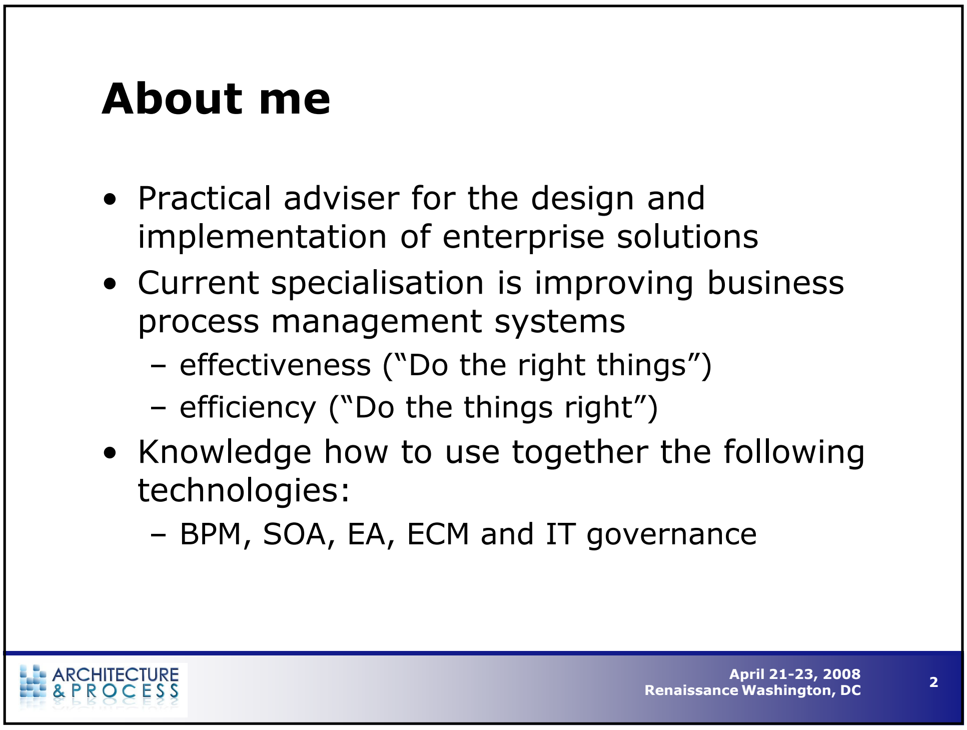
ARCHITECTURE & PROCESS

Welcome

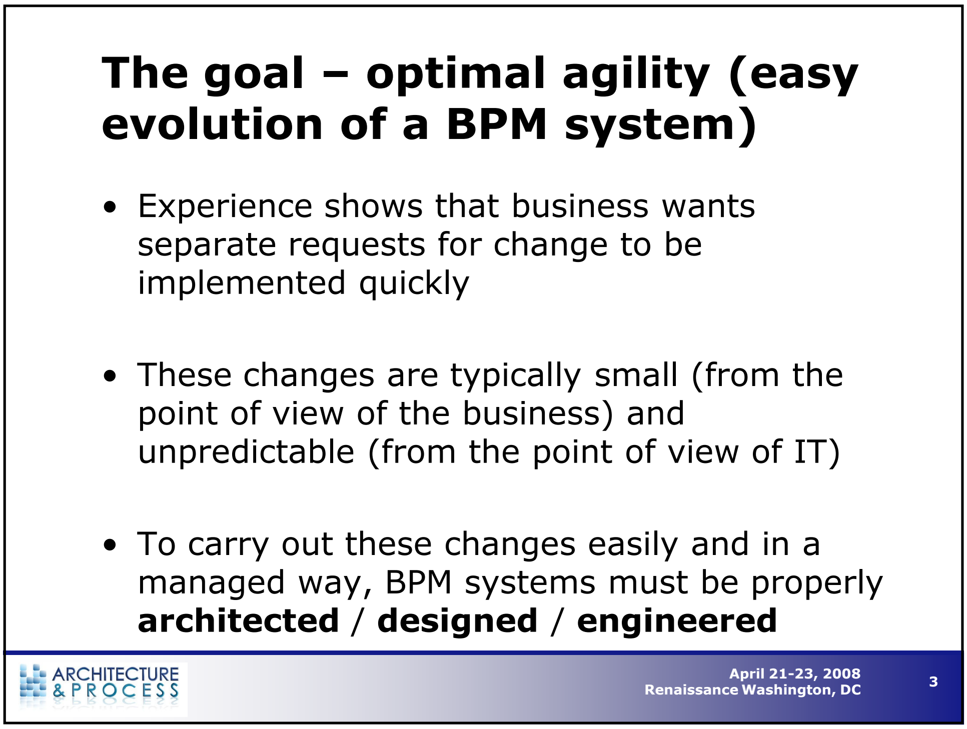
Dr Alexander Samarin
Enterprise Solutions Architect
www.samarin.biz

Architecting enterprise BPM systems for optimal agility



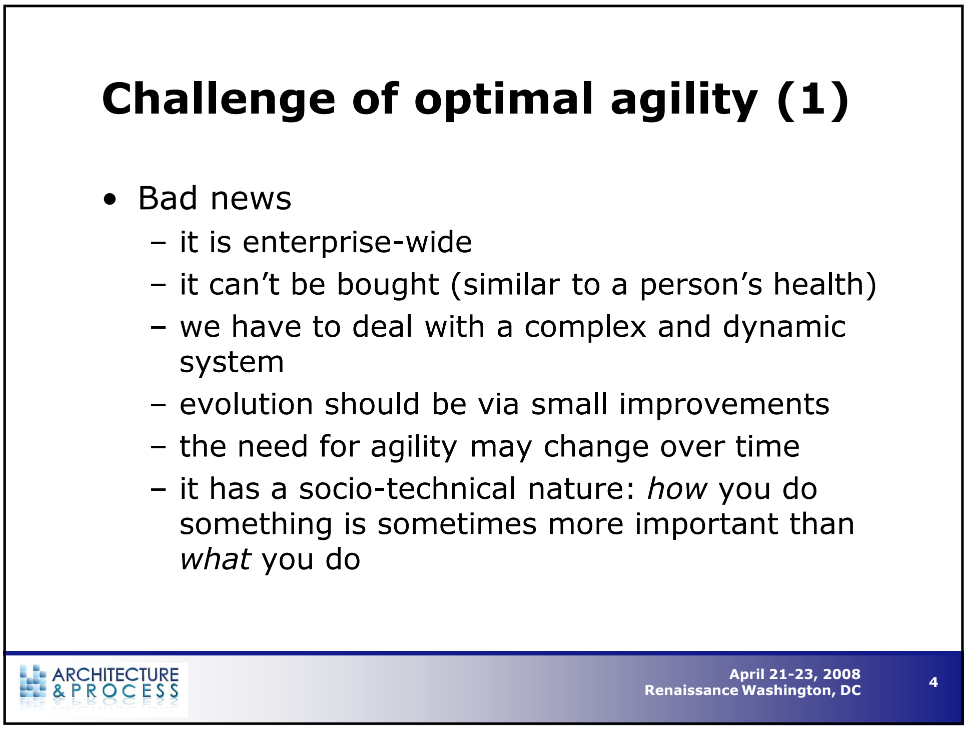
About me

- Practical adviser for the design and implementation of enterprise solutions
- Current specialisation is improving business process management systems
  - effectiveness (“Do the right things”)
  - efficiency (“Do the things right”)
- Knowledge how to use together the following technologies:
  - BPM, SOA, EA, ECM and IT governance



The goal – optimal agility (easy evolution of a BPM system)

- Experience shows that business wants separate requests for change to be implemented quickly
- These changes are typically small (from the point of view of the business) and unpredictable (from the point of view of IT)
- To carry out these changes easily and in a managed way, BPM systems must be properly **architected** / **designed** / **engineered**

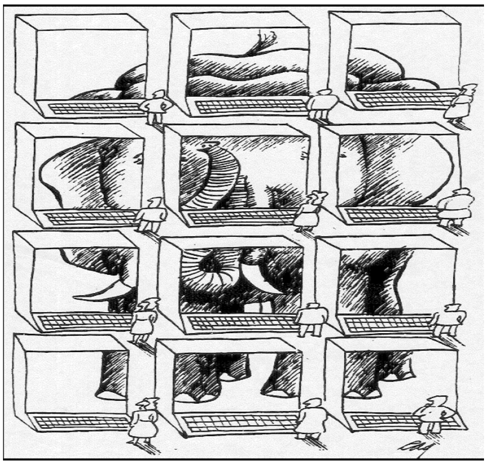


Challenge of optimal agility (1)

- Bad news
  - it is enterprise-wide
  - it can't be bought (similar to a person's health)
  - we have to deal with a complex and dynamic system
  - evolution should be via small improvements
  - the need for agility may change over time
  - it has a socio-technical nature: *how* you do something is sometimes more important than *what* you do

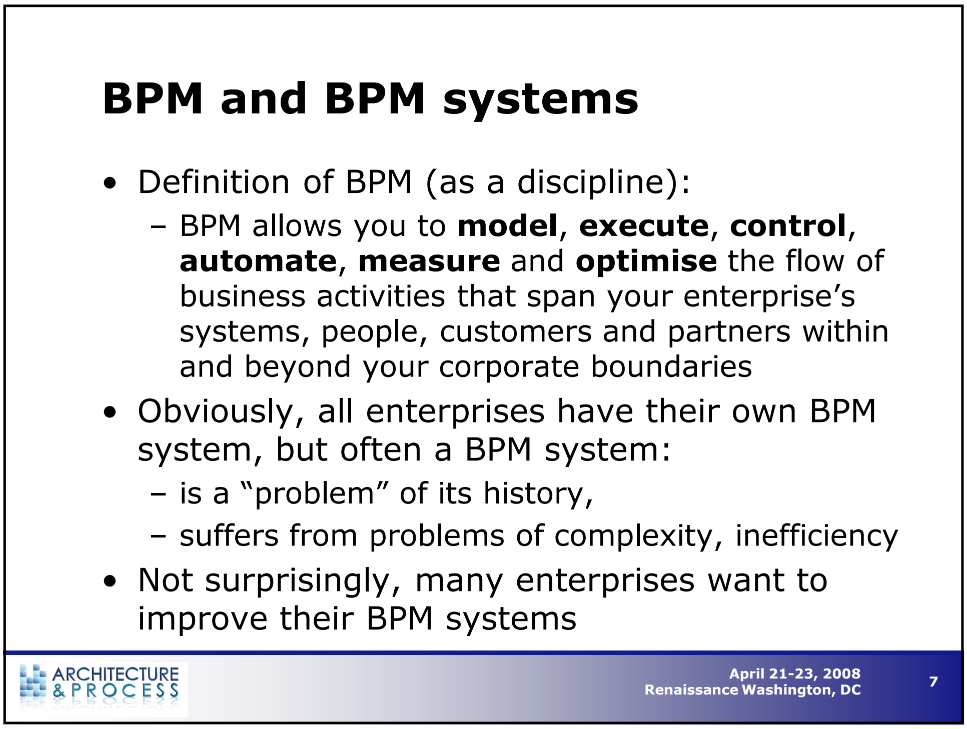
Challenge of optimal agility (2)

- Many stakeholders
  - top manager
  - business manager
  - process owner
  - super-users
  - users
  - business analysts
  - IT managers
  - architects
  - developers
  - operators



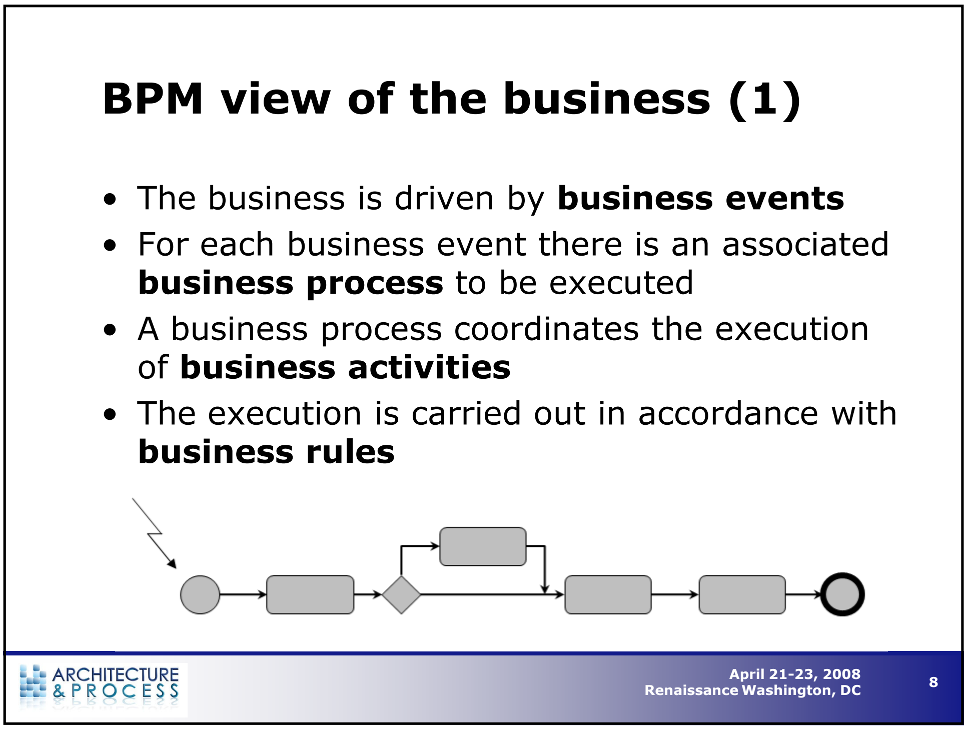
Challenge of optimal agility (3)

- Good news
  - there are many good business process improvement methods
  - BPM is appreciated an enterprise-wide management discipline
  - there is understanding of the relationship between BPM and other business process improvement methods
  - “BPM suite” software products are available
  - agile development has been proven to be feasible
  - Service-Oriented Architecture (SOA) is maturing



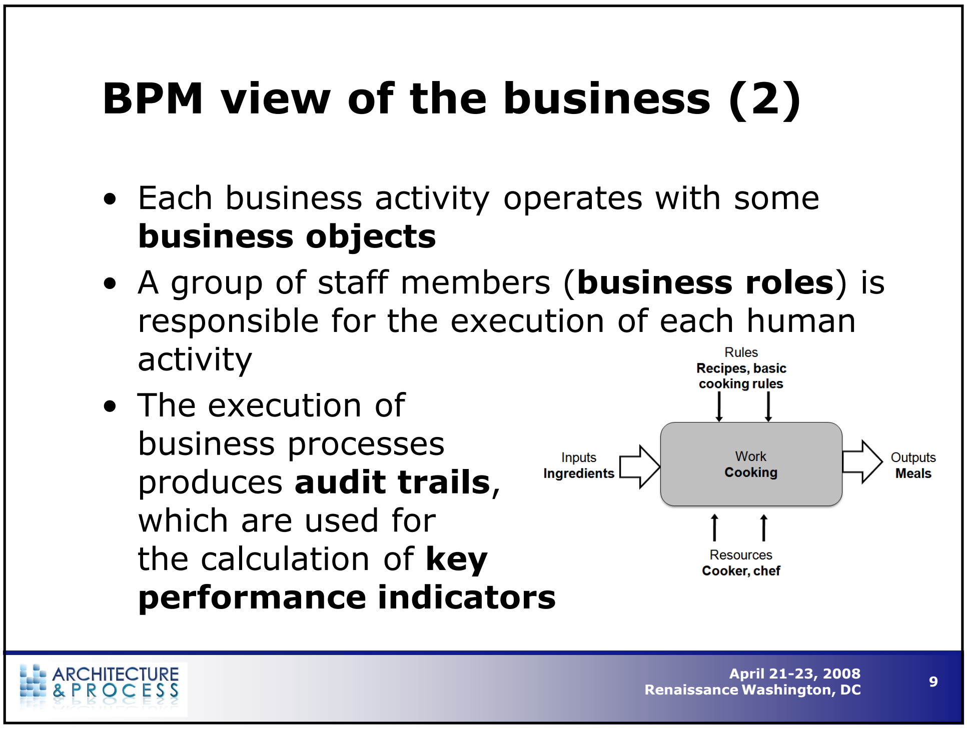
BPM and BPM systems

- Definition of BPM (as a discipline):
  - BPM allows you to **model**, **execute**, **control**, **automate**, **measure** and **optimise** the flow of business activities that span your enterprise's systems, people, customers and partners within and beyond your corporate boundaries
- Obviously, all enterprises have their own BPM system, but often a BPM system:
  - is a "problem" of its history,
  - suffers from problems of complexity, inefficiency
- Not surprisingly, many enterprises want to improve their BPM systems



BPM view of the business (1)

- The business is driven by **business events**
- For each business event there is an associated **business process** to be executed
- A business process coordinates the execution of **business activities**
- The execution is carried out in accordance with **business rules**



BPM view of the business (2)

- Each business activity operates with some **business objects**
- A group of staff members (**business roles**) is responsible for the execution of each human activity
- The execution of business processes produces **audit trails**, which are used for the calculation of **key performance indicators**

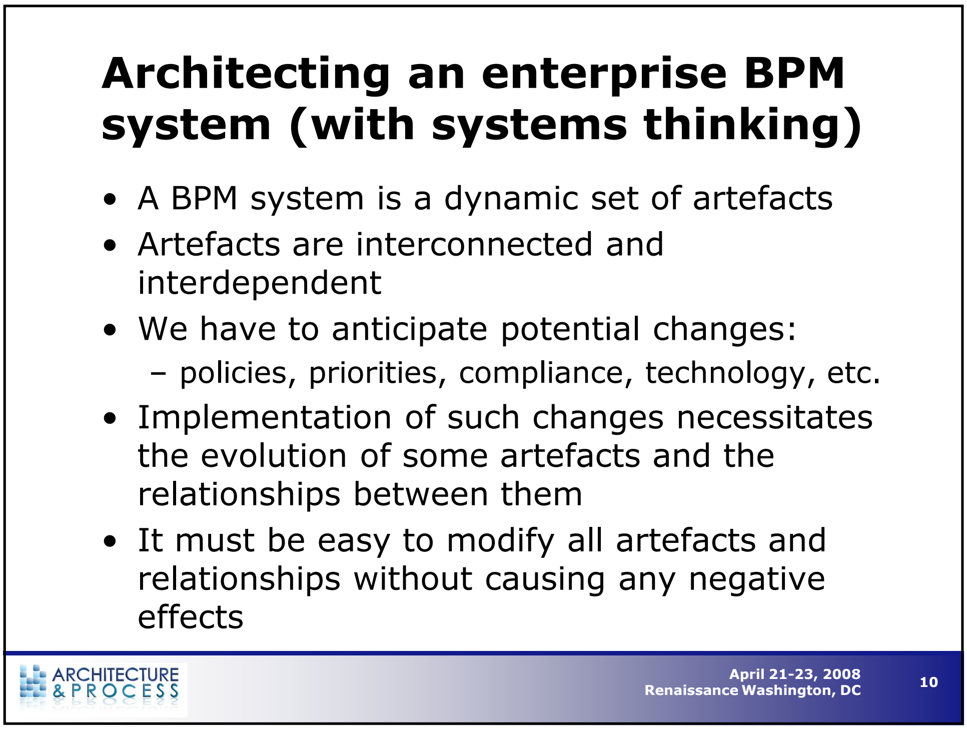
Rules
Recipes, basic cooking rules

Inputs
Ingredients

Work
Cooking

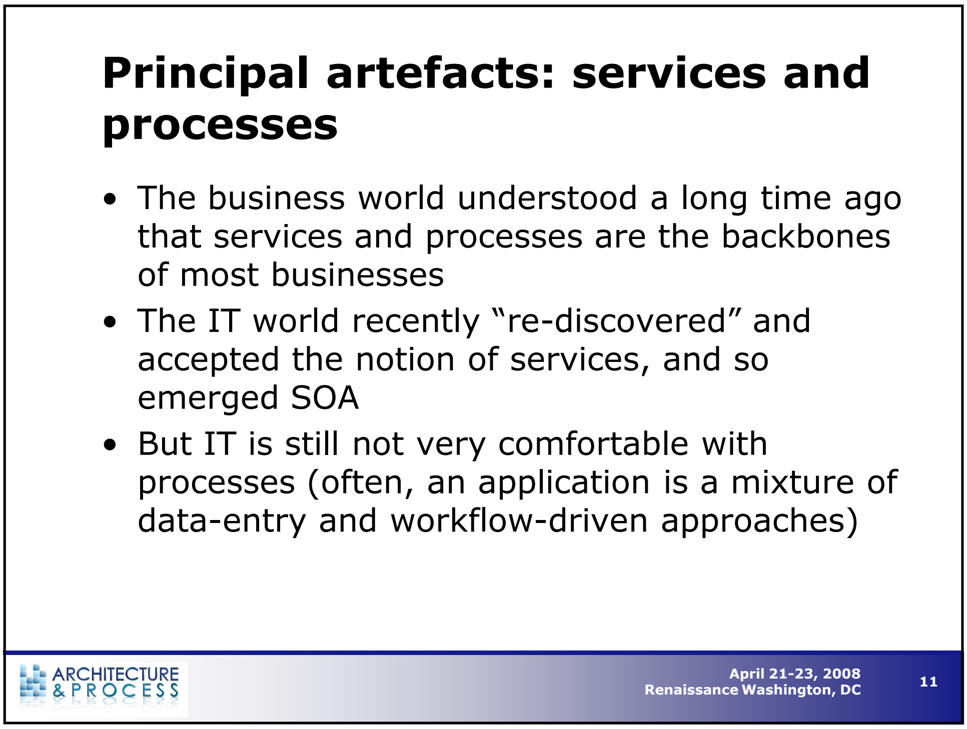
Outputs
Meals

Resources
Cooker, chef



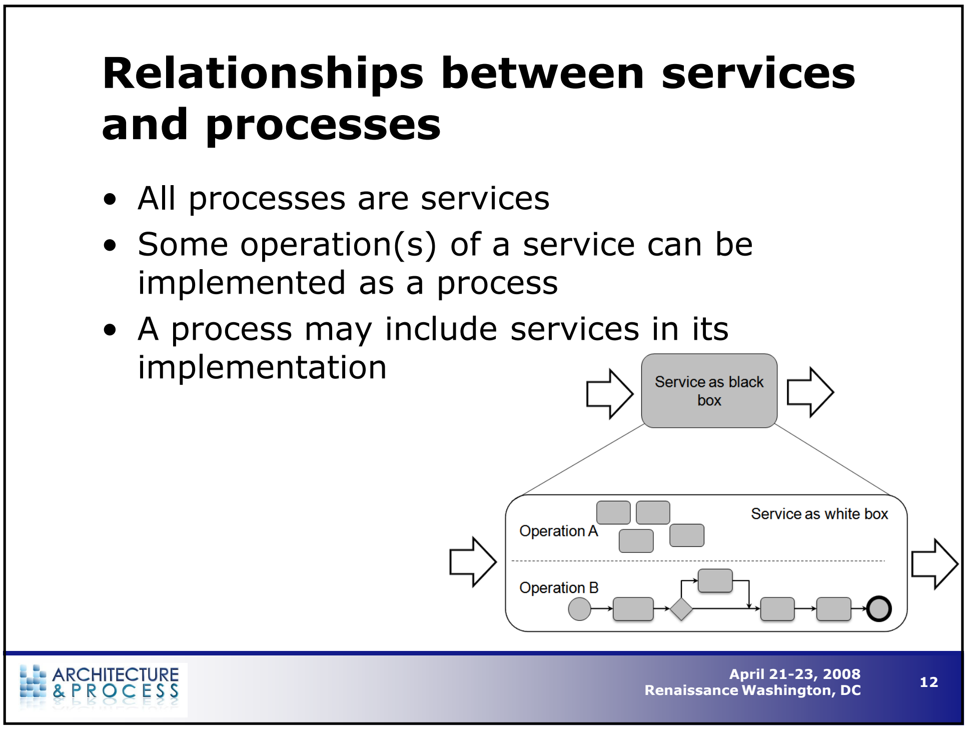
Architecting an enterprise BPM system (with systems thinking)

- 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



Principal artefacts: services and processes

- The business world understood a long time ago that services and processes are the backbones of most businesses
- The IT world recently “re-discovered” and accepted the notion of services, and so emerged SOA
- But IT is still not very comfortable with processes (often, an application is a mixture of data-entry and workflow-driven approaches)

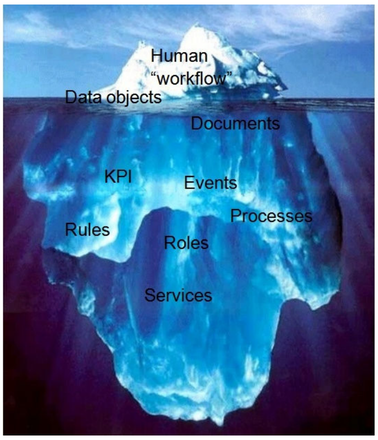


Relationships between services and processes

- All processes are services
- Some operation(s) of a service can be implemented as a process
- A process may include services in its implementation

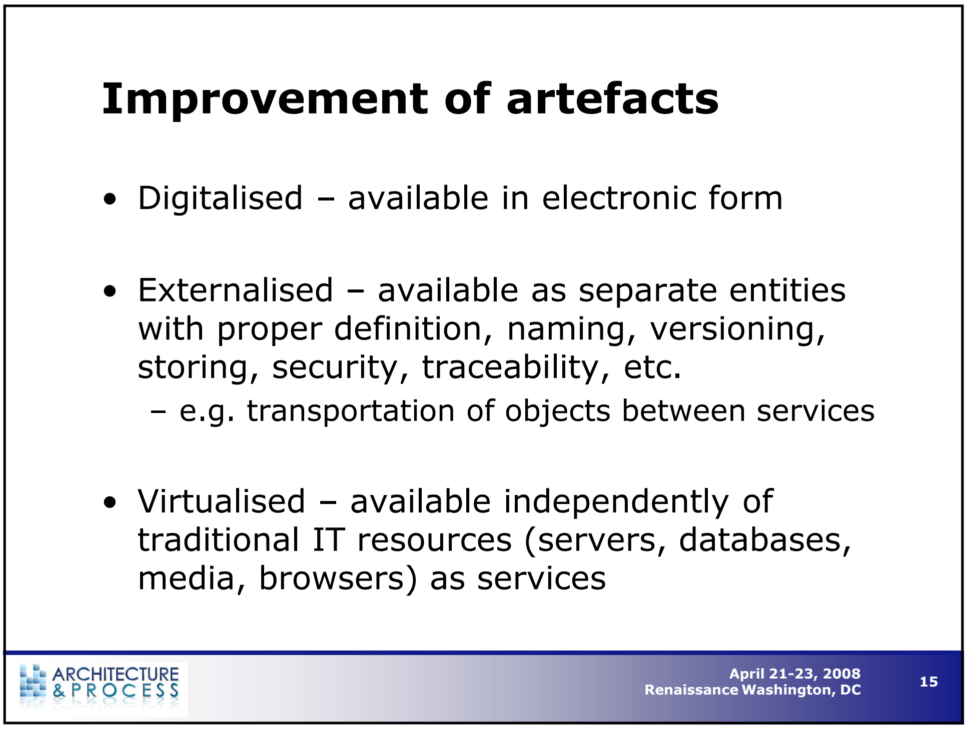
All BPM artefacts

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



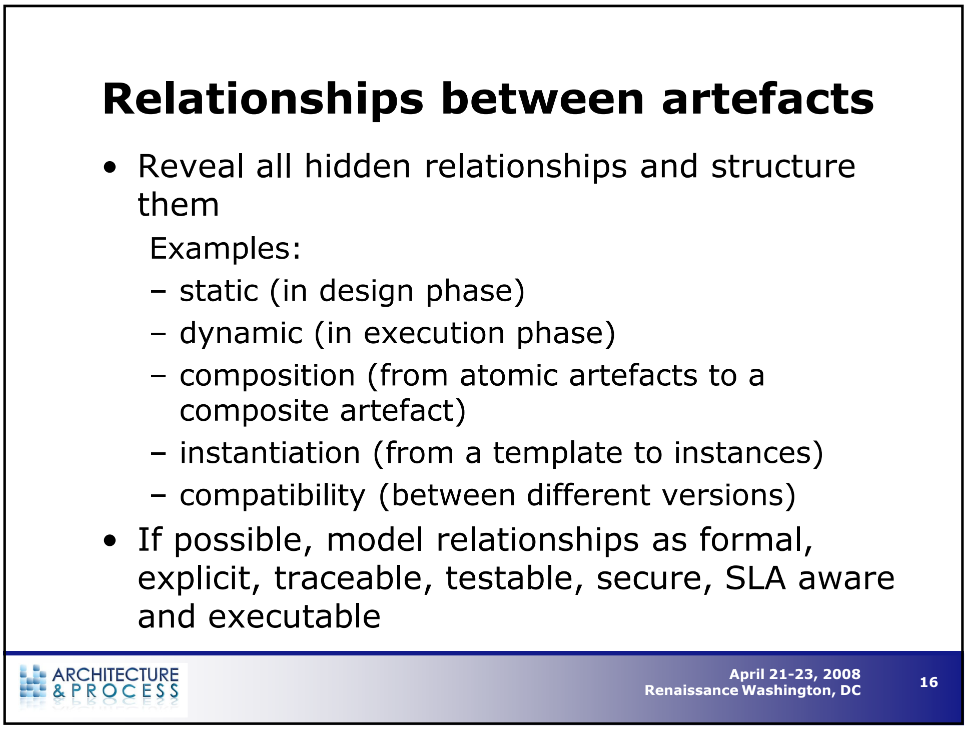
Main architecting principles

- All artefacts must be improved 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**



Improvement of artefacts

- Digitalised – available in electronic form
- Externalised – available as separate entities with proper definition, naming, versioning, storing, security, traceability, etc.
  - e.g. transportation of objects between services
- Virtualised – available independently of traditional IT resources (servers, databases, media, browsers) as services



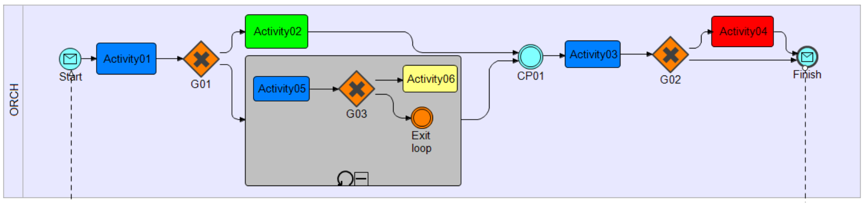
Relationships between artefacts

- 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

Explicit models

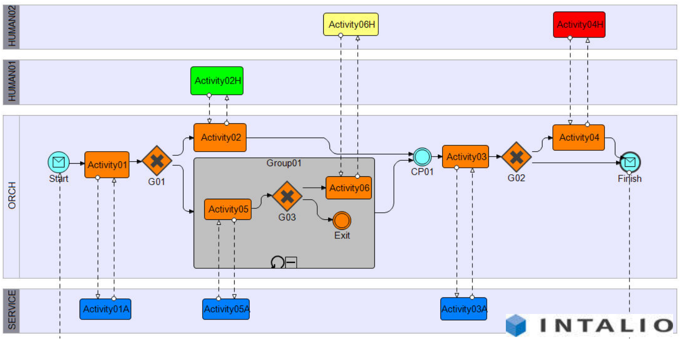
- Process model is an aggregation of
  - events, human and automated activities, roles, objects, rules, audits, etc.
- Versioning is vital

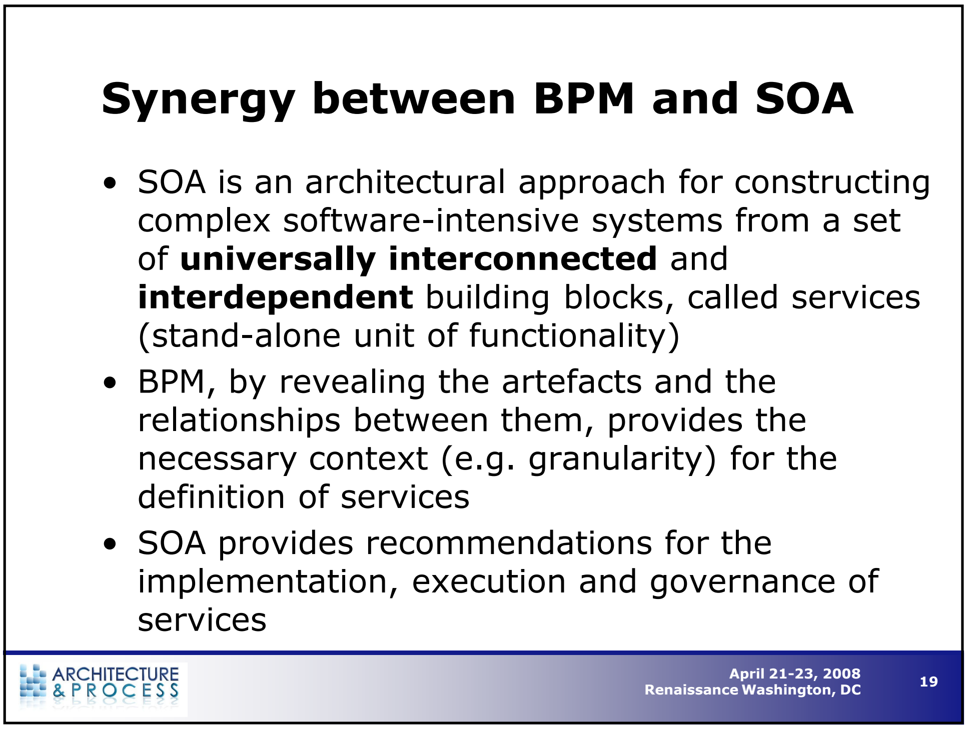


INTALIO

Executable models

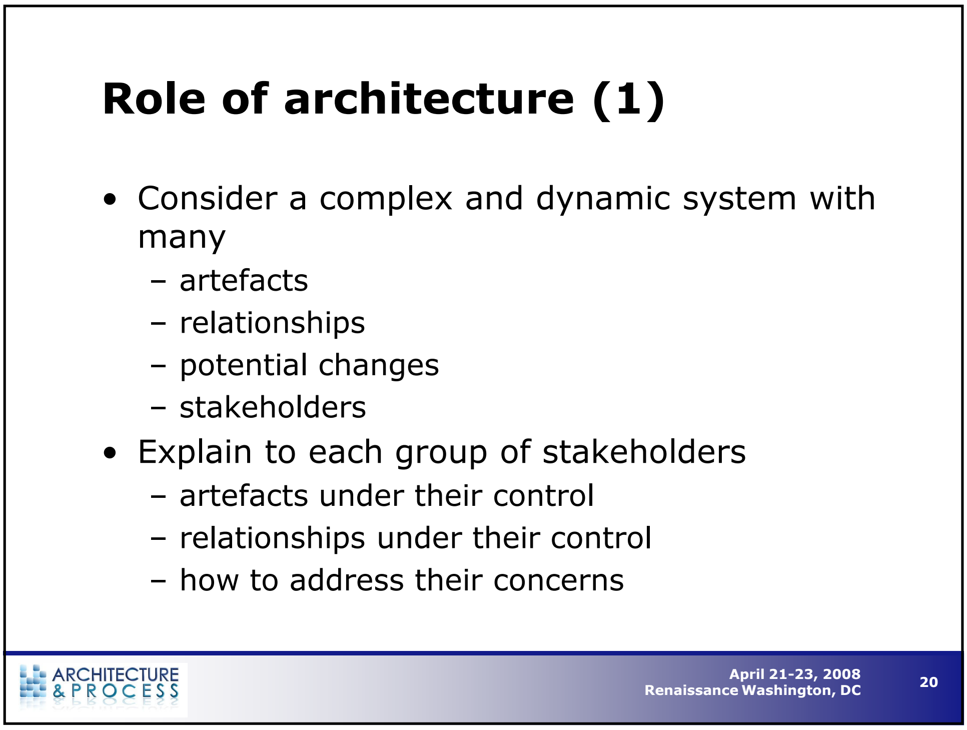
- In an implementation, a model acts as a skeleton or foundation to which we attach services (**what you model is what you run**)





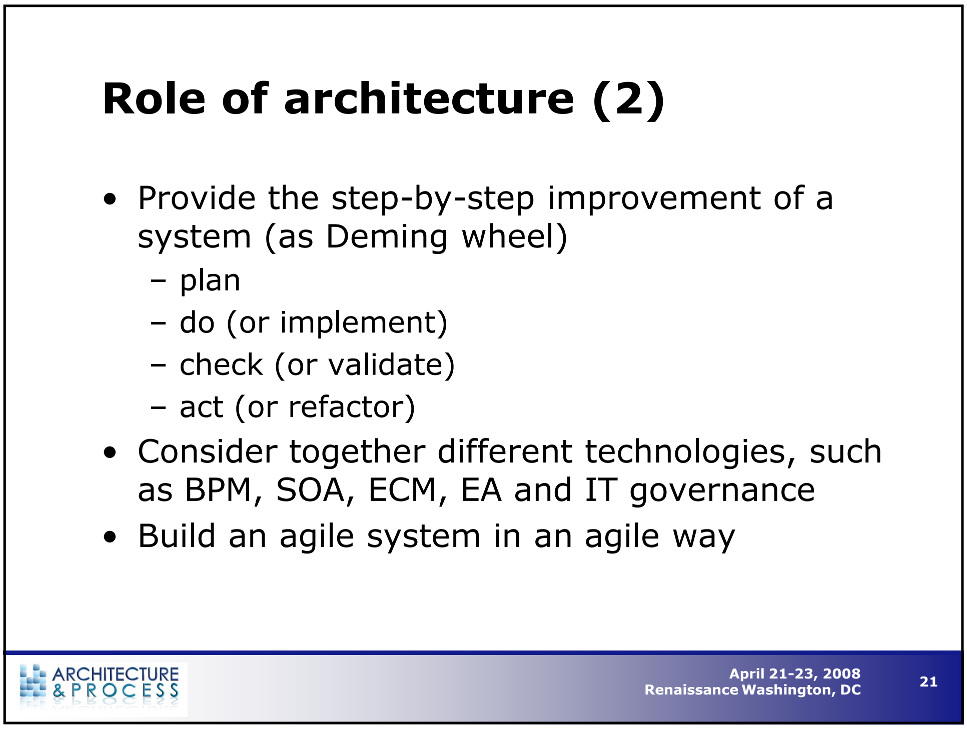
Synergy between BPM and SOA

- SOA is an architectural approach for constructing complex software-intensive systems from a set of **universally interconnected** and **interdependent** building blocks, called services (stand-alone unit of functionality)
- BPM, by revealing the artefacts and the relationships between them, provides the necessary context (e.g. granularity) for the definition of services
- SOA provides recommendations for the implementation, execution and governance of services



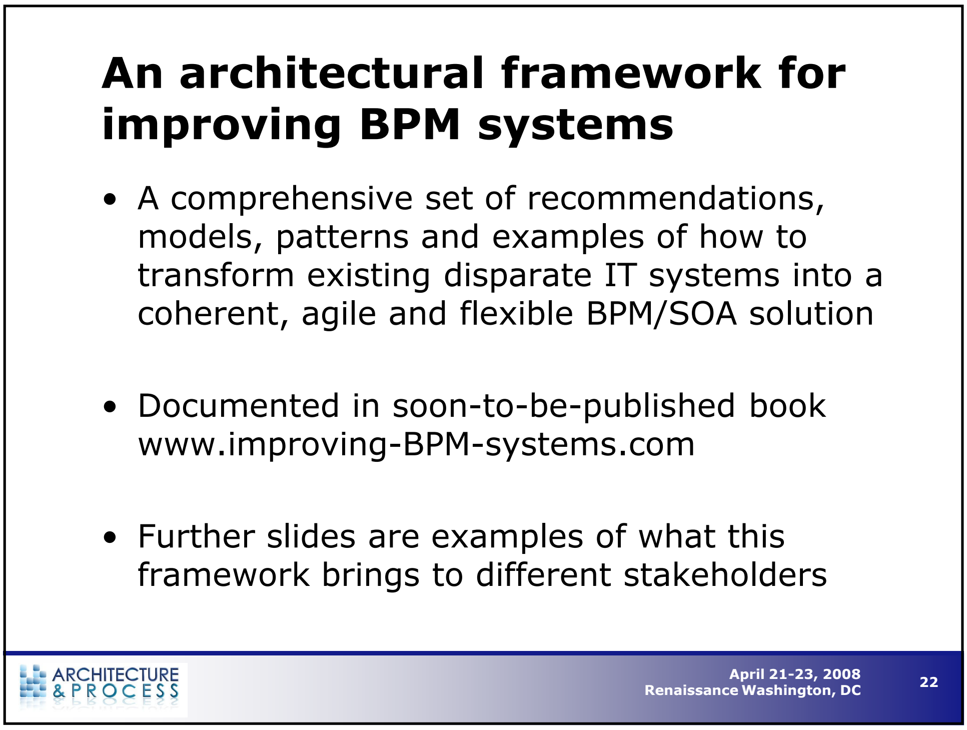
Role of architecture (1)

- Consider a complex and dynamic system with many
  - artefacts
  - relationships
  - potential changes
  - stakeholders
- Explain to each group of stakeholders
  - artefacts under their control
  - relationships under their control
  - how to address their concerns



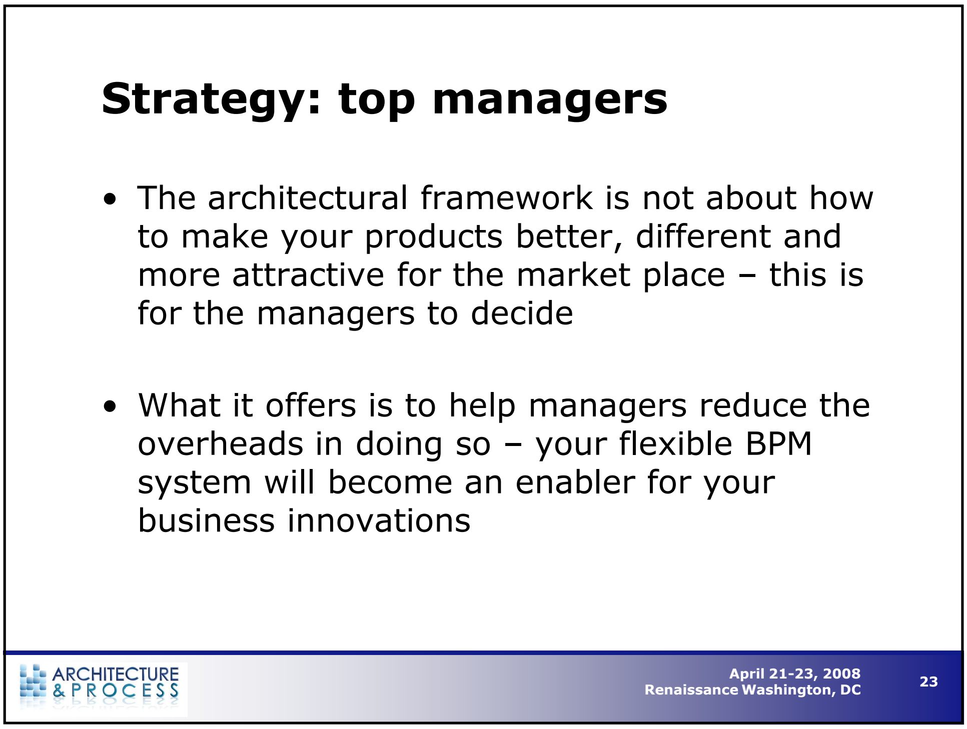
Role of architecture (2)

- Provide the step-by-step improvement of a system (as Deming wheel)
  - plan
  - do (or implement)
  - check (or validate)
  - act (or refactor)
- Consider together different technologies, such as BPM, SOA, ECM, EA and IT governance
- Build an agile system in an agile way



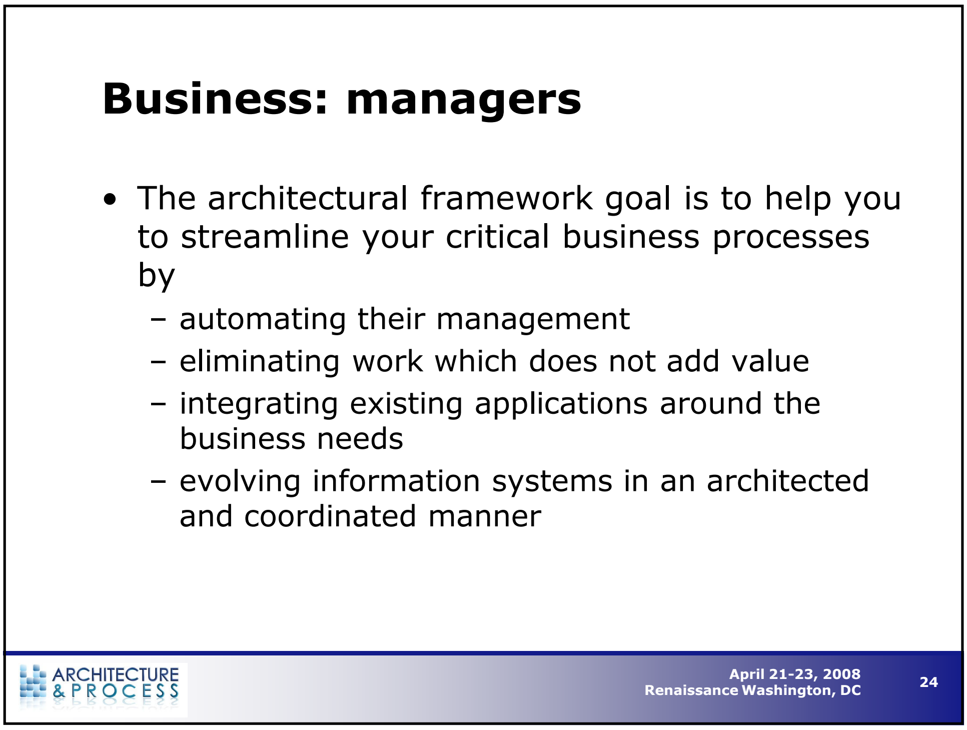
An architectural framework for improving BPM systems

- A comprehensive set of recommendations, models, patterns and examples of how to transform existing disparate IT systems into a coherent, agile and flexible BPM/SOA solution
- Documented in soon-to-be-published book www.improving-BPM-systems.com
- Further slides are examples of what this framework brings to different stakeholders



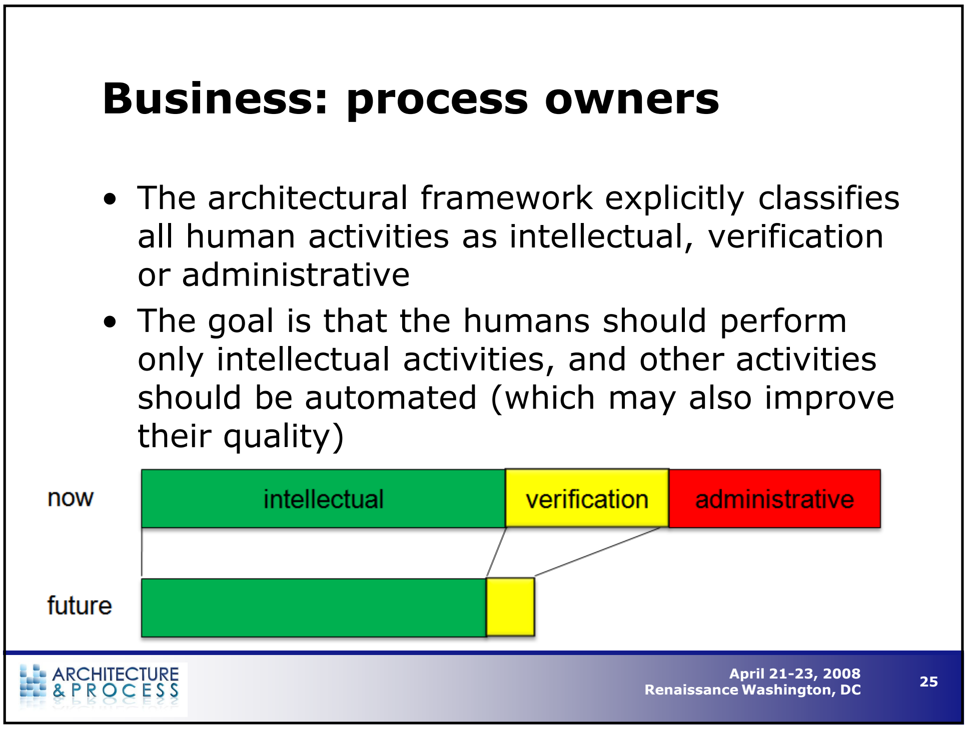
Strategy: top managers

- The architectural framework is not about how to make your products better, different and more attractive for the market place – this is for the managers to decide
- What it offers is to help managers reduce the overheads in doing so – your flexible BPM system will become an enabler for your business innovations



Business: managers

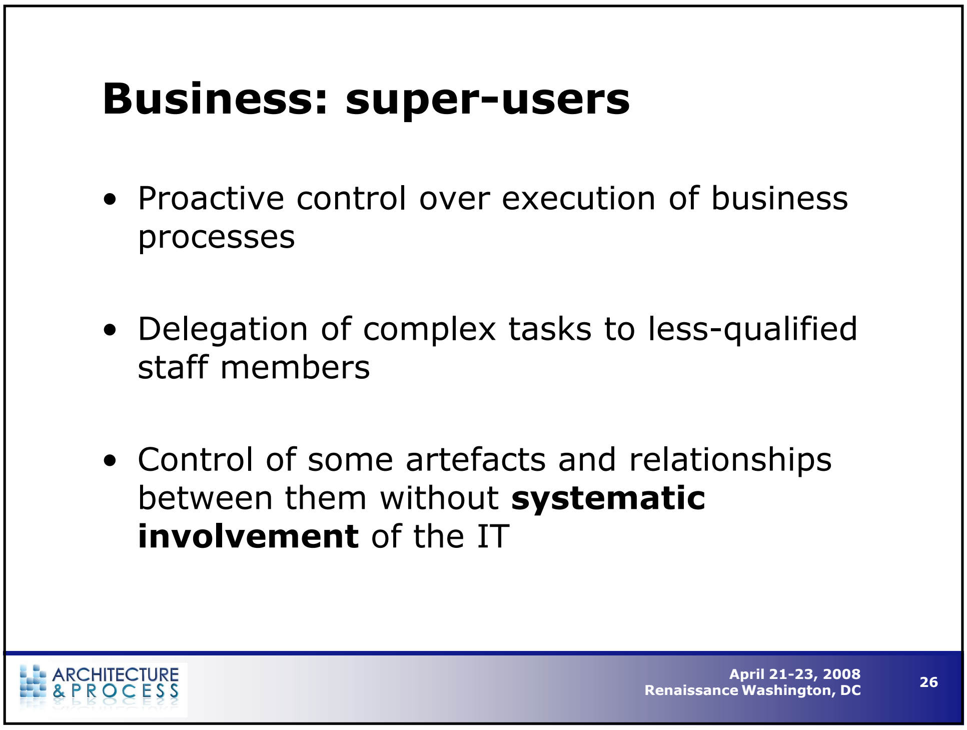
- The architectural framework goal is to help you to streamline your critical business processes by
  - automating their management
  - eliminating work which does not add value
  - integrating existing applications around the business needs
  - evolving information systems in an architected and coordinated manner



Business: process owners

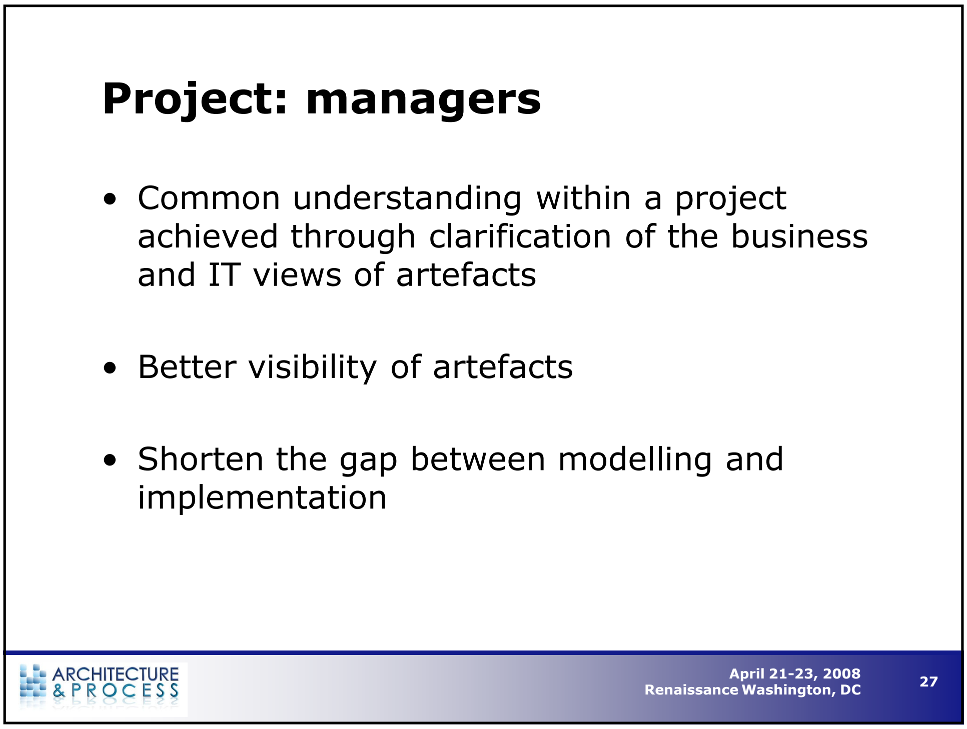
- The architectural framework explicitly classifies all human activities as intellectual, verification or administrative
- The goal is that the humans should perform only intellectual activities, and other activities should be automated (which may also improve their quality)

now
intellectual
verification
administrative
future



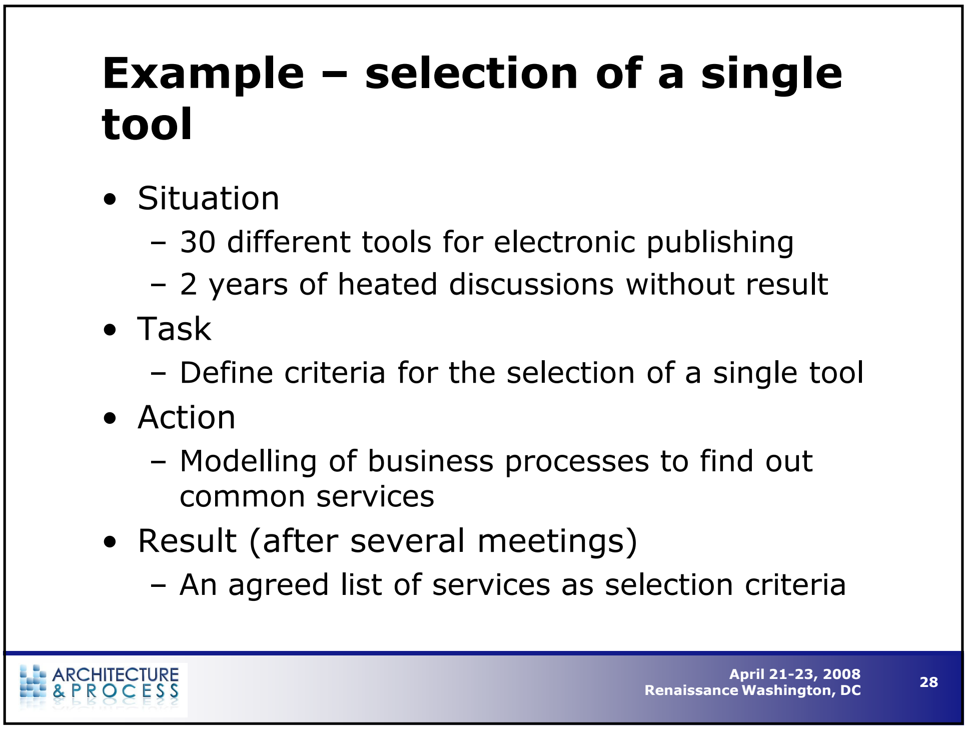
Business: super-users

- Proactive control over execution of business processes
- Delegation of complex tasks to less-qualified staff members
- Control of some artefacts and relationships between them without **systematic involvement** of the IT



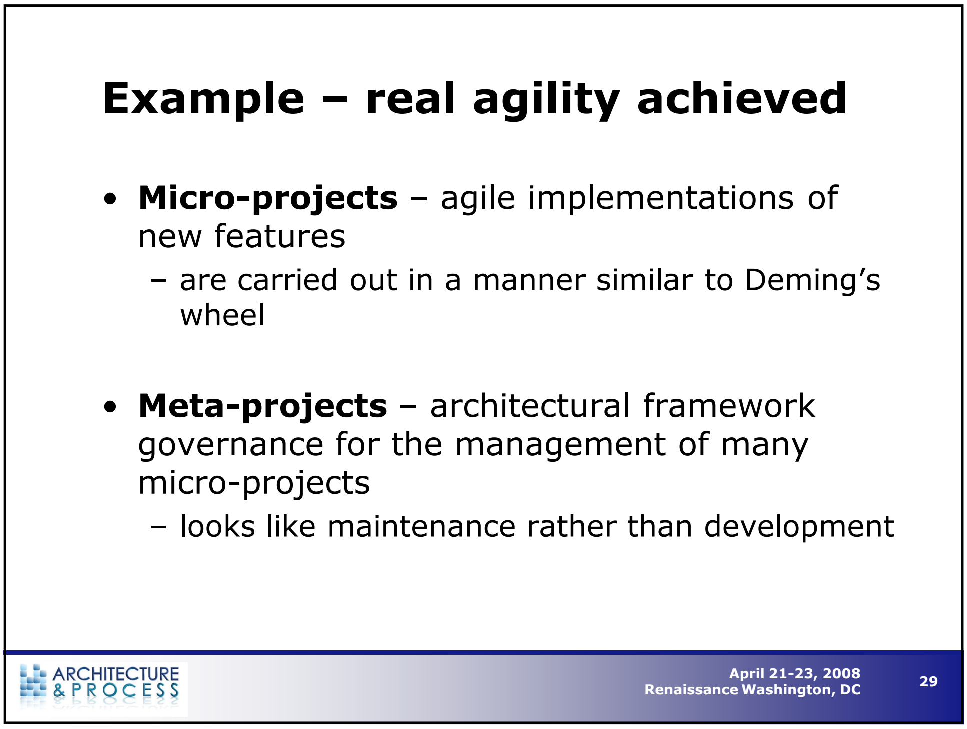
Project: managers

- Common understanding within a project achieved through clarification of the business and IT views of artefacts
- Better visibility of artefacts
- Shorten the gap between modelling and implementation



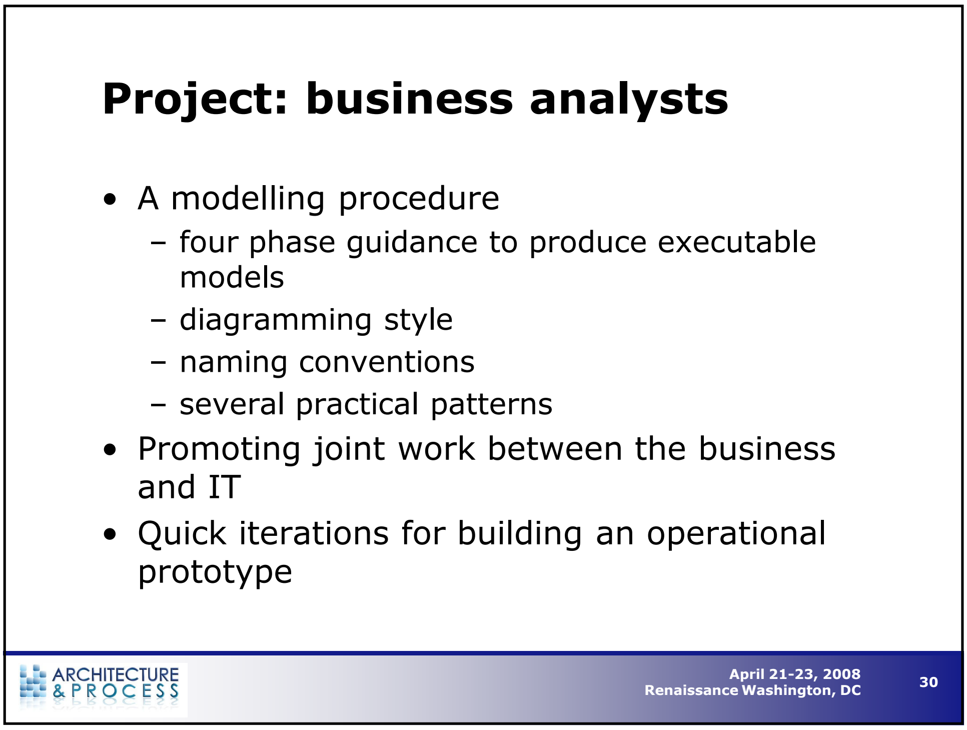
Example – selection of a single tool

- Situation
  - 30 different tools for electronic publishing
  - 2 years of heated discussions without result
- Task
  - Define criteria for the selection of a single tool
- Action
  - Modelling of business processes to find out common services
- Result (after several meetings)
  - An agreed list of services as selection criteria



Example – real agility achieved

- **Micro-projects** – agile implementations of new features
  - are carried out in a manner similar to Deming’s wheel
- **Meta-projects** – architectural framework governance for the management of many micro-projects
  - looks like maintenance rather than development



Project: business analysts

- A modelling procedure
  - four phase guidance to produce executable models
  - diagramming style
  - naming conventions
  - several practical patterns
- Promoting joint work between the business and IT
- Quick iterations for building an operational prototype

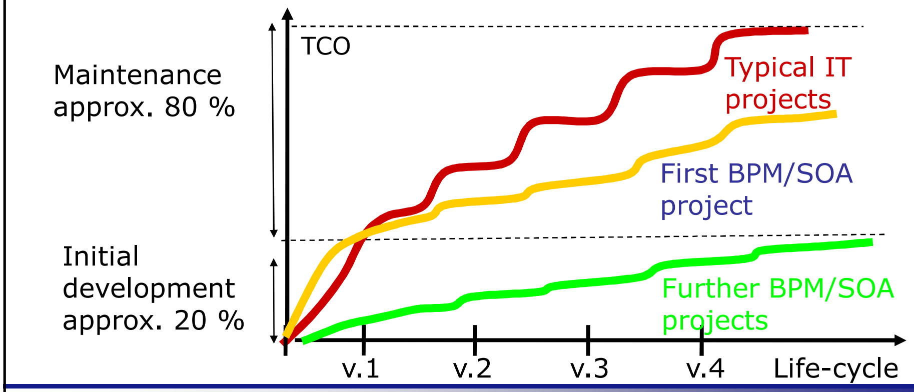
Example – early industrialisation of a business system

- Intensive training for business process modelling
- Use of open source BPM suite for modelling in BPMN
- Tailoring of the modelling procedure for organisational needs
- Common modelling in two major projects
  - new ECM
  - new ERP

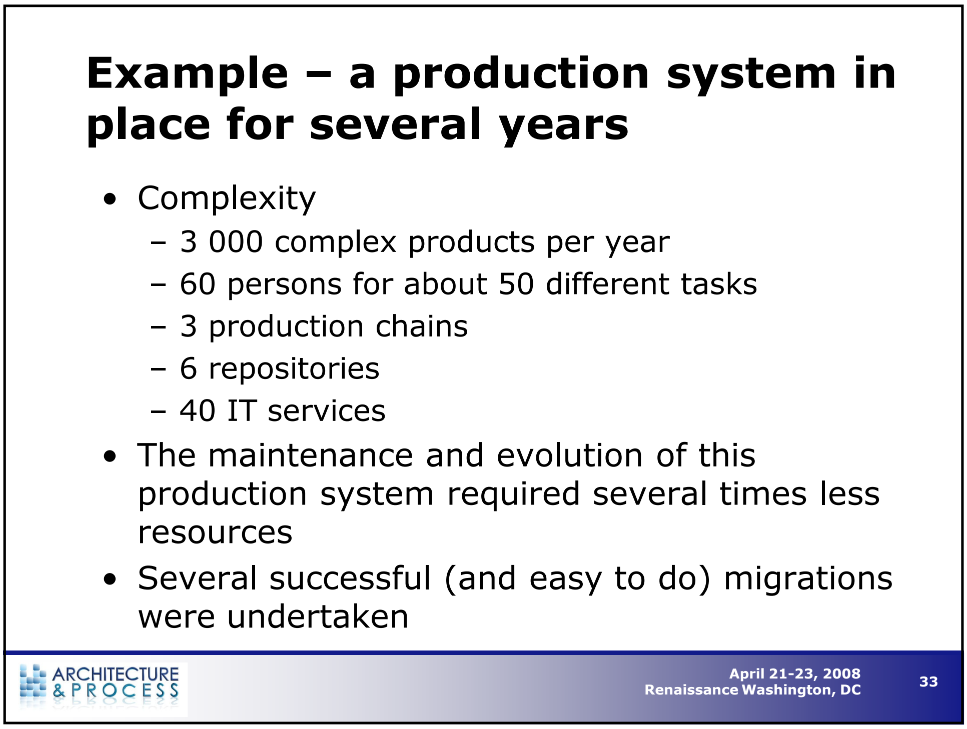
IT: managers

Each subsequent solution is cheaper because it reuses the same tools, the same services, the same architecture

- Considerable reduction of Total Cost of Ownership (TCO)

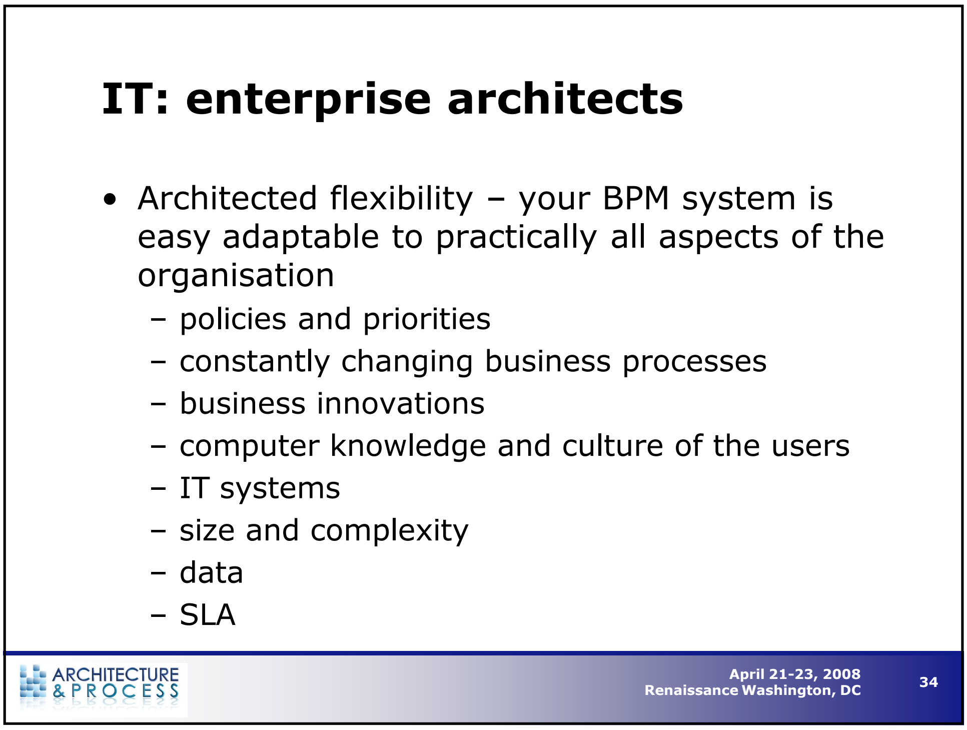


Visit www.architecture08.com for full screen presentation files



Example – a production system in place for several years

- Complexity
  - 3 000 complex products per year
  - 60 persons for about 50 different tasks
  - 3 production chains
  - 6 repositories
  - 40 IT services
- The maintenance and evolution of this production system required several times less resources
- Several successful (and easy to do) migrations were undertaken

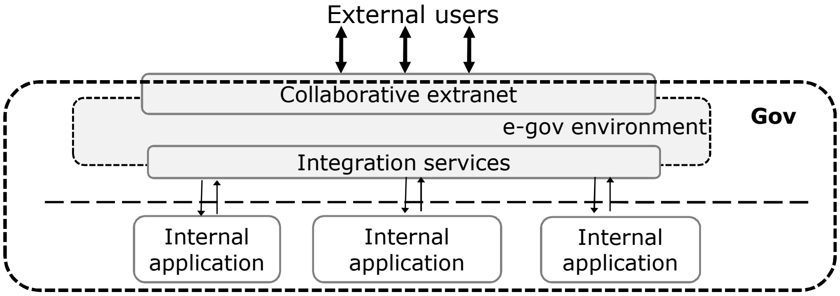


IT: enterprise architects

- Architected flexibility – your BPM system is easy adaptable to practically all aspects of the organisation
  - policies and priorities
  - constantly changing business processes
  - business innovations
  - computer knowledge and culture of the users
  - IT systems
  - size and complexity
  - data
  - SLA

Example – Solution architecture for an e-gov project (1)

- Minimum disruptions for internal applications
- Direct participation of external users in internal business processes
- Maximum traceability (easily certified)



External users

Collaborative extranet

e-gov environment

Gov

Integration services

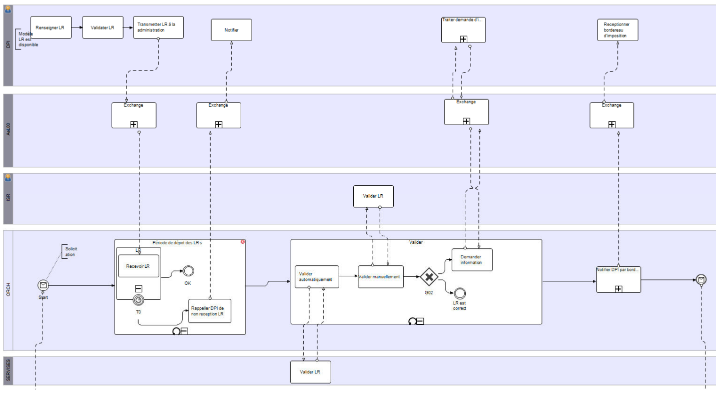
Internal application

Internal application

Internal application

Example – Solution architecture for an e-gov project (2)

- One of the pools (second from the top) serves as an insulation layer

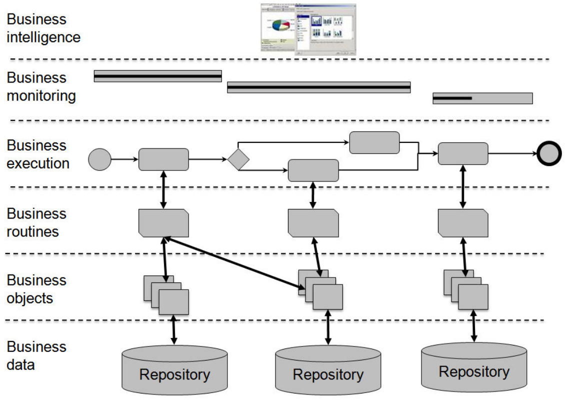


Example – Solution architecture for an e-gov project (3)

- Classification of services
  - **Business-specific**
    to be used in a particular solution
  - **Business-generic**
    to be used in several solutions
  - **Technology-generic**
    protecting business from technological changes
  - **Technology-specific**

IT: architects (1)

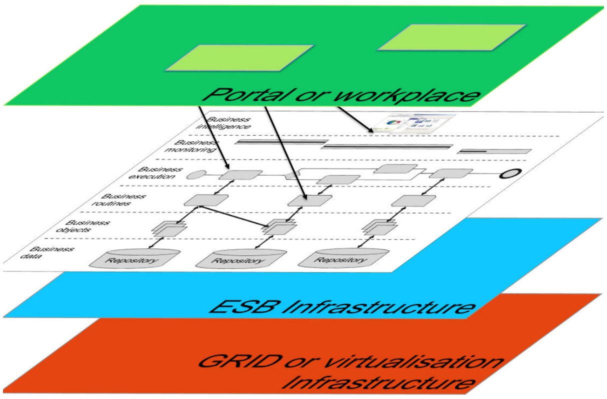
- Relationship between artefacts as implementation layers



Visit www.architecture08.com for full screen presentation files

IT: architects (2)

- Relationship of BPM/SOA with other technologies

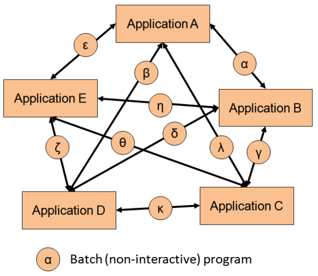


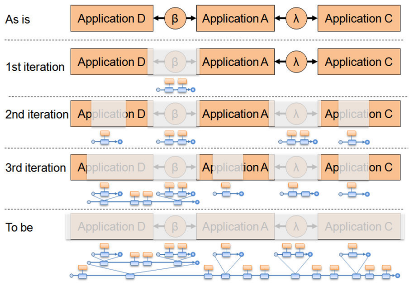
Example – complete redesign of a business system

- The following recommendations were provided
  - principles for building BPM systems
  - typology of BPM artefacts for the understanding and construction of artefacts
  - architecting flexibility of BPM systems, e.g. rules for versioning, conventions for WSDL and XSD, etc.
  - design consideration for implemention of artefacts

IT: developers

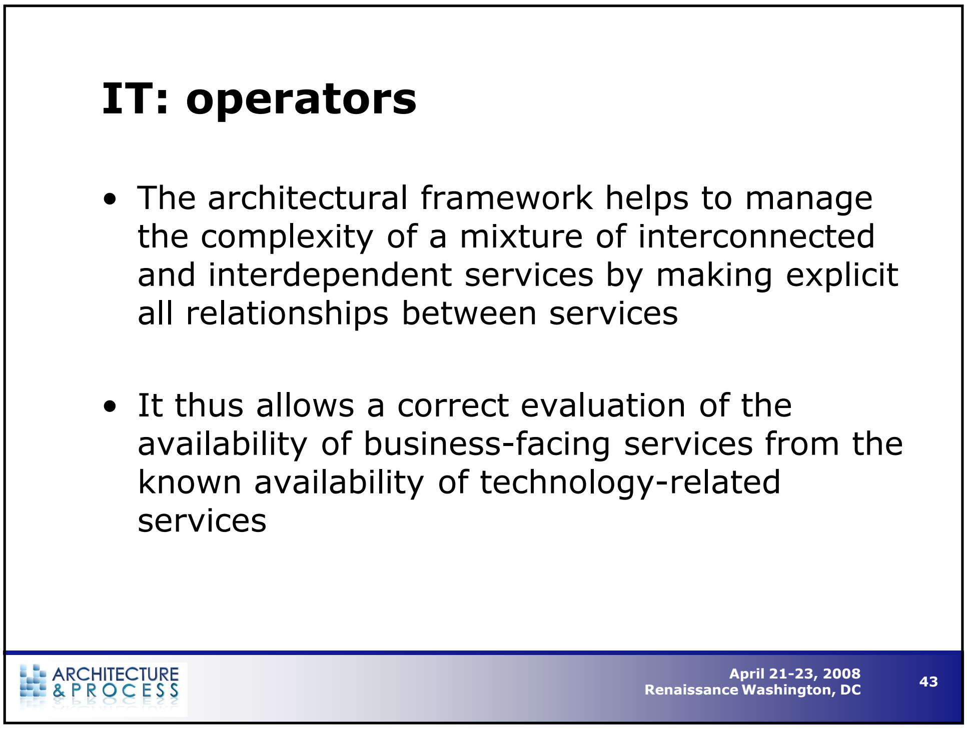
- Incremental transformation from typical inter-application data flows to end-to-end coordination of services





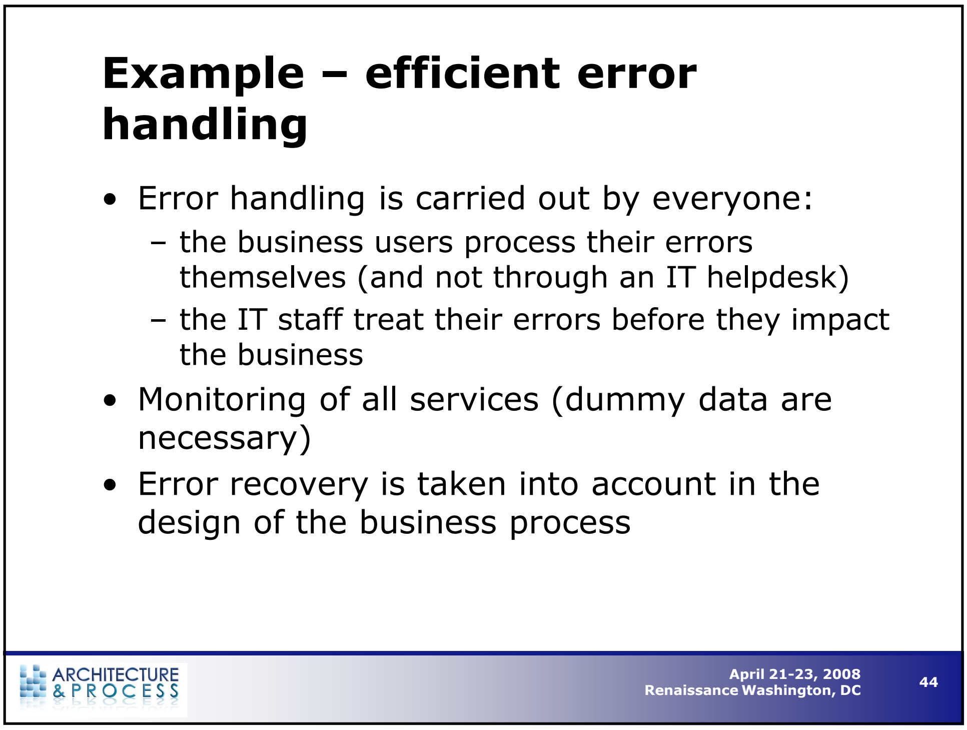
Example – typical timing of micro-projects

- The architectural framework provides the big picture which is
  - represented graphically (and therefore easily understood), agreed internally by consensus, addressing BPM and not "parachuted in" by consultants or a vendor
- Many projects become very agile
  - Definition phase: 1 hour
  - Specification / conception phases: a few hours
  - Development / test / validation phases: a few hours / days (depending on the user's availability)
  - Production phase: practically instant



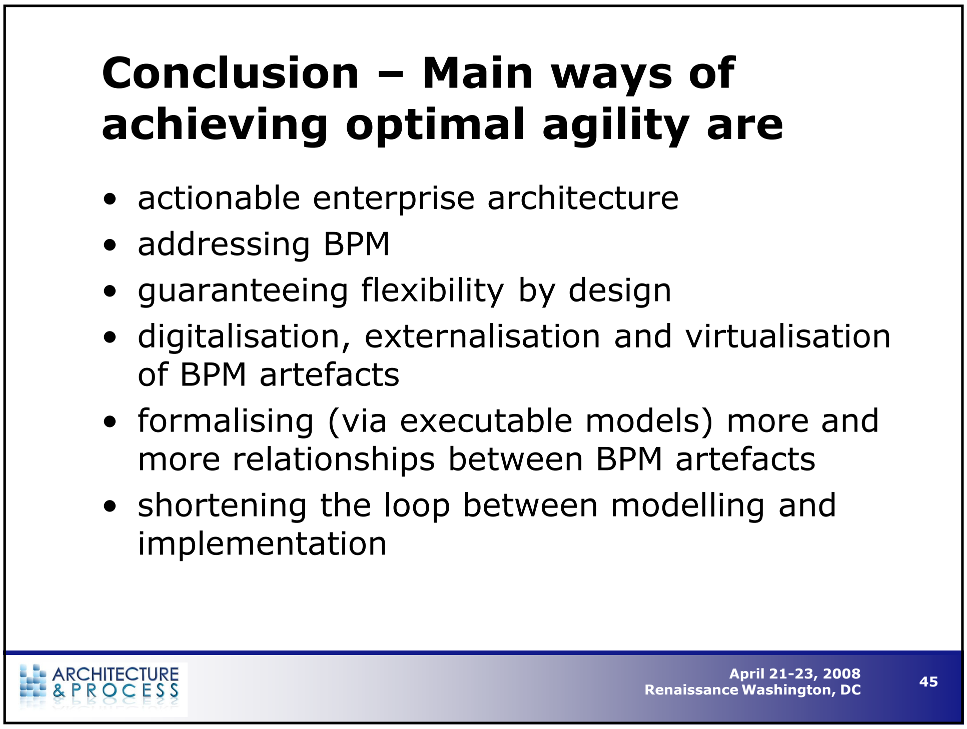
IT: operators

- The architectural framework helps to manage the complexity of a mixture of interconnected and interdependent services by making explicit all relationships between services
- It thus allows a correct evaluation of the availability of business-facing services from the known availability of technology-related services



Example – efficient error handling

- Error handling is carried out by everyone:
  - the business users process their errors themselves (and not through an IT helpdesk)
  - the IT staff treat their errors before they impact the business
- Monitoring of all services (dummy data are necessary)
- Error recovery is taken into account in the design of the business process



Conclusion – Main ways of achieving optimal agility are

- actionable enterprise architecture
- addressing BPM
- guaranteeing flexibility by design
- digitalisation, externalisation and virtualisation of BPM artefacts
- formalising (via executable models) more and more relationships between BPM artefacts
- shortening the loop between modelling and implementation

Thank you!

Dr Alexander Samarin
Enterprise Solutions Architect
www.samarin.biz

Contact information:
+41 76 573 40 61
samarin@bluemail.ch