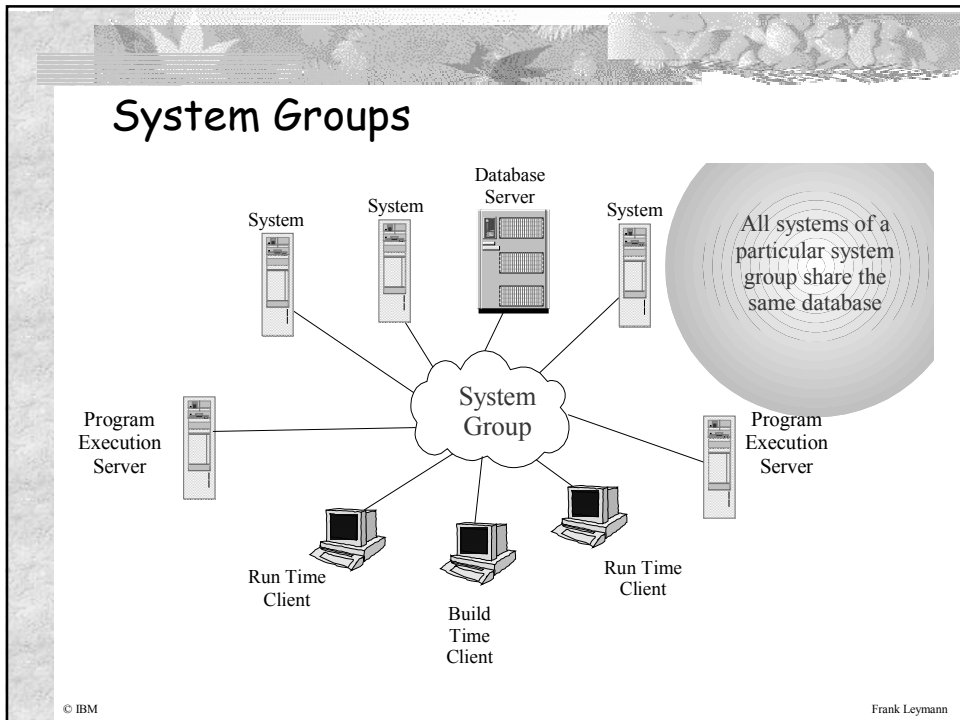
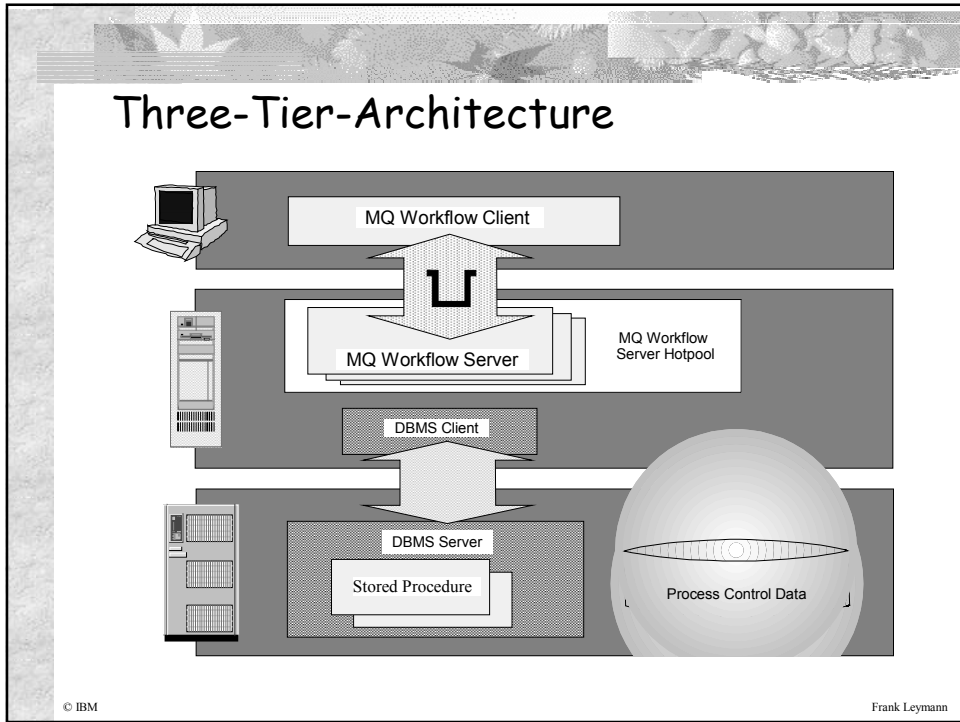
Agenda

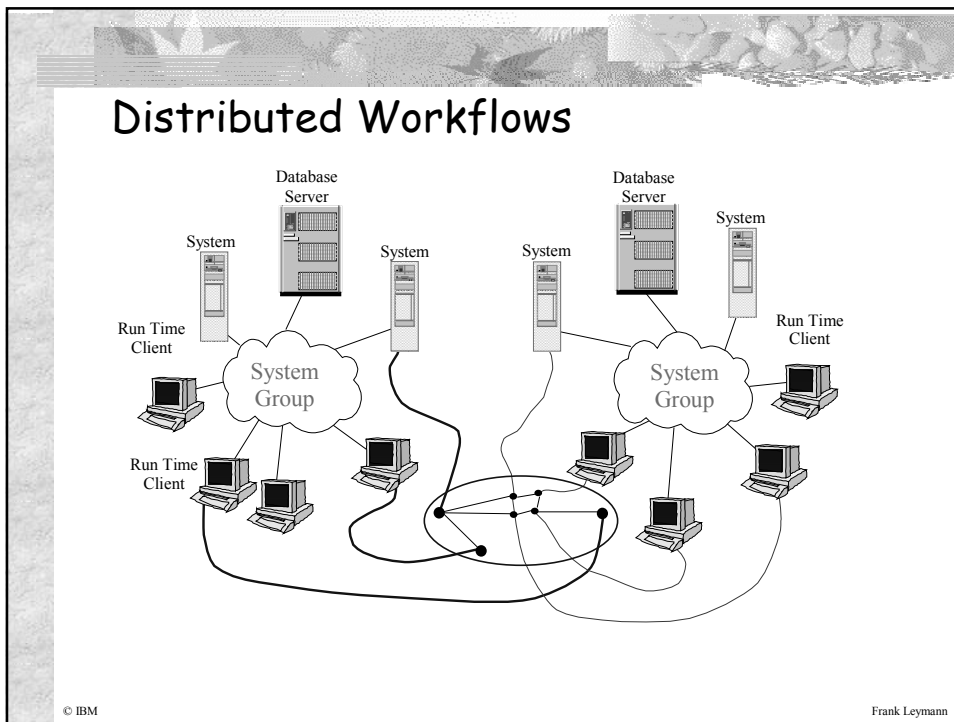
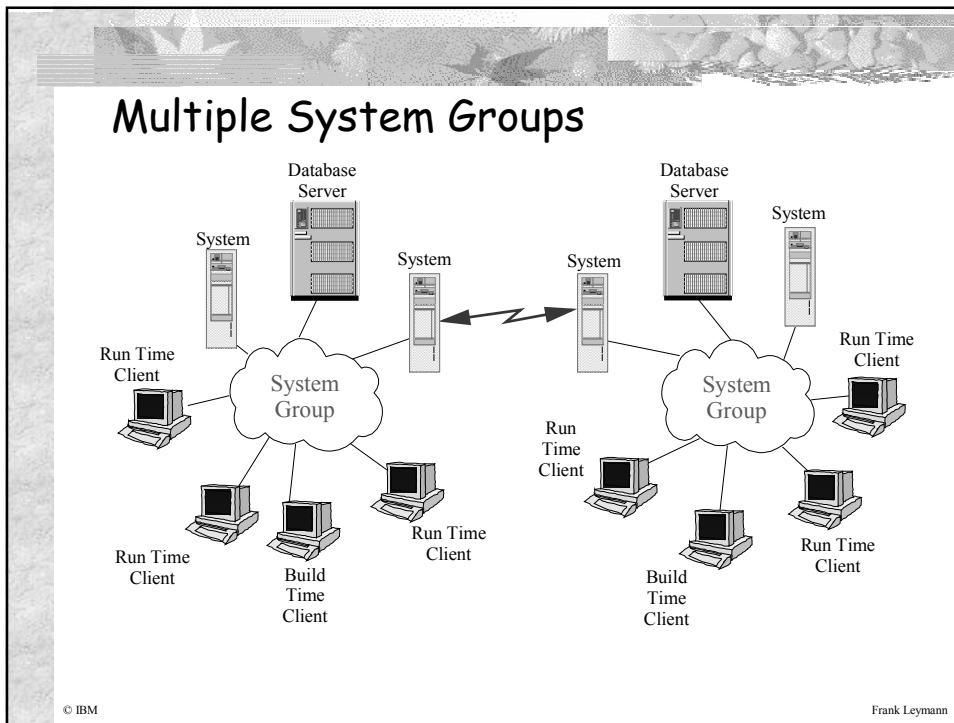
1. History
2. Workflow Basics
3. Some Plumbing
- 4. WFMS Architecture**
5. Transactional Workflows
6. Application Structures
7. Web Services

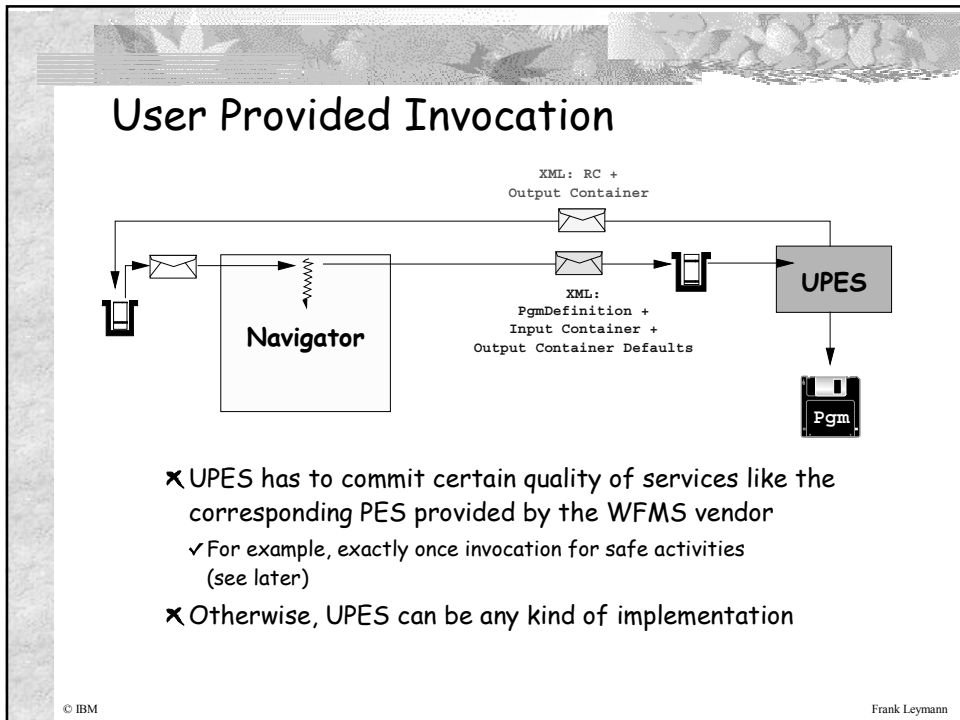
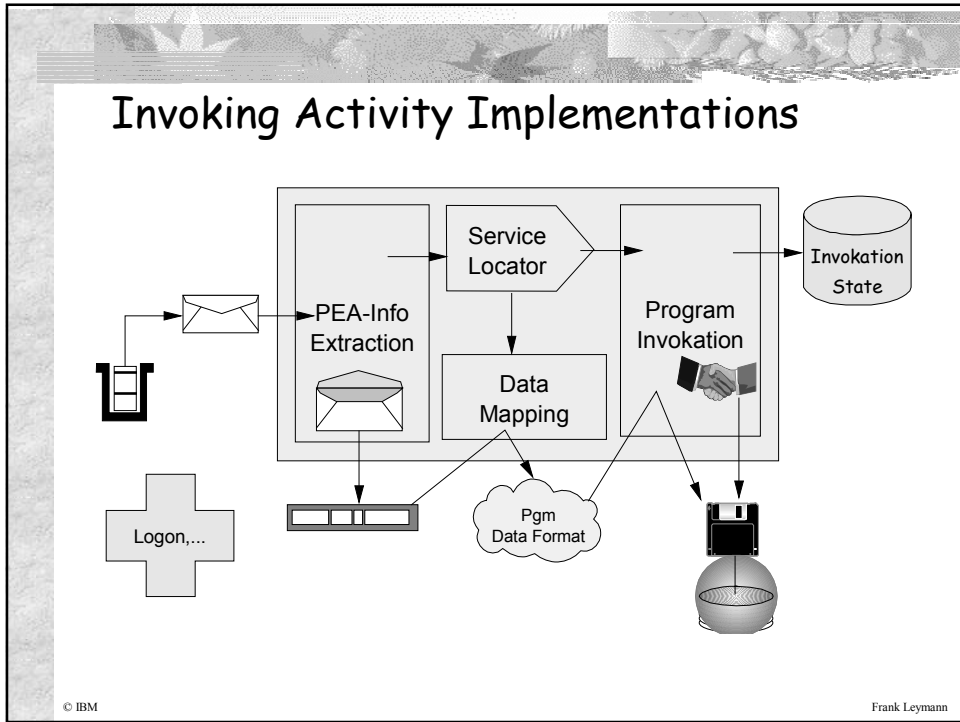
© IBM

Frank Leymann









Agenda

1. History
2. Workflow Basics
3. Some Plumbing
4. WFMS Architecture
- 5. Transactional Workflows**
6. Application Structures
7. Web Services

© IBM

Frank Leymann

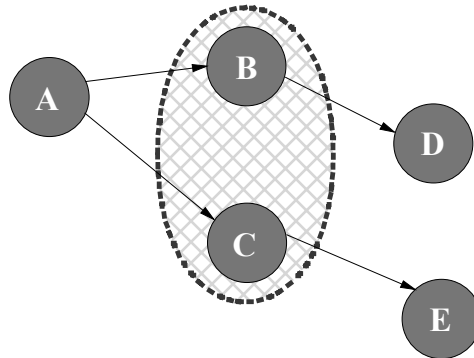
Transaction Models

- Variation of ACID properties
 - The origin: Spheres of Control
 - Nested transactions (weakens „D“)
 - Sagas (weakens „I“)
 - ConTracts
 - ...

© IBM

Frank Leymann

Atomic Spheres (Global Transactions)



- Each activity is implemented as resource manager program
- Either all activities performed with sphere commit or none
- Very often, microflows as a whole are atomic spheres

© IBM

Frank Leymann

Global Transactions: Practice

- Transaction with multiple participants
- Atomic commitment is the issue
 - E.g. 2-phase-commit protocol
- Efficiency problems when used across machine boundaries
- Not realistic across organization boundaries
 - Not only „efficiency“ issues but additional legal-, ownership-, privacy-,... Issues
 - Especially not in Internet scenarios

© IBM

Frank Leymann

„Long“ Transactions

- „Long“ is a couple of seconds to years
- Basic characteristics are:
 - Must survive (planned as well as unplanned) interrupts
 - Including power-off
 - Backout of whole transaction due to local failure not tolerable
- Often, corresponds to a business process
- Implemented via workflow technology

© IBM

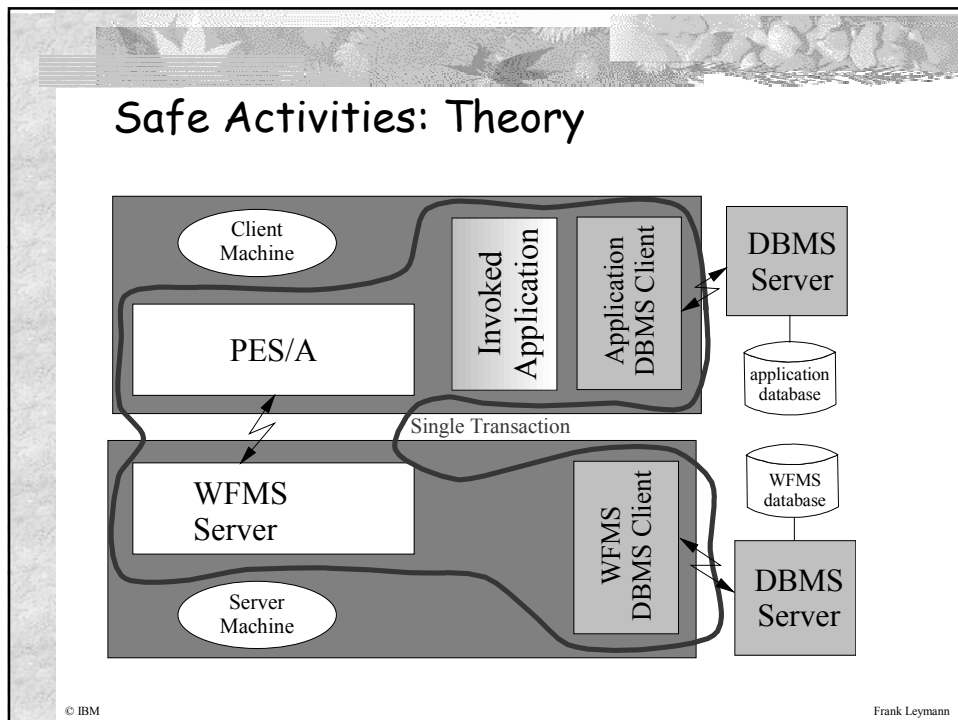
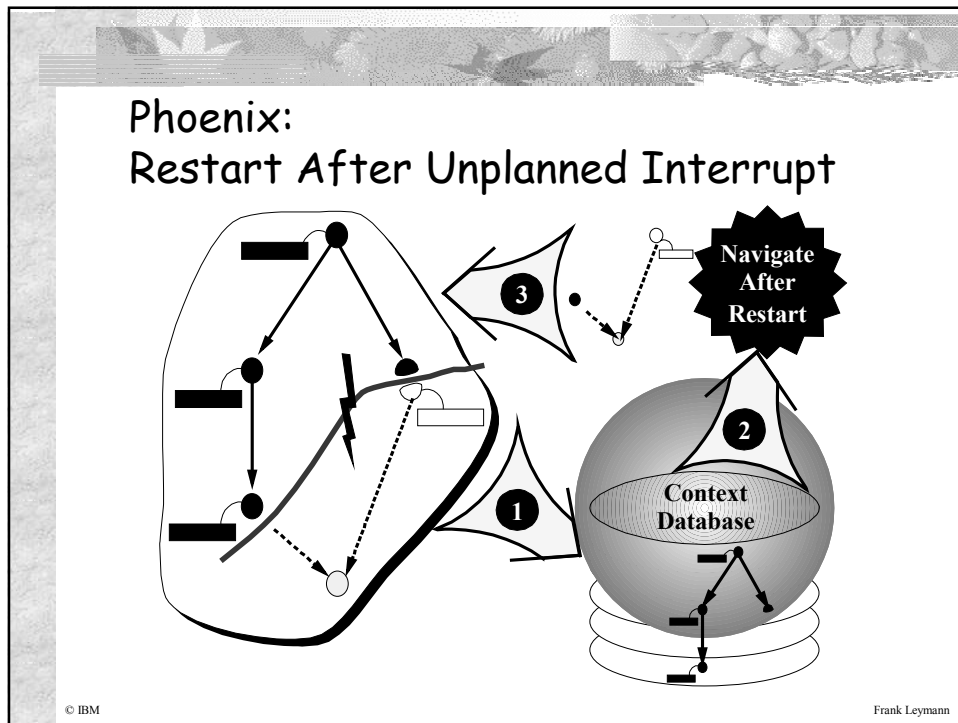
Frank Leymann

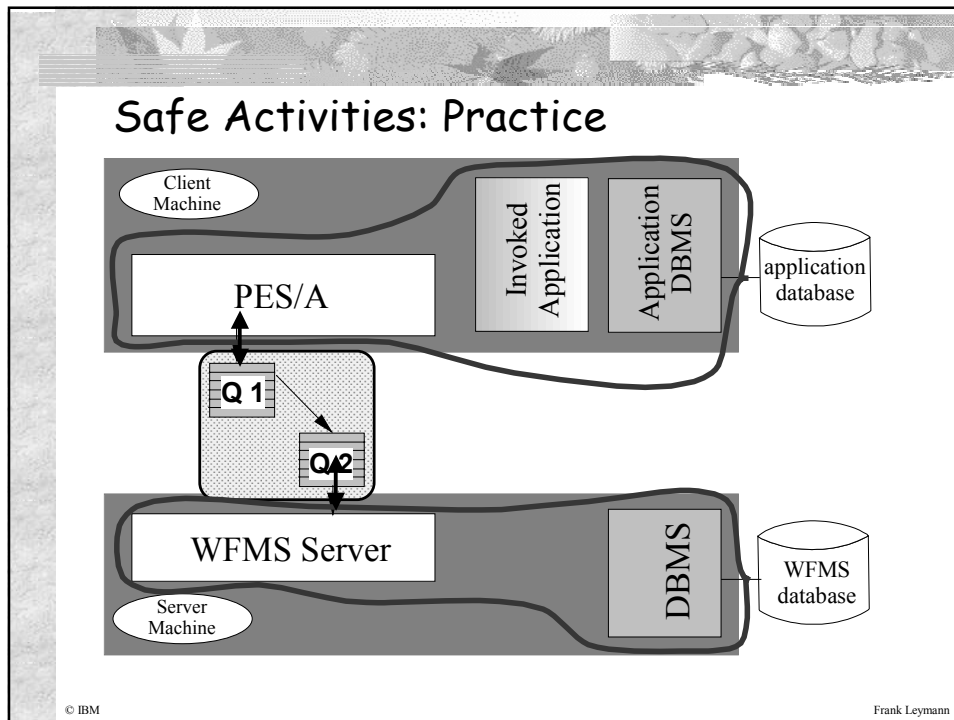
Consequence of „Interruptability“

- Hosting environment must provide „Phoenix“-Behavior
- Workflow recovers out of the ashes
 - Work continues after interrupt where left
- „Relevant“ state changes of workflow must be made persistent
- (Transactional) „steps“ and state-changes of workflow must be performed atomic

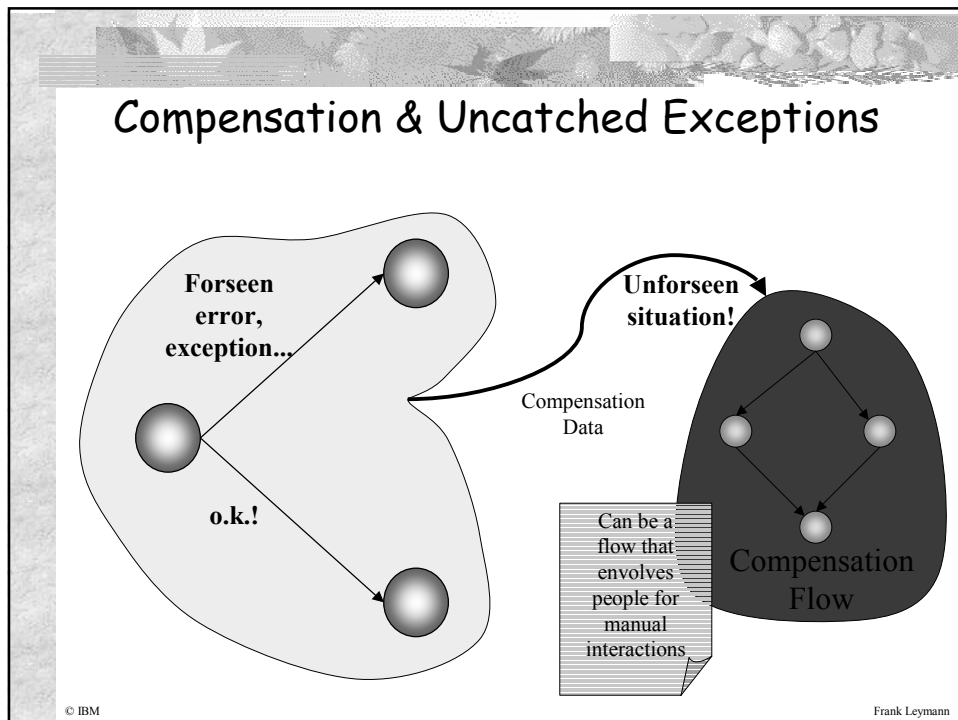
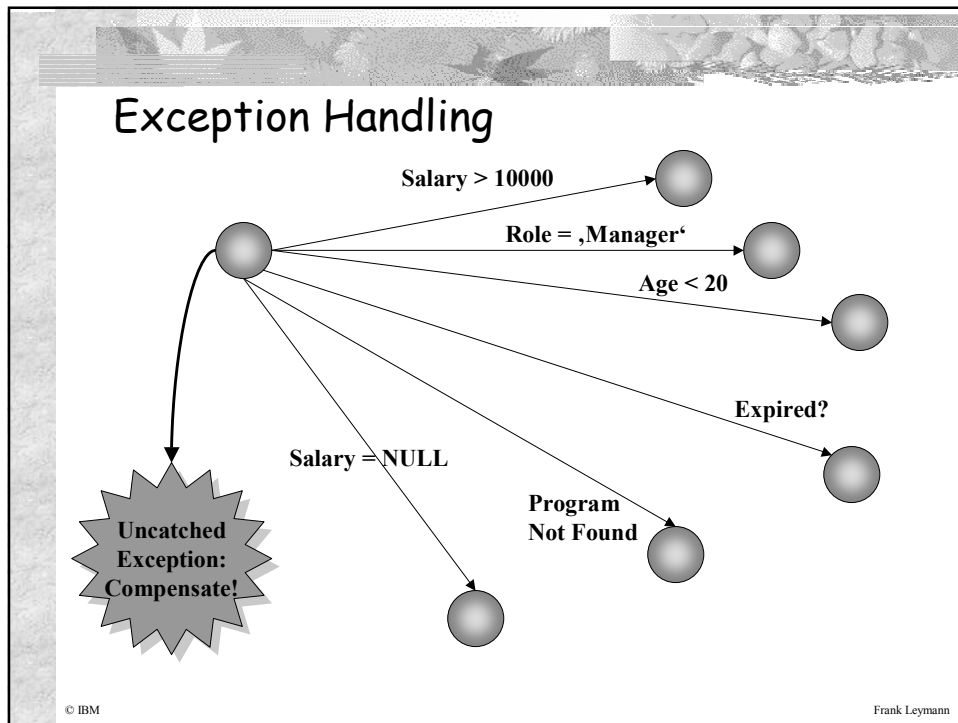
© IBM

Frank Leymann





- ### Consequence of „No System Initiated Backout“
- Handling of exceptions is explicitly modeled
 - By means of flow constructs
 - Compensation actions
 - System can initiate human intervention to repair erroneous situations
 - Failed steps set in „error“ state
 - Hosting environment stops regular processing
 - Notification to specified personell
 - Fix container content
 - Fix activity implementation
 - Skip, retry,... activity
 - Suspend, resume, terminate, compensate,... Workflow
 - Also important for automatic workflows
- © IBM Frank Leymann



Compensation: Concepts

- Not every action has a reverse (real action)
- In reality, the effects of an arbitrary action cannot be simply undone, i.e. the initial state cannot be recreated
- An action used to reverse the effects of another action is called compensation action
- Semantic Recovery: Recovery schema base on compensation

- CAVEAT: Compensation very likely one of today's most frequently exploited techniques in transaction processing

© IBM

Frank Leymann

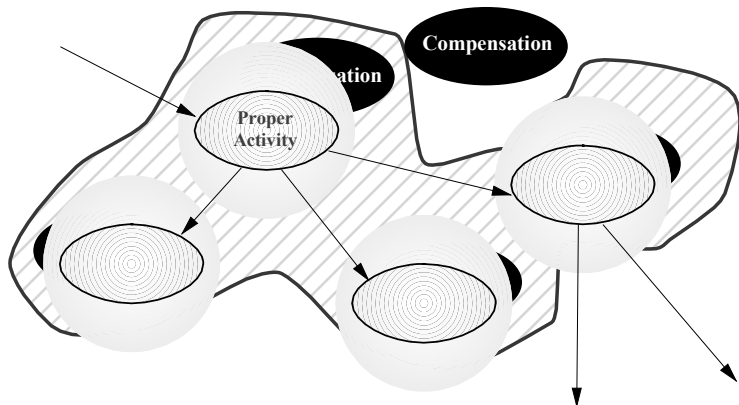
Compensation: Examples

- Compensation attempts to repair actions that cannot be simply undone
 - E.g. an already committed update on a database, sending an e-mail, dispensing money by an automatic teller machine, etc.
- Compensation action is often dependent on context
 - E.g. writing an offer and sending it via mail to a customer
 - If letter is still in outbasket, simply remove it from outbasket
 - If letter is already received by the customer, write and send a countermanding letter
- Compensation often cannot recreate the same state that existed before the proper action had been performed
 - E.g. canceling a flight might cost a cancellation fee
 - Even more complicated, the cancellation fee might depend on the point in time, i.e. it is higher the later the cancellation is requested

© IBM

Frank Leymann

Compensation Spheres



The diagram illustrates 'Compensation Spheres' as a collection of four overlapping circles, each containing a smaller circle with concentric lines. One of these circles is labeled 'Proper Activity'. A larger, irregularly shaped area with diagonal hatching encompasses all four circles. A black oval labeled 'Compensation' is positioned above the circles. Arrows point from the 'Proper Activity' circle to the other three circles, and from the 'Compensation' oval to the top-right circle. Two arrows also point downwards from the bottom-right circle.

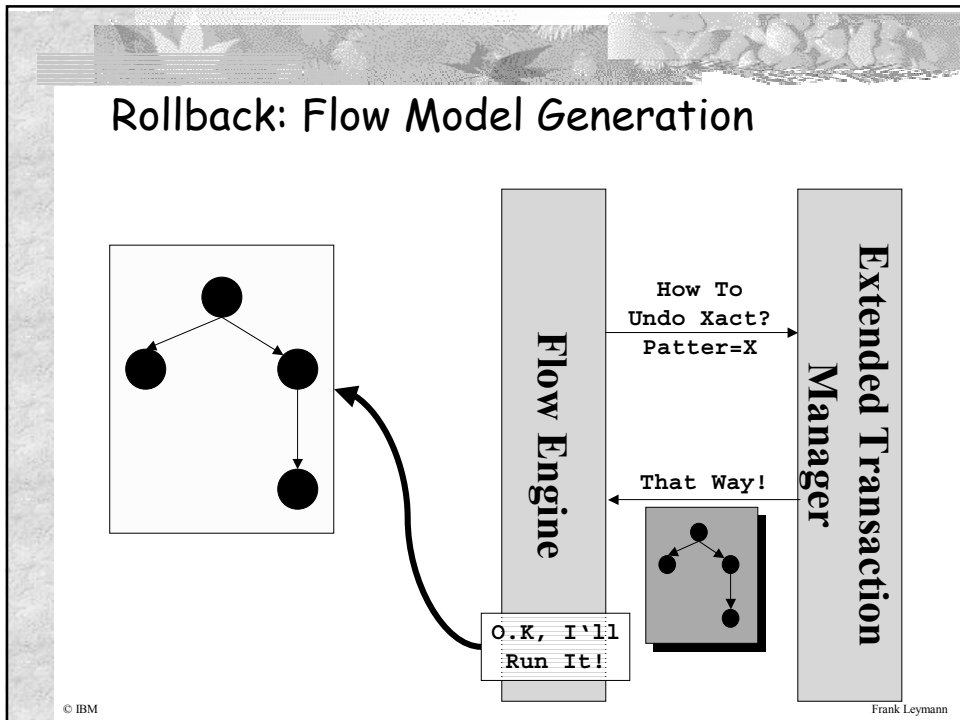
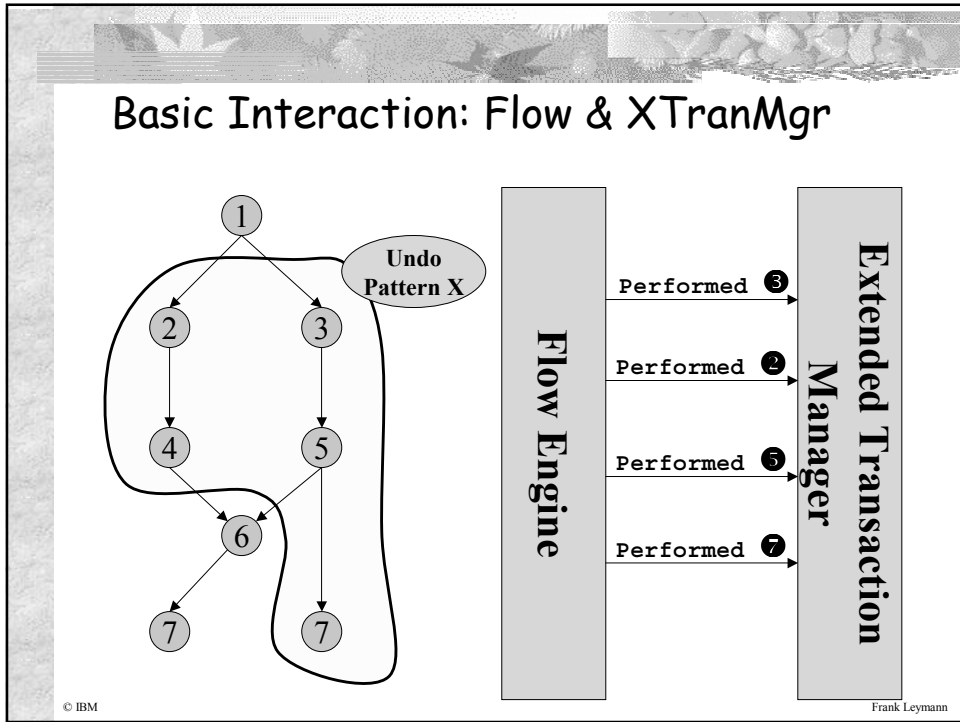
- Set of activities that must complete successfully as a whole
 - Otherwise it must be undone semantically

© IBM Frank Leymann

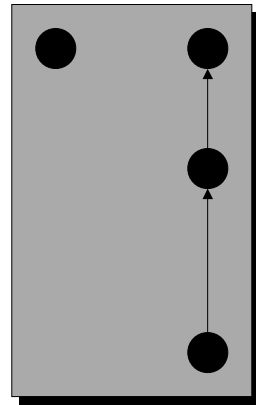
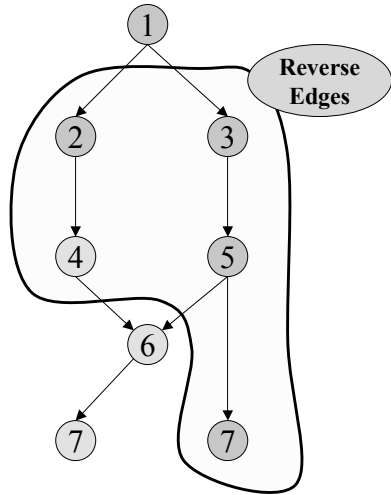
Extended Transaction Manager

- An „Extended Transaction Manager“ is for Extended Transactions what a Transaction Manager is for Distributed Transactions (e.g. the TM in X/Open DTP)
 - Manages participants
 - Signals what to do at EoT
- But it does not run and manage the participating transactions itself!

© IBM Frank Leymann



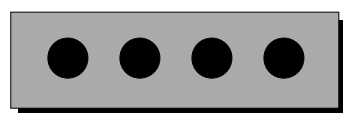
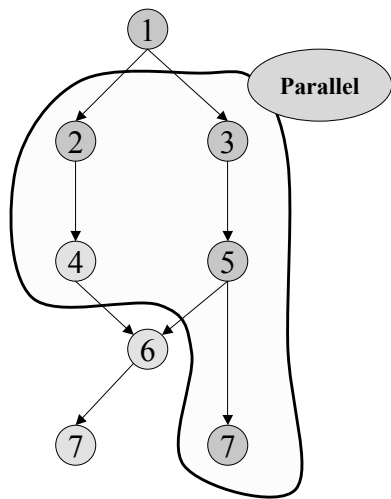
Undo Pattern: Example 1



© IBM

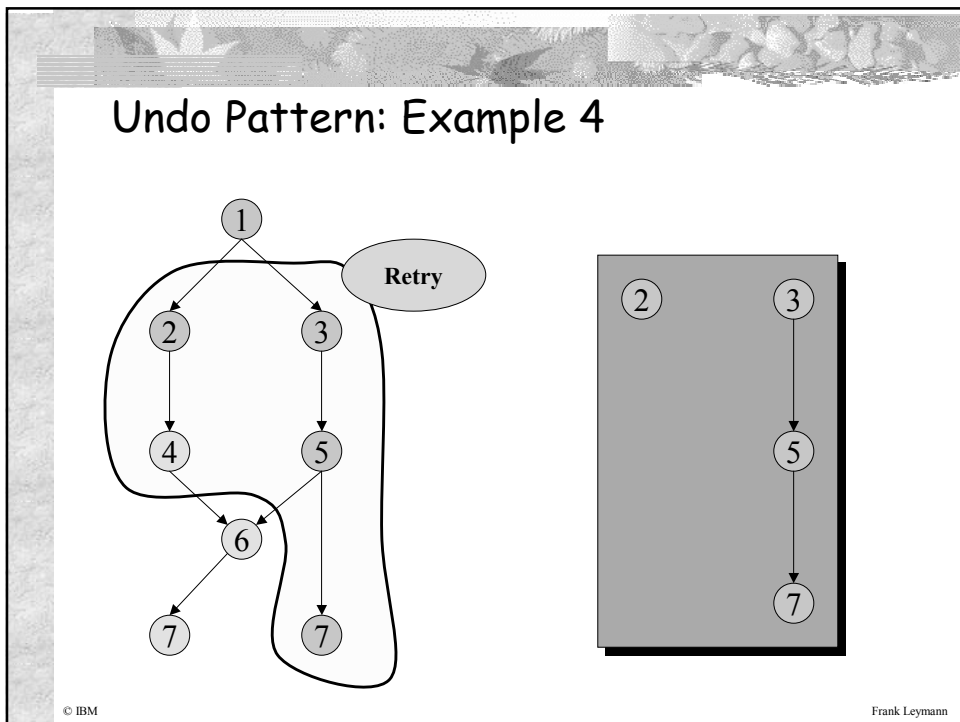
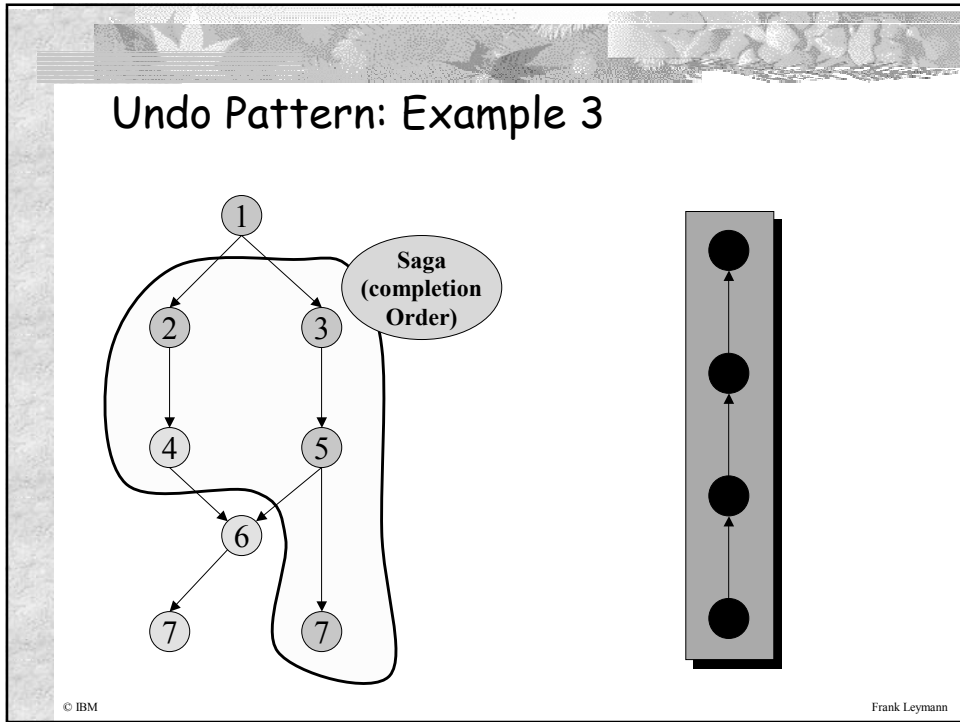
Frank Leymann

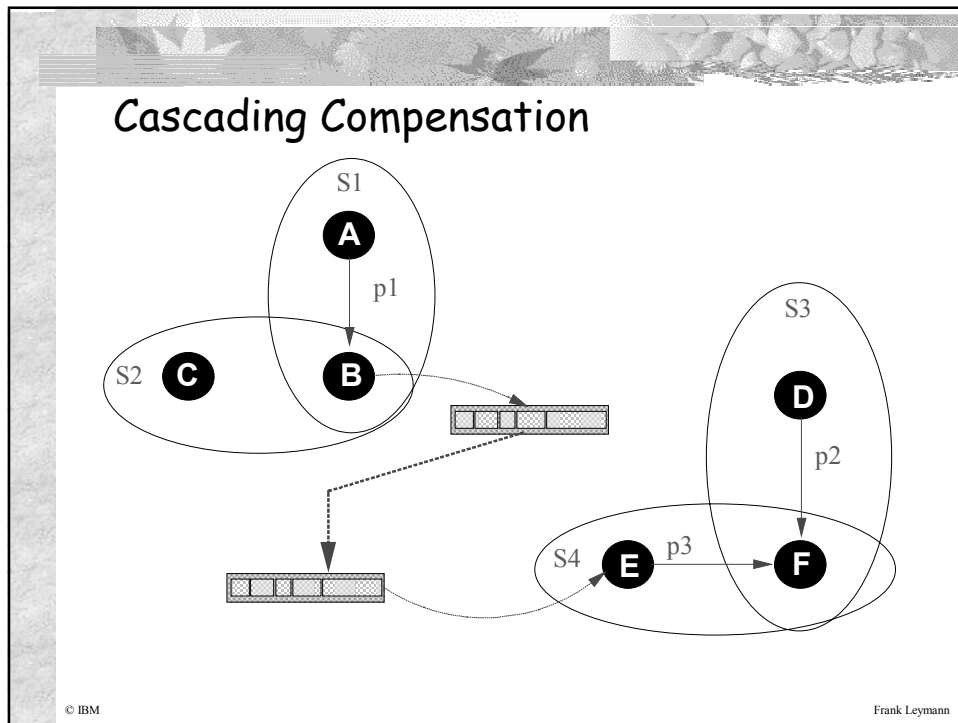
Undo Pattern: Example 2



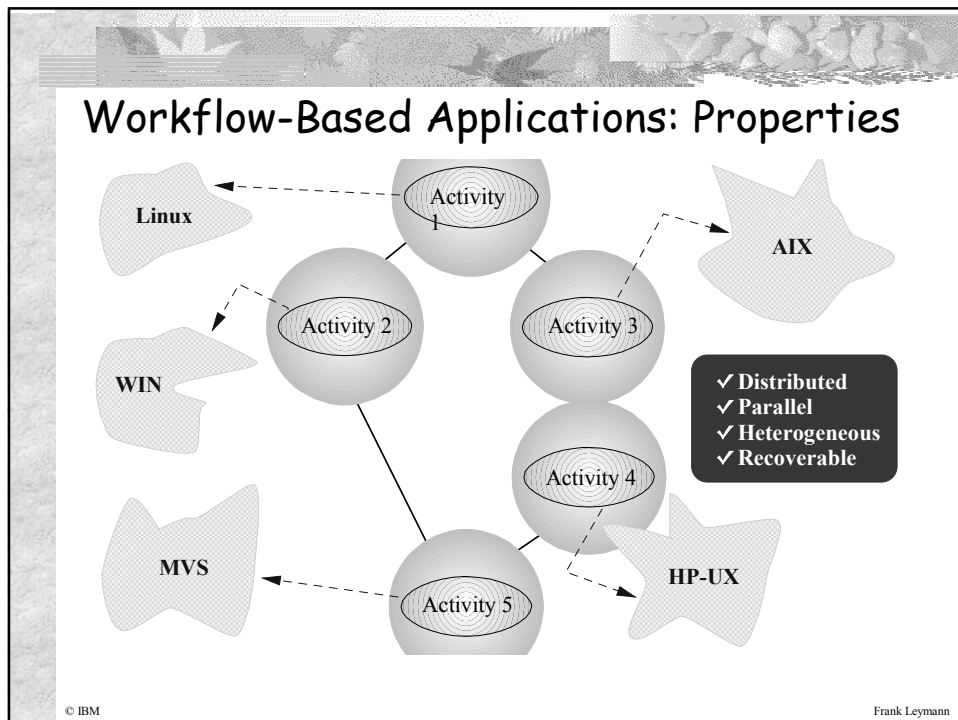
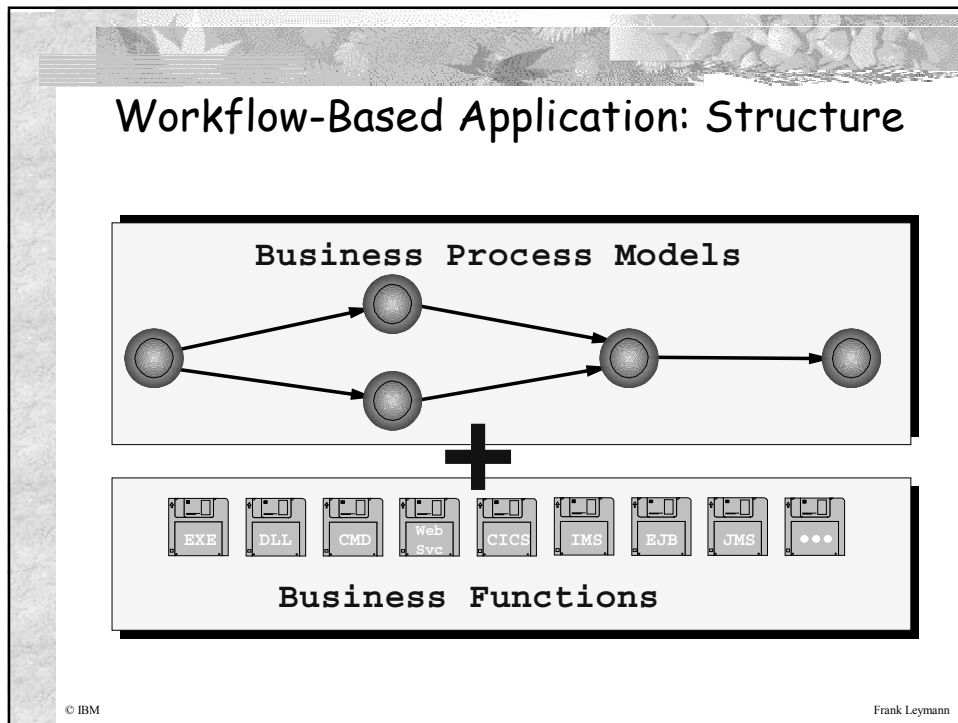
© IBM

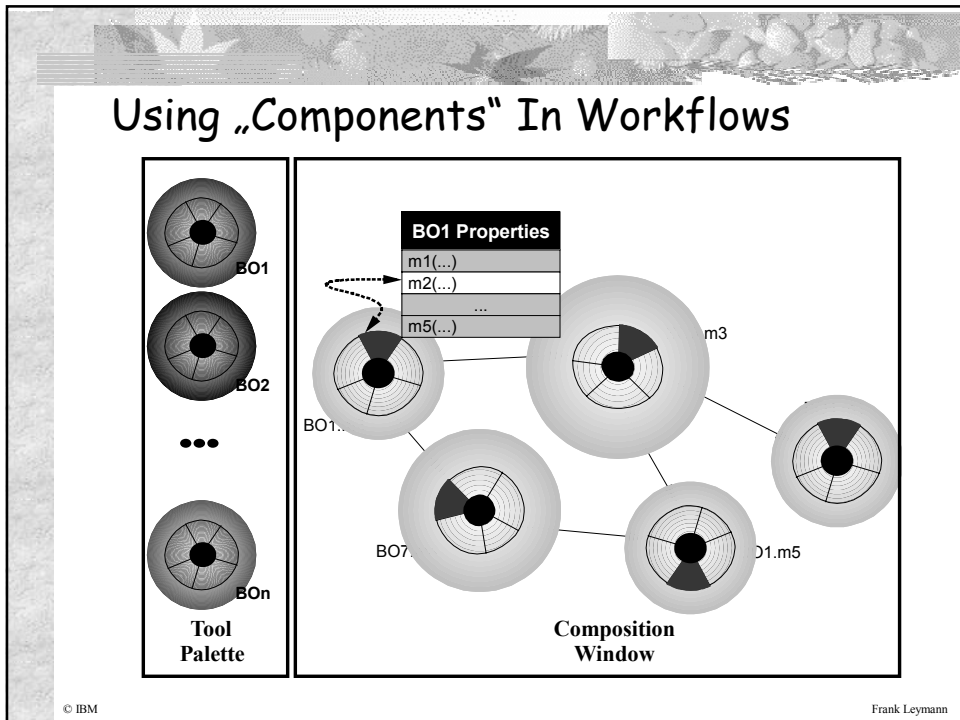
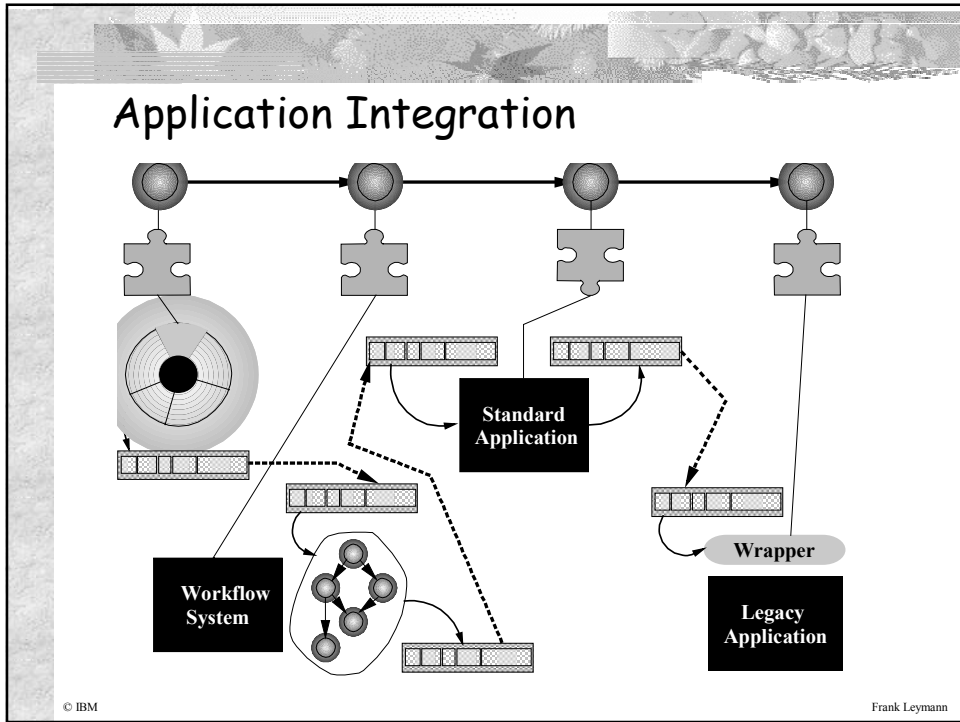
Frank Leymann

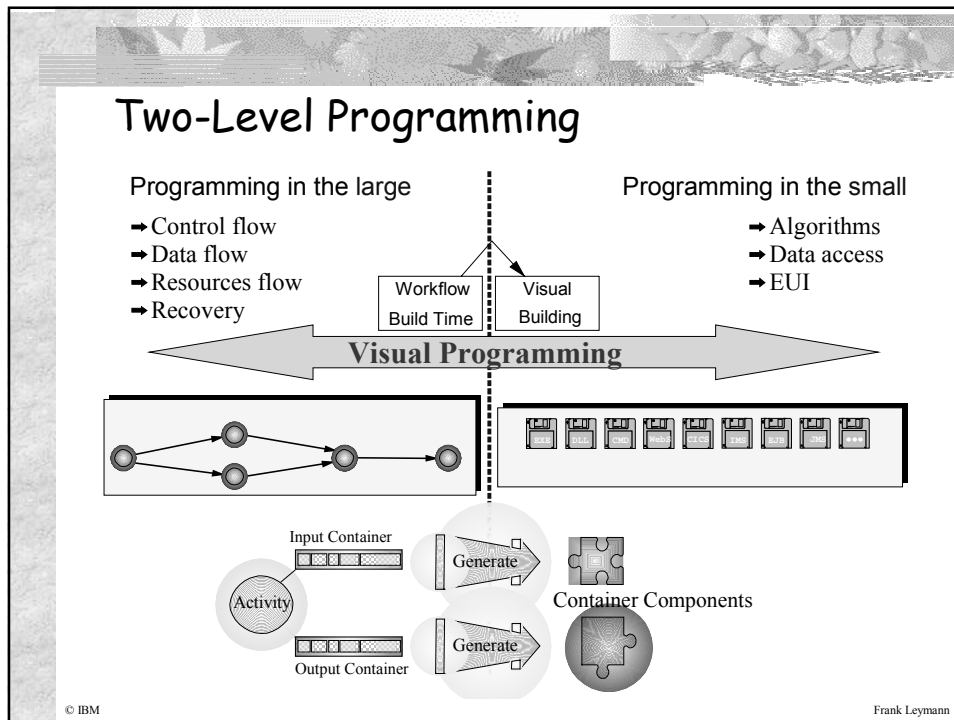




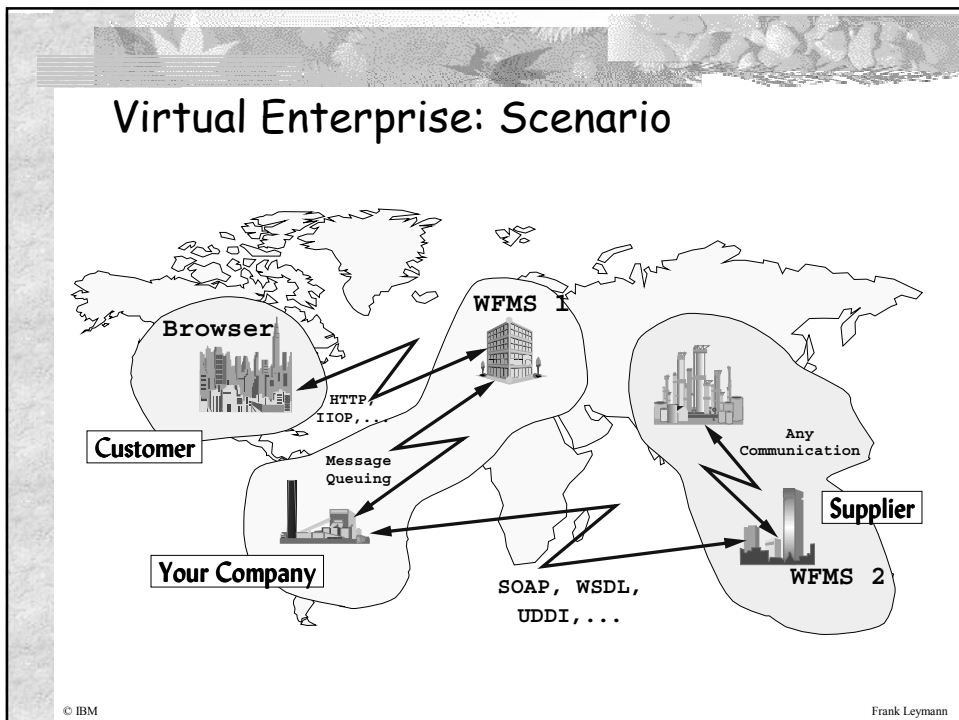
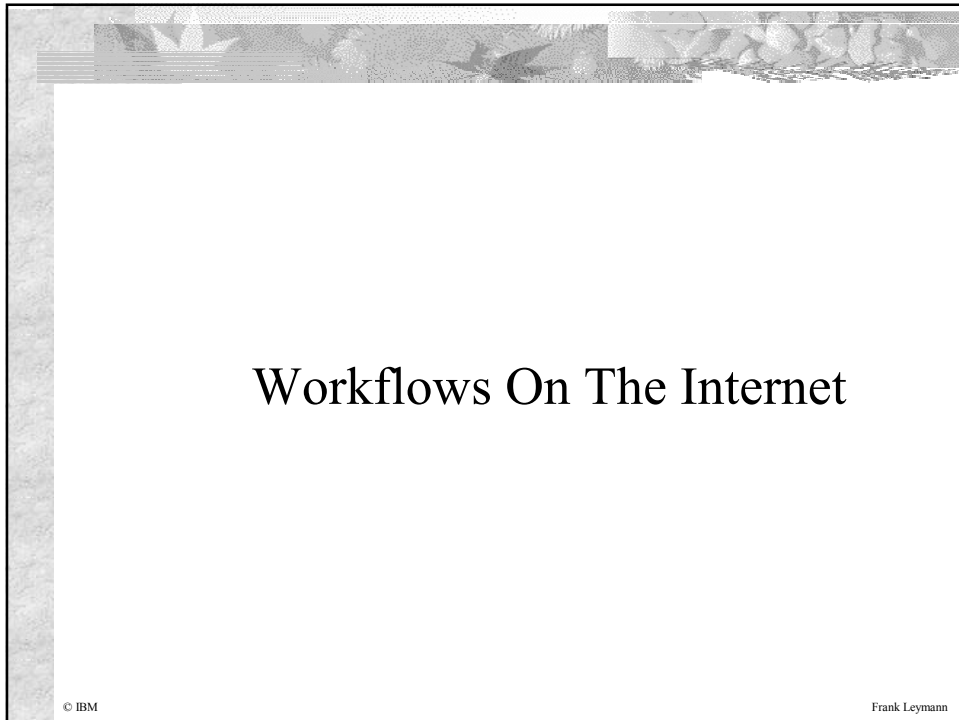
- ## Agenda
1. History
 2. Workflow Basics
 3. Some Plumbing
 4. WFMS Architecture
 5. Transactional Workflows
 - 6. Application Structures**
 7. Web Services
- © IBM Frank Leymann







- ## Agenda
1. History
 2. Workflow Basics
 3. Some Plumbing
 4. WFMS Architecture
 5. Transactional Workflows
 6. Application Structures
 - 7. Web Services**
- © IBM Frank Leymann



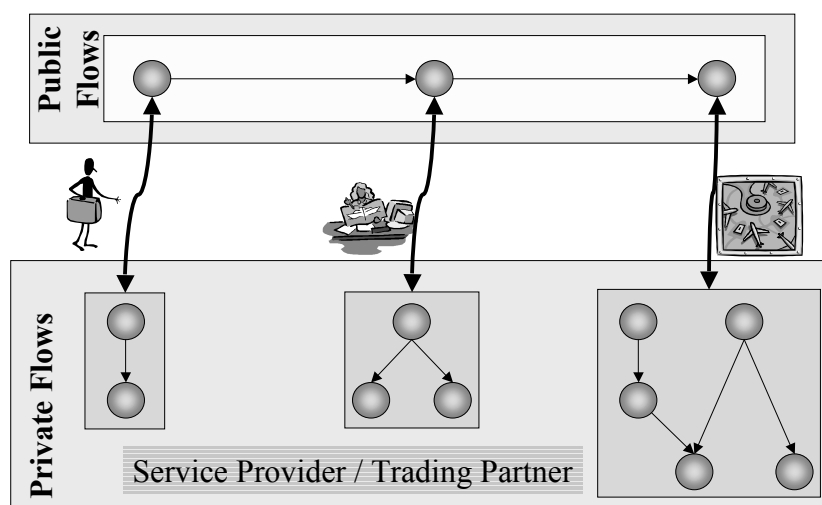
Workflows Between Partners

- It's painful to monitor, control,... manually the processing at partner sides and react on exceptional situations
- Workflow technology can help
 - Model regular partner interactions
 - „Conversations“ are different (!!!)
 - „Arbitrary“ exchange of messages between two partners to complete a single interaction in the overall flow
 - Specify exception handling
 - Time out
 - Not-performed requests
 - ...

© IBM

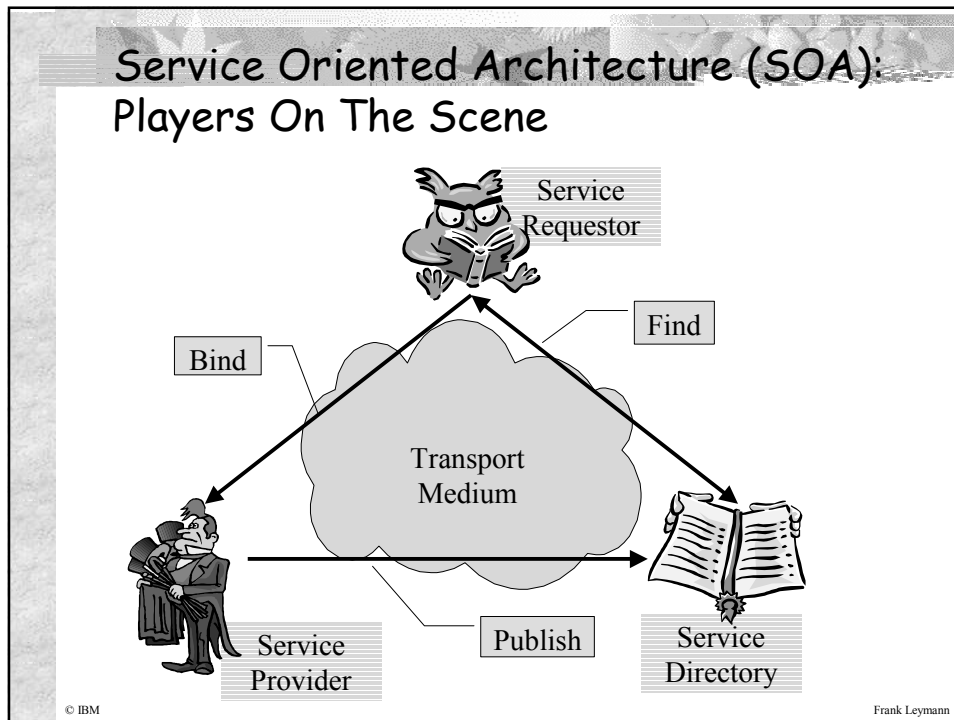
Frank Leymann

Types Of Workflows

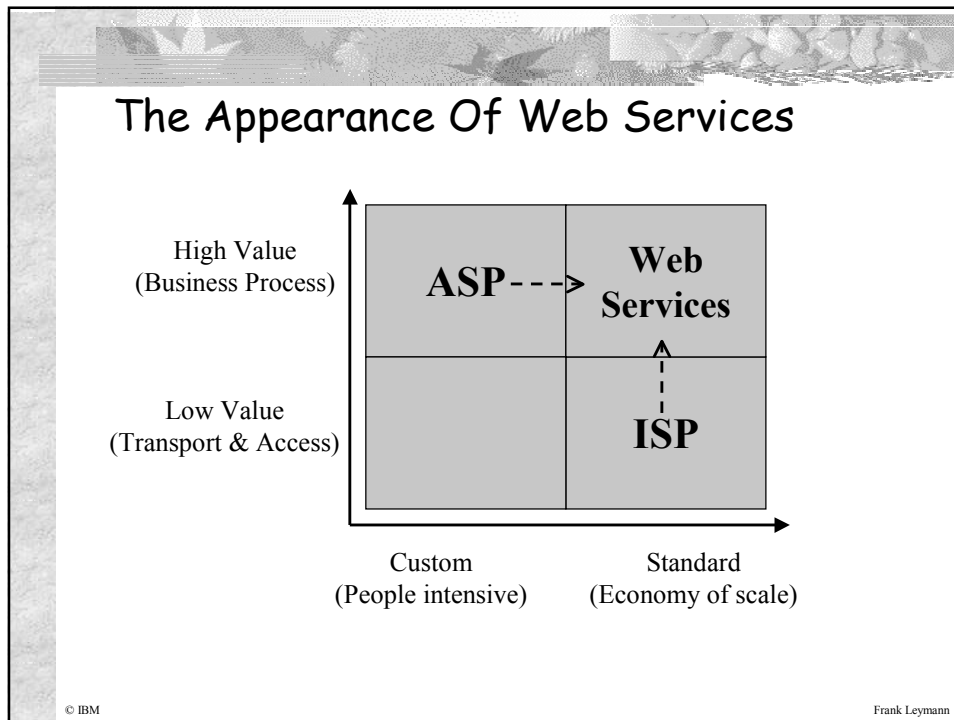


© IBM

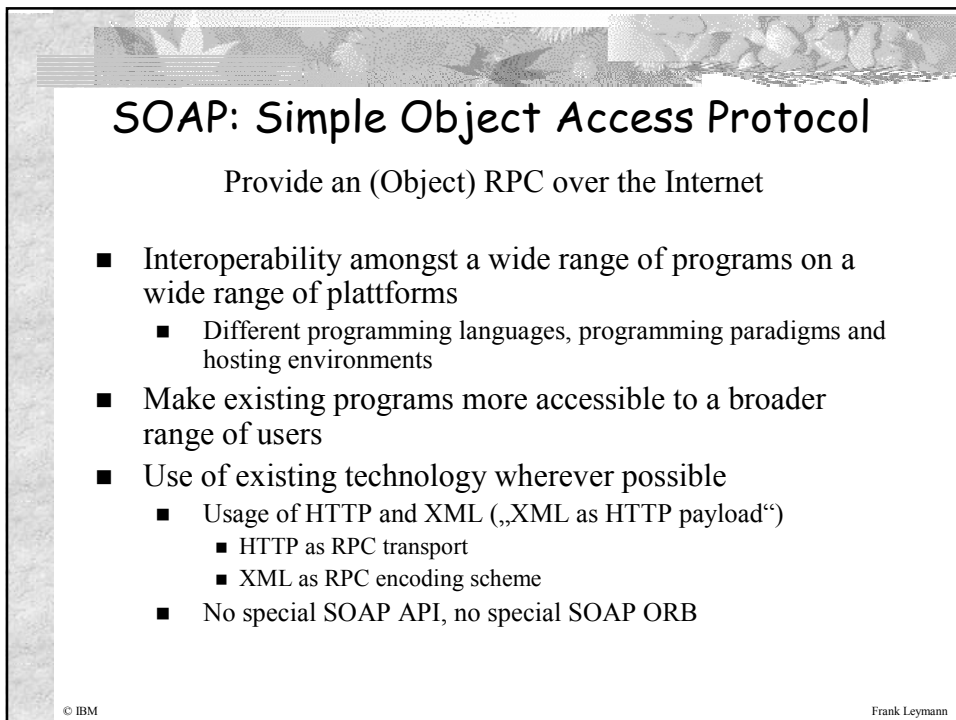
Frank Leymann

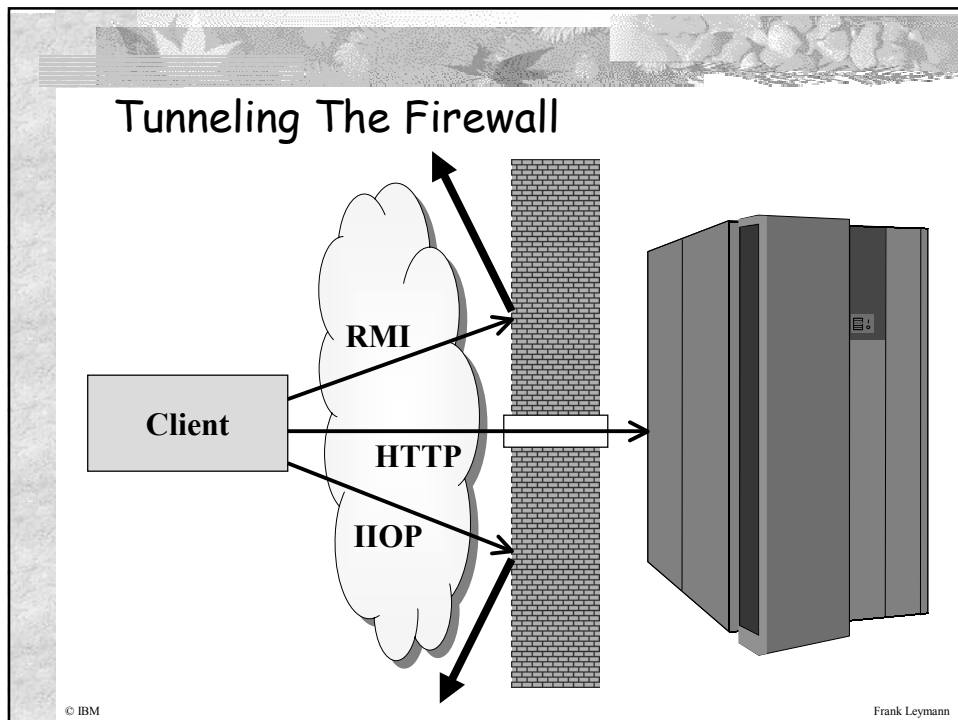
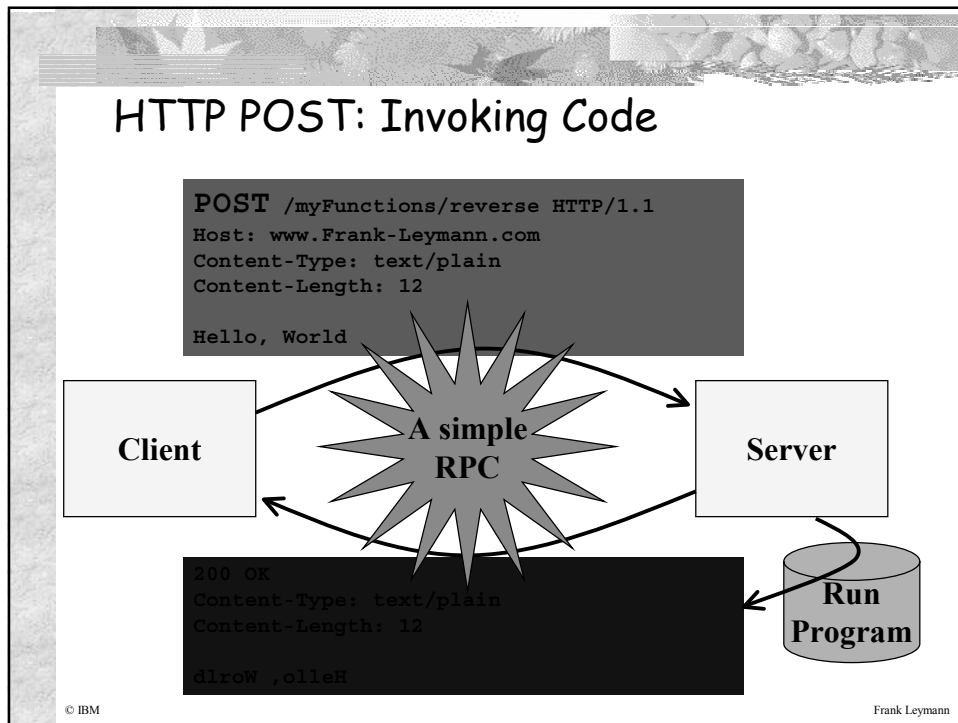


- ### Consequences
- You can...
 - ...outsource „services“ (e.g. where you are not competitive)
 - ...offer „services“ (e.g. make money with functions & processes you perform for others)
 - These services reach from
 - ...simple, fine grained functions...
 - ...to complete business processes, i.e. coarse grained functions
- Workflow technology is at the heart of it!**
- © IBM Frank Leymann



- ### O.K., But: How To Access Services?
- Firewalls are obstructions to ubiquitous access to services
 - Wide range of different programming languages, programming paradigms and hosting environments,... makes orchestration a nightmare
 - ...
- We need a globally available invocation mechanism!
- © IBM Frank Leymann





An XSD Example

```

<element name="Journey">
  <complexType>
    <all>
      <element name="Stage"
        minOccurs="1" maxOccurs="unbounded">
        <complexType>
          <sequence>
            <element name="Location" type="string"/>
            <element name="Begin" type="date"/>
            <element name="End" type="date"/>
          </sequence>
        </complexType>
      </element>
      <element ref="participants"/>
    </all>
  </complexType>
</element>
    
```

© IBM Frank Leymann

SOAP Request/Response Structure

Request	Response
<div style="display: flex; align-items: center;"> <div style="writing-mode: vertical-rl; transform: rotate(180deg); font-weight: bold; padding-right: 5px;">Endpoint Reference</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Object ID (Which object?)</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Interface ID (Which interface?)</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Method ID (Which method?)</div> </div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px; margin-left: 20px;">Extension Headers (What has been forgotten to be build into the protocol?)</div> <div style="border: 1px solid black; padding: 5px; margin-left: 20px;">Parameter (In and inout parms)</div>	<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px; text-align: center;">Status Code (Did it work?)</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px; text-align: center;">Extension Headers (What has been forgotten to be build into the protocol?)</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px; text-align: center;">Parameter (Out and inout parms)</div>

© IBM Frank Leymann

SOAP Endpoint Reference

IP Host Address
TCP Port No
Object Endpoint ID

209.111.234.34
80
/StockServer/getLastTradePrice

POST /StockServer HTTP/1.1
 Host: 209.111.234.34
 Content-Type: text/xml;
 charset="utf-8"
 Content-Length: nnnn
 <SOAP-ENV:Body>
 <m:GetLastTradePrice xmlns:m="Some-URI">

...

© IBM

Frank Leymann

A Simple SOAP RPC

```

POST /StockQuote HTTP/1.1
Host: www.stockquoteserver.com
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:GetLastTradePrice xmlns:m="Some-URI">
      <symbol>DIS</symbol>
    </m:GetLastTradePrice>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
    
```

Method Name

Input Parameter

© IBM

Frank Leymann

A Simple SOAP Response

```
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="Some-URI">
      <Price>34.5</Price>
    </m:GetLastTradePriceResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Standard
Suffix

© IBM

Frank Leymann

SOAP Header

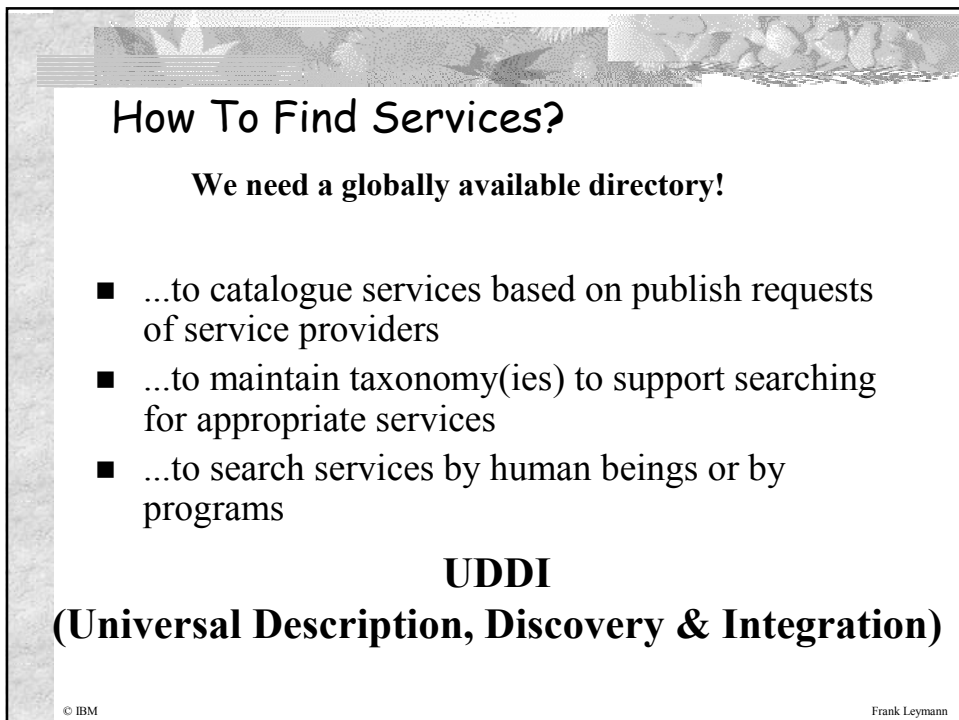
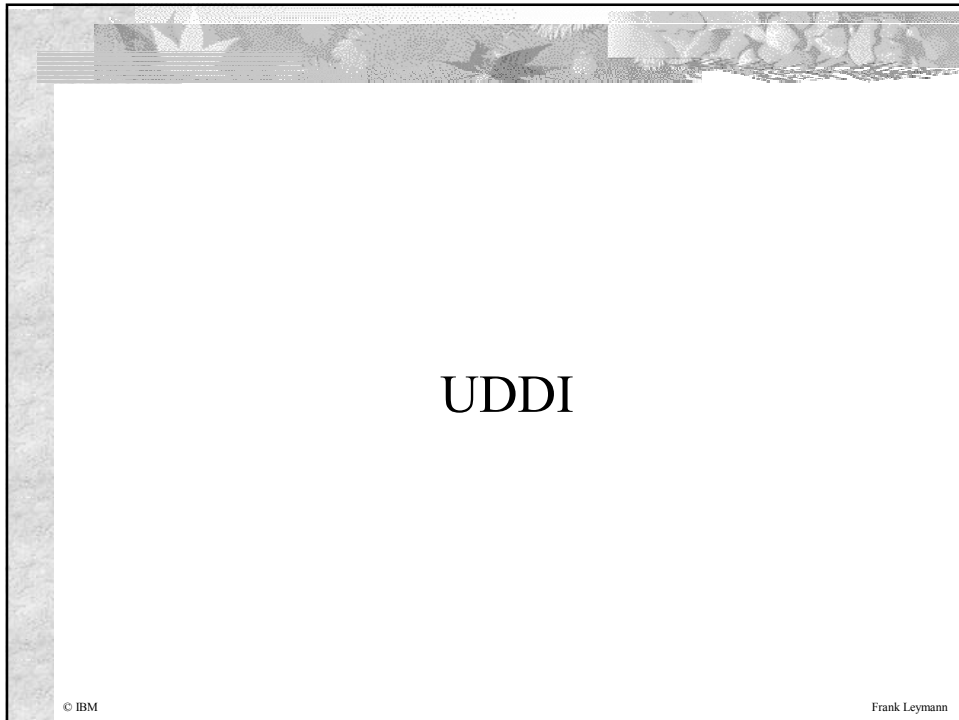
```
POST /StockQuote HTTP/1.1
Host: www.stockquoteserver.com
Content-Type: text/xml;
charset="utf-8,,
Content-Length: nnnn
SOAPAction: „Some-URI“
```

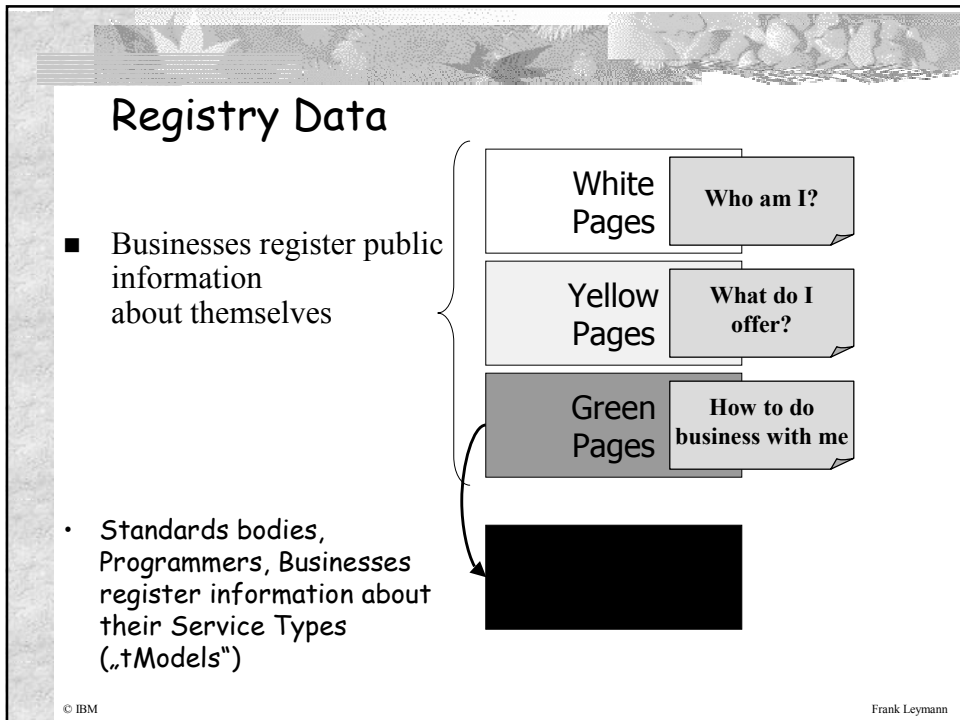
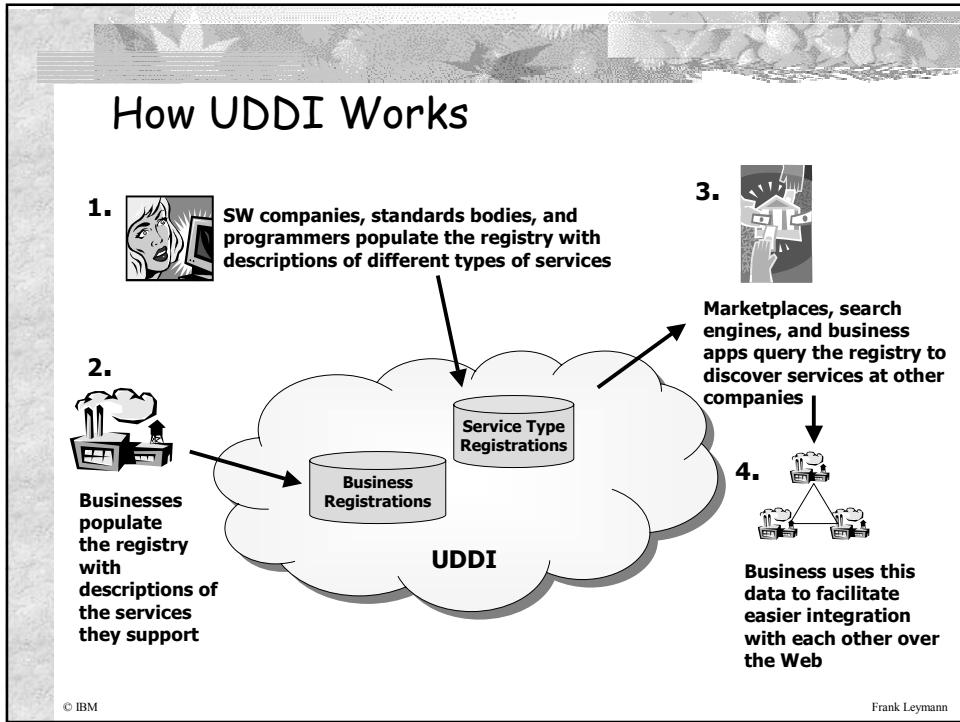
```
<SOAP-ENV:Envelope xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/"
SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
  <SOAP-ENV:Header>
    <t:Transaction xmlns:t="some-URI, SOAP-ENV:mustUnderstand="1">
      5
    </t:Transaction>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <m:GetLastTradePrice xmlns:m="Some-URI">
      <symbol>DEF</symbol>
    </m:GetLastTradePrice>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

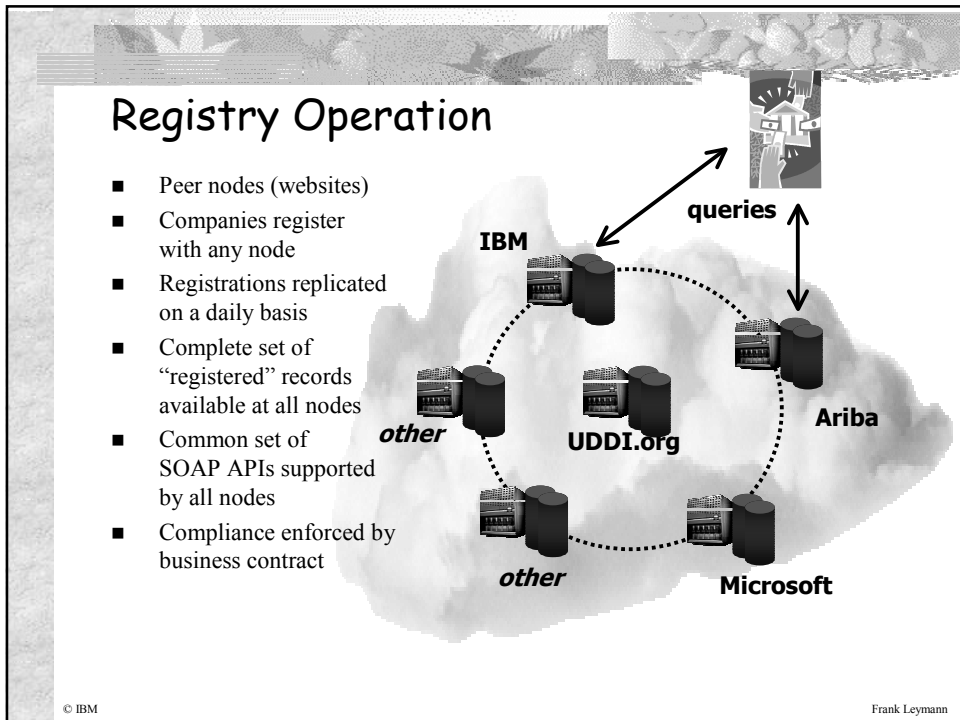
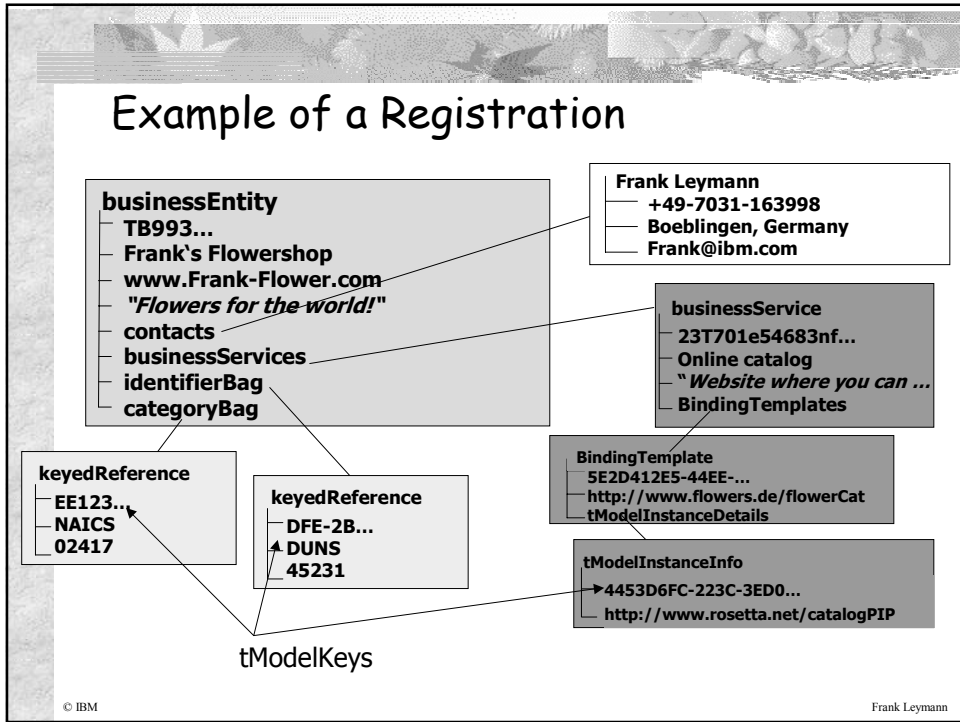
Protocol
Extensions

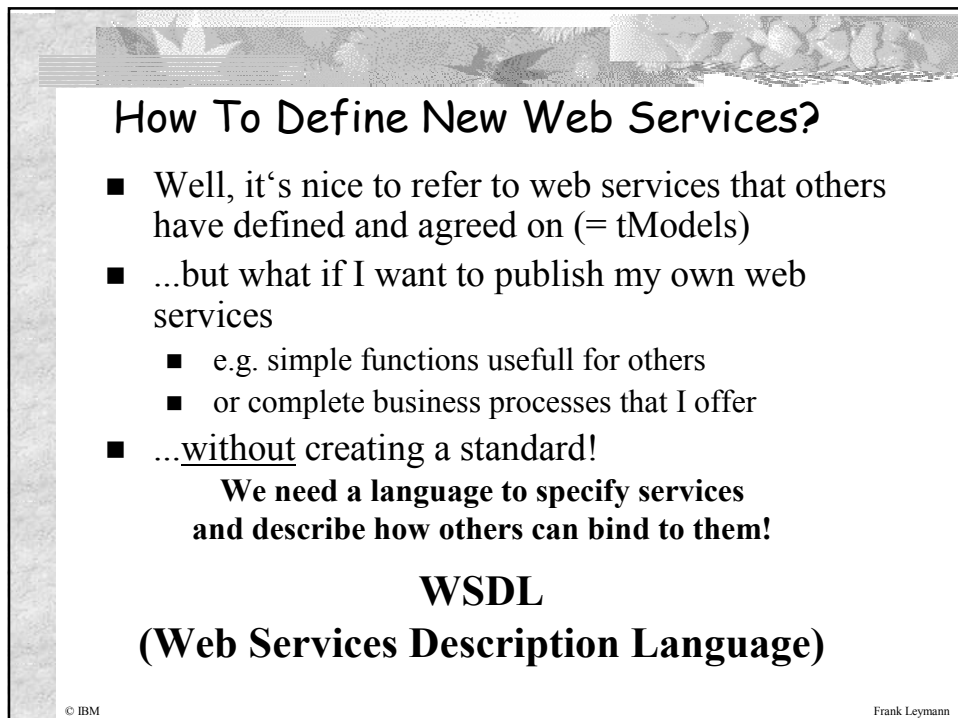
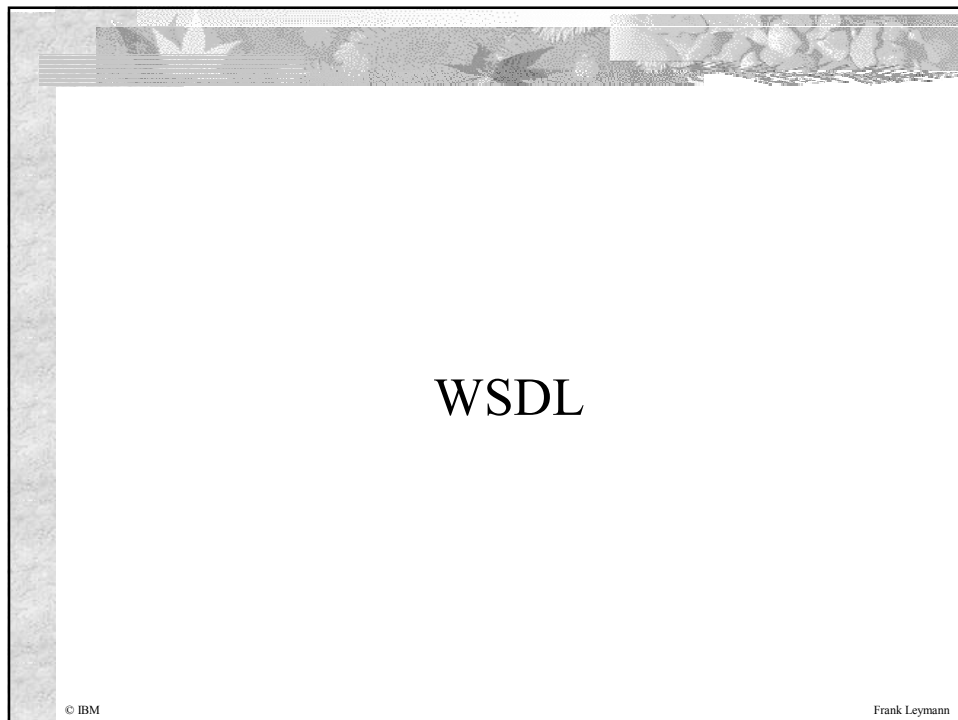
© IBM

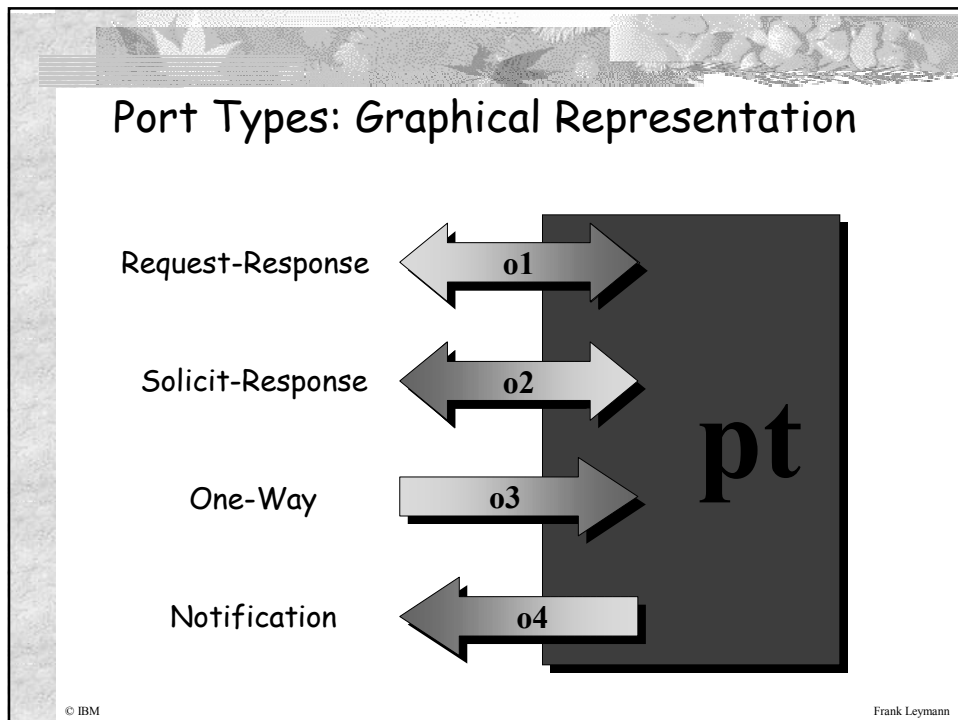
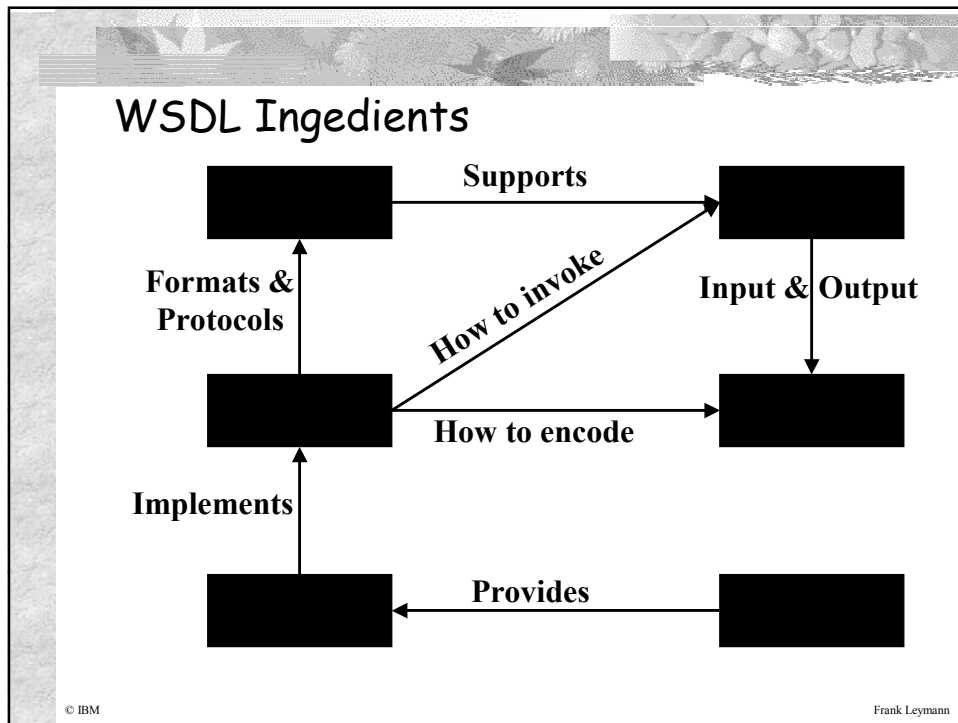
Frank Leymann











Example

```
<?xml version="1.0"?>
<definitions name="StockQuote"
xmlns:tns="http://leymann.com/stockquote.wsdl"
xmlns:xsd="http://leymann.com/stockquote.xsd"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns="http://schemas.xmlsoap.org/wsdl/">
  <message name="GetLastTradePriceRequest">
    <part name="tickerSymbol" element="xsd:string"/>
    <part name="time" element="xsd:timeInstant"/>
  </message>
  <message name="GetLastTradePriceResponse">
    <part name="result" type="xsd:float"/>
  </message>
  <portType name="StockQuotePortType">
    <operation name="GetLastTradePrice">
      <input message="tns:GetLastTradePriceRequest"/>
      <output message="tns:GetLastTradePriceResponse"/>
    </operation>
  </portType> ...
```

© IBM

Frank Leymann

Example

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

© IBM

Frank Leymann

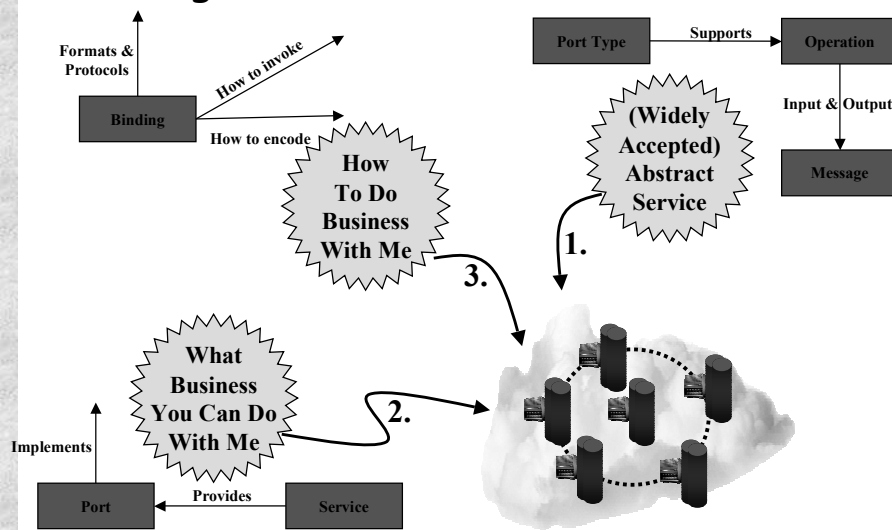
Example

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

© IBM

Frank Leymann

Making Services Available



© IBM

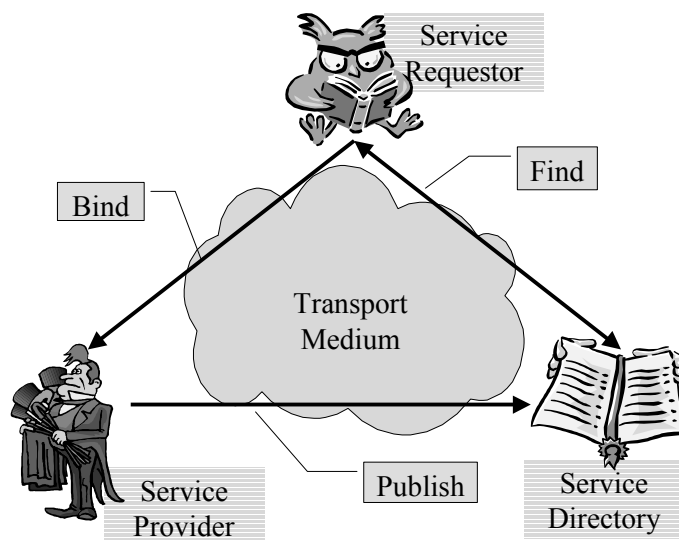
Frank Leymann

Relation To Service Oriented Architecture

© IBM

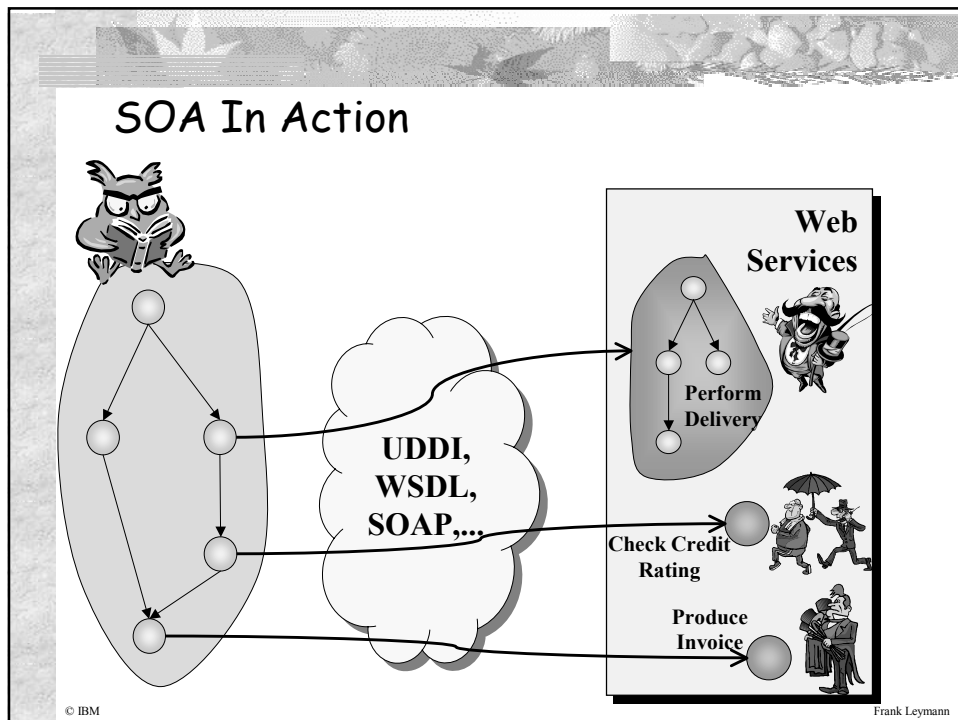
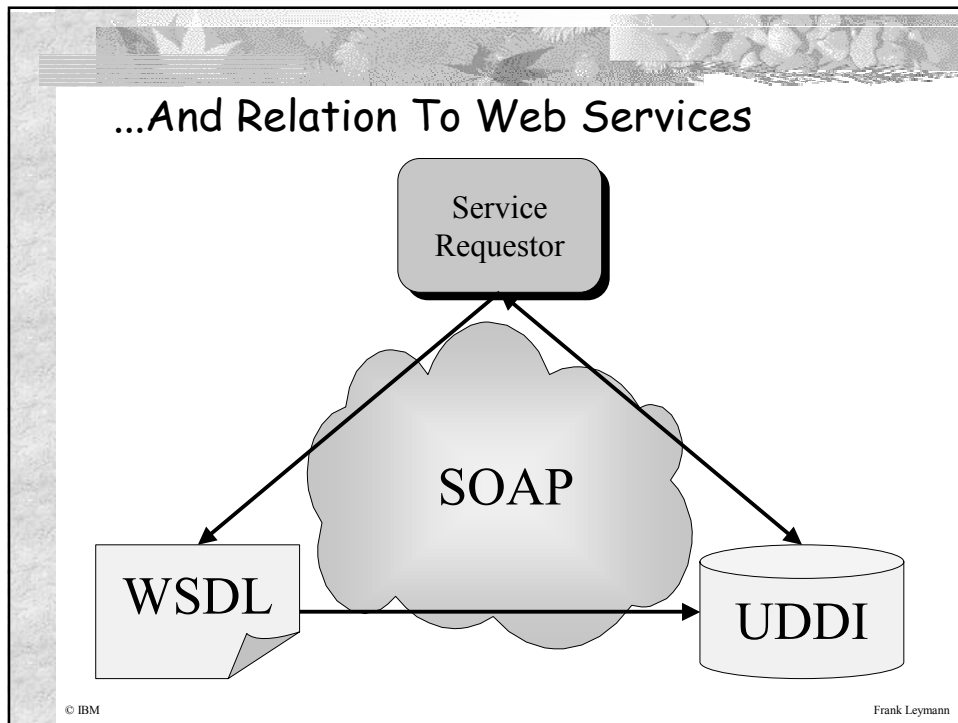
Frank Leymann

SOA Reminder



© IBM

Frank Leymann



What Is Needed?

We are looking for a language that...

- allows to specify flows as web services
- allows to specify flows between web services
- allows to specify new web services as aggregations of web services
- ...and any mixture of these aspects

Web Services Choreography

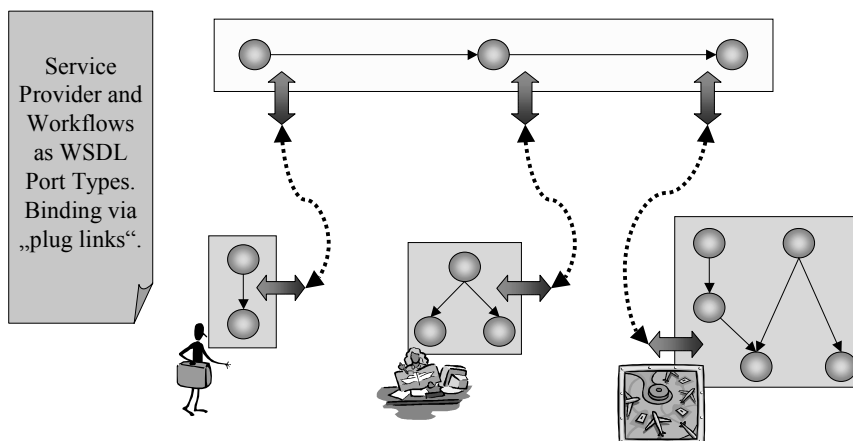
There are proposals for such a language, e.g.

- WSFL (Web Services Flow Language) - IBM
- XLANG - Microsoft

© IBM

Frank Leymann

Relation To Service Oriented Architectures



© IBM

Frank Leymann

From Outer Space: High-Level Scenarios

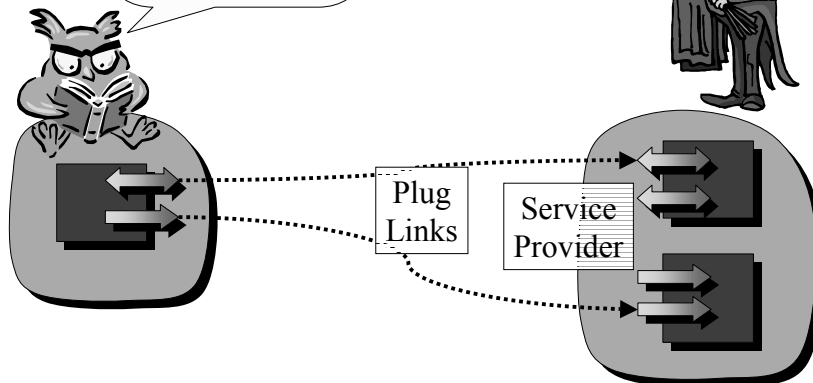
© IBM

Frank Leymann

Establishing Partnerships

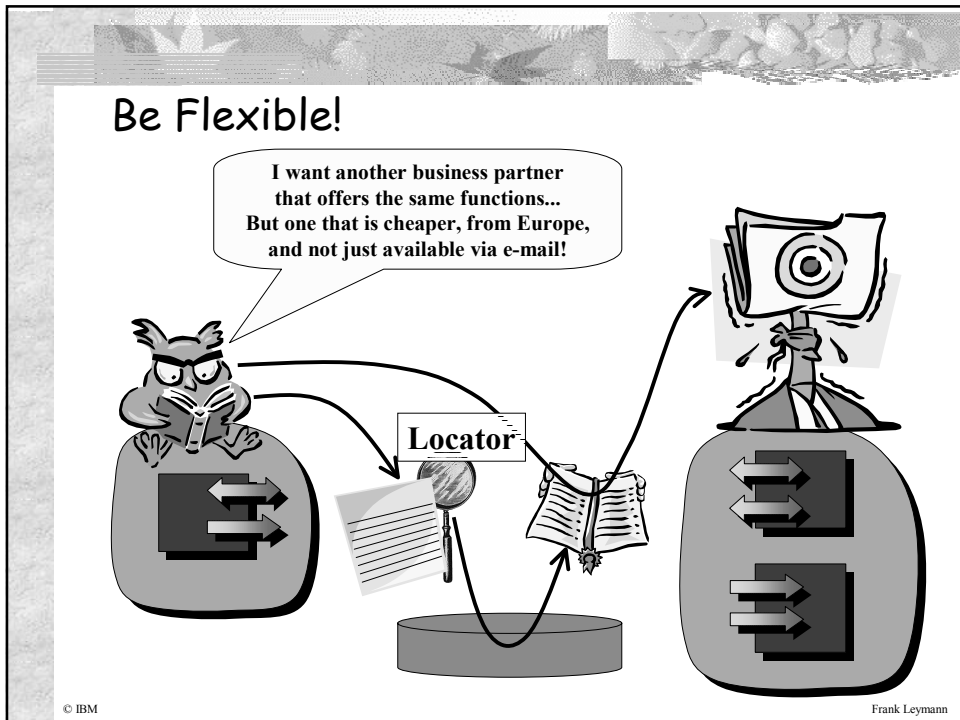
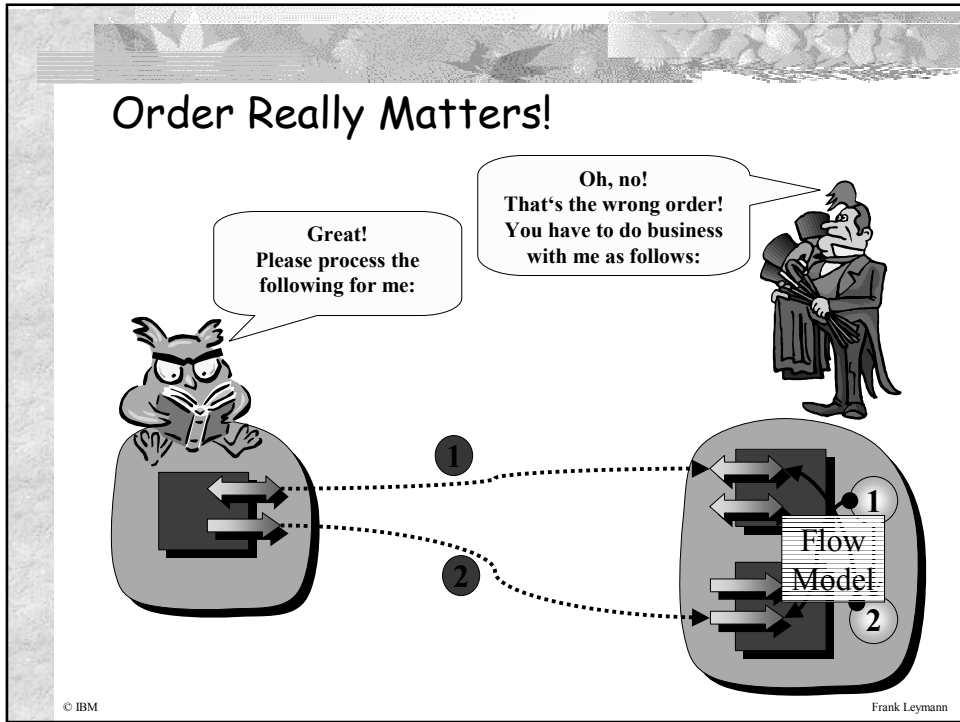
I think I need some help to do my business more efficiently!
Who is providing the following functions?

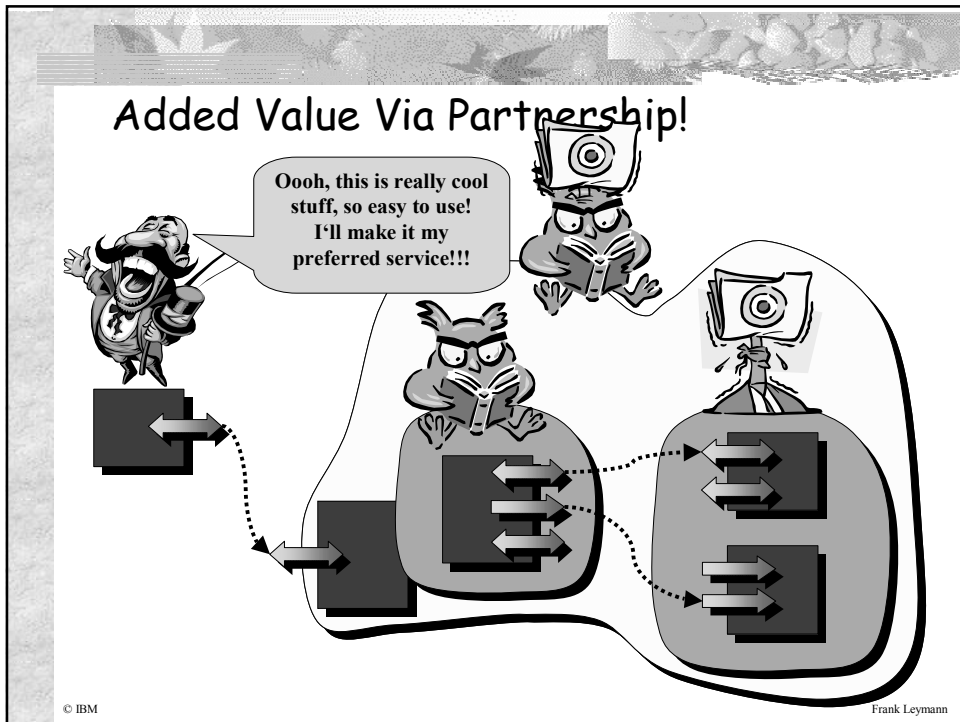
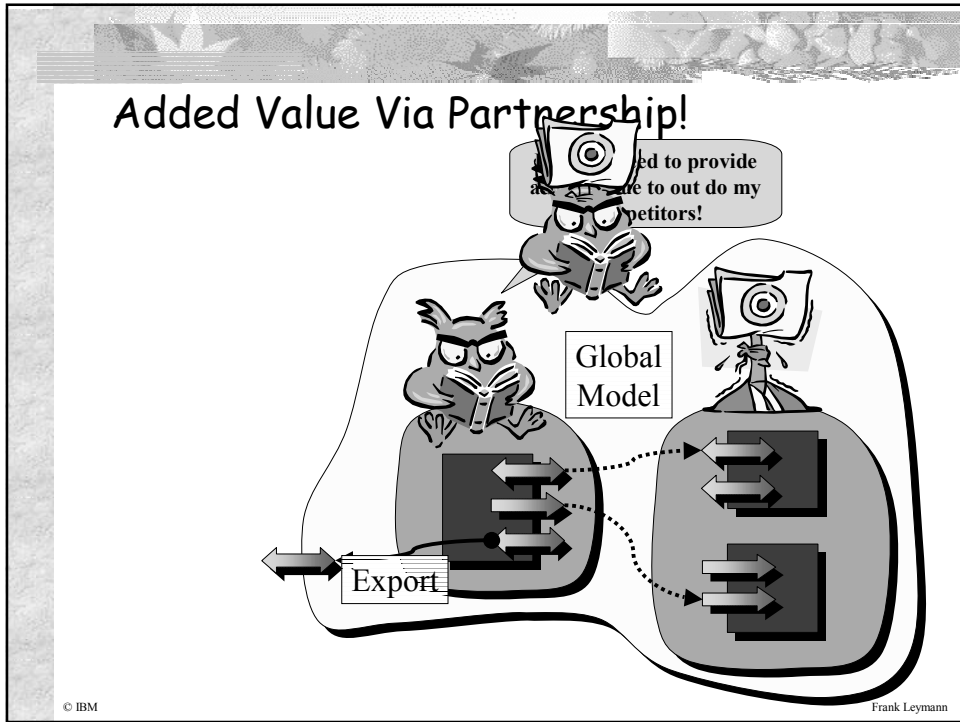
I offer the functions you need!
Here is how you can bind to them!

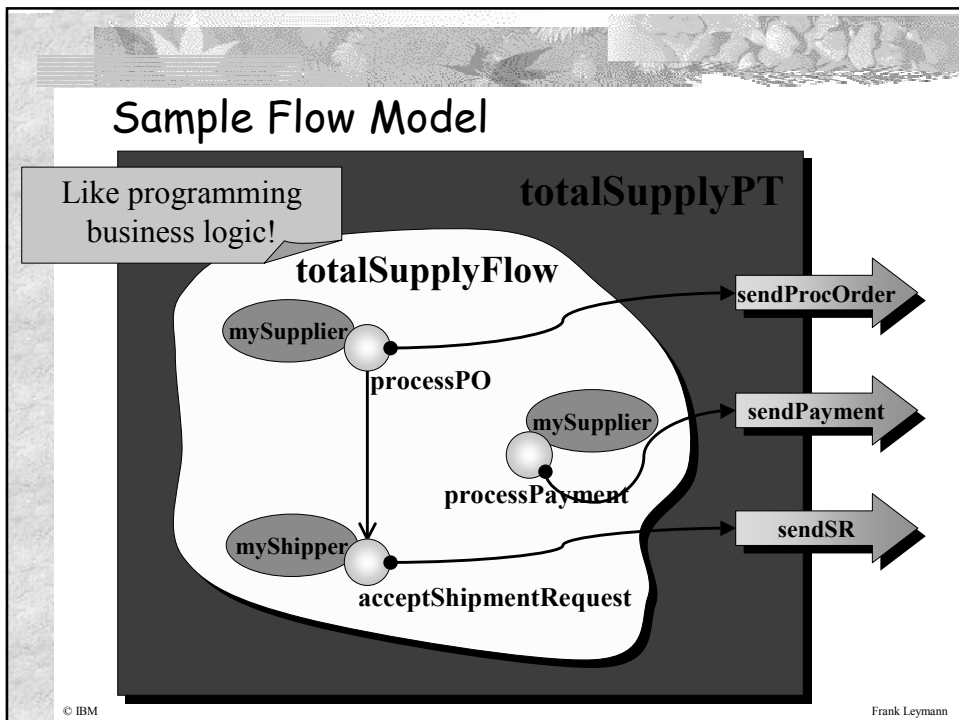
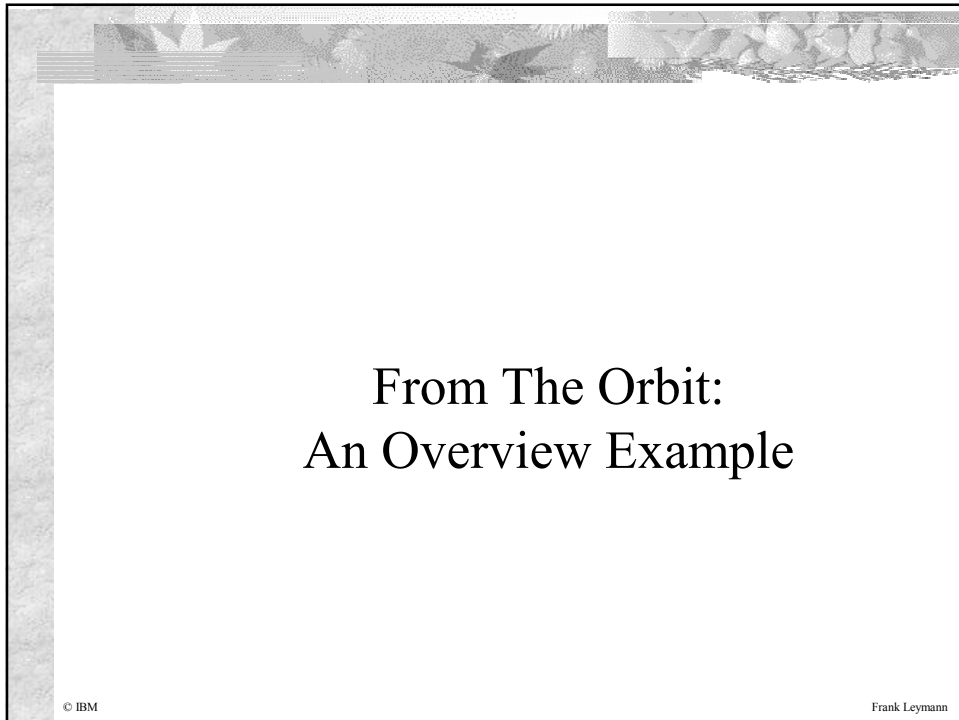


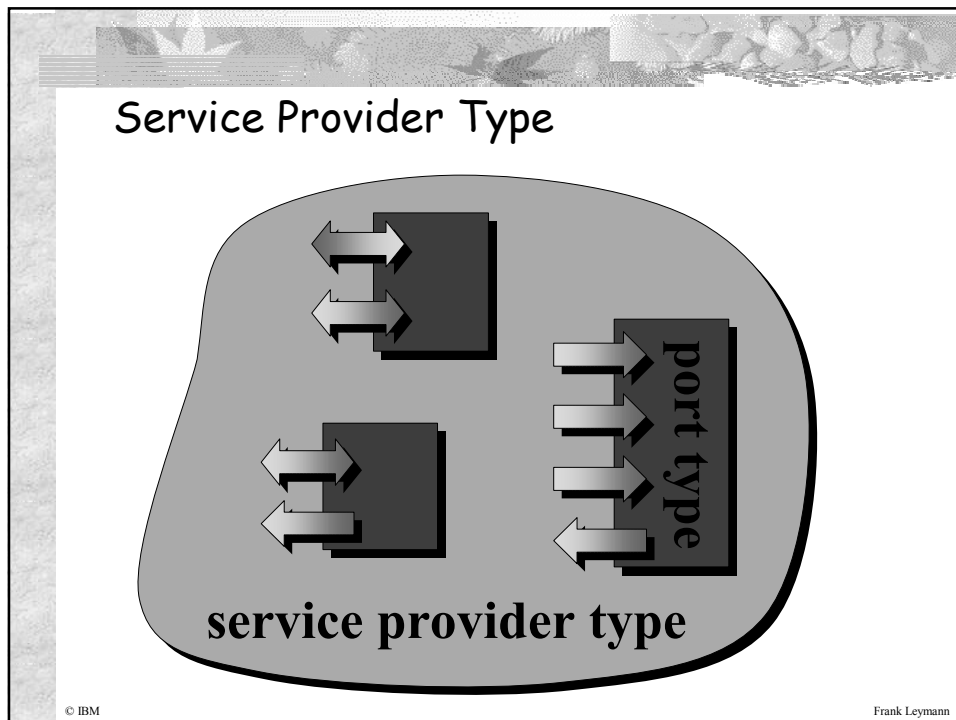
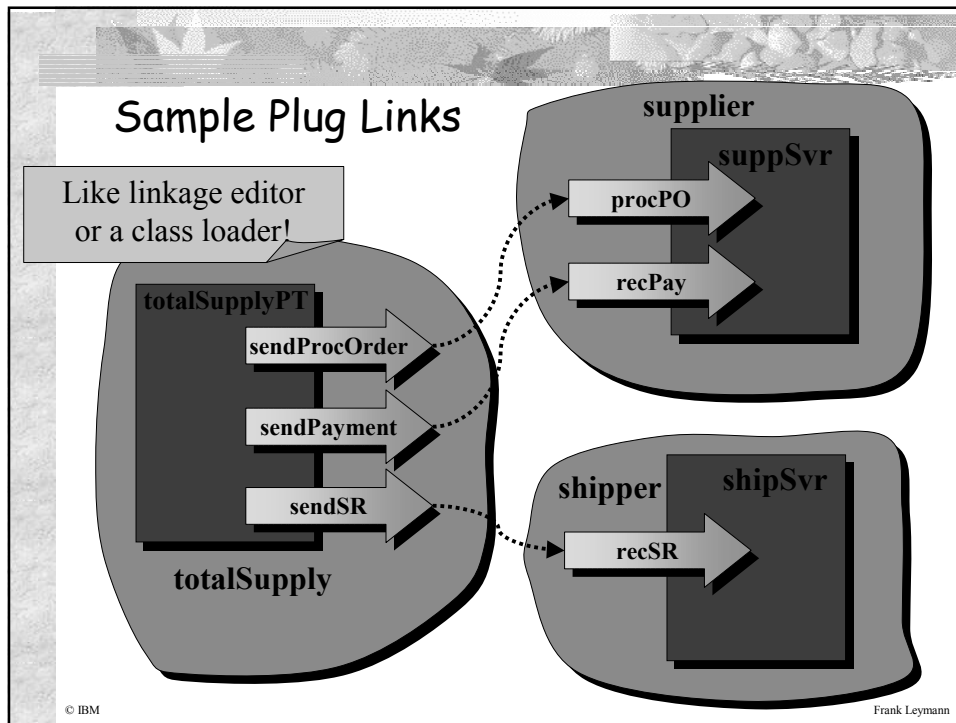
© IBM

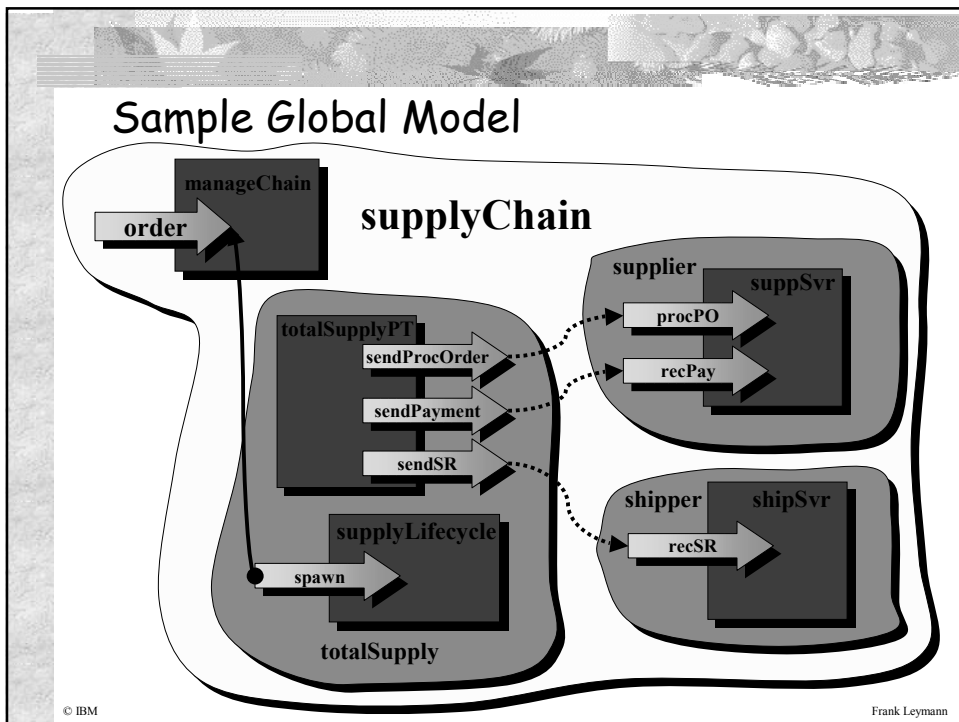
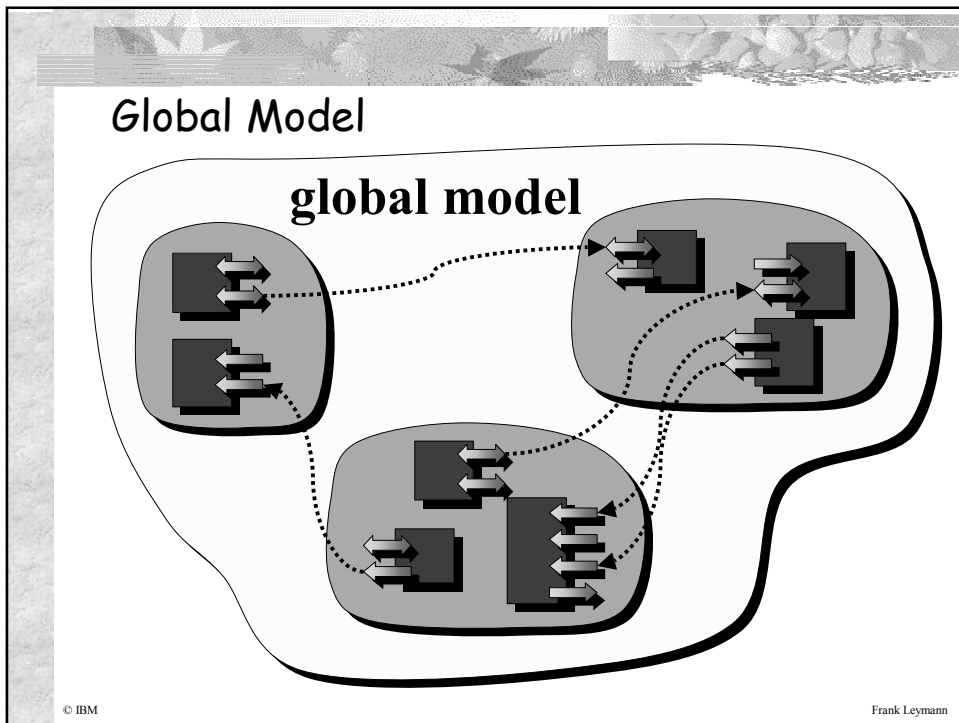
Frank Leymann

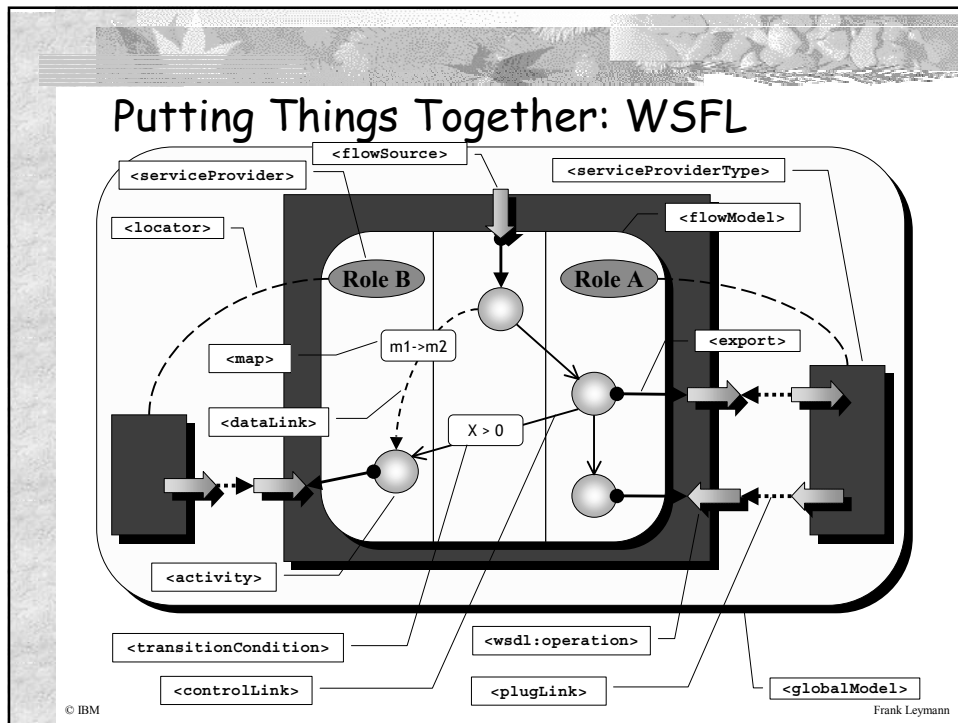






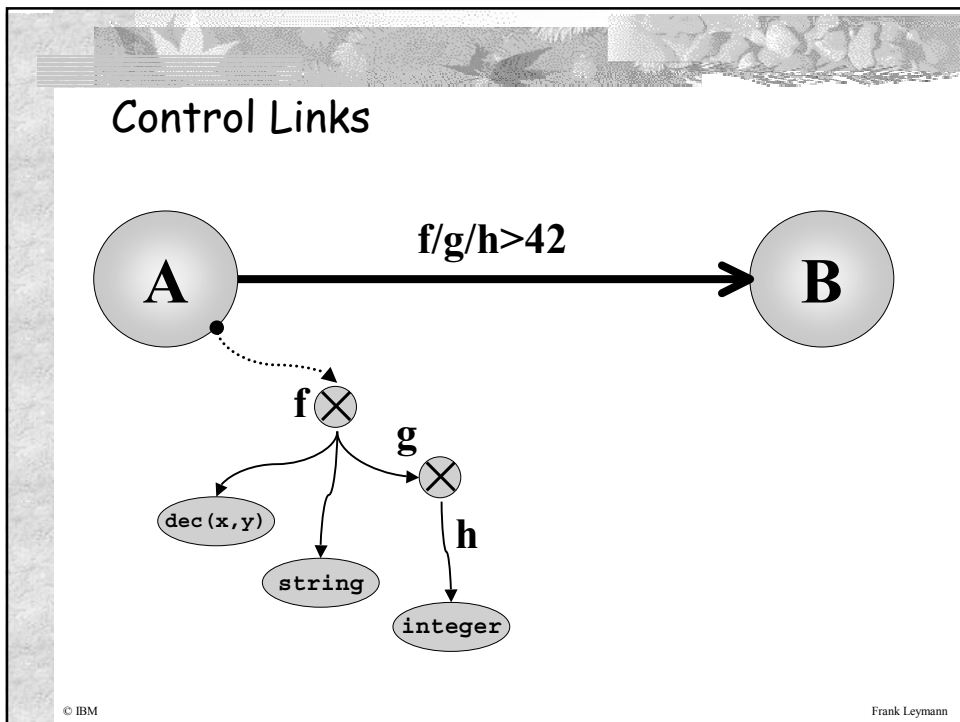
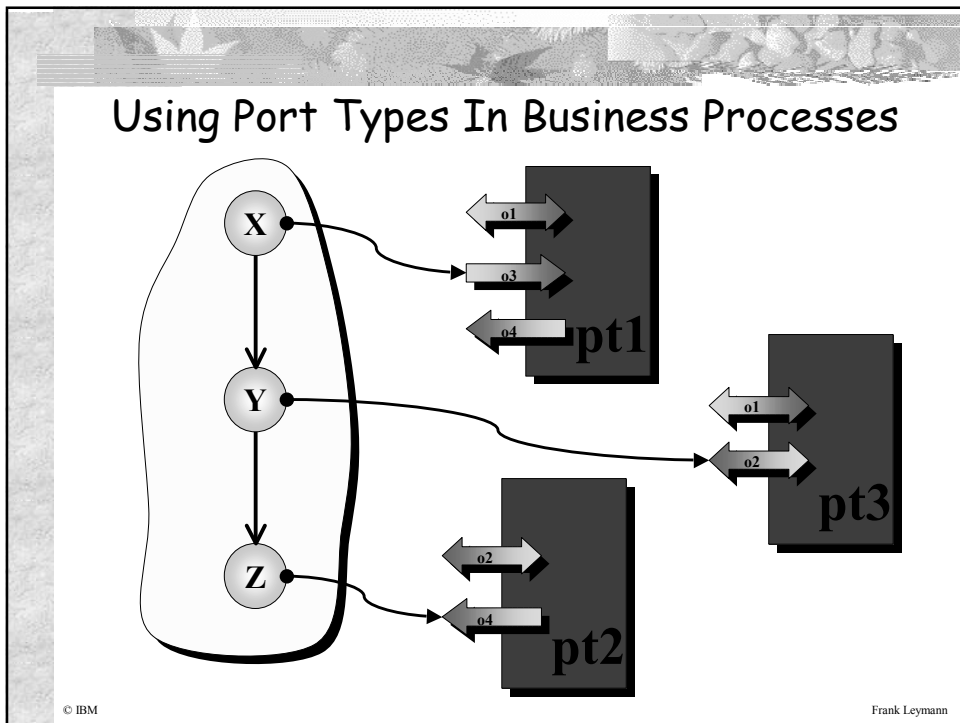


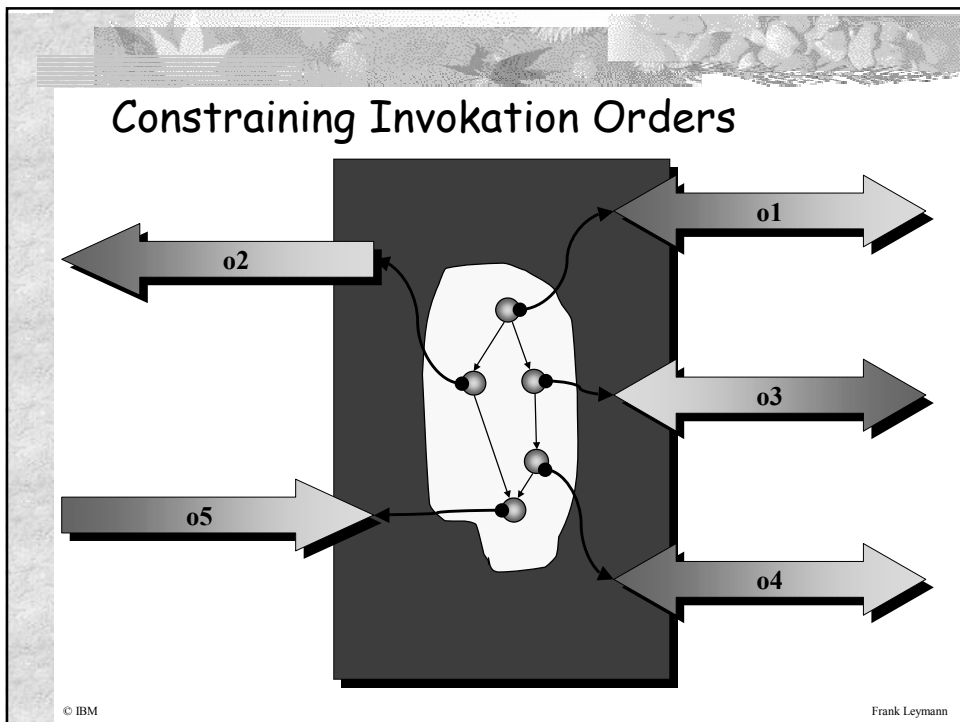
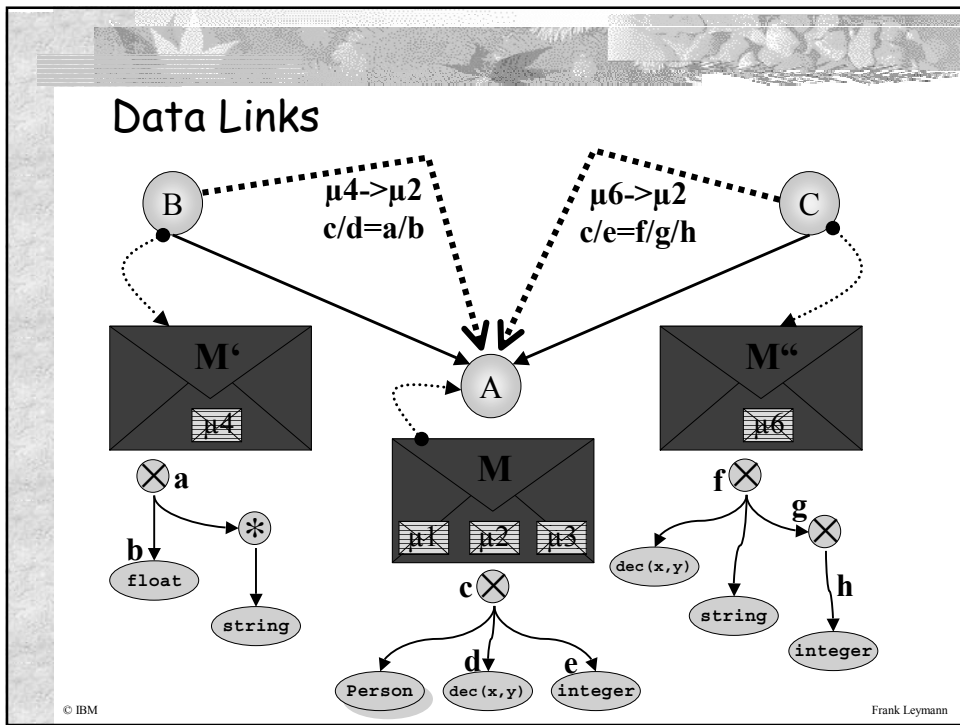


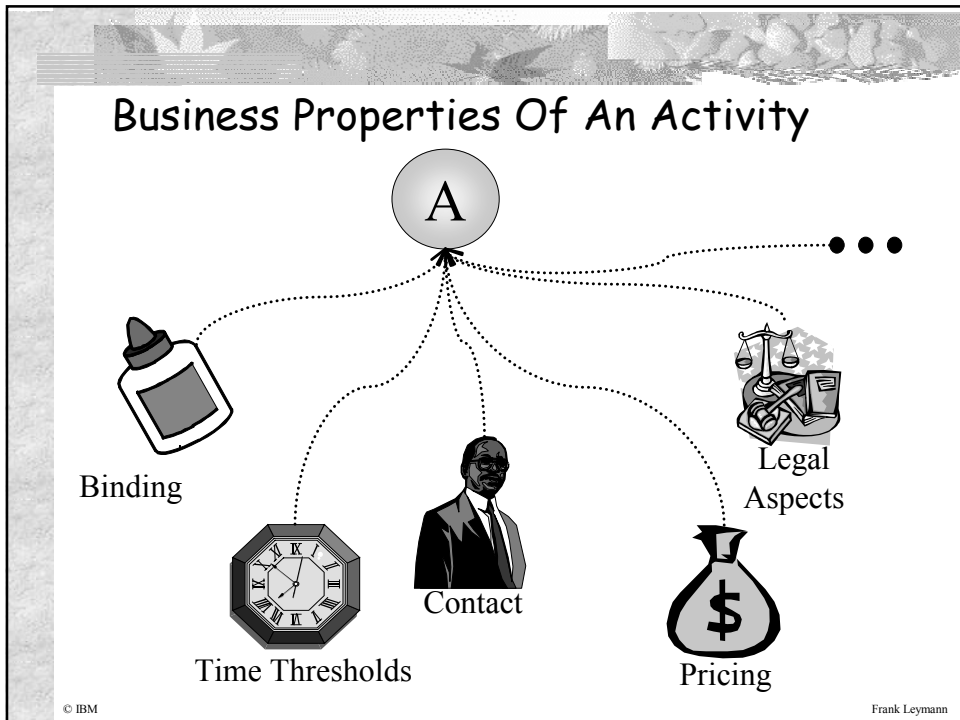
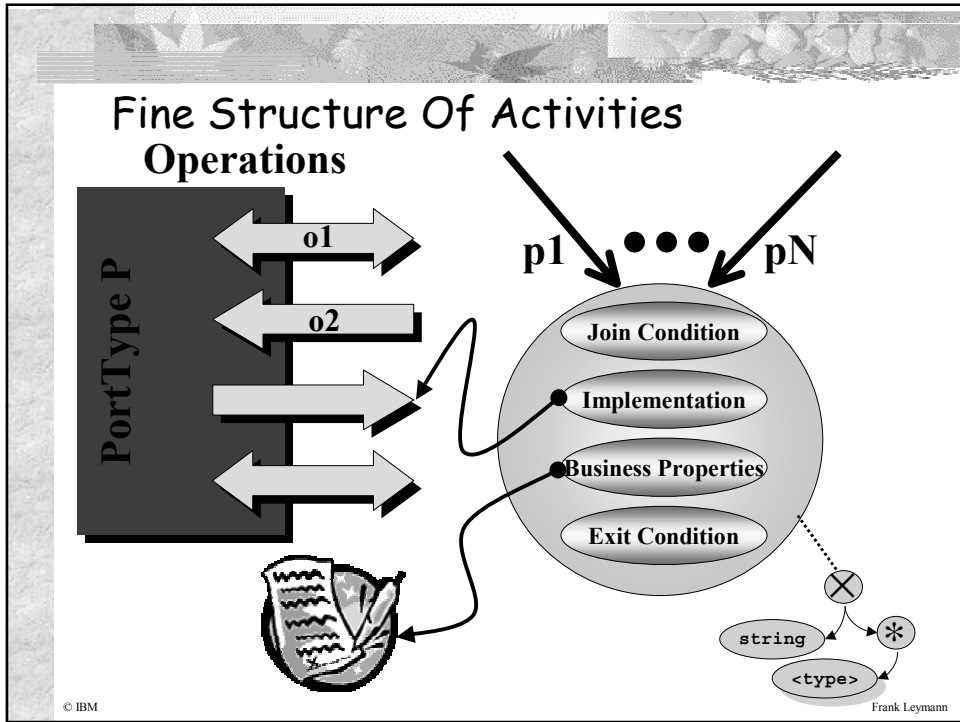


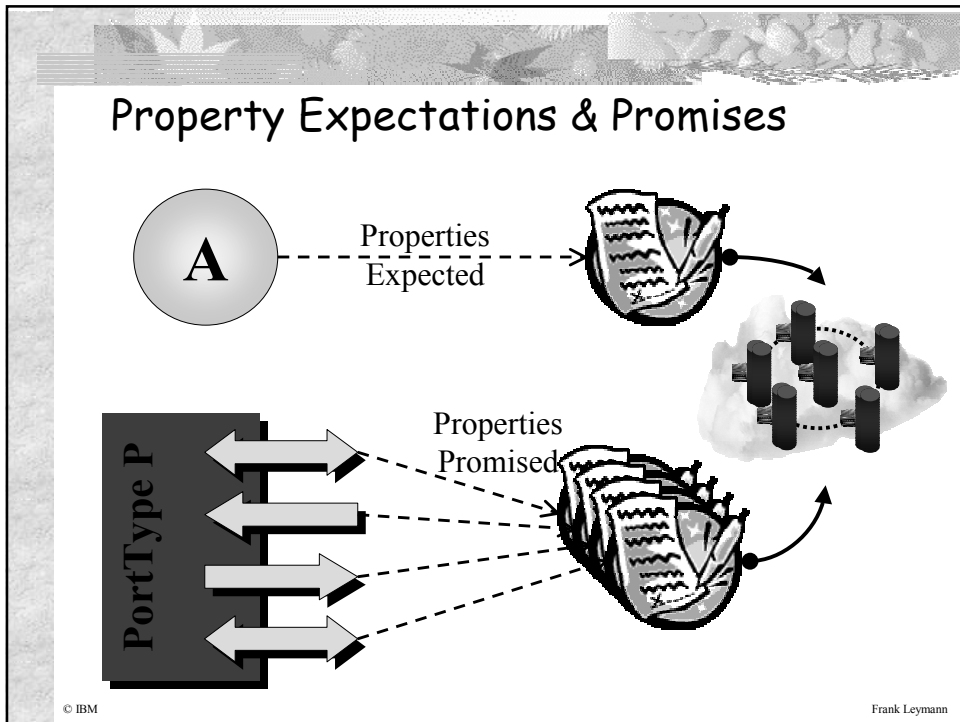
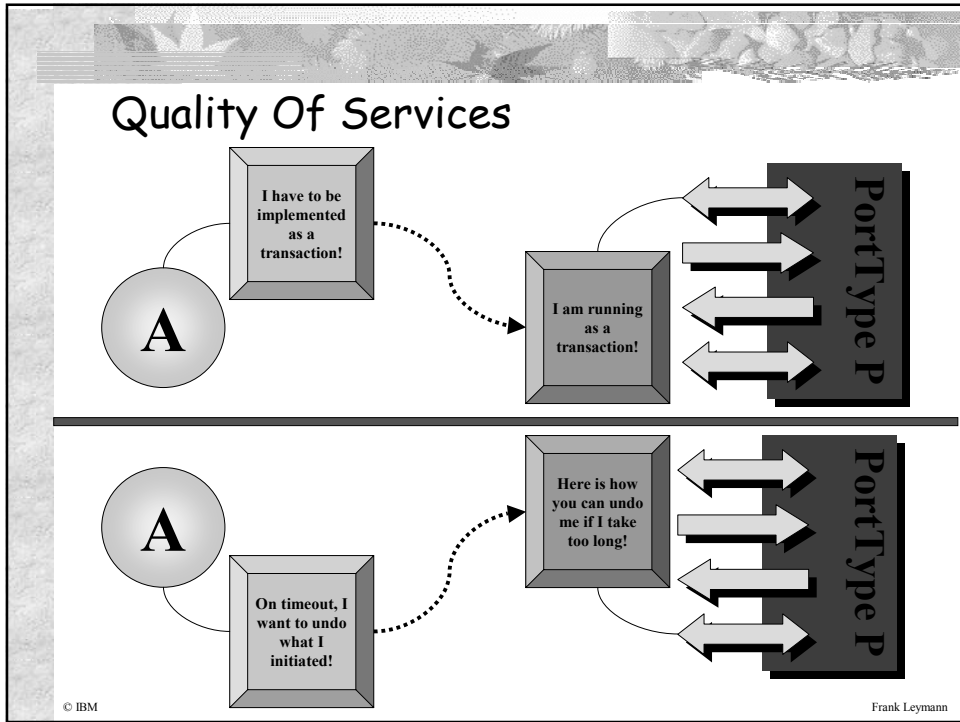
Touchdown: Some Details On WSFL Concepts

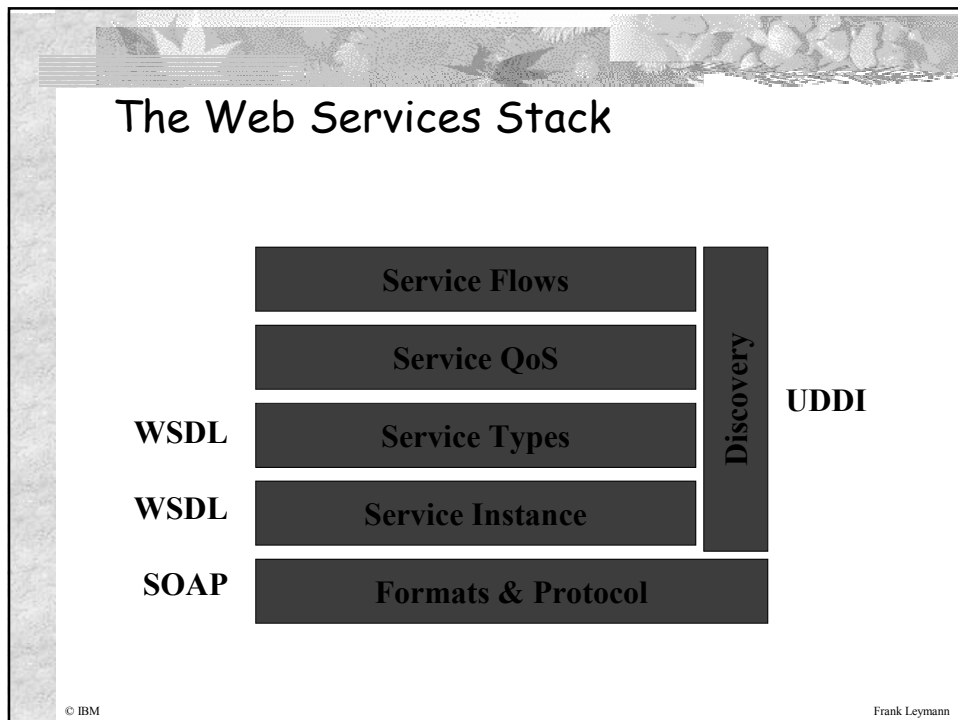
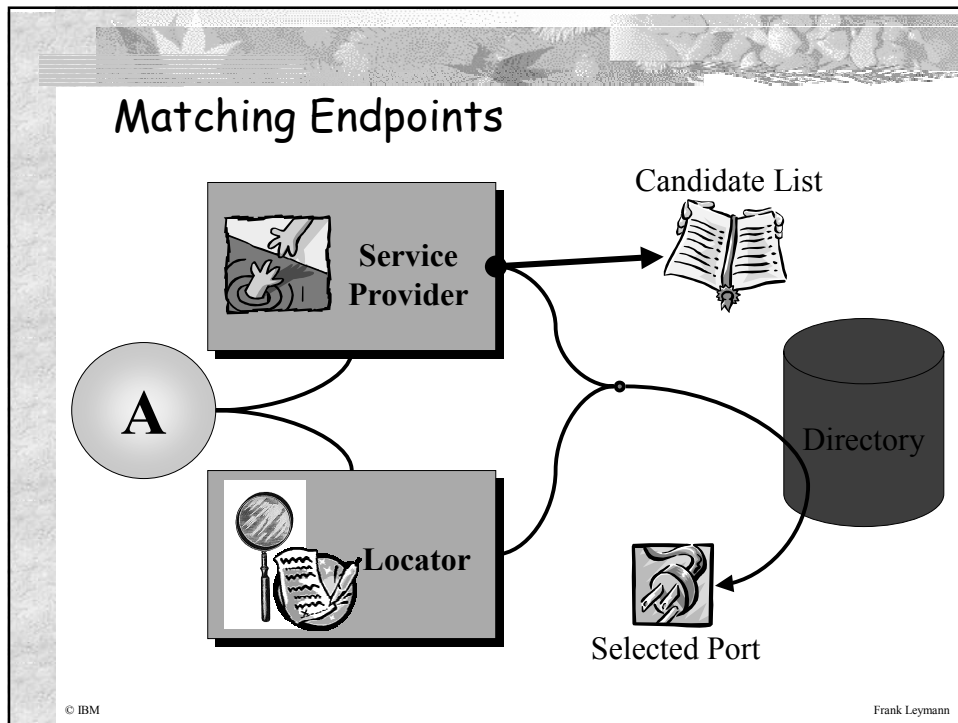
© IBM Frank Leymann

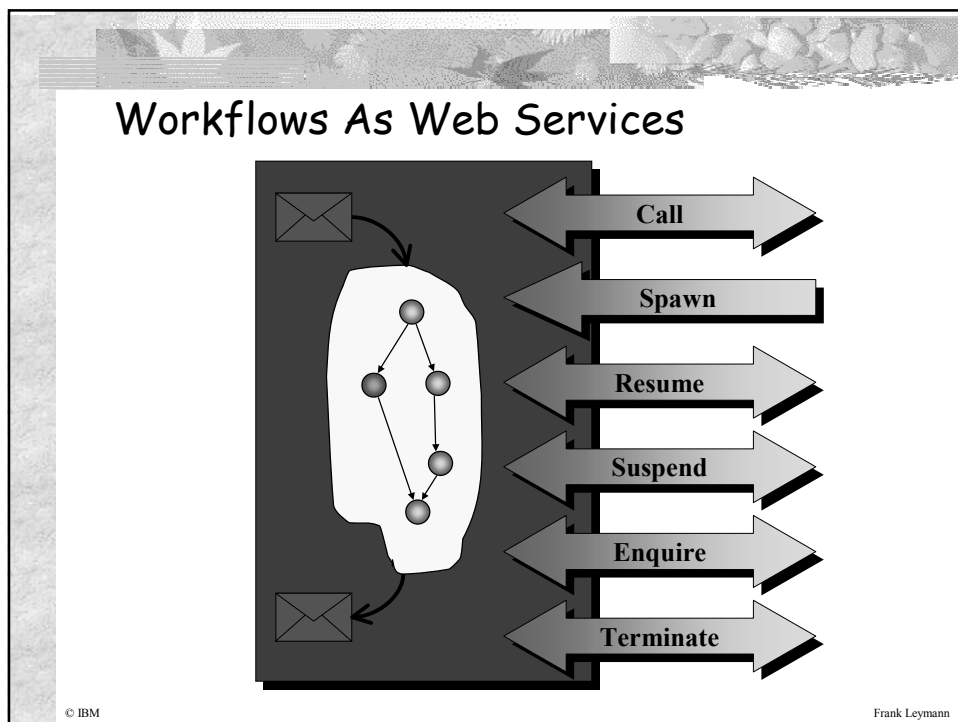
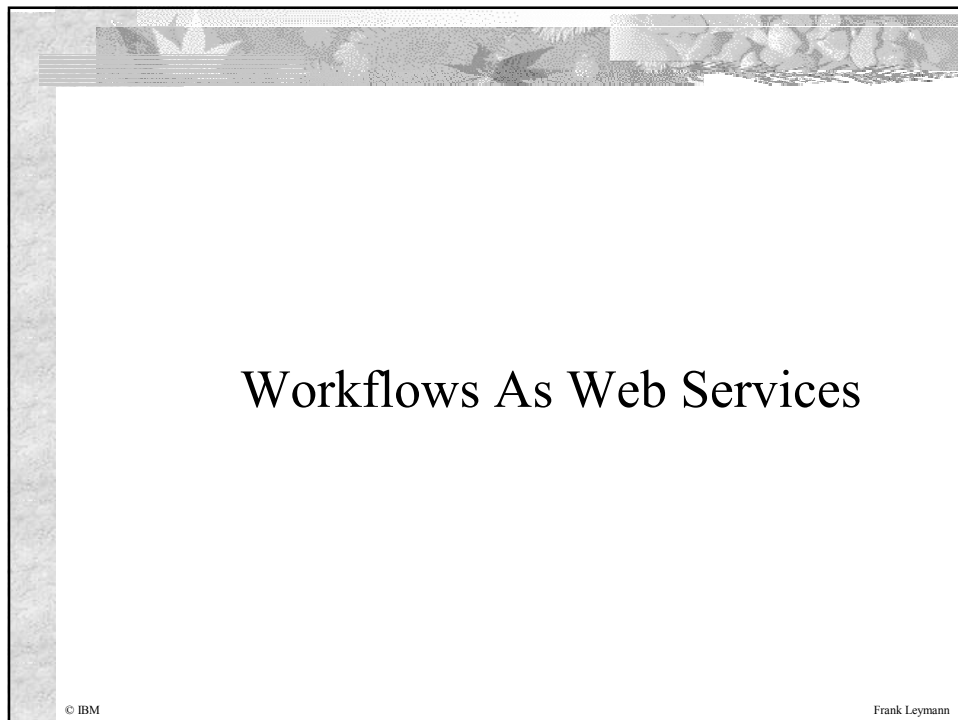


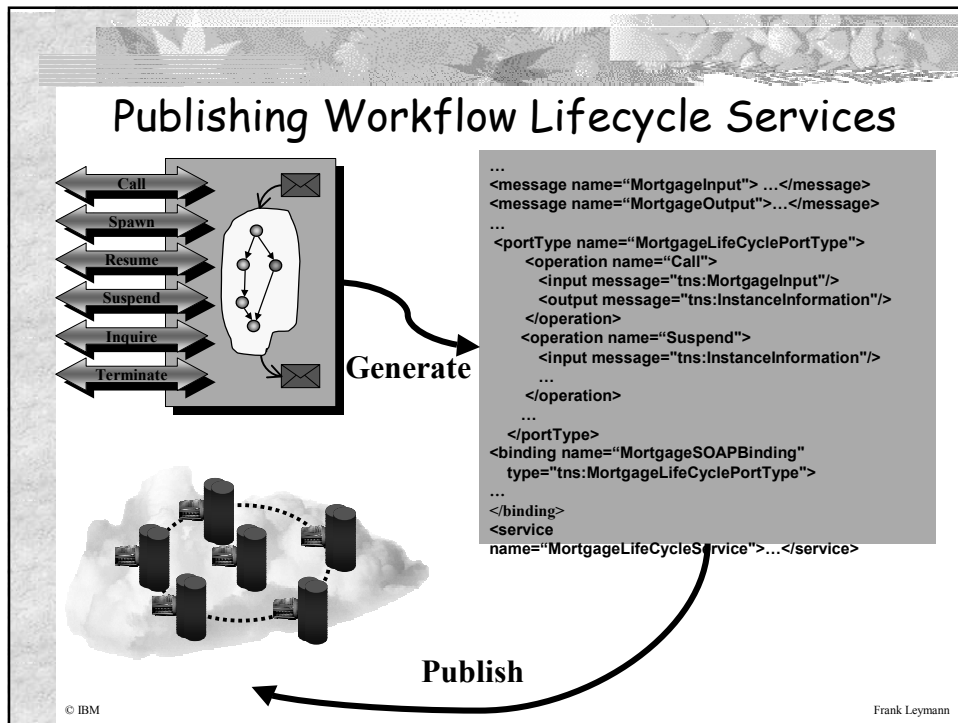






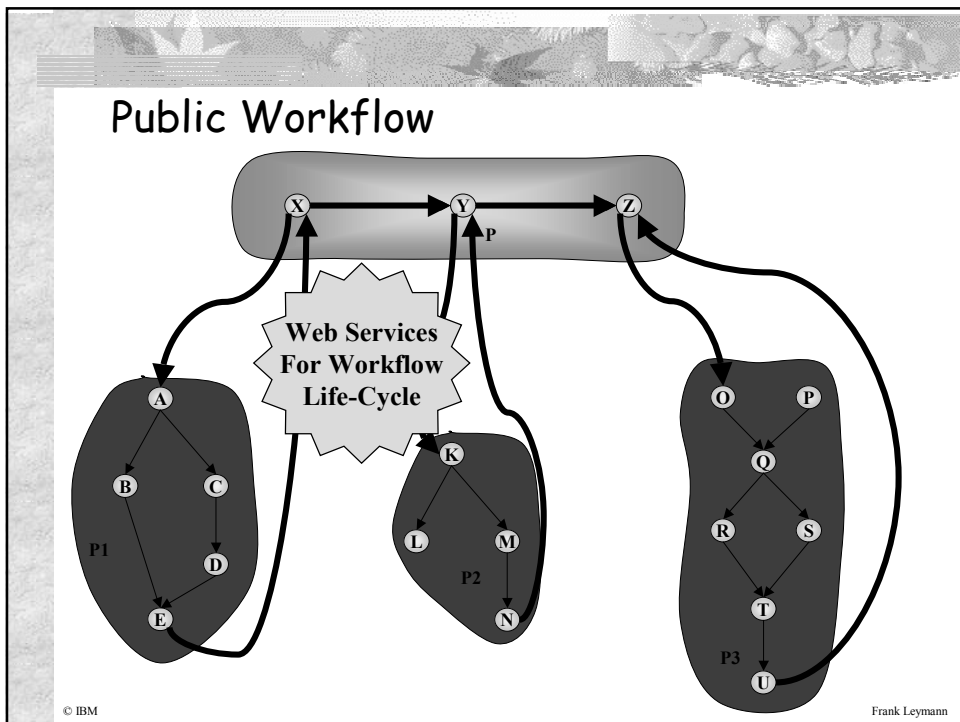
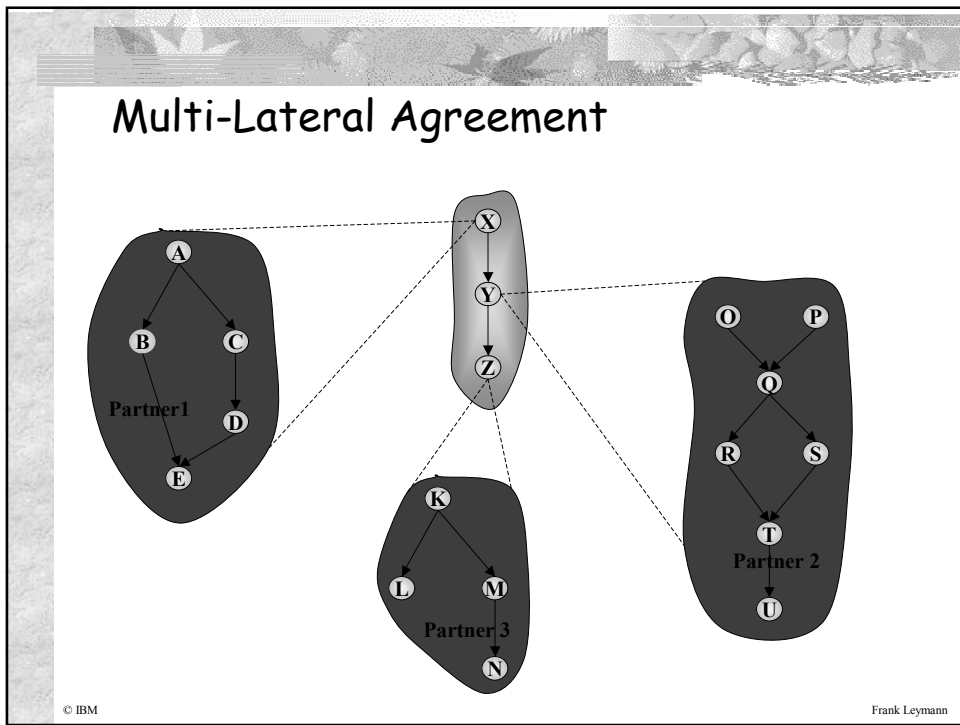


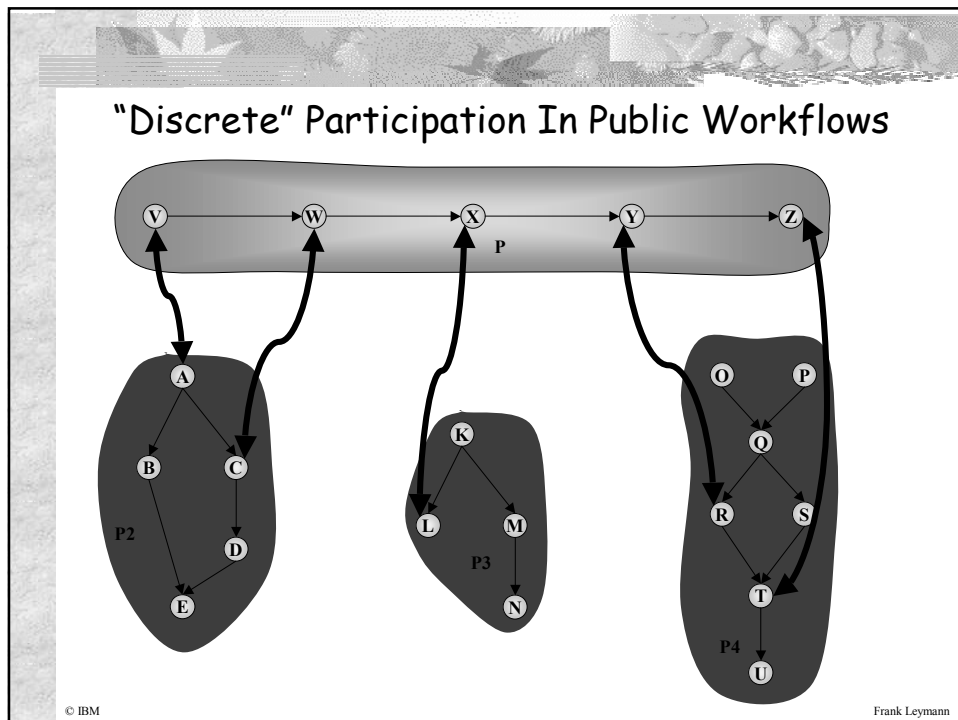
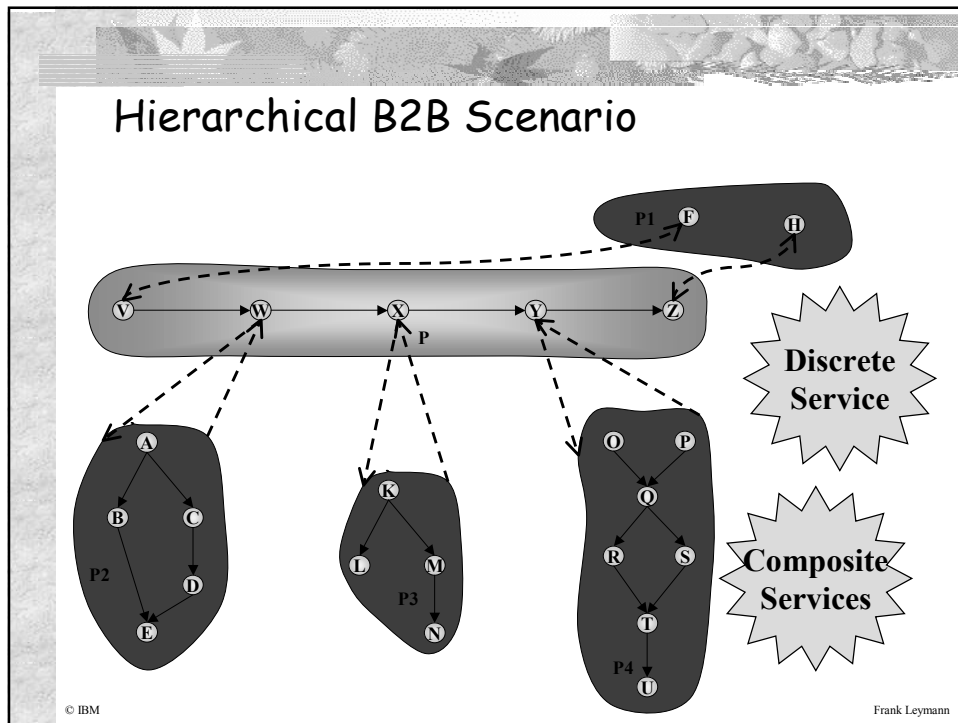


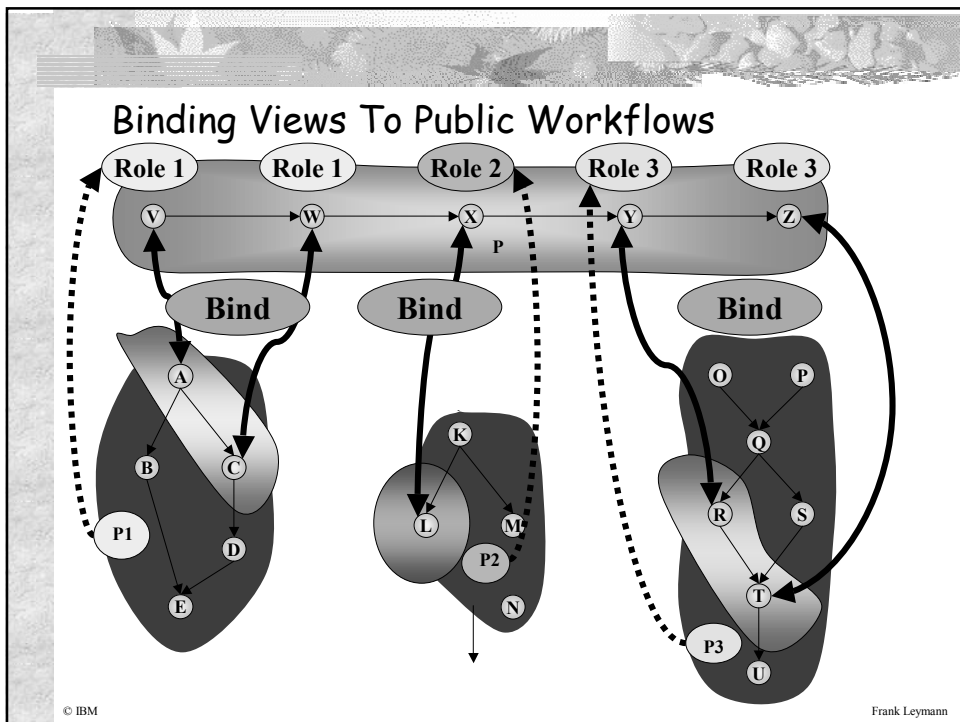
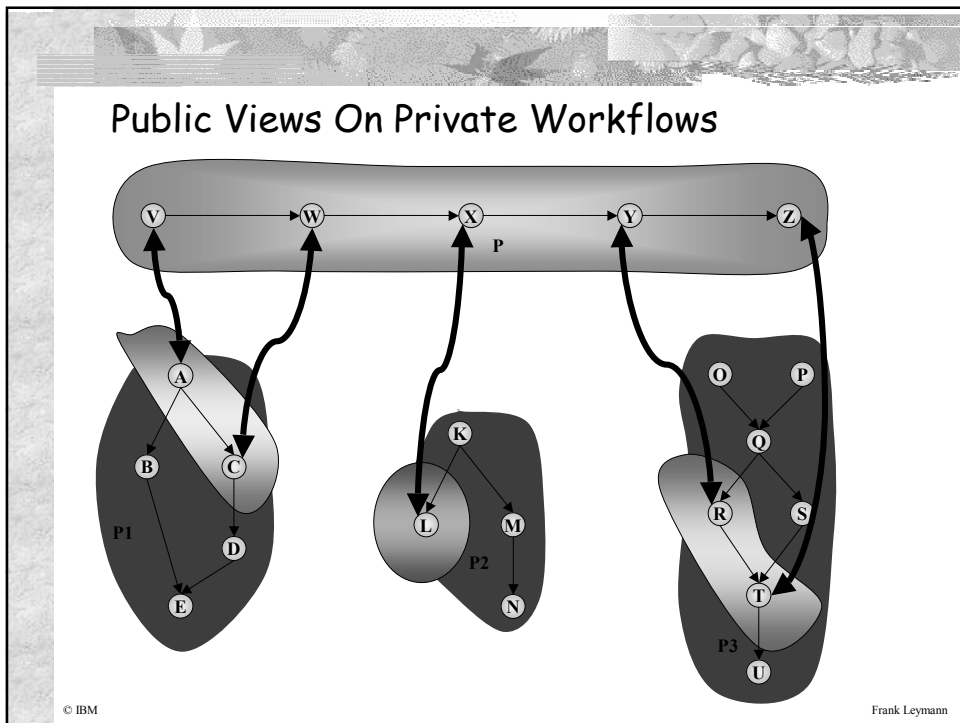


Web Services Choreography: Hierarchical Flows

© IBM Frank Leymann





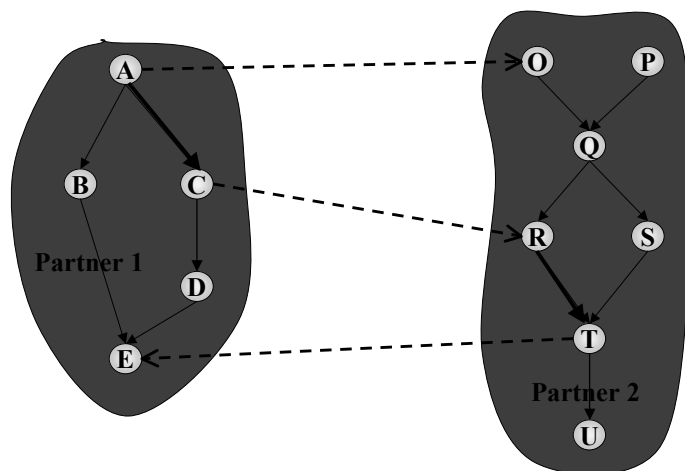


Web Services Choreography: Peer-To-Peer Flow

© IBM

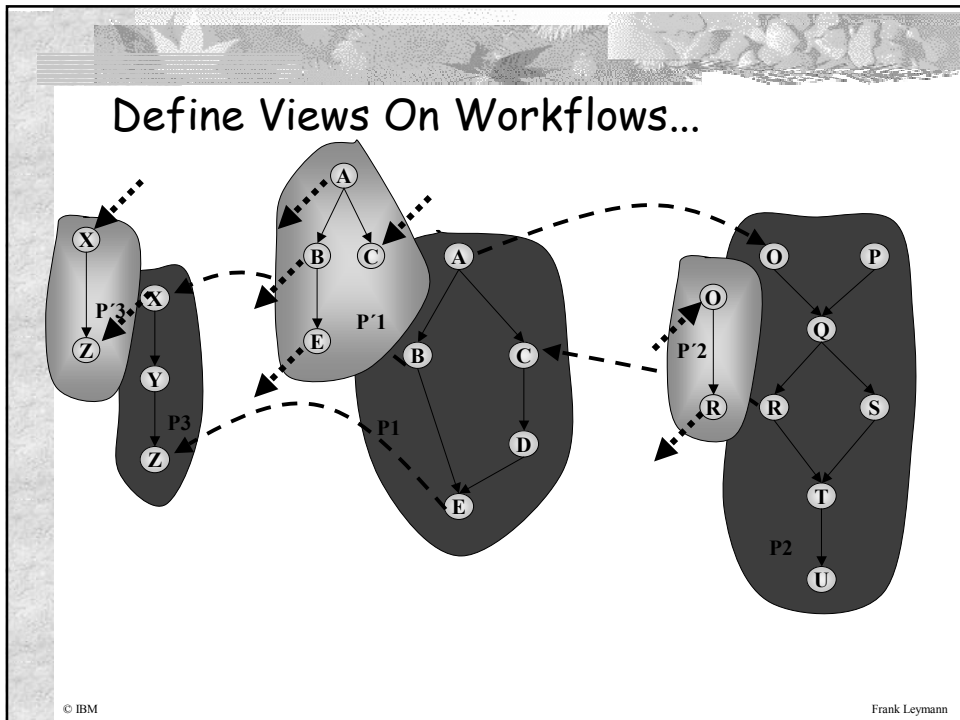
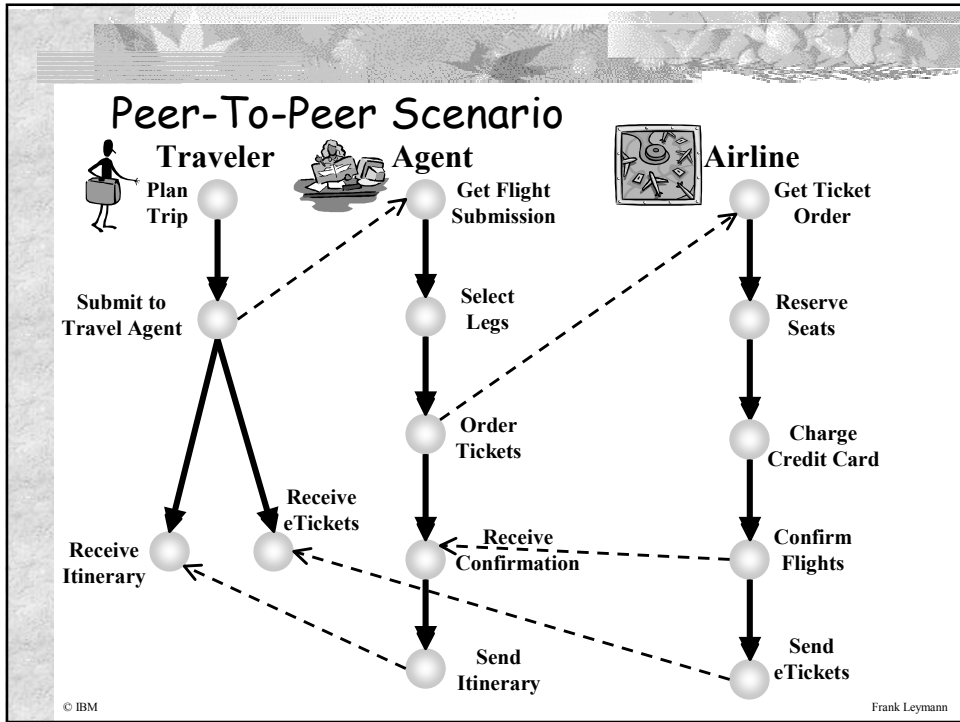
Frank Leymann

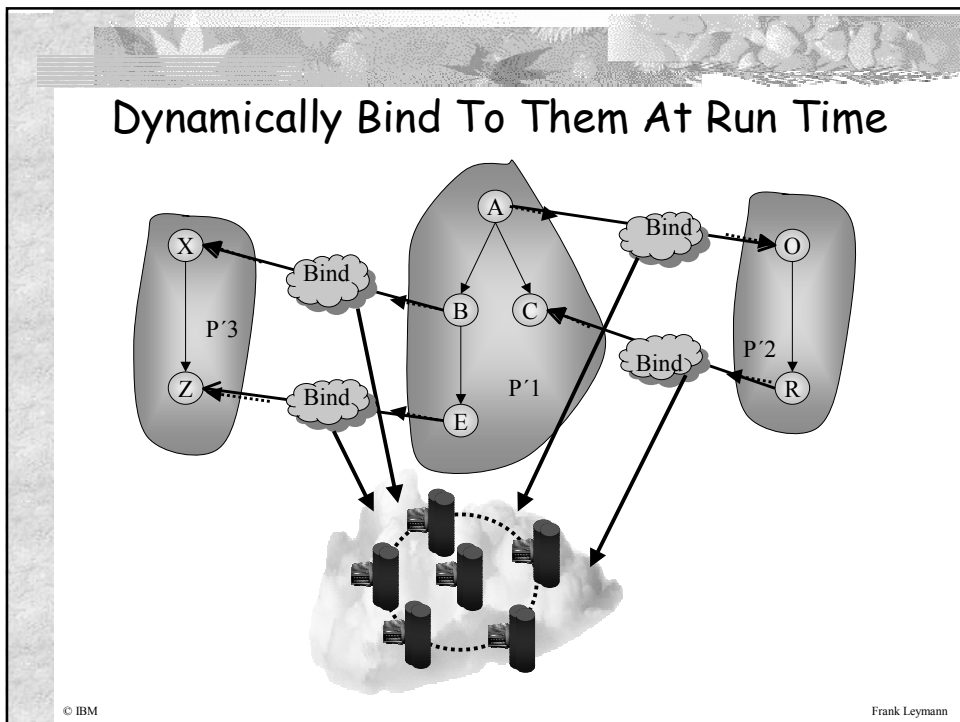
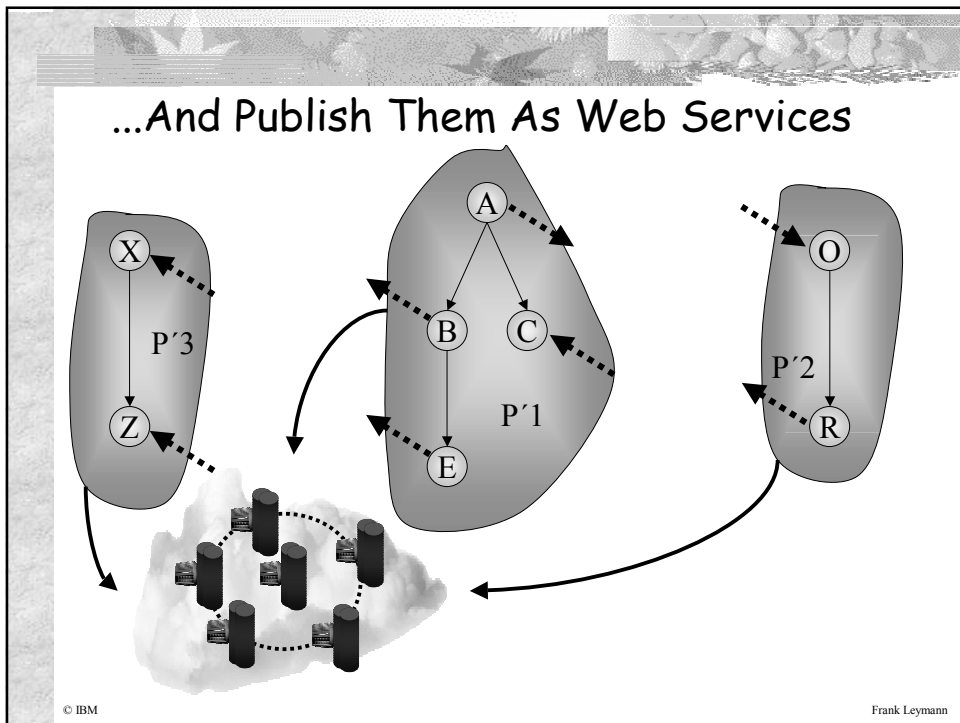
Peer-To-Peer B2B Scenario

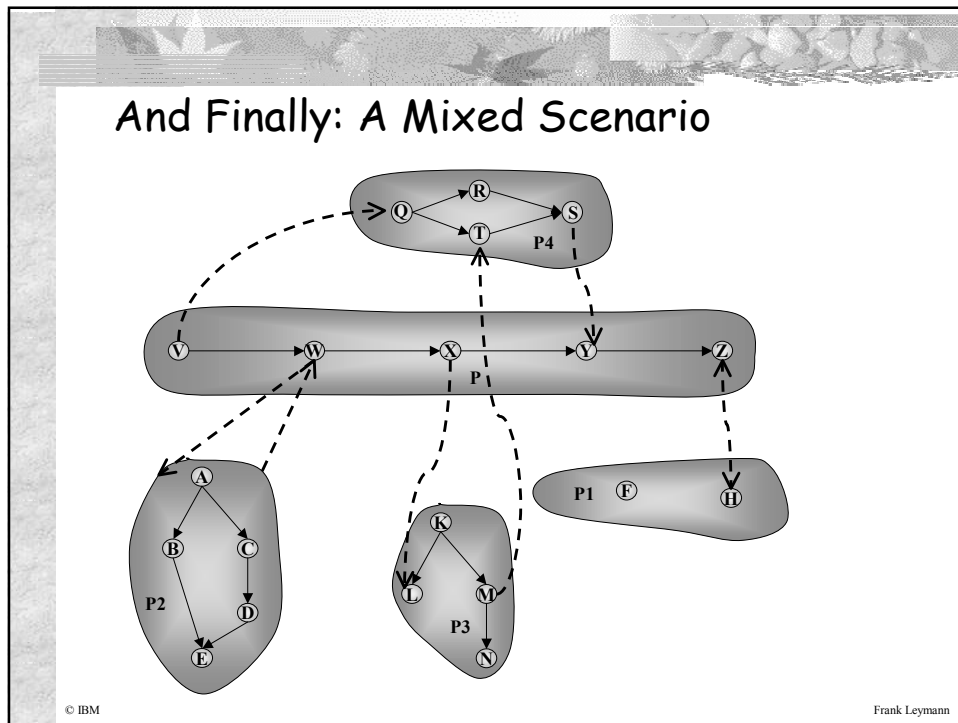


© IBM

Frank Leymann

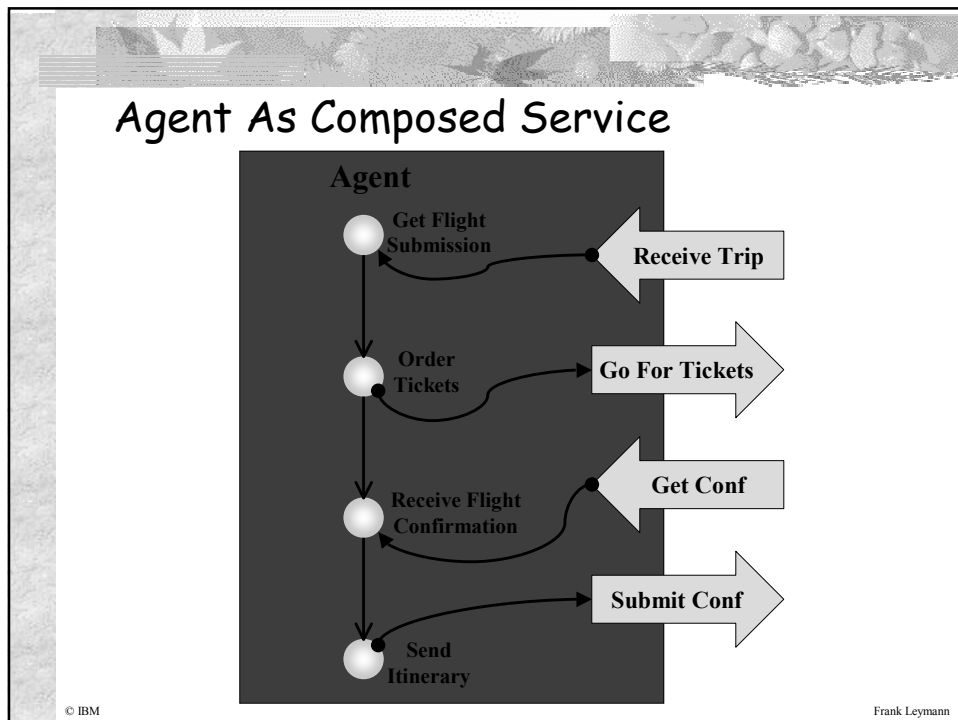
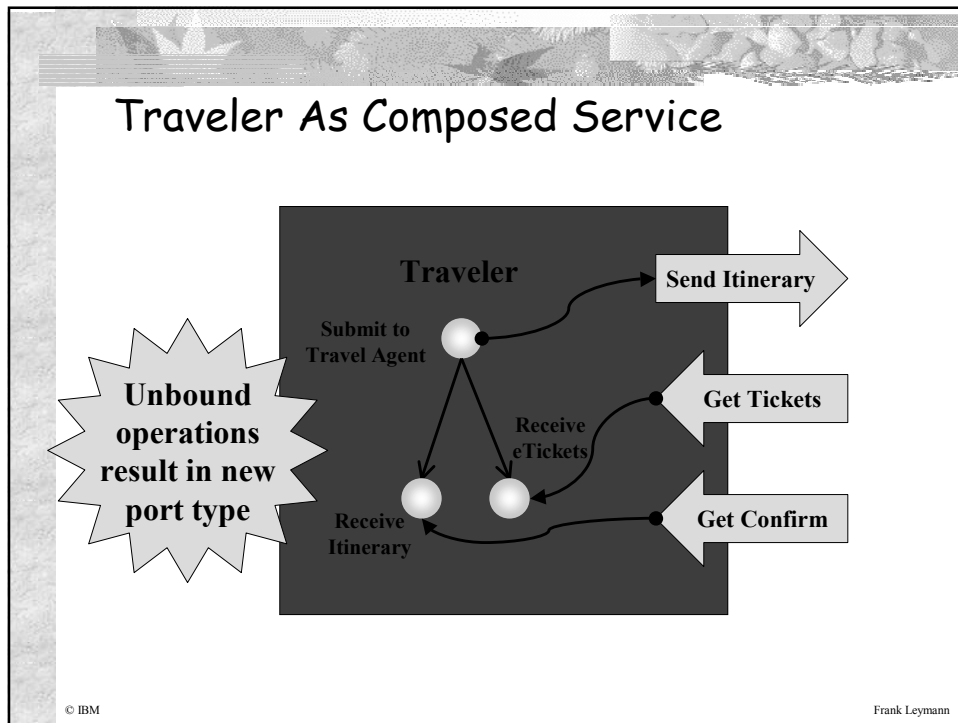


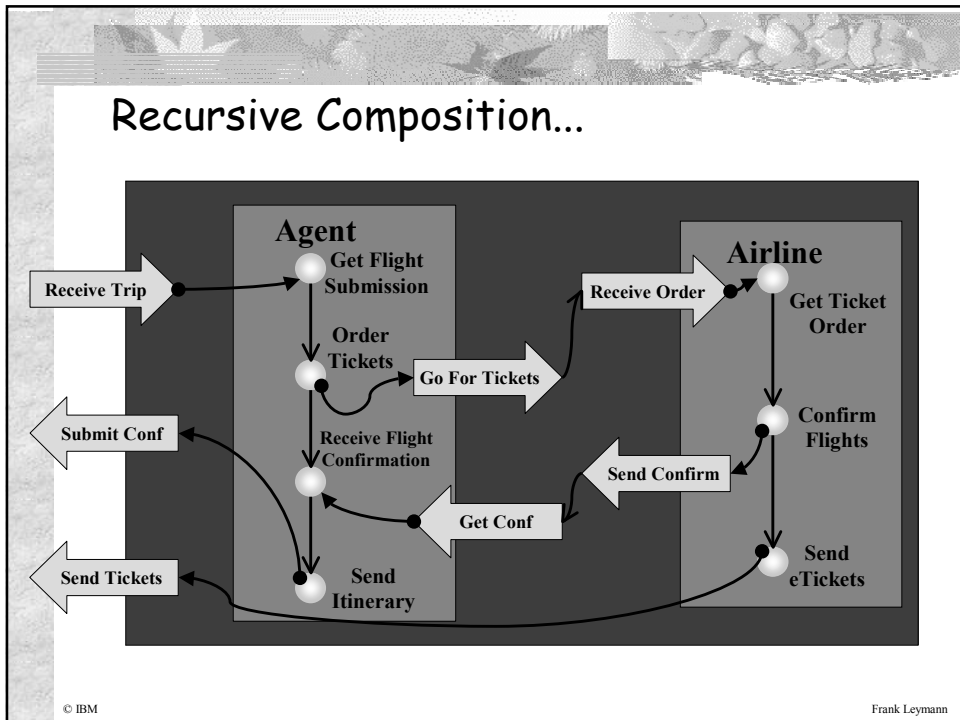
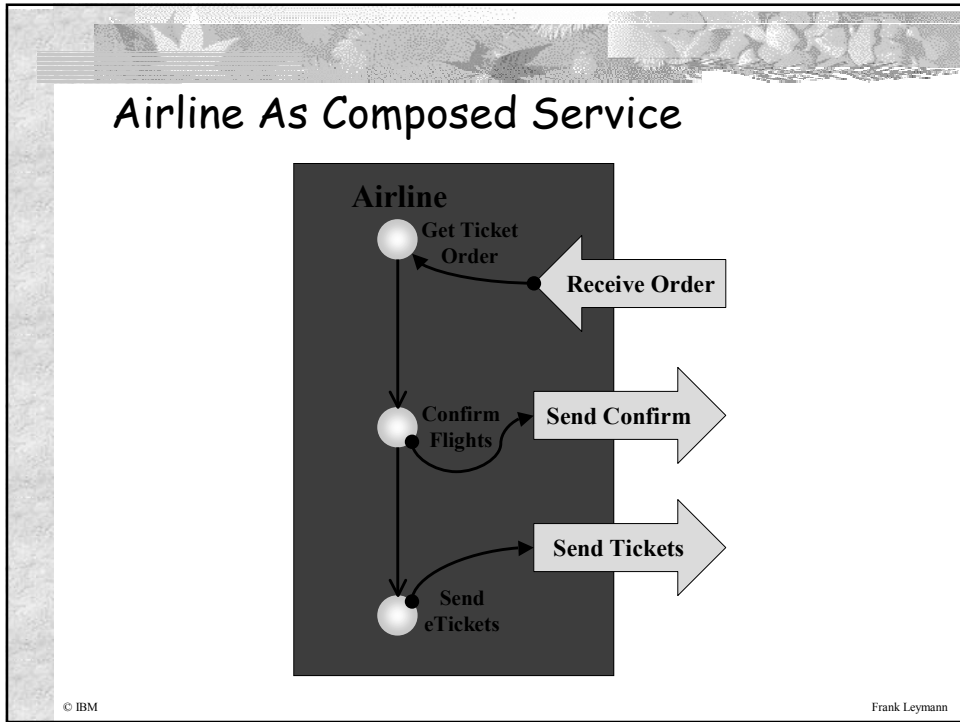


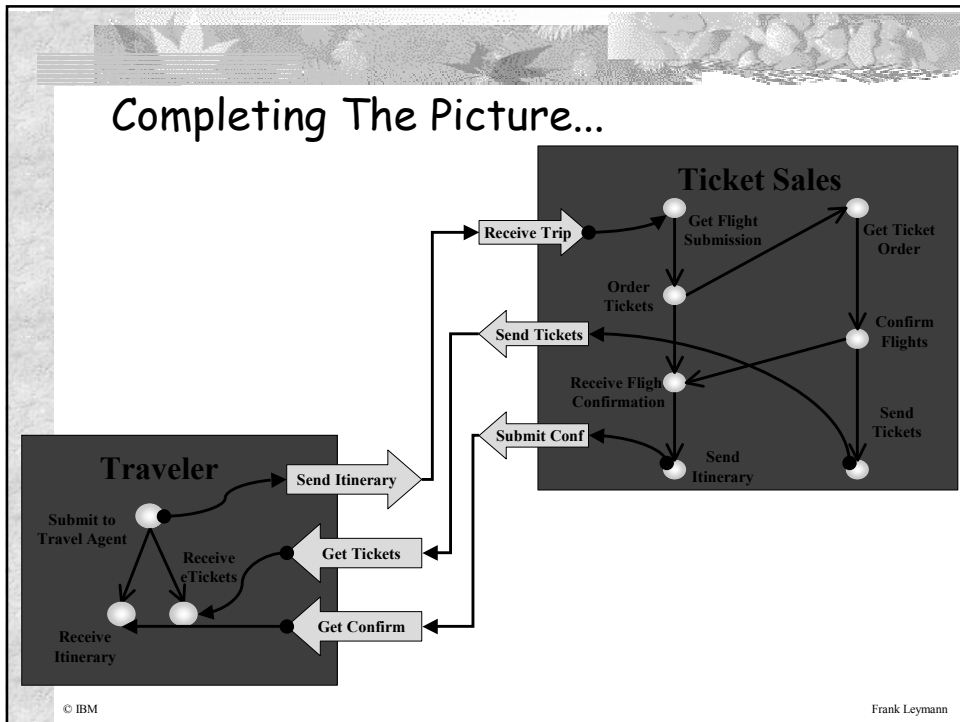
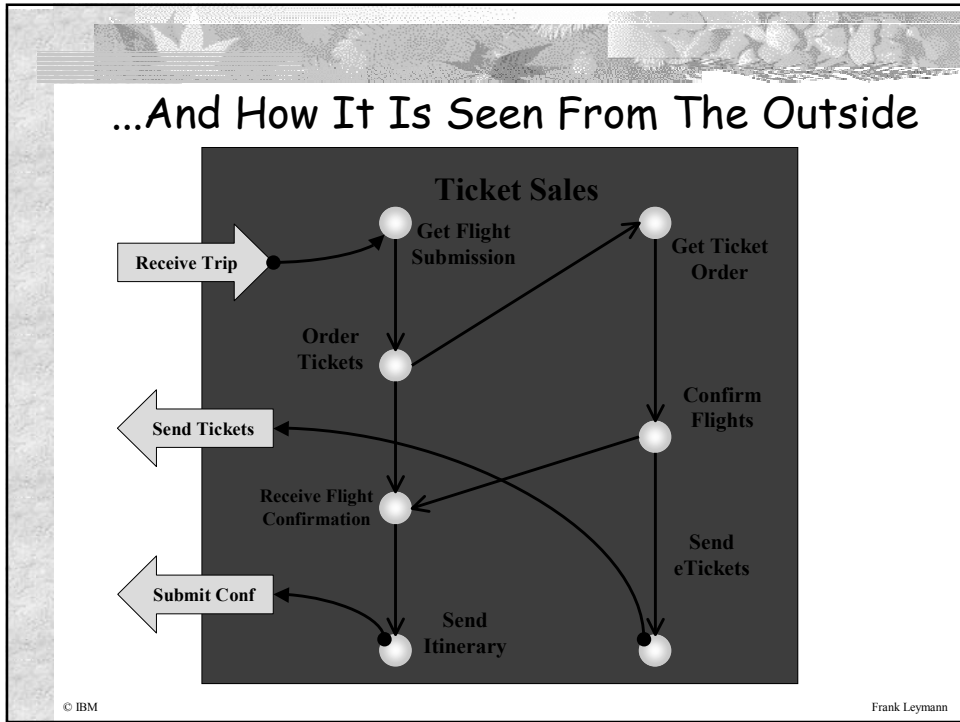


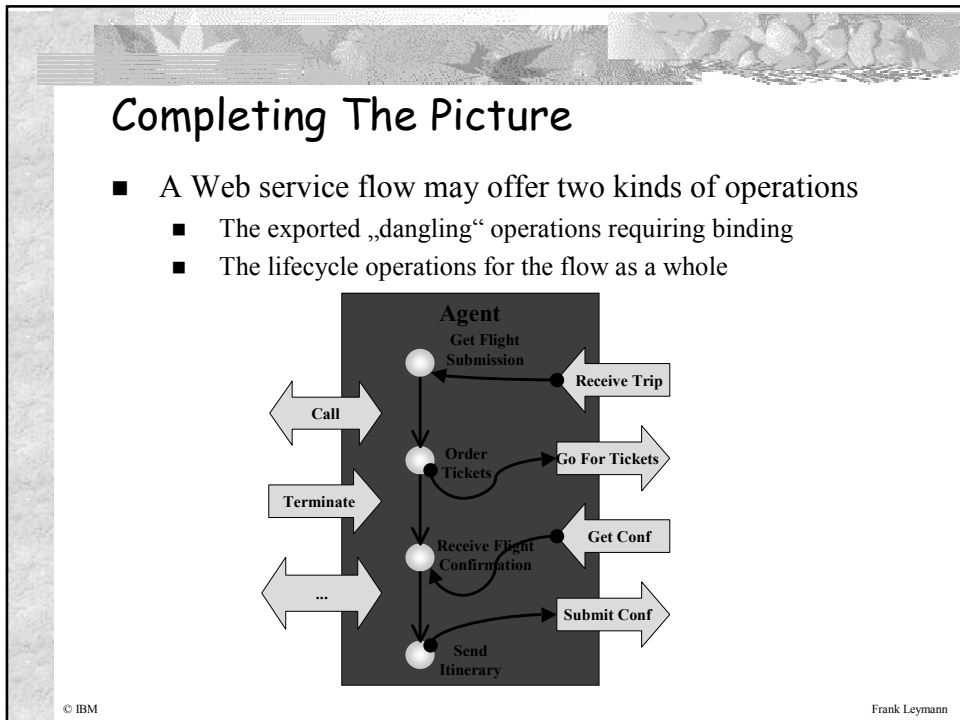
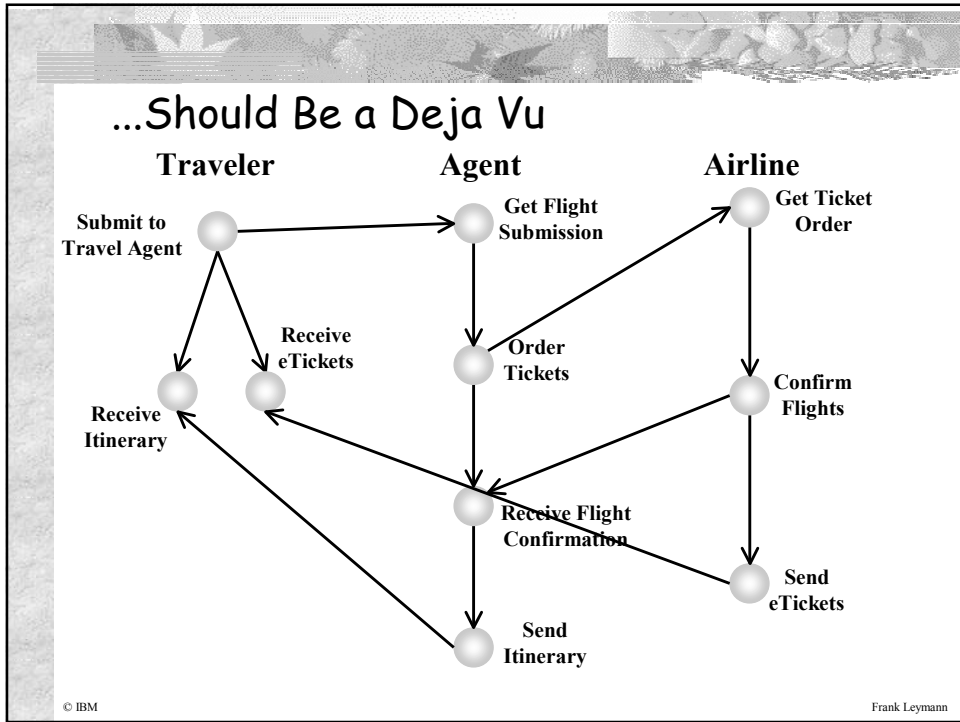
Web Services Choreography: Aggregating Web Services

© IBM Frank Leymann









Conclusion

© IBM

Frank Leymann

Summary

- BPR strives to optimize business processes
- Workflows are enacted business processes
- Workflows drive production of large companies
- Workflows across enterprise boundaries become important
- Workflow technology will become key aspect of SOA

© IBM

Frank Leymann

Many Open Problems

- Two-Level Programming Model
 - Boundary between modeling and programming
 - Debugging
 - Reuse of workflow models (specialization mechanism,...)
- Flow Metamodels
 - Suitability of different metamodels
 - Interoperability, transformation,...
 - WFMS federation
- Private/Public Flow Relation („View Problem“)
 - How to project public flows from private flows
 - How to extend public flows to private flows

© IBM

Frank Leymann

Many Open Problems (cont.)

- No accepted Extended Transaction Model
 - Compensation is agreed
 - Which „transactional properties“ are practical in Internet scenarios
- Internet Endpoints
 - Relevant QoS
 - Legal aspects
- Infrastructure for SOA
 - Match-making between expectations and promises
 - Establishing bindings (early, late, dynamic)

© IBM

Frank Leymann

Many Open Problems (cont.)

- Flexibility of workflows
 - Instance based modification of underlying model
 - Self-adapting workflow models
- Model mining
 - Derive workflow model from audit logs
- ...

© IBM

Frank Leymann

The End

© IBM

Frank Leymann