Architecture World '08

Enterprise architecture strategies, best practices from the Gurus!

Towards executable models within BPM

Dr Alexander Samarin
www.samarin.biz

Overview

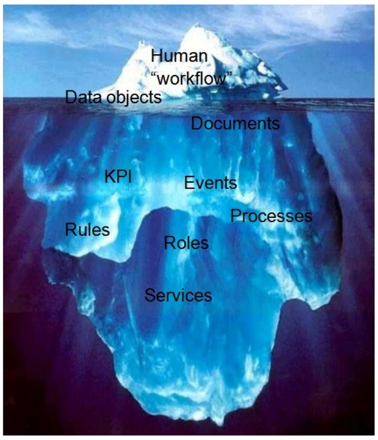
- □ Vision – *enterprise architecture* is a (semi-)exact science which provides **guidance and practical help** for the transformation of an enterprise to achieve certain desired characteristics (e.g. level of maturity, greater agility, better collaboration)
- □ Common understanding of artefacts
- □ Modelling of executable business processes
- □ IT governance as a BPM system

Architecting an enterprise BPM system (with systems thinking)

- A BPM system is a dynamic set of artefacts
- Artefacts are interconnected and interdependent
- We have to anticipate potential changes:
  - policies, priorities, compliance, technology, etc.
- Implementation of such changes necessitates the evolution of some artefacts and the relationships between them
- It must be easy to modify all artefacts and relationships without causing any negative effects

All BPM artefacts

- added-value chain
- events
- processes
- rules
- activities
- roles
- objects (data structures)
- objects (documents)
- audit trails
- performance indicators
- services



Main architecting principles

- □ All artefacts must be evolved to become **digital**, **external** and **virtual**
- □ All artefacts must be **versionable** throughout their lifecycle
- □ All relationships between these artefacts are **modelled explicitly**
- □ All models are made to be **executable**

Relationships between artefacts

- □ Reveal all hidden relationships and structure them – examples:
  - ◆ static (in design phase)
  - ◆ dynamic (in execution phase)
  - ◆ composition (from atomic artefacts to a composite artefact)
  - ◆ instantiation (from a template to instances)
  - ◆ compatibility (between different versions)
- □ If possible, model relationships as formal, explicit, traceable, testable, secure, SLA aware and executable

Knowledge about artefacts

- For each artefact
  - definition and categories, if any
  - naming convention, if any
  - attributes
  - volume
  - dependencies
  - security (e.g. ownership)
  - life-cycle
  - versioning
  - examples

Artefact: Event

- A construct that represents an incident occurring in the business environment, which warrants some action from the business
- Categories
  - temporal
  - external
  - internal
  - spontaneous
- Usage
  - Decoupling of processes (relation with EDA)
  - Records management
- Challenges
  - How to extract the events from existing applications

Artefact: Role

- A construct that represents the actions and activities assigned to, or required or expected of, a person or group
- Possible categories of roles
  - organizational
  - functional
  - special expertise
  - project
  - security
- Usage
  - Defines who can carry out a particular operation with a particular artefact
  - Separation of duties
- Challenges
  - How to derive a role from others (an example of a relationship)

Artefact: Object

- Business objects are formal information descriptions of real things and people which constitute the business
- Categories
  - Data structures
  - Documents
- Usage
  - Information encapsulation
- Challenges
  - Transportation between different applications (exchange formats)
  - Documents as data structures
  - Data structures as documents

Artefact: Rule

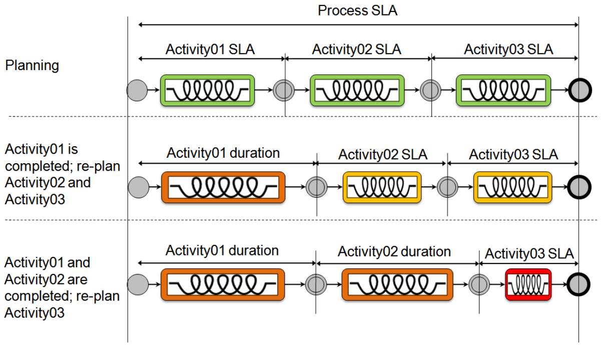
- Business rules are constraints and conditions under which the enterprise operates
- Categories
  - Facts
  - Relationships between facts
  - Constraints
  - Derivations
- Usage
  - Decision management
- Challenges
  - How to enable maintenance by the business owners (i.e. the users)

Artefact: Audit trail

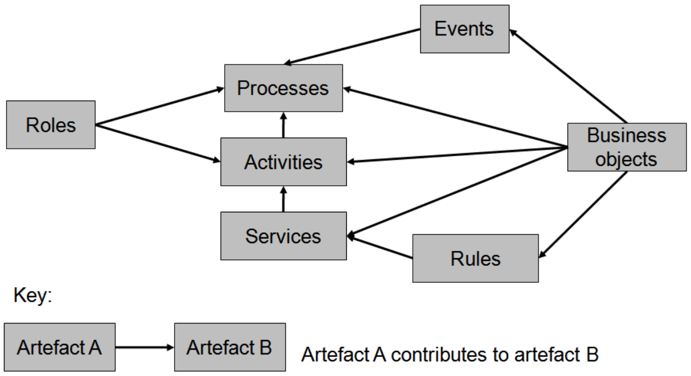
- Record of some information about a BPM system to be able to analyse its behaviour at a later date
- Categories
  - Technical
  - Business
- Usage
  - Traceability
  - Performance measurement
  - Post-mortem analysis
- Challenges
  - How to embed explicitly an audit trail into a process

Artefact: KPI

- Key Performance Indicators (KPIs) are a limited number of (agreed) quantifiable measurements that measure how well something or somebody is achieving its or his/her objectives
- In other words, KPIs measure the performance against the Service Level Agreement (SLA)



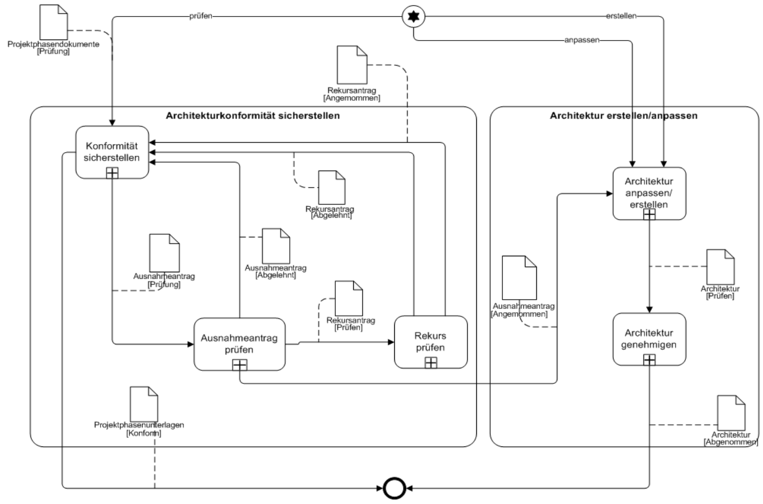
Dependencies between artefacts



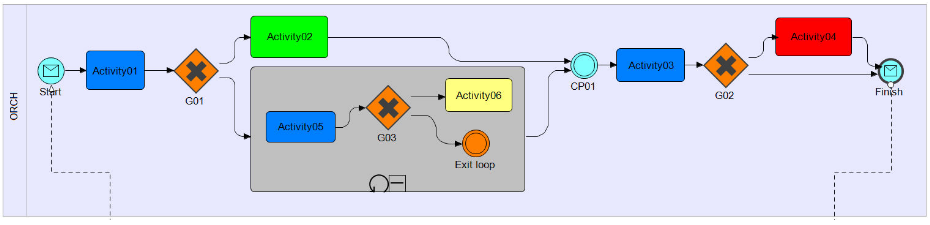
Modelling of executable business processes

- Diagramming style in BPMN
- A dozen practical patterns
- Structuring for better "executability"
- Modelling procedure
- Use a common tool (business and IT) for prototyping – Intalio BPM suite

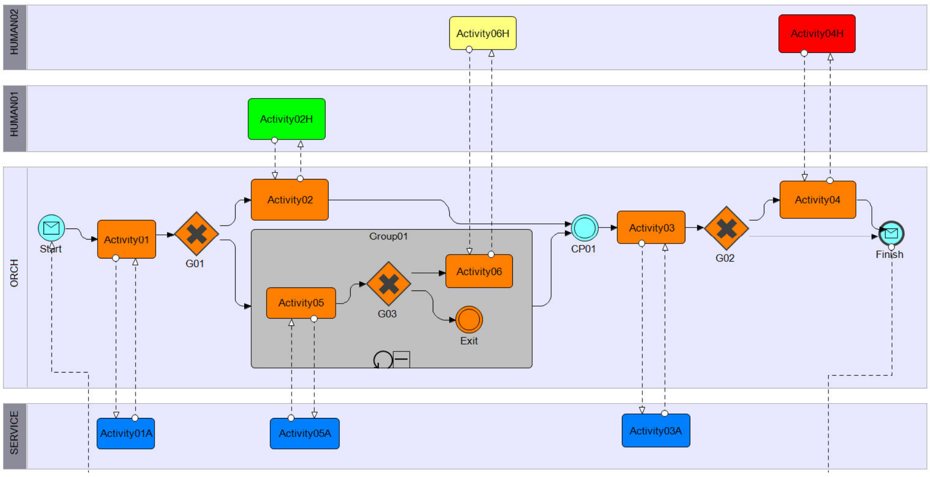
Example of unstructured BPMN


Projektphasendokumente [Prüfung]
prüfen
erstellen
anpassen
Rekursantrag [Angemommen]
Architekturkonformität sicherstellen
Architektur erstellen/anpassen
Konformität sicherstellen
Rekursantrag [Abgelehnt]
Architektur anpassen/ erstellen
Ausnahmeantrag [Prüfung]
Ausnahmeantrag [Abgelehnt]
Architektur [Prüfen]
Ausnahmeantrag [Angemommen]
Rekursantrag [Prüfen]
Ausnahmeantrag prüfen
Rekurs prüfen
Architektur genehmigen
Projektphasenunterlagen [Konform]
Architektur [Abgenommen]

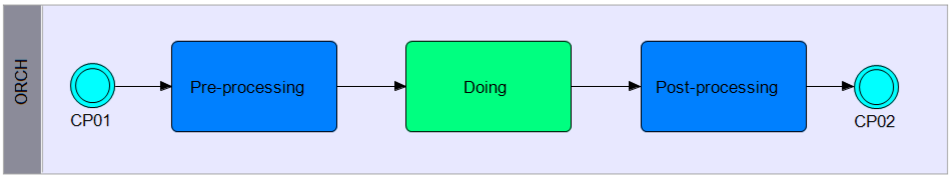
Diagramming style in BPMN (1)



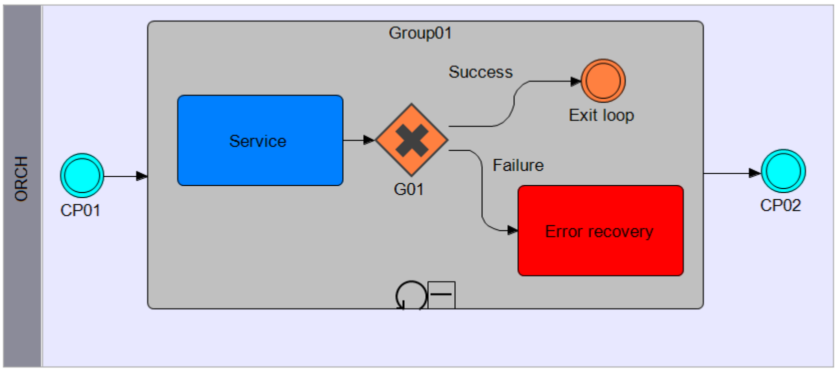
Diagramming style in BPMN (2)



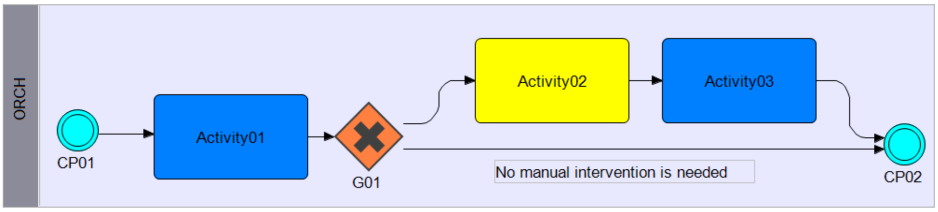
Pattern AHA – Automated, Human, Automated



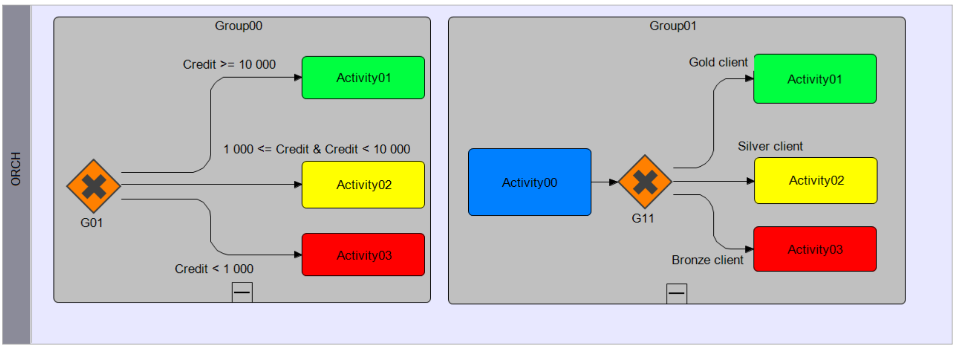
Pattern ERL – Error Recovery Loop



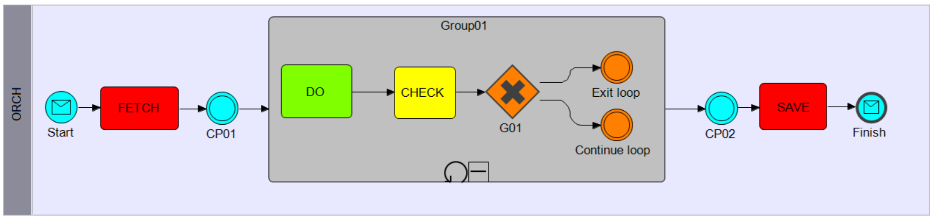
Pattern M&M – Man & Machine



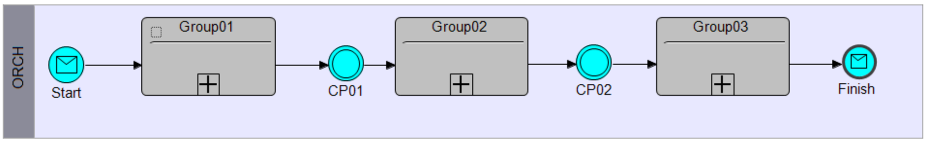
Pattern DBL – Decoupled Business Logic



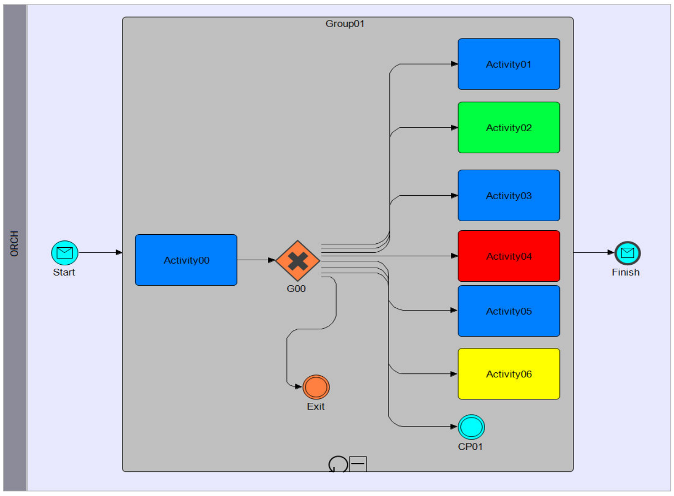
Pattern IPS – Initial Process Skeleton



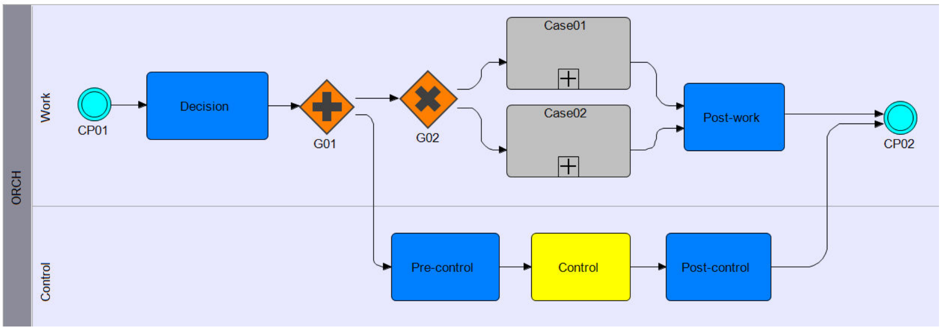
Pattern SYP – Structure Your Process



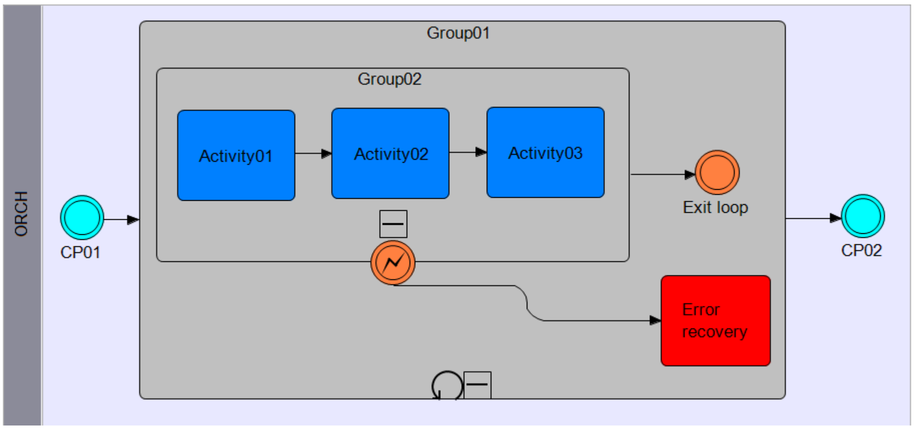
Pattern PRF – Process Realisation Faked



Pattern DEC – Decide, Execute, Control



Pattern IRIS – Integrity Reached via Idempotency of Services



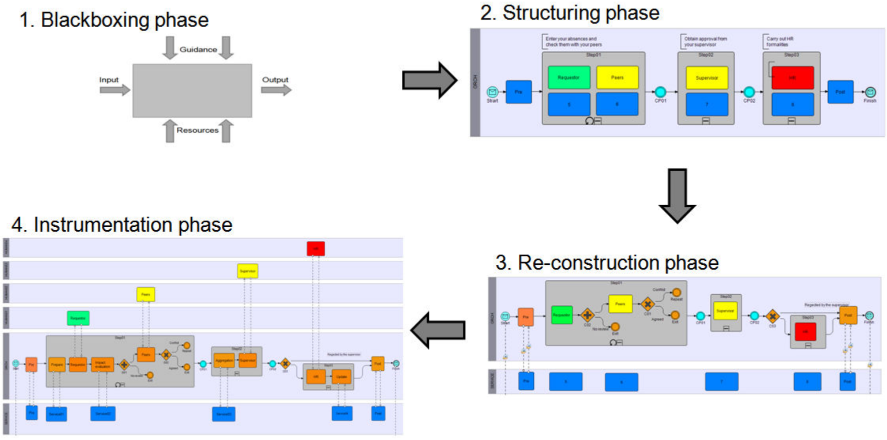
Principles of the modelling procedure

- it treats human and automated activities equally
- it is primarily for capturing the flow of control within a building block, and not for optimisation
- it is a tool for both the business and the IT (maybe with coaching by a process analyst)
- it provides validation by simulation
- it provides validation by quick prototyping – real services can be invoked
- it is “visual programming”

The modelling procedure

- Its purpose is
  - to analyse a building block (what it is supposed to do)
  - to synthesise its implementation (how it does this) as the explicit coordination of other building blocks (processes or activities)
- It is iterative – we can apply it until we have left only indivisible building blocks (i.e. activities)
- Artefacts are constructed recursively, like Russian dolls

Four phases



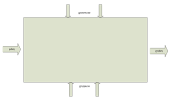
1. Blackboxing phase

2. Structuring phase

3. Re-construction phase

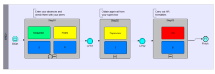
4. Instrumentation phase

Blackboxing phase

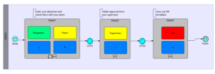


- The purpose
  - to analyse a building block as a whole
  - to discover its functional characteristics and some related artefacts
- The method
  - the business story behind this building block should be carefully analysed to determine some of its artefacts
- Recommendations
  - at this point, don't go into excessive detail for each artefact; this should be done later

Structuring phase (1)

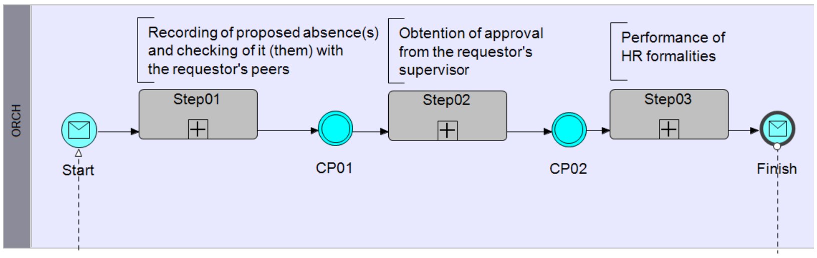


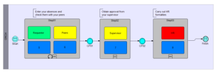
- The purpose
  - to analyse a building block from within to determine its internal structure and its major artefacts
- The method
  - determine the main functional (or logical) steps
  - add check-points between steps
  - classify artefacts for these steps
- Recommendations
  - don't have more than 7 steps
  - avoid loop-back over check-points



Structuring phase (2)

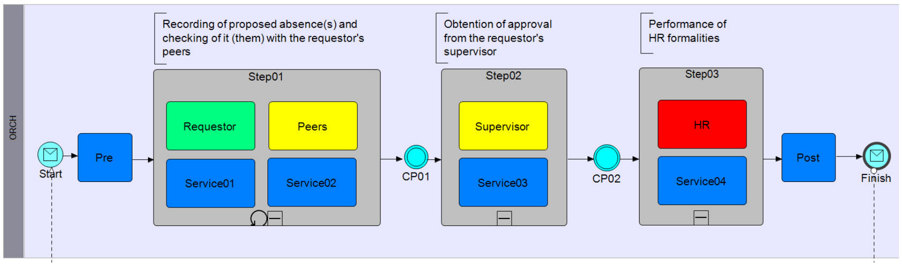
- Steps and check points


ORCH
Recording of proposed absence(s) and checking of it (them) with the requestor's peers
Obtention of approval from the requestor's supervisor
Performance of HR formalities
Step01
Step02
Step03
Start
CP01
CP02
Finish

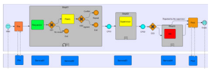


Structuring phase (3)

- Steps, check points and artefacts

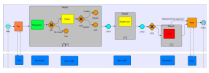

ORCH
Recording of proposed absence(s) and checking of it (them) with the requestor's peers
Obtention of approval from the requestor's supervisor
Performance of HR formalities
Step01
Step02
Step03
Start
Pre
Requestor
Peers
Service01
Service02
CP01
Supervisor
Service03
CP02
HR
Service04
Post
Finish

Re-construction phase (1)

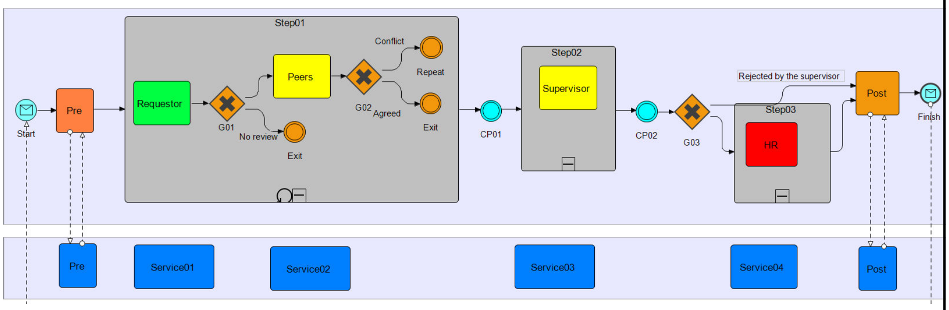


- The purpose
  - to synthesize an initial version of the formal coordination: some kind of process skeleton
- The method
  - add intra-step logic
  - start formalising the business objects involved
  - collect test scenarios
- Recommendations
  - consider implementation of human activities as interactive forms

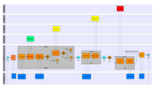
Re-construction phase (2)



- The diagram

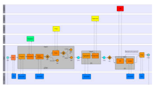


Instrumentation phase (1)

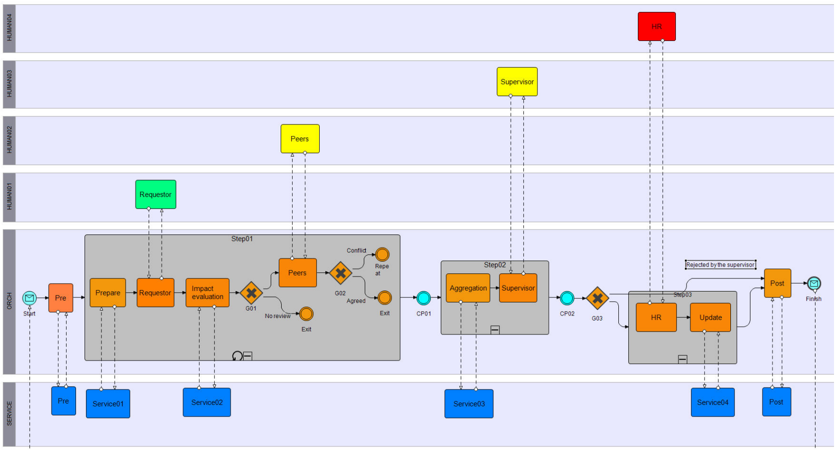


- The purpose
  - to enrich the process skeleton by adding more automated activities
- The method
  - add pools
  - apply different practical patterns
  - use a business rule engine if available
  - collect test scenarios
- Recommendations
  - work iteratively (step-by-step)

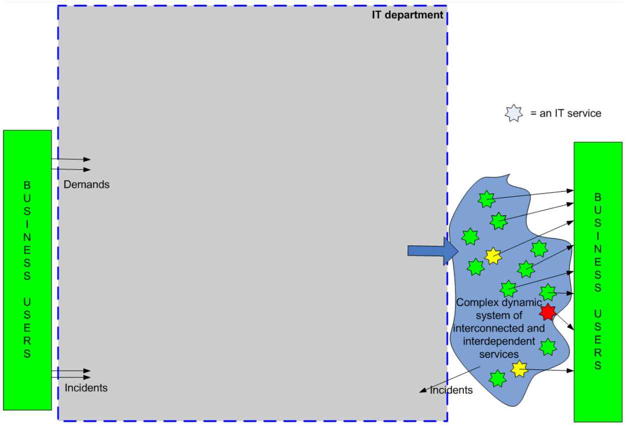
Instrumentation phase (2)



- The diagram



IT governance as a BPM system (1)



IT department

= an IT service

BUSINESS USERS

Demands

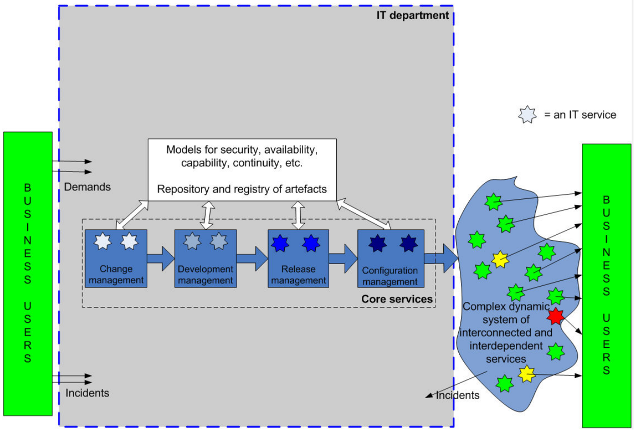
Incidents

Complex dynamic system of interconnected and interdependent services

Incidents

BUSINESS USERS

IT governance as a BPM system (2)



IT department

= an IT service

BUSINESS USERS

Demands

Models for security, availability, capability, continuity, etc.

Repository and registry of artefacts

Change management

Development management

Release management

Configuration management

Core services

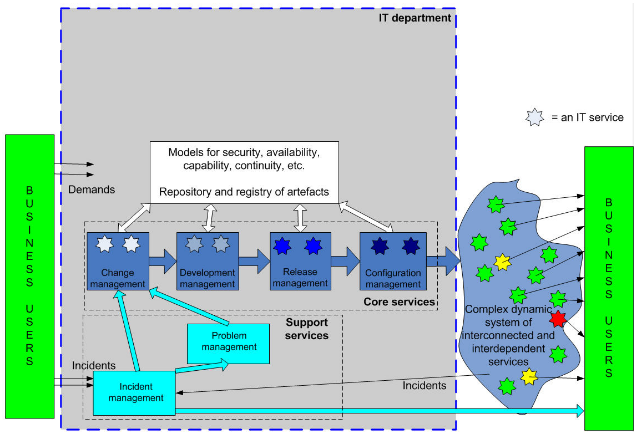
Complex dynamic system of interconnected and interdependent services

Incidents

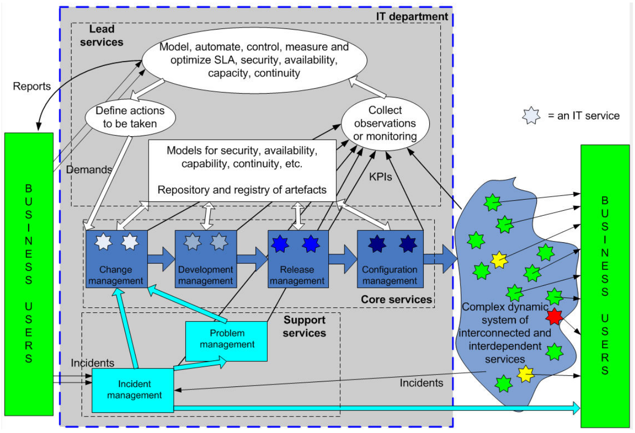
Incidents

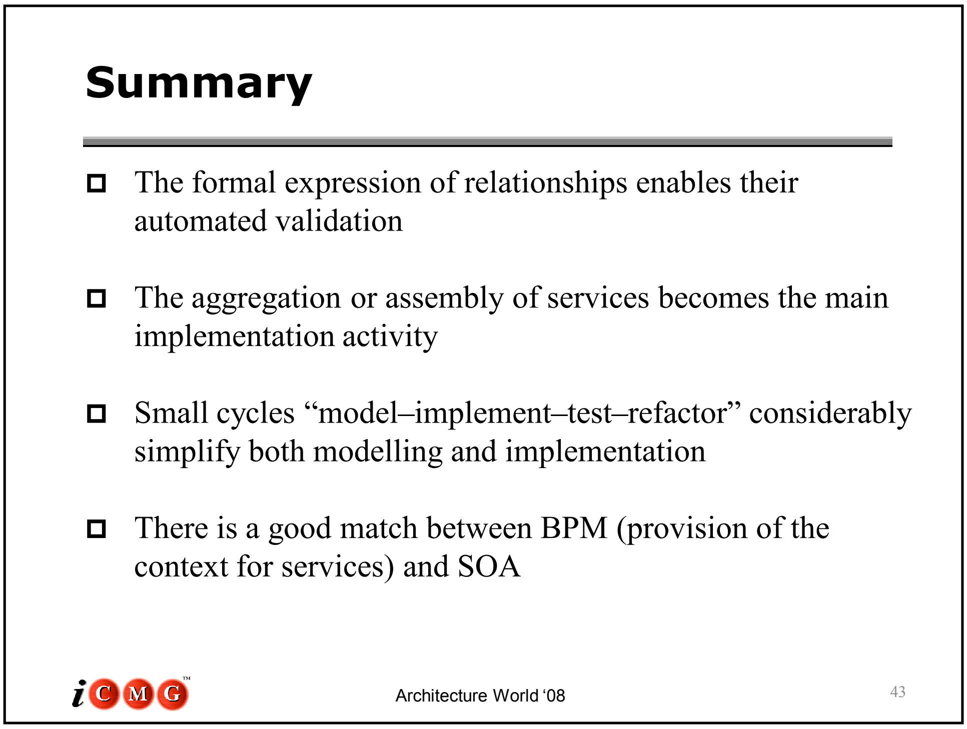
BUSINESS USERS

IT governance as a BPM system (3)


IT department
= an IT service
Models for security, availability, capability, continuity, etc.
Repository and registry of artefacts
Demands
B U S I N E S S U S E R S
Change management
Development management
Release management
Configuration management
Core services
Support services
Problem management
Incidents
Incident management
Incidents
Complex dynamic system of interconnected and interdependent services
B U S I N E S S U S E R S

IT governance as a BPM system (4)


IT department
Lead services
Model, automate, control, measure and optimize SLA, security, availability, capacity, continuity
Reports
Define actions to be taken
Collect observations or monitoring
= an IT service
Demands
Models for security, availability, capability, continuity, etc.
Repository and registry of artefacts
KPIs
B U S I N E S S U S E R S
Change management
Development management
Release management
Configuration management
Core services
Support services
Problem management
Incidents
Incident management
Incidents
Complex dynamic system of interconnected and interdependent services
B U S I N E S S U S E R S



Summary

- □ The formal expression of relationships enables their automated validation
- □ The aggregation or assembly of services becomes the main implementation activity
- □ Small cycles “model–implement–test–refactor” considerably simplify both modelling and implementation
- □ There is a good match between BPM (provision of the context for services) and SOA