

Chapter 7 – Business Process Modeling and Workflow Management

Introduction & Motivation
Business (Re-)Engineering
Workflow Management Systems
WF and Transactions



Role of Workflow Technology

- Applications support business processes and have to ensure compliance with business processes
 - => Application = Business Process + Business Functions
 - Large applications often use special "control programs" to ensure the appropriate/correct sequencing of business functions
- Changes in how to perform business must be reflected as soon as possible in applications
 - Requires code changes [which part to change?...], recompilation, redistribution of code,... to reflect new business processes
- What if users of standard applications want to reflect their own processes?
 - very difficult, cumbersome, expensive (service specialists, consultancy)
- Consequence: Implementation of control programs via workflows
 - Application consists of collection of business processes and collection of business functions (= "usual" programs)
 - Business processes are enacted by workflow system that invoke business functions "appropriately", i.e. according to process model
- No coding,... to adapt application to changed business process

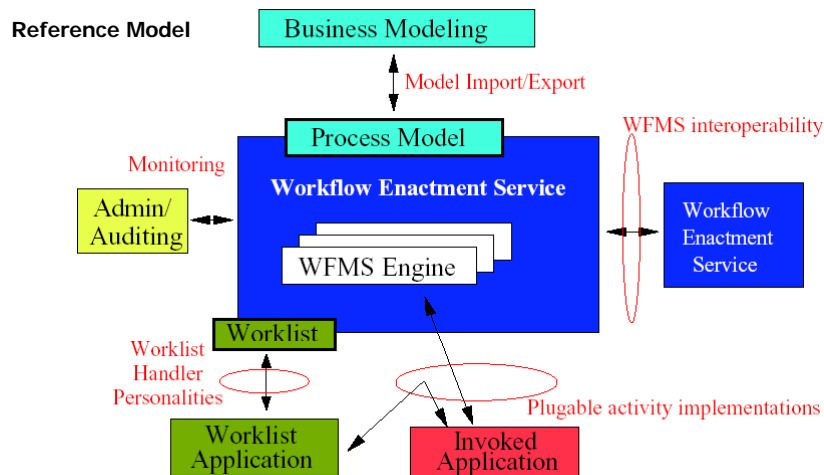


Historical Background

- Electronic document and folder routing in *office automation*
 - Routing through enterprise's organizational structure
 - Potential flow of documents prescribed in advance
 - Routing conditions in terms of document content or document properties
- Business processes involving functions provided by application systems
 - Launch-pad for executables, work item management
 - Launching executables requires parameter passing
 - Data flow features complemented available control flows
 - Control flows expressed in terms of these new parameters ("business rules")
- "Production Workflow" driving operational aspects of an enterprise
 - Modeling of business process logic and separation from business functions
 - Significant improvements regarding high availability, scalability, robustness
 - advanced transaction management (forward/backward recovery, compensation)
 - WFMS become an EAI-tool (often integrated with other EAI middleware)
 - business process logic includes integration logic



Standards: Workflow Management Coalition



Terminology

- **Material process**
 - assembles physical components, delivers physical products
 - may include moving, storing, transforming, measuring, assembling objects
- **Information process**
 - relates to automated and partially automated tasks that create, process, manage, provide information
 - involves programs, humans (interacting with computers)
 - infrastructure provided by database, transaction processing, and distributed systems technology
- **Business process**
 - market-centered description of an organization's activities
 - implemented as an **information process** and/or material process
- A **workflow** is a business process in execution (an instance of a process model) in a computing environment
 - Not all parts of a process are run in a computing environment - some processes are not run on a computer at all!
 - Often, "workflow" and "process" is identified



© Prof. Dr.-Ing. Stefan DeBloch

5

Enterprise Information Systems

Business Process (Re-)Engineering

- **Make company more flexible, react faster to change**
 - outsourcing of processes, supply-chains, virtual enterprises,...
- **Speed up, reduce costs**
 - eliminate unnecessary tasks, employ parallelism, deadline processing, monitoring/auditing, analysis, automation
- New processes are defined, existing are changed or abandoned
 - existing business processes must be analyzed, specified and modeled, optimized (includes simulation)
- Important to include resources used to perform processes
 - organizations, roles, people
 - IT resources
- Scope is not only intra-enterprise but also inter-enterprise
 - Business-to-Business, Consumer-to-Business, Business-to-Administration,...
- Reengineered processes may be implemented using a WFMS

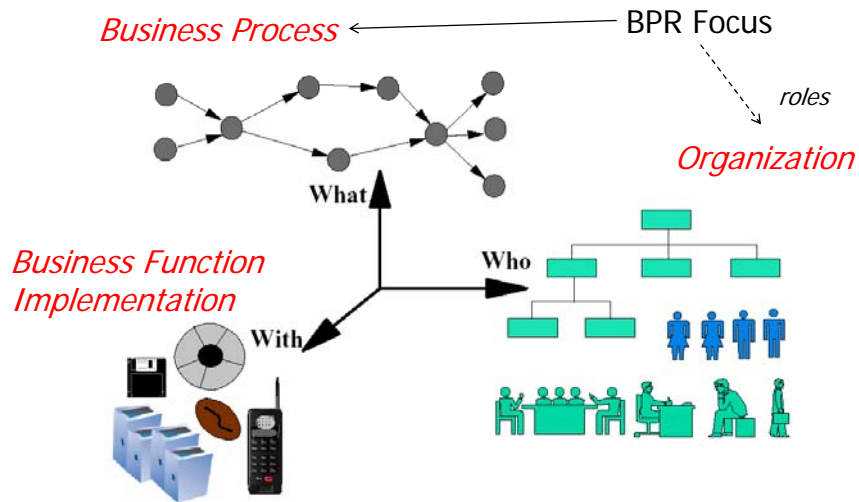


© Prof. Dr.-Ing. Stefan DeBloch

6

Enterprise Information Systems

The Three Dimensions Of Workflow

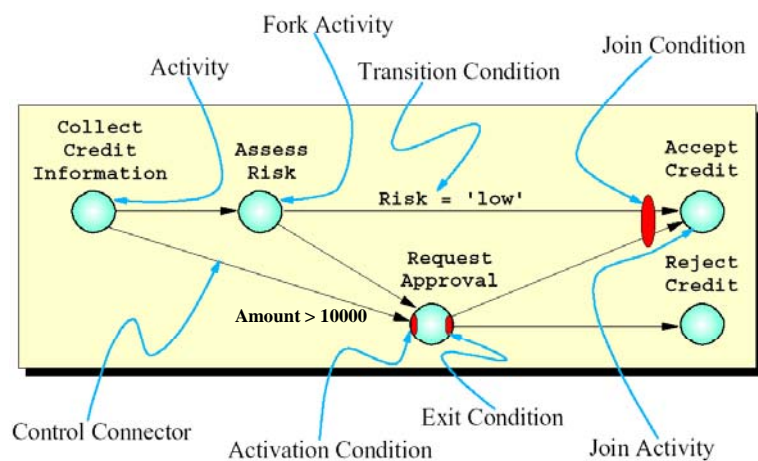


© Prof. Dr.-Ing. Stefan DeBloch

7

Enterprise Information Systems

Business Process: Control Flow



© Prof. Dr.-Ing. Stefan DeBloch

8

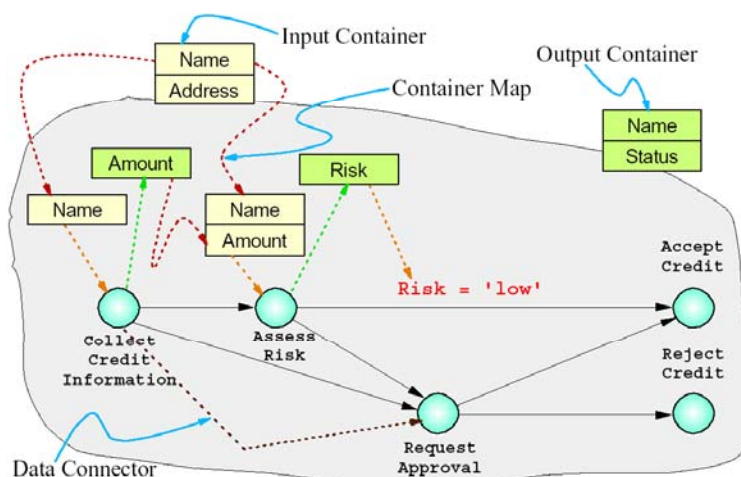
Enterprise Information Systems

Control Flow – Details

- Activities
 - describe task to be performed
 - different types of activity
 - program activity, person activity, process activity (subprocess), block (do-until loop)
- Control Connectors
 - describe potential sequence in which activities are carried out
 - connect **source** and **target** activities
 - **start/end** activities have no incoming/outgoing connectors
 - parallelism supported through **fork/join** activities
 - multiple outgoing/incoming connectors
 - join conditions act as synchronization points for parallel activities
- Conditions
 - **join** condition defines whether an activity can be started
 - **activation** condition specifies when an activity can be started
 - evaluated after the join condition
 - **exit** condition confirms that the activity has been successfully completed
 - **transition** condition describes a condition for following a control connector



Business Process: Data Flow



Process Optimization

- Static analysis of flows through organization
 - restructure to achieve high level of parallelism
 - minimize occurrences of control flow crossing organization boundaries
 - reorder and combine multiple activities into a single one activity (or a stream of activities) per role
 - optimize the organization (i.e., change organization-department-employee aspects)
- Dynamic analysis (simulation)
 - used to compare and select from alternative models of a given business process the "optimal" one (in terms of metrics like cost, duration,...)
 - involves quantitative aspects
 - requires process instrumentation
 - number of processes per time unit, probabilities of routing conditions being satisfied, average duration of activities, processing power and availability of resources
 - analytical simulation
 - calculates how often an activity has to be performed, probability of execution paths, duration and probability of process execution paths
 - discrete (event) simulation
 - calculation/simulation at the level of individual resources
 - considers availability, resource competition

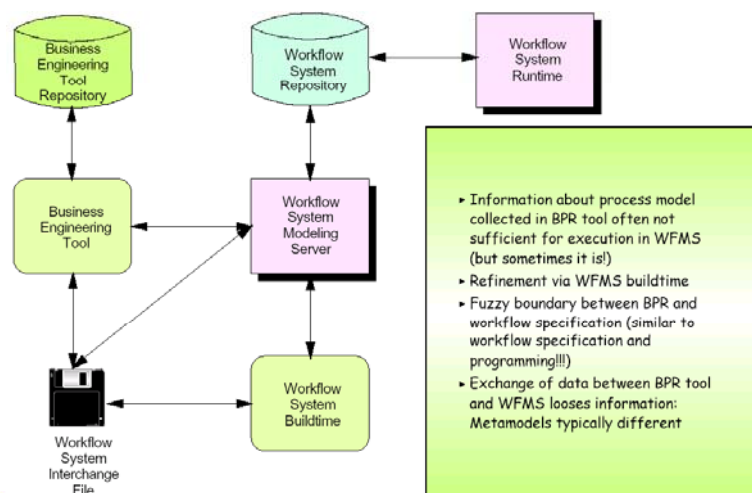


© Prof. Dr.-Ing. Stefan DeBloch

11

Enterprise Information Systems

BPR-WFMS Exchange Of Information

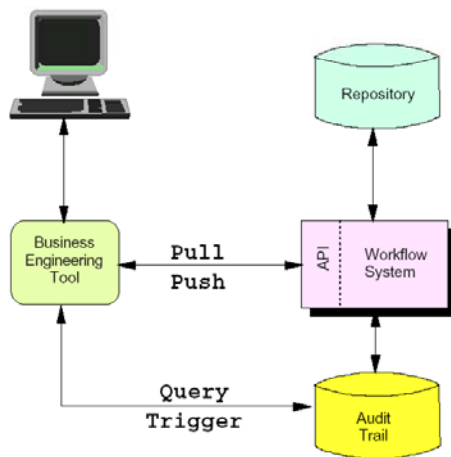


© Prof. Dr.-Ing. Stefan DeBloch

12

Enterprise Information Systems

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,...): **Leitstand**

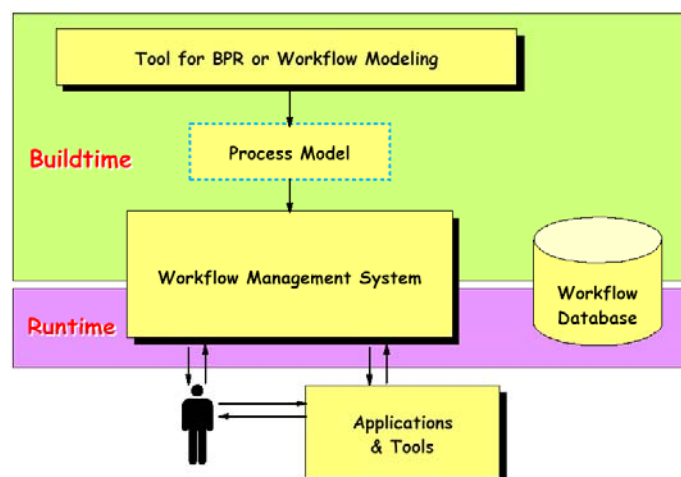


© Prof. Dr.-Ing. Stefan DeBloch

13

Enterprise Information Systems

Major Building Blocks Of A WFMS



© Prof. Dr.-Ing. Stefan DeBloch

14

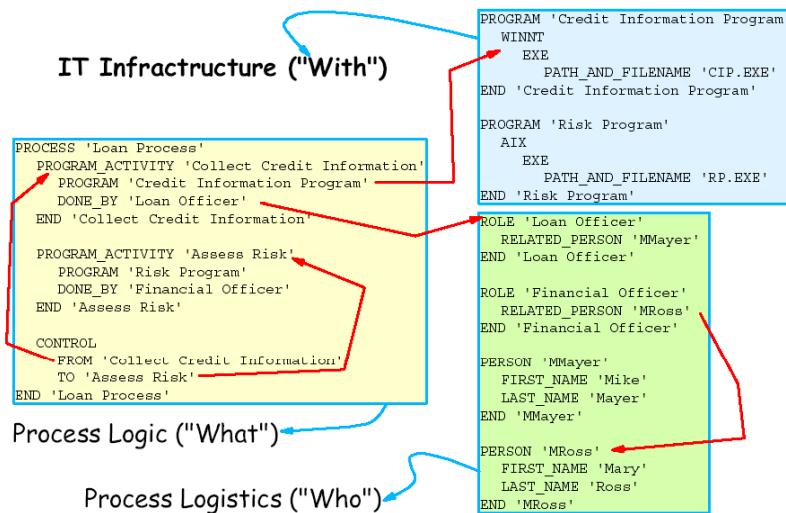
Enterprise Information Systems

Buildtime

- Component providing all functions and capabilities to define, test and manage all workflow related information
 - Especially, all three workflow dimensions are covered
 - Often, administrative and systems management information is included, e.g.
 - Session threshold, i.e. maximum period of time a user can work with the WFMS
 - Actions to be taken when average response time exceeds threshold
 - All information stored in WFMS own database ("buildtime database")
- Two different kinds of interfaces
 - Graphical end user interface
 - Workflow Definition Language
 - ASCII text with special syntax/semantics
 - Most often vendor specific
 - e.g., IBM's FDL (Flow Definition Language)
 - Standard developed by Workflow Management Coalition (WFMC)
 - WPDL (Workflow Process Definition Language)
 - XPDL (XML Process Definition Language)
 - Both GUI and WFDL cover all concepts of the WFMS Meta Model

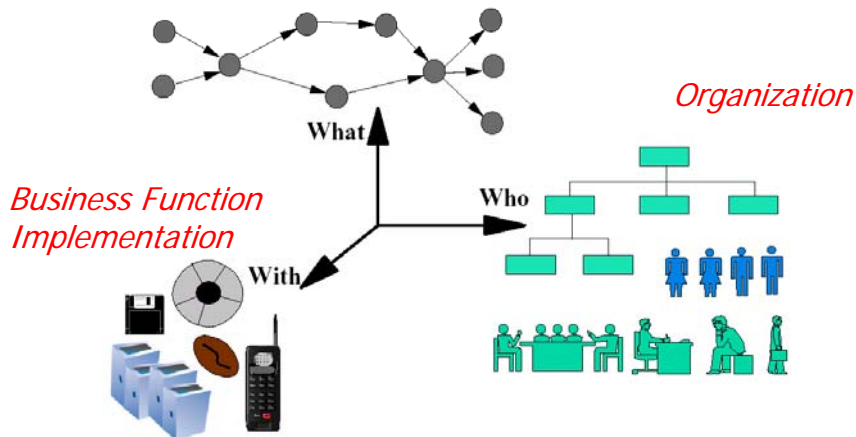


Workflow Definition Language: Example (FDL)



The Three Dimensions Of Workflow

Business Process



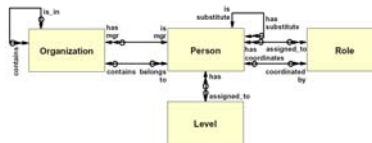
© Prof. Dr.-Ing. Stefan DeBloch

17

Enterprise Information Systems

"Who" Dimension: Organizational Aspects

- WFMS organization schema
 - either **fixed** (built-in) by the WFMS vendor
 - often simple, can be implemented efficiently
 - or **dynamic**, allowing to change the entities and relationships
 - very flexible, but hard to achieve efficiency
 - requires mapping (see discussion below)
- Organizational data is
 - either managed by WFMS in its own database
 - optimized schema, no performance impact on source systems, but: often replica of "real" org. database, might run out of sync
 - or shared with other systems
 - no redundant data, but: performance impact, usually involves dynamic mapping
- Staff Resolution: performing staff queries at runtime
 - staff query need to be attached to each activity for staff assignment
 - may require interaction with external org data system using a staff resolution exit (external program), mapping WFMS org schema to external schema, or directly using external DBMS interface/schema



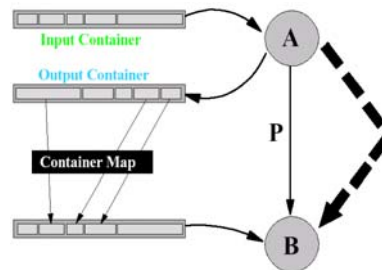
© Prof. Dr.-Ing. Stefan DeBloch

18

Enterprise Information Systems

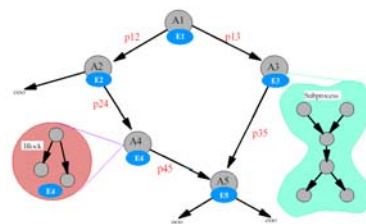
"What" Dimension: Data Flow

- Input/Output Container
 - defines data passed to/returned by process or activity
 - based on simple/structured types
 - definitions can be shared
 - can also specify default values
 - provides the execution context
- Data Connectors
 - specify which data needs to be copied where
 - details provided by container map
 - field/data type mapping
 - data transformations
- WFMS at runtime
 - materializes input container instance before activity is started
 - may utilize so-called dead data maps
 - makes output container instance persistent



"What" Dimension: Control Flow

- Different Types of Activities
 - Information Activity
 - inform user to take some actions
 - no implementation
 - Program Activity
 - implemented by a program
 - different types of binding
 - Process Activity
 - activity implemented as a sub-process
 - different types of connection
 - Bundle Activity
 - the same activity is implemented on a set of objects
 - parallel execution as an option
 - Block Activity
 - provides DO-UNTIL behavior as special construct
 - process model often restricted to DAG
 - exit condition determines looping behavior



Subprocesses

- Subprocess may be **local** (performed by the same WFMS) or **remote**
 - remote WFMS may be from the same or a different vendor (requiring standardized formats and protocols)
- Autonomy of subprocesses (governed by autonomy rules)
 - **autonomous**: once started subprocess cannot be influenced by the parent
 - **controlled**: life-cycle of the subprocess is determined by the parent process, e.g.
 - suspension of the implemented activity forces the subprocess to suspend
 - tight administrative coupling
 - whole spectrum between these extremes can be defined
- Nested subprocesses
 - controlled subprocess returns control to parent after completion
 - nesting hierarchy



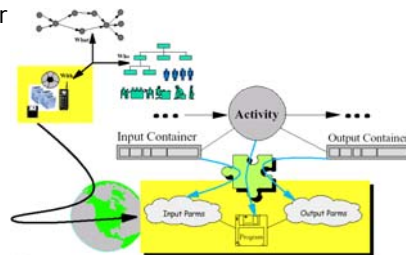
Deadlines

- Most processes must be performed in a certain time
 - E.g. for legal reasons or to meet company specific quality goals
- To support this, the WFMS allows to specify...
 - ...time limits at both, the process model level and the activity level
 - ...actions that should happen when a time limit is exceeded
 - Typical action is to notify somebody who has to take corrective actions
 - This facility is called "notification"
 - The processing of deadlines is called "escalation"
 - Deadlines can also be specified for actions associated with escalations
 - Escalations are escalated via notifying the process administrator
- The time measured for detecting out-of-line situations can be
 - ...the absolute time passed since the beginning of the situation to be monitored ("soccer semantics")
 - Time since activity has been scheduled, arrived on worklist, started to be worked on,...
 - ...the time passed on working on the activity or process to be monitored ("basket ball semantics")



"With" Dimension: Program Registration

- Decoupling activities and implementation
 - business process modeler can focus on process models, programs can be linked to activities later
 - programs depend on the environment they are running in
 - often program signatures depend on the environment
 - mapping from container to signature must be specified: "Data Mapping Language"
 - programs should be able to be exchanged without requiring to modify process models ("late binding")
- WFMS can resolve actual program to call when activity implementation must be invoked, based on meta-data for
 - program calls
 - method invocation
 - message queuing
 - TP-monitor interactions



Web service to the rescue!

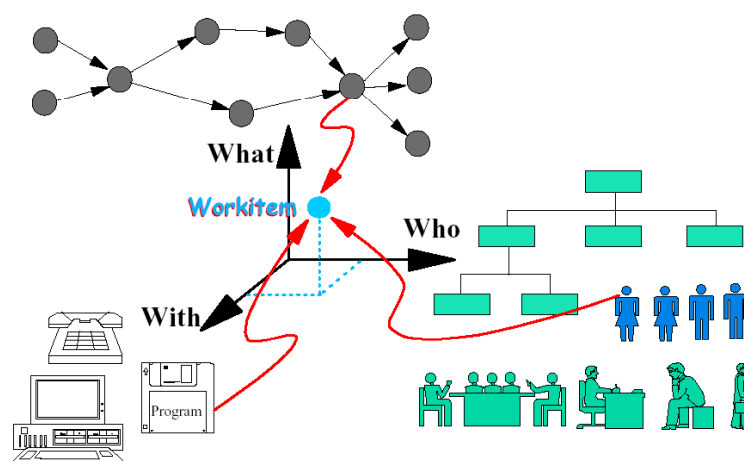


© Prof. Dr.-Ing. Stefan DeBloch

23

Enterprise Information Systems

The Three Dimensions Of Workflow



© Prof. Dr.-Ing. Stefan DeBloch

24

Enterprise Information Systems

Defining Worklists

- Worklist: collection of workitems that have the same common characteristics
- Characteristics are defined via queries on workitem properties
 - Especially, a workitem can be on multiple worklists
 - Worklists of different agents
 - Different worklists of the same agent
- Not only people or program executors may have worklists but also each instance of any element of the org metamodel
 - Worklists associated with an org instance that collects multiple people is called a group worklist
 - All users belonging to the group associated with the group worklist can pick a workitem from that list
- Modes of worklists
 - Pull
 - explicitly request refresh
 - suitable for high throughput environments, where certain worklists might be in constant flux!
 - Push
 - are always up to date
 - Grab
 - deliver a matching workitem on request ("get next workitem")
 - convenient for group worklists



Putting Process Models Into Production

- When modeling a process is finished it can be put into production
- Putting a process model into production means
 - ...to "freeze" the model, i.e. nobody can change it any more
 - Only "what" dimension (the activities and control-/dataflow between them) is really frozen
 - Organization model ("who dimension") can of course be modified
 - E.g. people can change departments
 - Might impact staff queries (e.g. dropping a department a query refers to): If no agent is found process administrator is notified
 - Often, organizational structure is completely maintained via separate application (e.g. Human Resource) and replicated periodically into the WFMS database in batch mode
 - Activity implementations ("with dimension") can be "early bound" or "late bound"
 - Early bound process model is frozen too, late bound process model is resolved at runtime
 - ...often to TRANSLATE the corresponding data into a different format
 - Modeling tool and WFMS runtime might use database structure optimized to their needs
 - ...often to create a new version of an already existing process model ("valid from")
 - Existing instances of earlier versions are run according to the model which was valid when the instance has been created (auditability is a key requirement!)
 - New instances are created according to the new version
- Once put into production, instances can be made from a model



Runtime

- WFMS proactively drives the processes
 - process navigation
 - interaction with end users, applications
- Support of process queries to locate a particular process or set of processes
 - may be based on operational (e.g., start date) or business selection criteria (e.g., customer name)
 - a *key container* may be defined and filled with interesting data for that purpose
- Audit trail
 - recording of important events in the life of a process
 - possible usages
 - legal requirements
 - analysis process reengineering
- Monitoring process collections for out-of-line situations
 - Leitstand
- Process Repair
 - administrator gets notifications about erroneous situations
 - may manually fix errors (e.g., content of containers, state of activity, assigned implementation or resource)



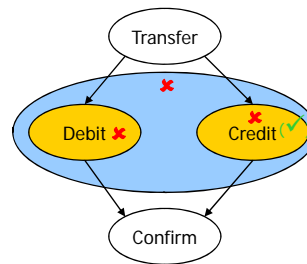
Workflows And ACID Transactions

- ACID properties
 - Atomicity, consistency, isolation, durability
- Distributed transactions
 - (distributed) two-phase commit
 - DTP X/Open
- Flat transaction model
- Foundation for DBMS, TP monitors
- Hidden assumption: transactions are short
 - ~ 0.1 – 2 seconds
- ACID TAs and workflows
 - allow automated recovery, based on ACID properties
 - most feasible for highly automated (sub-)processes with short-duration activities
 - support for automated activity "streams"
 - can be modeled explicitly using *atomic spheres*



Atomic Spheres (global TAs)

- Set of activities where either all activities in a sphere commit, or none
- Properties:
 - Each activity in an atomic sphere is **transactional**
 - Manipulates resources in RM according to X/OPEN DTP
 - Does not establish TA boundaries by itself
 - All connectors entering the sphere have the same activity as their origin
 - Ensure fast execution of sphere (distributed TA)
 - If an activity is rolled back, then all previously completed activities in the sphere are rolled back as well
- In case the sphere is rolled back, it can be automatically rerun by the WFMS
 - based on persistent input containers
- Atomic sphere allow reuse of existing transactional programs



© Prof. Dr.-Ing. Stefan DeBloch

29

Enterprise Information Systems

Atomic Sphere (cont.)

- Specifying atomic spheres may also involve
 - number of retries in case of failure
 - action if the sphere finally fails
 - notify administrator, "skip" the activities of the sphere, ...
- WFMS implementation
 - Start global TA when control flow enters atomic sphere
 - Wait for running activities in sphere to complete when control flow leaves the sphere, and commit global TA
 - If commit fails, carry out further steps (repeat, exception WF, ...) based on sphere parameters
- Global Transactions: Practice
 - Transaction with multiple participants
 - Atomic commitment is the issue
 - E.g. 2-phase-commit protocol
 - Not realistic across organization boundaries
 - Not only „efficiency“ issues but additional legal-, ownership-, privacy-,... issues
 - Especially not in Internet scenarios



© Prof. Dr.-Ing. Stefan DeBloch

30

Enterprise Information Systems

Long Transactions

- "Long" is a couple of seconds to years
 - Batches
 - Multi-step transactions
 - Design activities
 - ...
- Basic characteristics are:
 - Must survive (planned as well as unplanned) interrupts
 - Including power-off
 - Rollback of the whole transaction due to local failure is not tolerable
- Often correspond to a business process
 - business transaction
 - based on advanced TA models that 'weaken' the ACID properties
 - define groups of operations within transactions, dependencies between groups, transactional properties for the groups (in addition to transaction as a whole)
 - can be modeled explicitly using compensation spheres in the context of WFMS

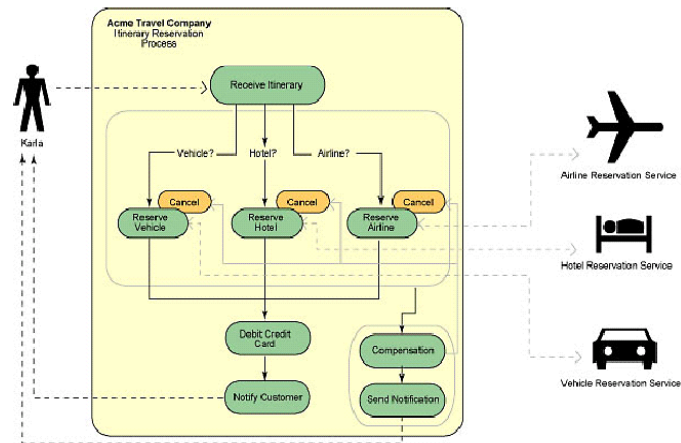


Compensation Spheres

- Set of activities that must complete successfully as a whole
 - Otherwise it must be undone semantically
- Activities can be arbitrary
 - Don't have to be realized as transactions
- Each activity in the sphere and the compensation sphere itself is associated with a compensating action
 - Could be the NULL action
- A compensating action may be an activity or (complex) business process
- If an activity fails
 - Compensating actions of all completed activities in the sphere are executed in 'reverse' order
 - Compensating action associated with the compensation sphere is executed



Compensation Spheres – Example



© Prof. Dr.-Ing. Stefan DeBloch

33

Enterprise Information Systems

Phoenix Behavior

- Forward Recovery ("recover out of the ashes")
 - workflow itself must be recoverable
 - persistent, recoverable workflow context (using DBMS)
 - reliable messaging for communicating workflow events
 - implementations of activities must be included in the recovery processing of the workflow
 - "safe" activities
 - utilizes stratified transactions
 1. generate request for program execution agent
 2. transport request from server queue to program execution agent queue
 3. read message, execute activity implementation, generate completion message
 4. transport completion message to server queue
 5. read completion message, store workflow state change



© Prof. Dr.-Ing. Stefan DeBloch

34

Enterprise Information Systems

Conclusions

- Business (Re-)Engineering, Business Process Modeling
 - goal: efficient execution of core business processes
 - explicit specification of process models
 - focus on control flow, rudimentary data flow, organizational aspects
 - process optimization and analysis
 - static optimization
 - simulation (analytical, discrete event/based)
- Workflow Management Systems
 - middleware for management, control and execution of business processes
 - build time
 - extend process models created using BPR tools
 - data flow, control flow details, organization aspects, activity implementation
 - run time
 - work item lists, process life cycle and process management, audit trail



Conclusions (cont.)

- Workflows and transactions
 - ACID is too strict for long transactions
 - only appropriate for individual activities or restricted subset of activities (atomic spheres)
 - advanced transaction concepts
 - complex transaction structures
 - compensation
 - forward recovery
 - compensation spheres: sets of semantically linked transactional (sub-)activities
- Advantages compared to explicit modeling of exception/failure handling steps as part of the process model
 - Reduces complexity of the process
 - Separation of regular business logic from exception/failure handling
 - Increased flexibility
 - compensation of spheres that have completed successfully

