Bernus, P. (2008) (Half-day tutorial) SOA for the Whole of Business. IQPC (International Quality and Productivity Center) : Service Oriented Architecture. Sydney 18-21 February 2008

How to Implement SOA for the Whole of Business

Associate Professor Peter Bernus
Griffith University
Enterprise Integration Group
School of ICT
P.Bernus@bigpond.com

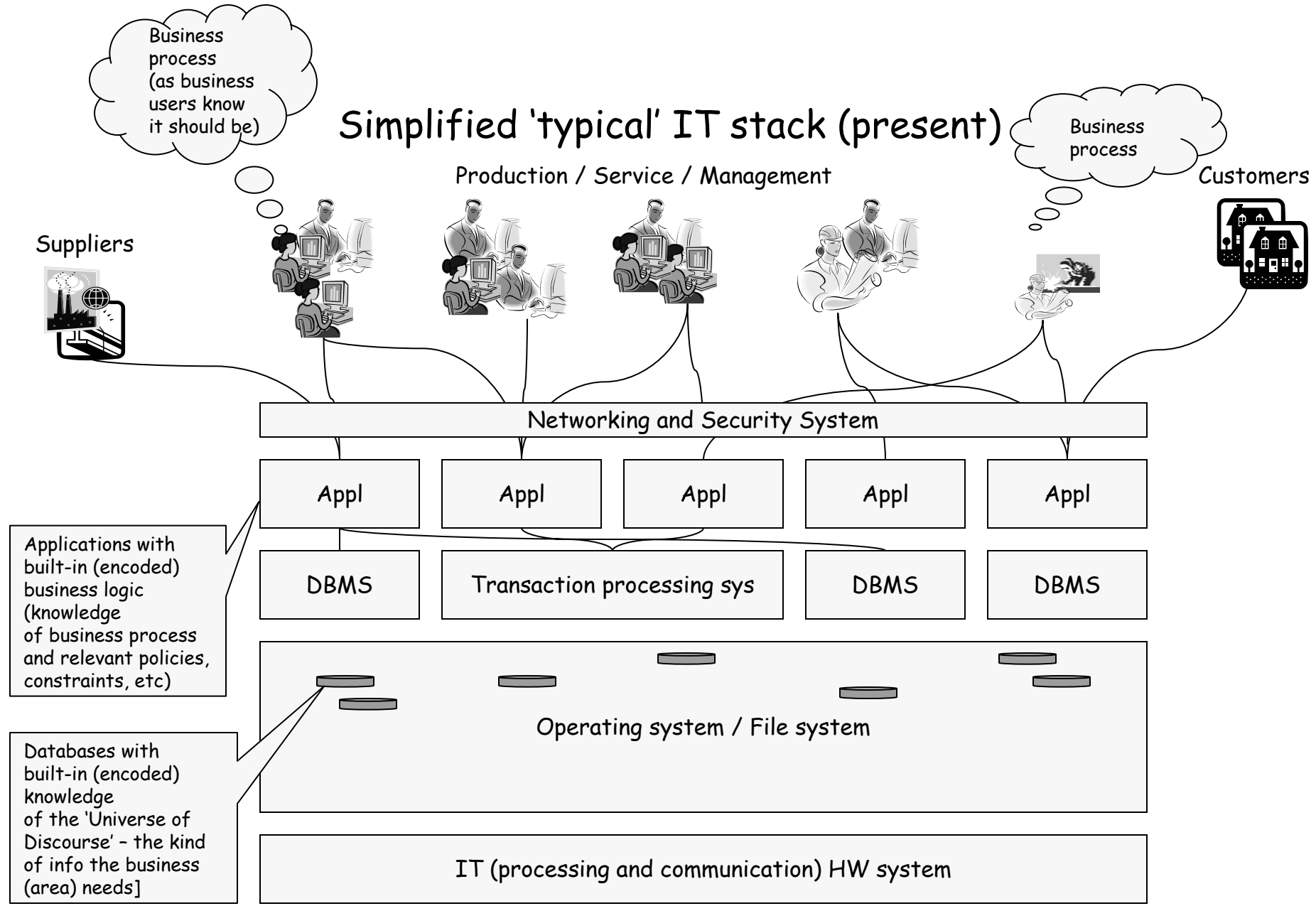
Overview

1. SOA as a strategic business transformation initiative
2. How to build the business case and set a realistic transformation strategy
3. The scope of SOA and project governance
4. The scope of the technical development
5. External and internal participants in the transformation program
6. Prioritising and staging the development effort

1. SOA as a strategic business transformation initiative

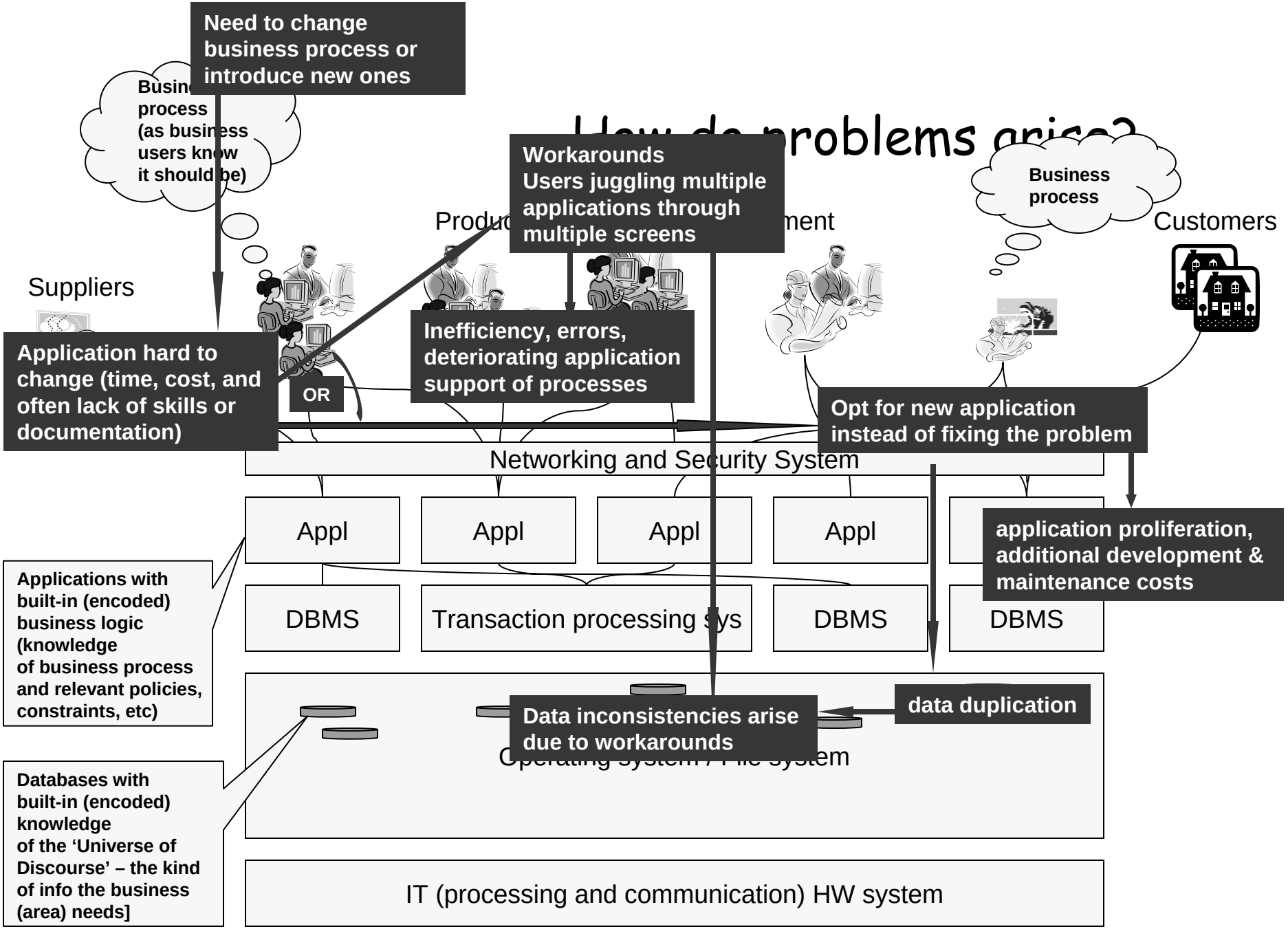
- Agility is a business imperative
- A Business can be organised as a set of loosely coupled service entities (SEs) and Business Entities (BEs) where BEs perform value-producing business processes using the services of SEs
- SEs can be internal or external to the enterprise
- Each Service Entity is like a 'mini enterprise' with its own mission and management
- This view is in broader than the present view of SOA as a way to maintain a lean set of Application programs and Databases, de-coupled from business process management (design and execution)

- Services provide reusable *functions* implemented as
  - Existing or new service-enabled applications *from within* the enterprise or *from outside* providers
  - Legacy applications with some (or all) of their functions transformed into services through new interfaces
  - Information Stores / Sources
    - Databases or data warehouses extended with a service interface to provide answers to queries, updates to existing information, or insertion of new facts – whether these databases store management- or product(ion)-related information
    - Other information repositories (including free text and multimedia), such as Geographic Information Systems (GIS), Process and policy repositories, metadata repositories, etc.

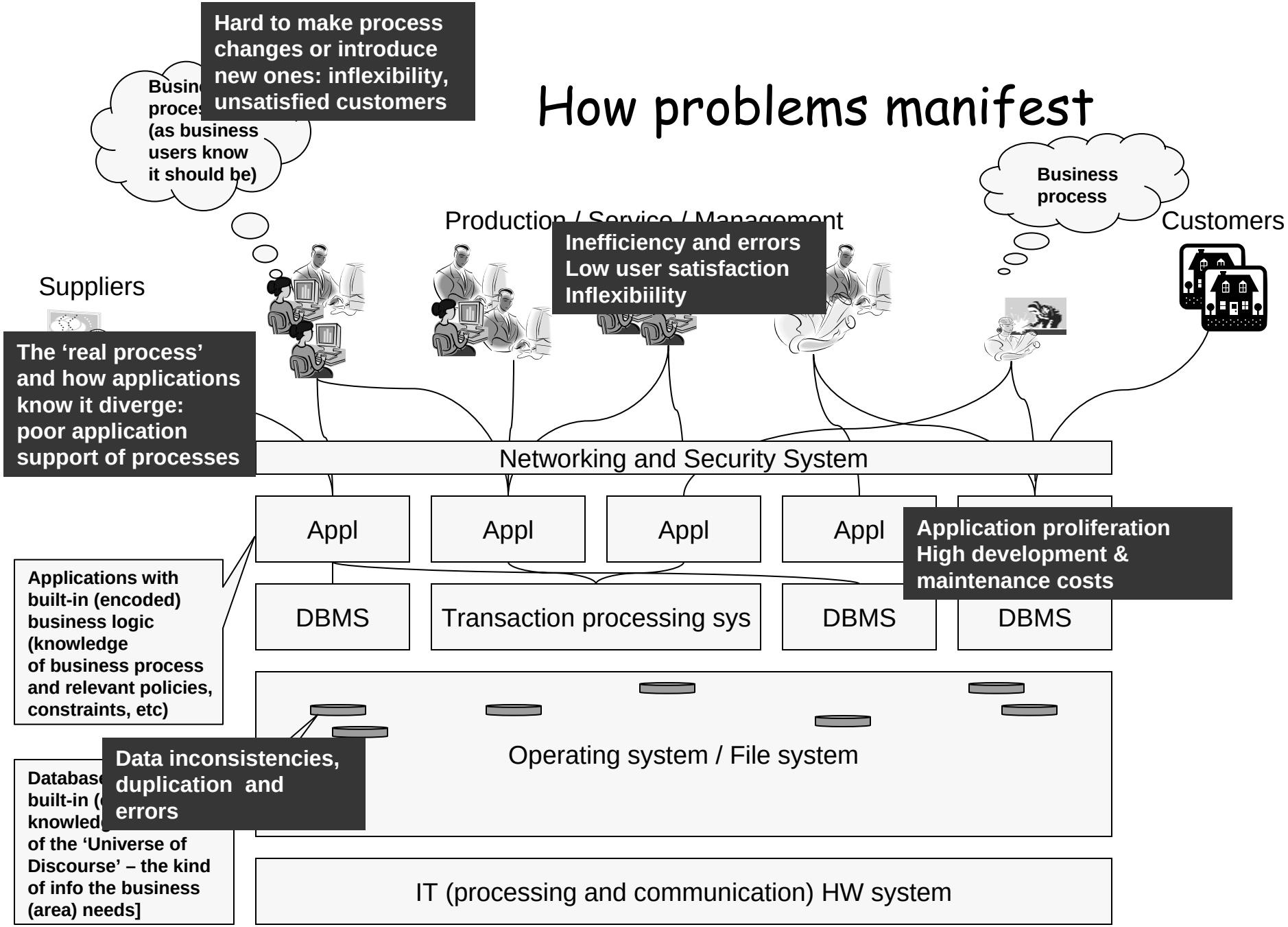


Characteristics of this typical architecture

- Business Processes, Policies and Constraints *change frequently* (data definitions do not change as fast, *if* specified by a good analyst)
- Application programs are *hard to change* (change is slow and costly)
- As a result, companies that created new processes rapidly, end up with
  - Employees using *multiple screens* (*multiple application programs*) copying data between these screens (introducing *errors* and *inefficiency*), or
  - *Poor application support of business processes:* the real business process is no longer reflected by the application program (leading to *workarounds*),
  - *Databases being used in 'unconventional ways'* (such as data stored in places where they do not belong) – leading to *data inconsistencies* or *duplication* and the *lack of a 'single point of truth'*
- Application programs do not support the business *as* they should
- Pressing needs have created *even more new applications* and even more of the above problems – at *additional recurring costs*



How problems manifest



2. Build the business case and set a realistic transformation strategy

- Strategic needs to dictate the change
  - Process flexibility, efficiency improvement
  - Reuse of IT resources (applications and data)
  - Lower cost of operation and maintenance

- Enterprise Architecture practice has the capacity to provide answers to such strategic transformation

- The Architecture of the enterprise (i.e., how the structure of the enterprise implements its business functions / processes) includes

  - Process Architecture
  - Information Architecture
  - Technology Architecture – HW+ SW (Applications & Databases)
  - Organisation ('Human Architecture')

So how does EA help us achive this transformation?

- It is impractical to try and change the *entire* business (and it is also not necessary)
- EA helps *identify* and *structure* the activities that must be done to implement a strategic initiative, such as adopting SOA on the business level, or on the technical level
- EA helps maintain a close link between the strategic objective and the eventual implementation of the change

to explain some basic concepts…

An Architecture Framework (AF) organises / describes everything involved in architecture practice

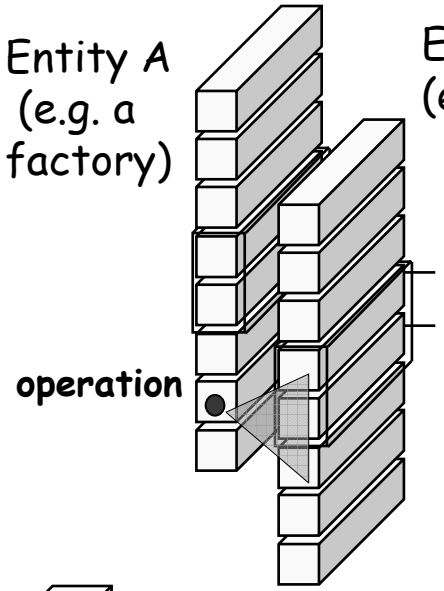
Here are the basic concepts [*cf* ISO 15704:2000] that any AF should cover (even though their respective terminology may be different)

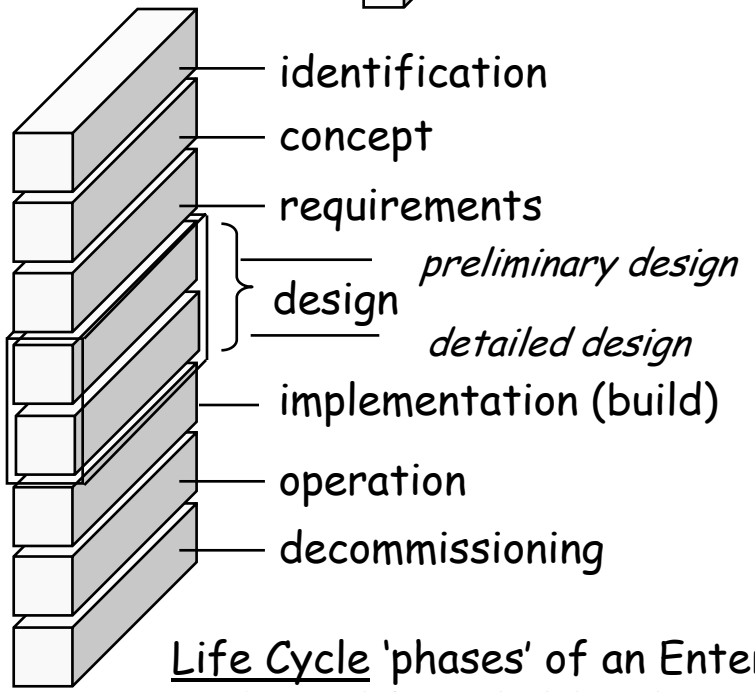
- Enterprise Entity
- Life-cycle, life-cycle relations
- Life history
- Modelling Framework
- (Particular) Enterprise Models
- Reference Models ('partial enterprise models')
- Enterprise Modules (components)
- Enterprise Engineering Methodology(ies)
- Enterprise Modelling / Enterprise Engineering Tools
- Enterprise Modelling Languages (and their semantic definitions: generic EM concepts)

GERA

GERAM Framework

The Life Cycles of Enterprise Entities are related

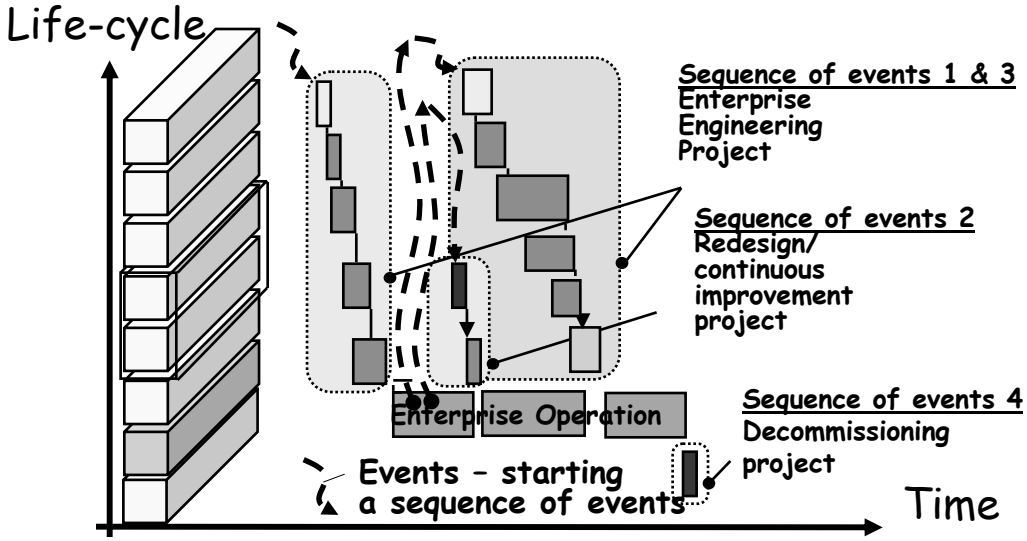




Life Cycle 'phases' of an Enterprise Entity

(note: life cycle 'phase' is *not* a temporal concept)

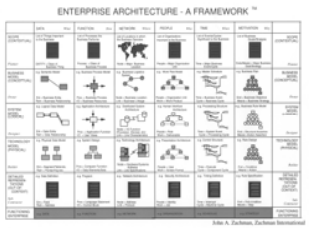
The Life History of an entity consists of multiple, potentially parallel, sequences of events during the life of the entity



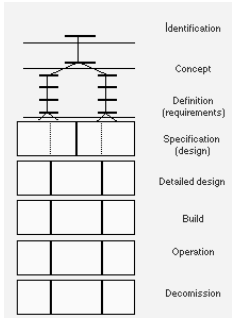
Life-cycle activities overlap (within / between sequences of events). Each sequence of events is subdivided into stages separated by milestones (stage gates)

DETOUR

Zachman MFK

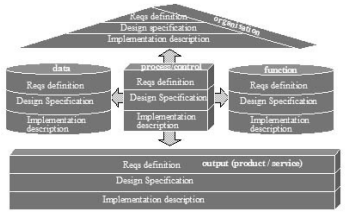


PERA MFK

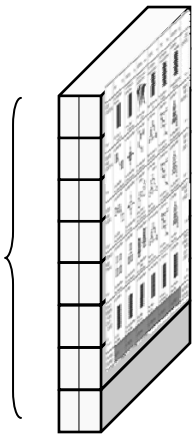


FEAF, TOGAF, etc
Modelling frameworks

ARIS MFK



ISO 15704:2000
GERAM
EA Framework
includes the
GERA MFK
(also see EN / ISO 19349)



C4ISR/DoDAF MFK

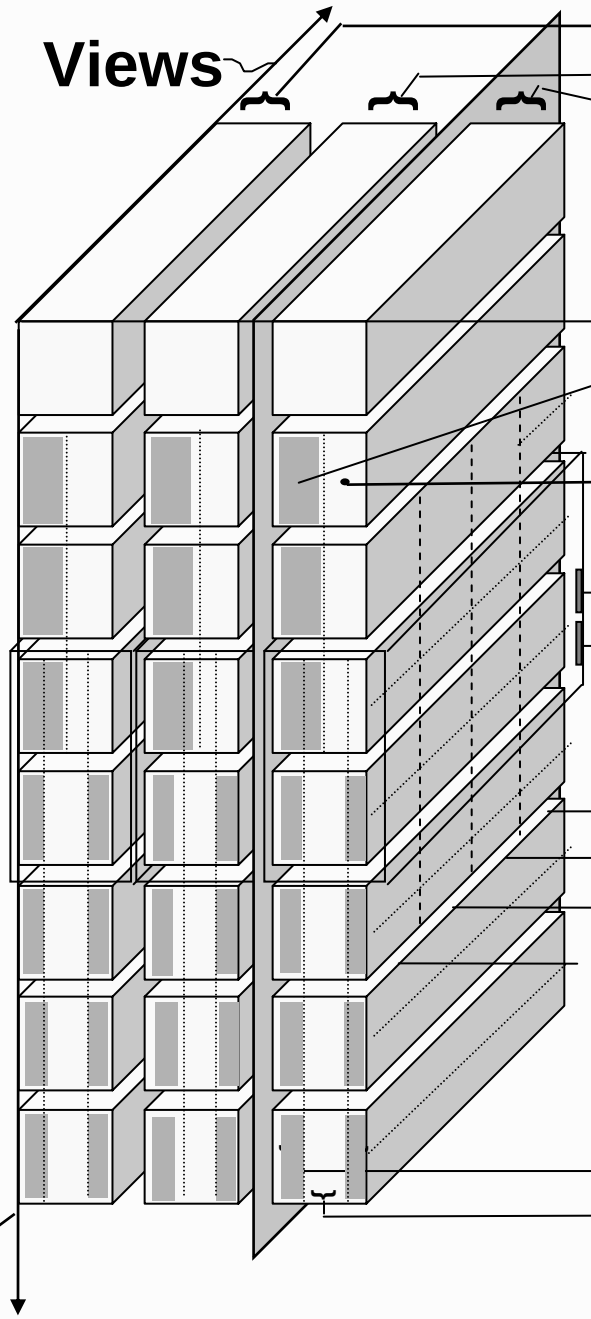
Modelling Frameworks (MFK) provide a means to organise the models / descriptions used throughout the life-cycle of the system in question

As long as ISO15704:2000 requirements are satisfied, these MFKs, or in-house combinations of these, are suitable

Most modelling frameworks tell us about the needed types of models / descriptions, but give different levels of advice about the scope

- ISO 15704:2000 definies scope requirements in addition to defining the types of models needed
- These requirements can be satisfied in conjunction with many known modelling frameworks

GERA modelling framework



Views

Generic
Partial
Particular

Subdivision according to genericity

Instantiation

Customer service
Management- and control

Subdivision according to purpose of activity

Software
Hardware

Subdivision according to physical manifestation

Resource
Organisation
Information
Function

Subdivision according to modelling views

Machine
Human

Subdivision according to means of implementation

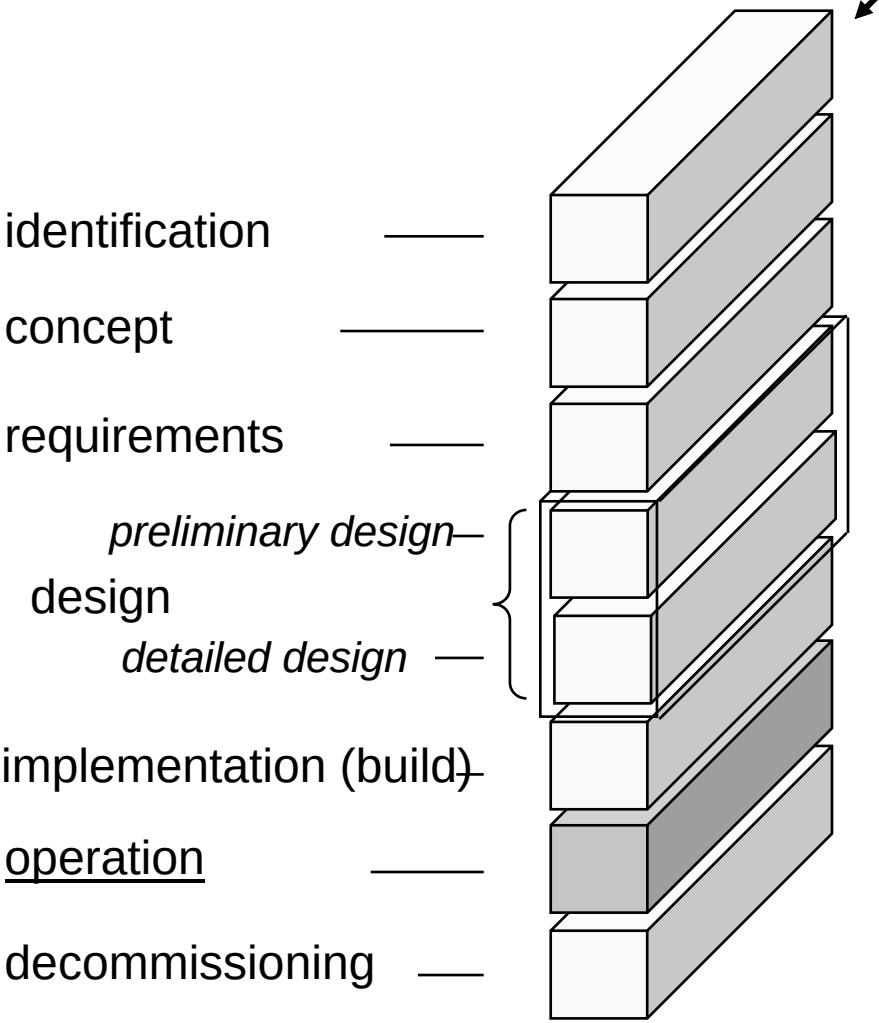
Life-cycle phases

END DETOUR

What is the Methodology that is *relevant* to the strategic objectives at hand?

Start with understanding your 'Business Model'

Suppose this is the 'entire' enterprise (and its life-cycle …)



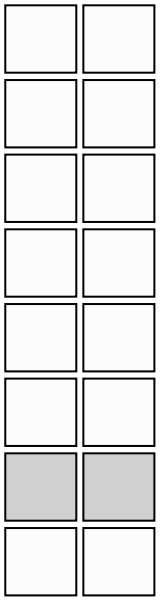
How do we determine all the steps of change needed to achive a strategic objective (which in turn is based on a vision)?

We want a Master Plan and an Implementation Plan

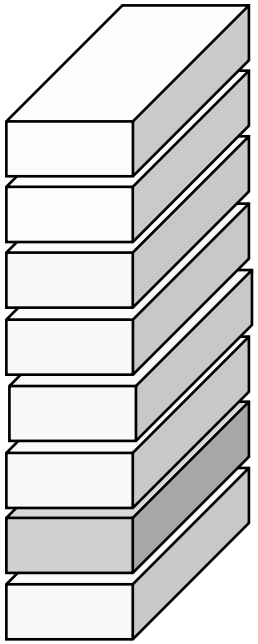
… which is feasible and
… to which the stakeholders are committed and understand

To change *everything* (Big Bang) is not feasible: we decompose the enterprise into its constituents and see *what needs to change* ?

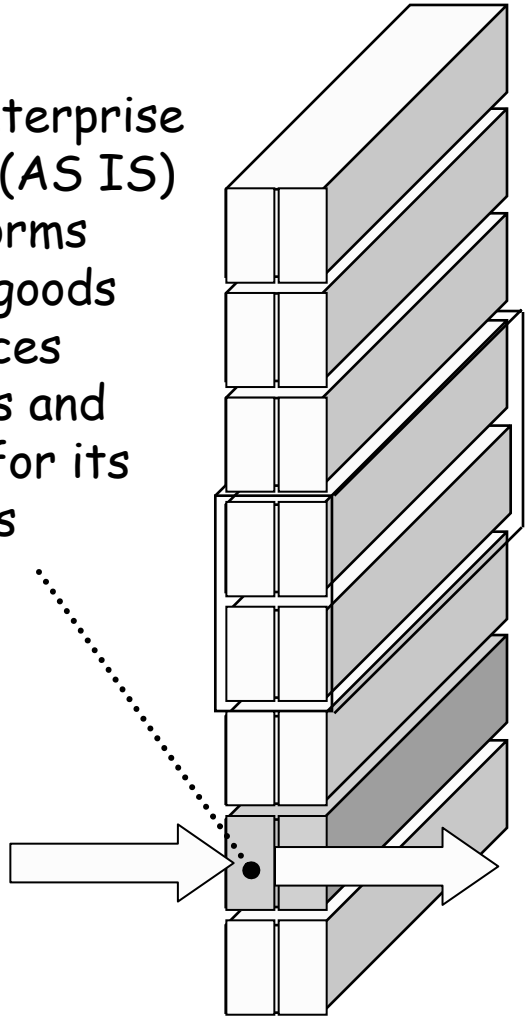
Legend for next pages



stands for



As the enterprise operates (AS IS) it transforms incoming goods and services into goods and services for its customers

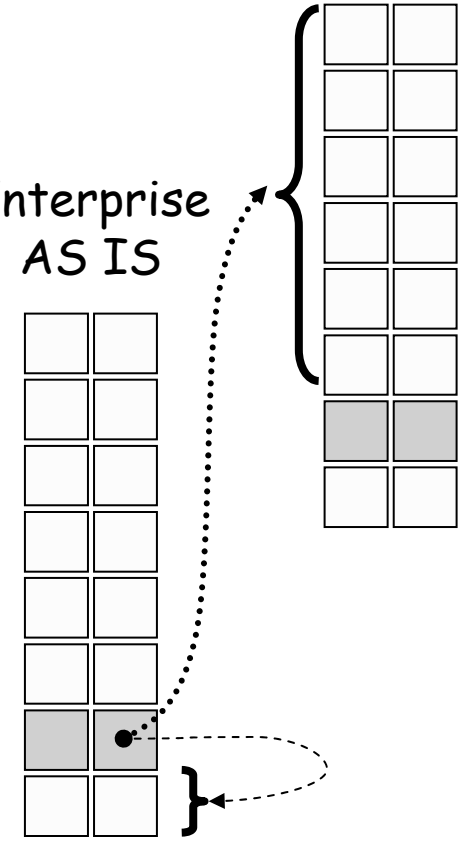


Who performs all these other activities?

identification
concept
requirements
preliminary design
detailed design
implementation
decommissioning

Enterprise TO BE

Enterprise AS IS



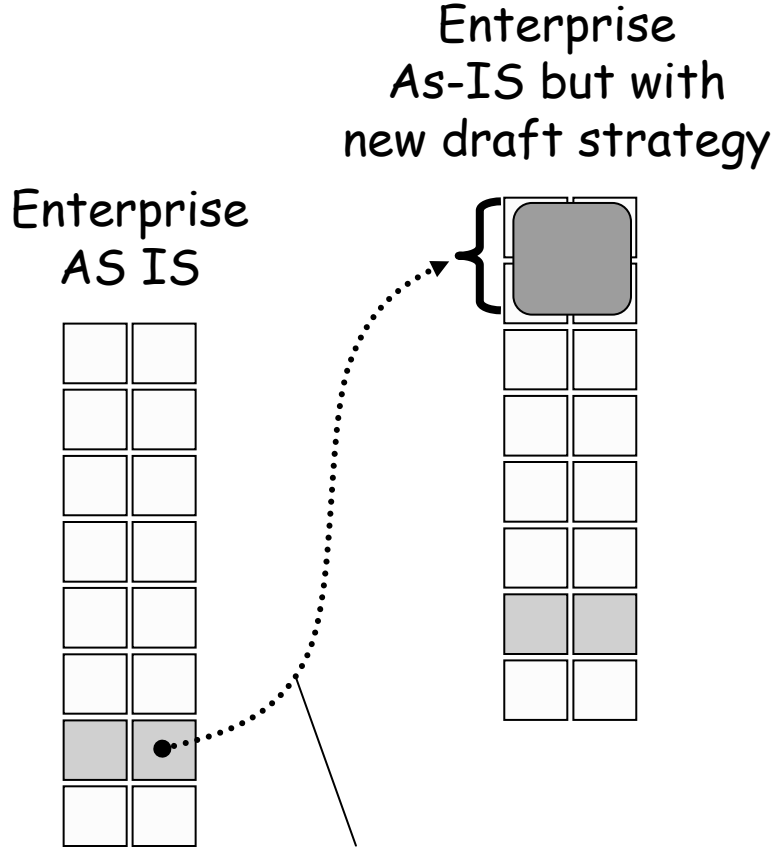
One task of management is to ensure that all of the above are covered by mgmt themselves, or by internal / external support and by establishing appropriate roles and responsibilities (individuals, committees)

- Before decomposing the enterprise into its constituents we need to remain just for a while on the whole-of-enterprise level

- The strategy is enterprise-wide and the change requirements need to be developed in light of the complete business

- We shall use a special type of 'Business Model diagram' to describe the scope and location of change

Developing the 'Business Model' for the future



s1. Identify the TO BE business with its mission, involved entities and their strategic relationships (take into account the extended enterprise, in-house and *relevant* external infrastucture)
s2. Develop the business Concept (vision, strategic objectives and major policies and principles and critical success factors that should guide, and argue the *why*, of the tranformation)

Leadership is essential: gain wide stakeholder support, consultation / common understanding

s1+s2 is strategy making (usual strategy making activities are involved)

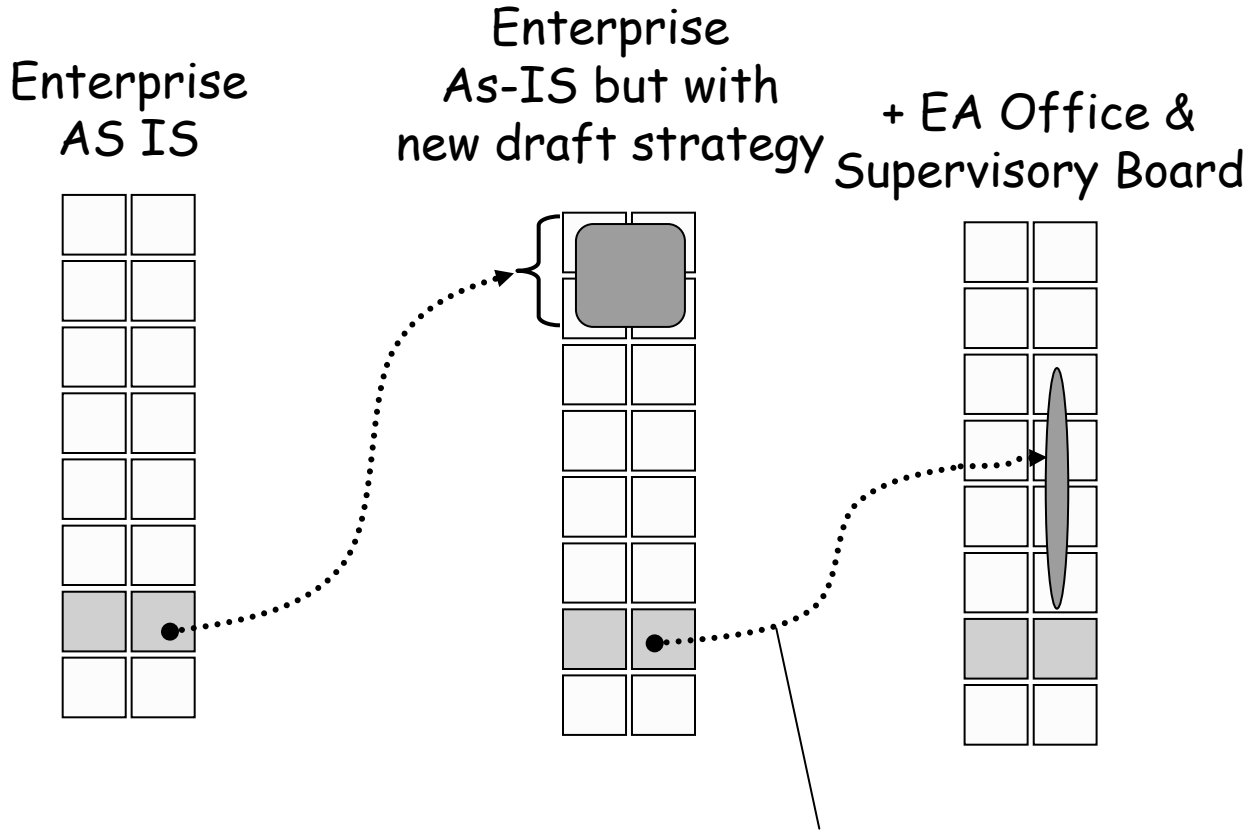
The arguments should be supported by an analysis of the impact of change

Need champion and sponsor of change
(NB the *origin* (idea) of the strategy may be at various levels of the organisation and is sometimes informally communicated – i.e. not only official channels)

At this stage …

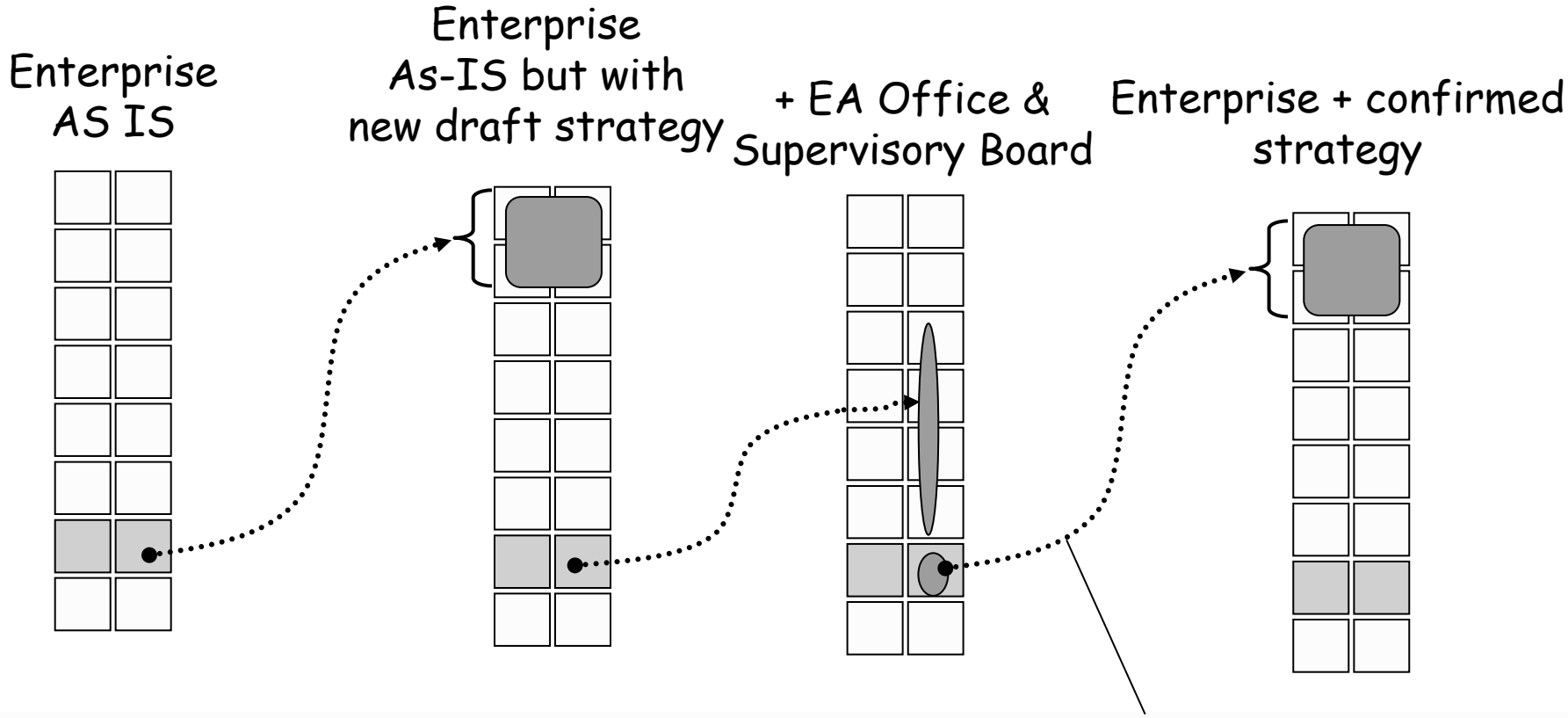
- There is a champion, a sponsor
- 'Pre-feasibility study' has been informally conducted and basic strategic analysis has confirmed strategy
- Stakeholder consultation has taken place and support has been gained
- At this point there is no formalised organisational host yet for the change (only the champion and the sponsor)
- Now the sponsor can help establish a formal organisational host – could be at Heqadquarters level or be part of a 'lead constituent' (division, department that has established influence within the organisation and is trusted by the rest of the organisation, usually the champion is from this element)

Example: EA Office + Supervisory Board



Establish a formal organisational host (We implement the relevant part of the life-cycle of this host (EA Office + Supervisory Board), i.e.

- Define their tasks (mandate), policies, responsibilities, decision making processes / procedures / authorities,
- Determine Personnel roles (including skills / knowledge requirements)
- Determine / Select IT and logistic support needs (office, admin, tools, budget)
- Determine training needs
- Train, assign personnell, deploy tools, establish accounts, assign admin personnel)



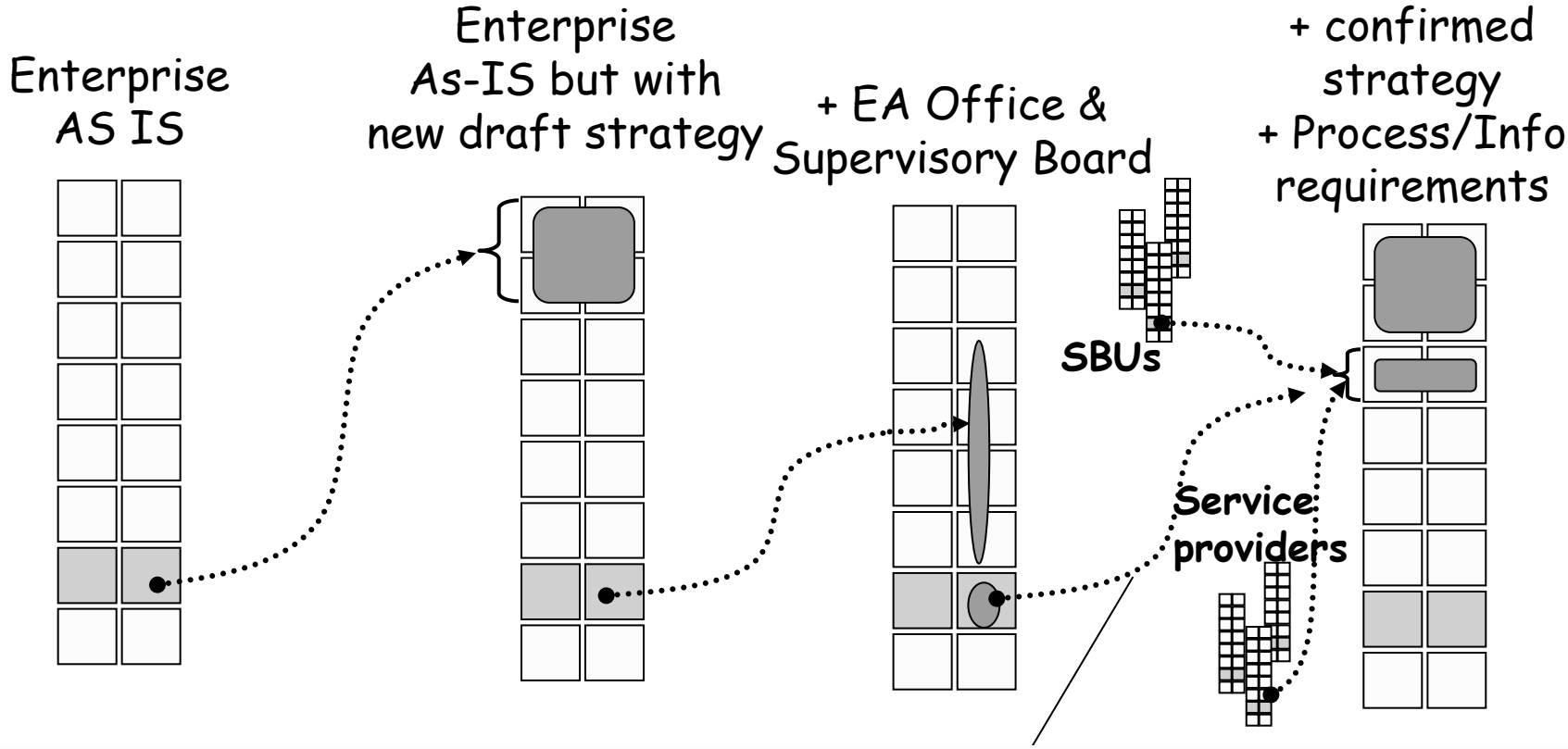
EA Office and Supervisory Board now start operating

Main task at this stage is to confirm the strategy

Conduct feasibility analysis and any other strategic analyses deemed necessary

May need to perform *some* AS-IS analysis as part of this activity
(e.g. maturity, market, technology, etc)

As a result of this, the confirmed / refined strategy has stakeholder support and understanding, and can be *actioned*



EA Office + Supervisory Board with the participation of Strategic Business Units (SBUs) and possibly (external) Service Providers needs to

Define Information and Process requirements

… for the business domain that is deemed to be in need of change
… on the whole-of-enterprise level (NB business processes cut across SBUs)

IMPORTANT NOTE: this step may not have to be performed before the next if change can be clearly localised to part of the enterprise!

s3. Requirements – Information and Process reqmts are best developed on the whole-of-enterprise level

- Information requirement models can be captured as Information Schemata (conceptual schemata – e.g. Entity Relatioship / UML Class Diagram, IDEF1X, etc)

- Process Requirements can be captured as Process Models. These typically cut across several SBUs and also call (rely on) APP Services

- The two are related, because processes use and produce information thus the above requirements need to be co-developed

  Reliance on existing database schemata
  Reliance on existing data definitions (of data not stored in databases)

- Information requirements: not only an integration of existing schemata – must apply quality criteria, design optimisation principles, plan for maintainability and possibilities for extension. Piecemeal, by domains of business function to carve out realistic projects

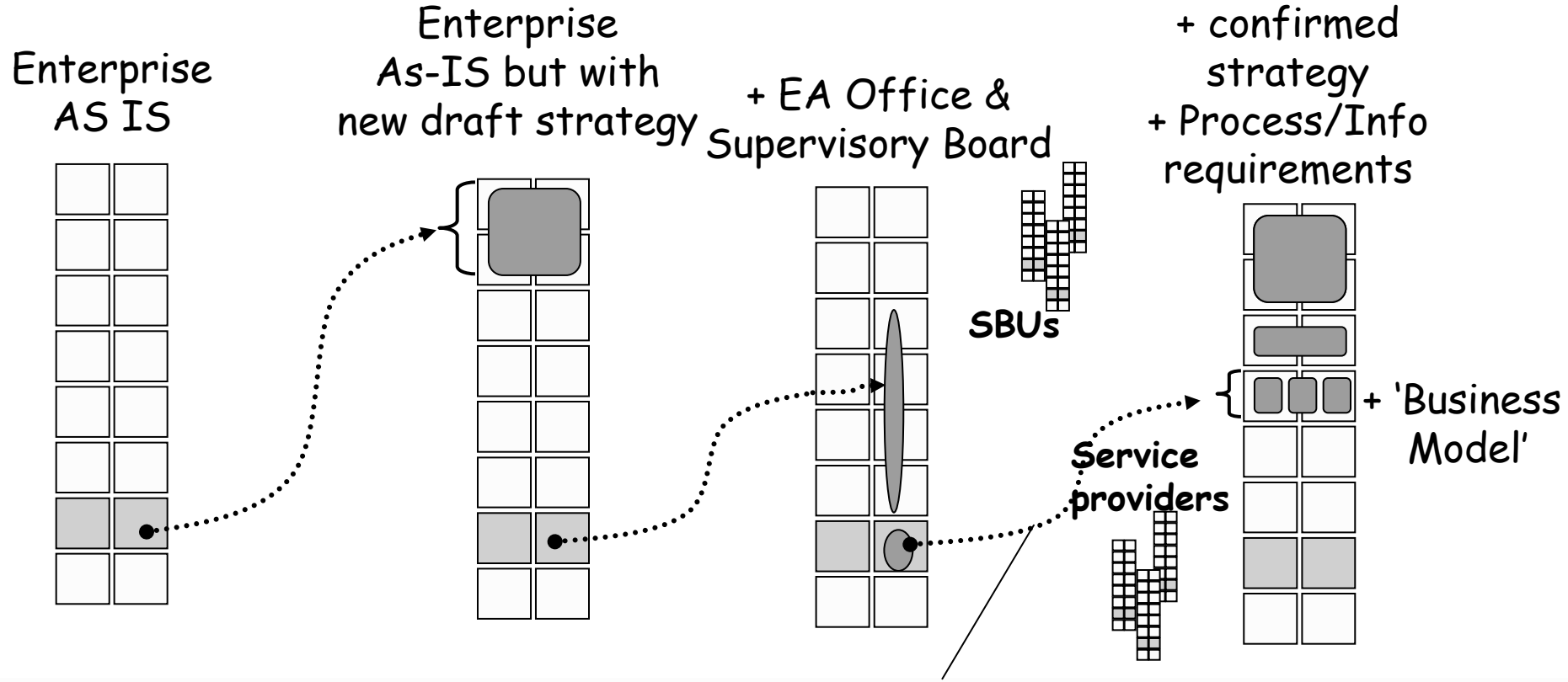
s4. Architecting – a 'Business Model' is a particularly useful way to represent *strategic relationships*, what needs to change, why , who should do it, when , where and how

The AS-IS and TO-BE Business Models would not be *completely* different – this *localises the change's scope*

Steps s1 + s2 can be done by the champion and sponsor (who can be the business owner or top mgmt with the approval of the business owner)

This step (s4) will need
a) A small group of people (who can involve and activate stakeholders from the rest of the business) and
b) A body that supervises the project / programme (governance)

EA group (a)
EA supervisory board (b)



EA Office with the participation of Strategic Business Units (SBUs) and possibly (external) Service Providers + Supervisory Board

Create an architectural (structural) decomposition of the Enterprise

On the high level this can be expressed as a 'Business Model' (BM)

The BM is a structural model of the enterprise but also shows the life cycle of the constituents

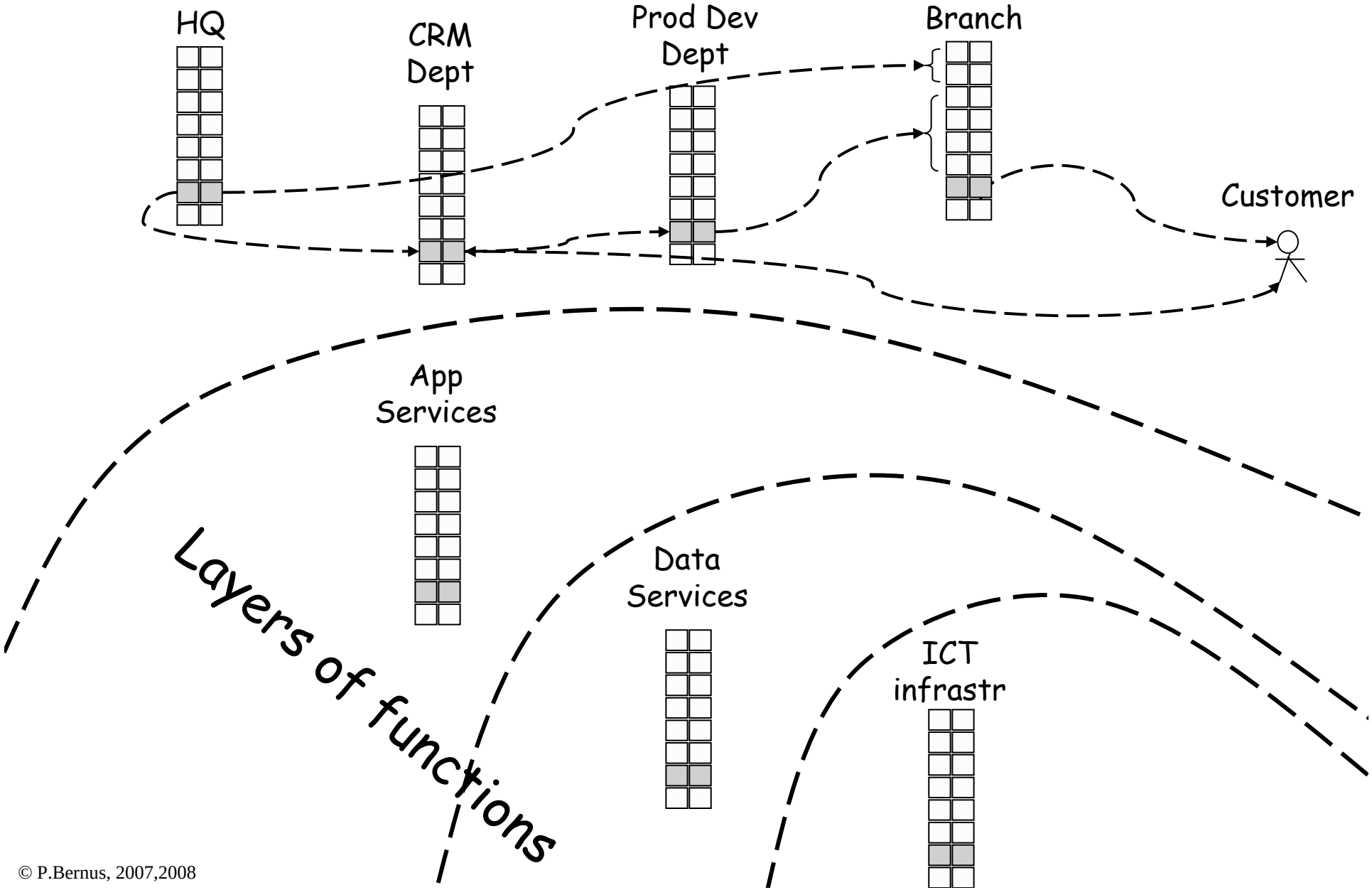
One can 'read' the methodology to be followed in the change process

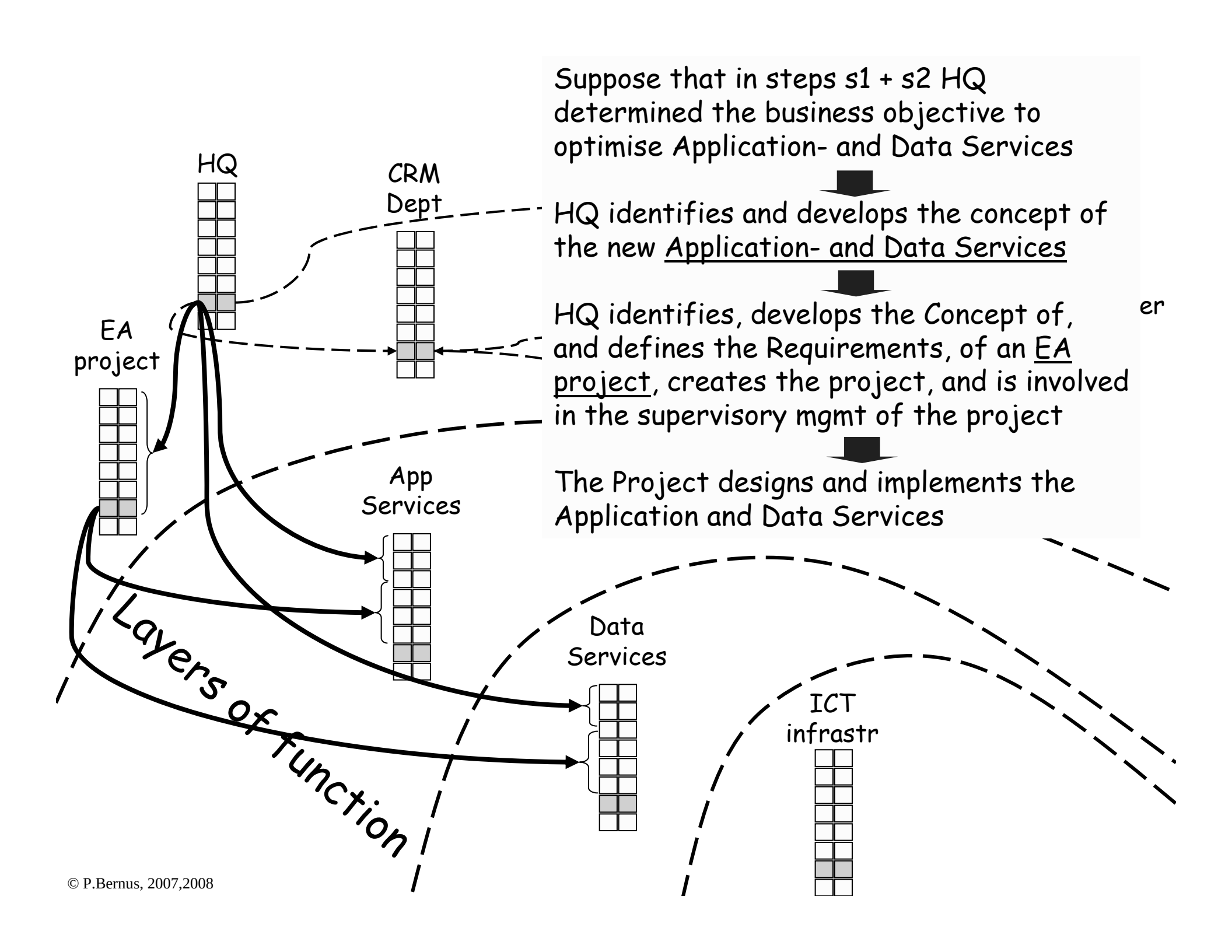
Note

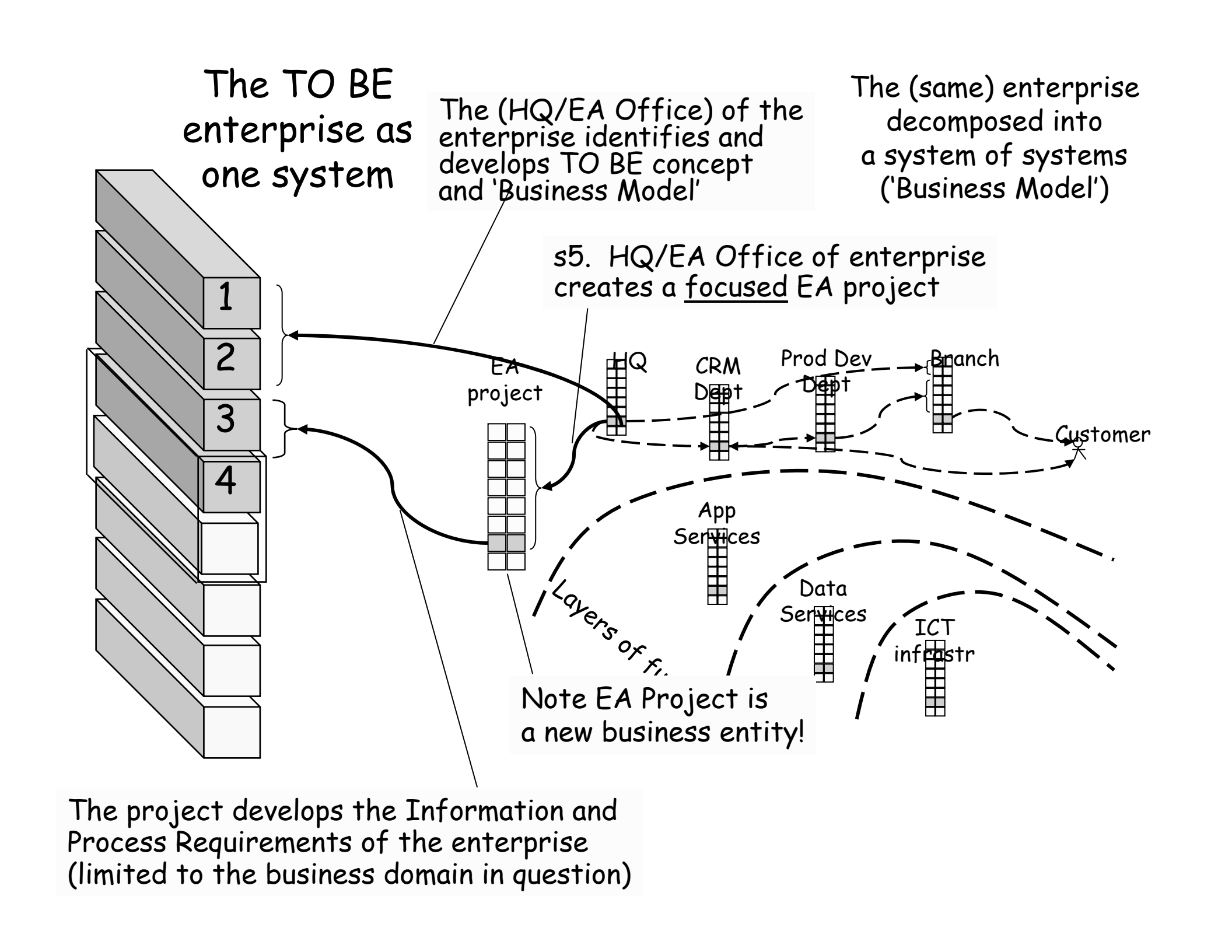
- s3 & s4 do not have to be carried out in sequence!
- The Business Model, which is a draft of (s4), can be developed ahead of time and through this the scope of change contained
- Subsequently it is possible to re-visit (s3) [the requirements] but localise the development of the requirements to the domains that need change (or are in some way involved in the change through 'spill effects')
- The outcome of (s4) is a Master Plan, including the Information-, Process-, Technology-, Application- and Human / Organisational Architecture(s)
- In practice we rarely carry out a complete enterprise wide Master Planning exercise (except perhaps for green field enterprises, however, in that case we rely on a previous similar enterprise's Master Plan)

A Business Model (TO BE example) –

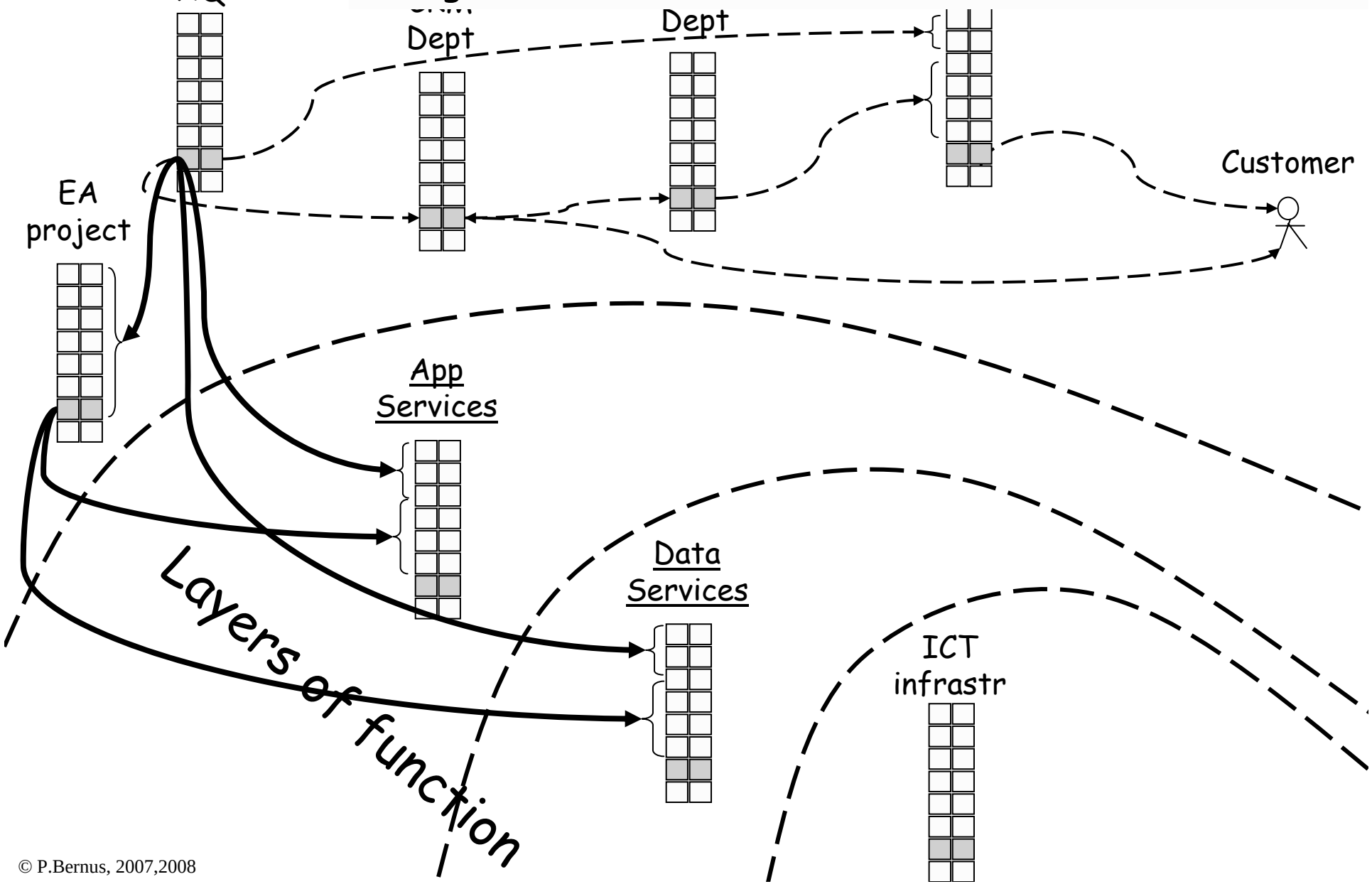
the enterprise is identified as a system of interrelated entities



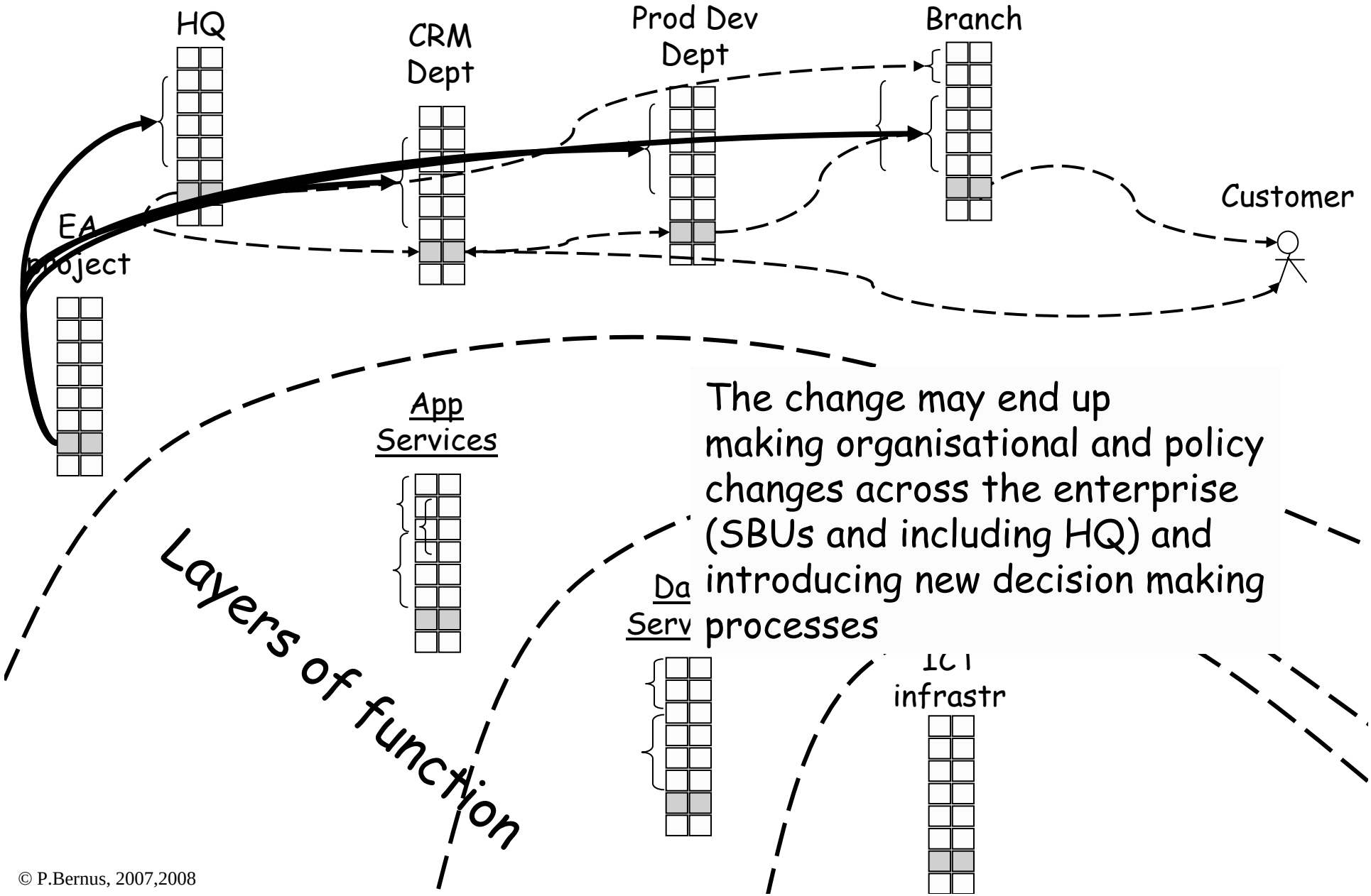



The TO BE enterprise as one system
The (HQ/EA Office) of the enterprise identifies and develops TO BE concept and 'Business Model'
The (same) enterprise decomposed into a system of systems ('Business Model')
s5. HQ/EA Office of enterprise creates a focused EA project
1
2
3
4
EA project
HQ
CRM Dept
Prod Dev Dept
Branch
Customer
App Services
Data Services
ICT infrastr
Layers of fi
Note EA Project is a new business entity!
The project develops the Information and Process Requirements of the enterprise (limited to the business domain in question)

It is possible to mark up this 'Business Model' (the model of the business) showing enterprise entities that need change or that need to be created

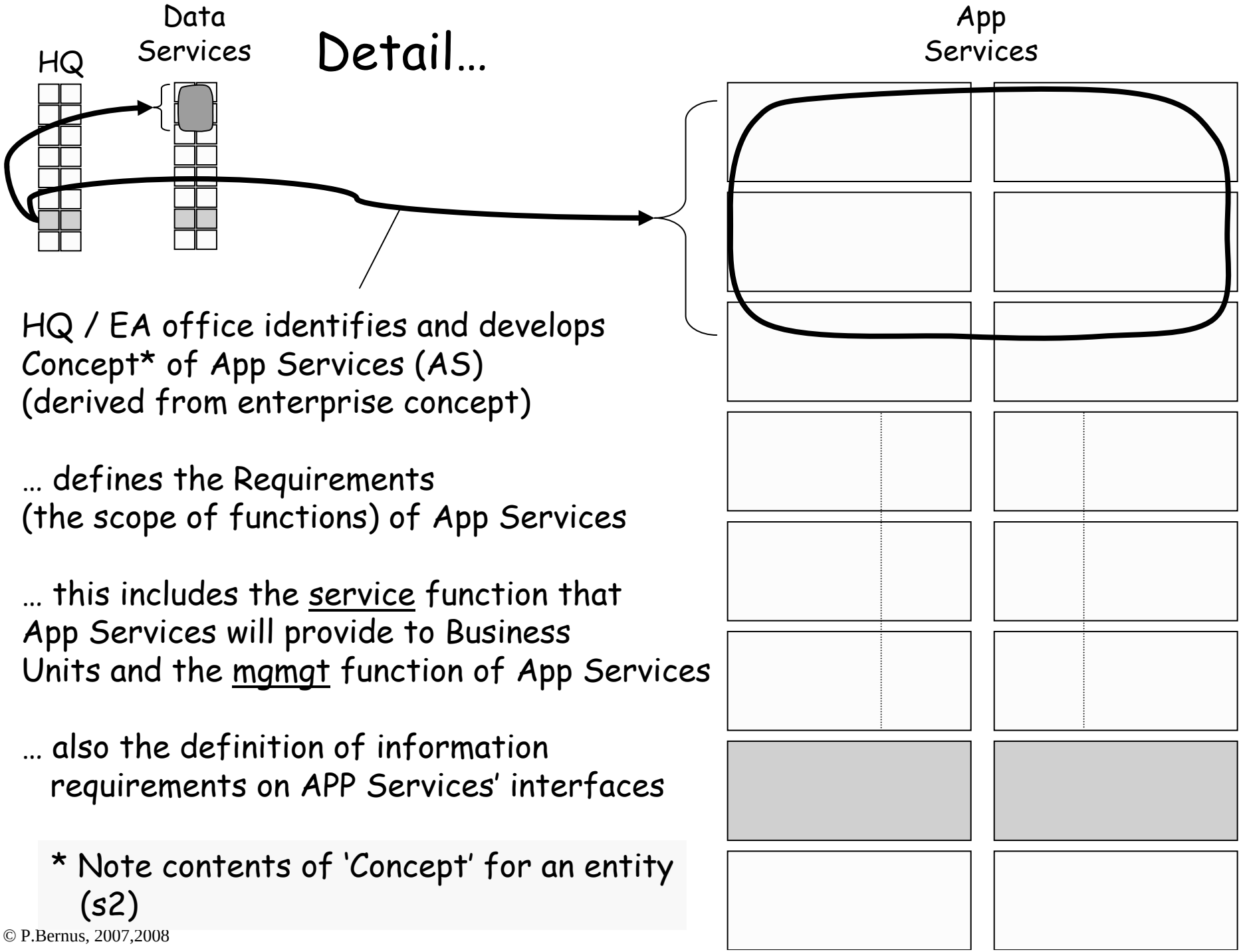


The 'Business Model' diagram representing the 'architecture of the business' includes operational and strategic relationships



The change may end up making organisational and policy changes across the enterprise (SBUs and including HQ) and introducing new decision making processes

Detail…



HQ / EA office identifies and develops Concept* of App Services (AS) (derived from enterprise concept)

… defines the Requirements (the scope of functions) of App Services

… this includes the service function that App Services will provide to Business Units and the mgmgt function of App Services

… also the definition of information requirements on APP Services' interfaces

* Note contents of 'Concept' for an entity (s2)

6 Execute Project …

(Informed by AS's Concept and Scope)

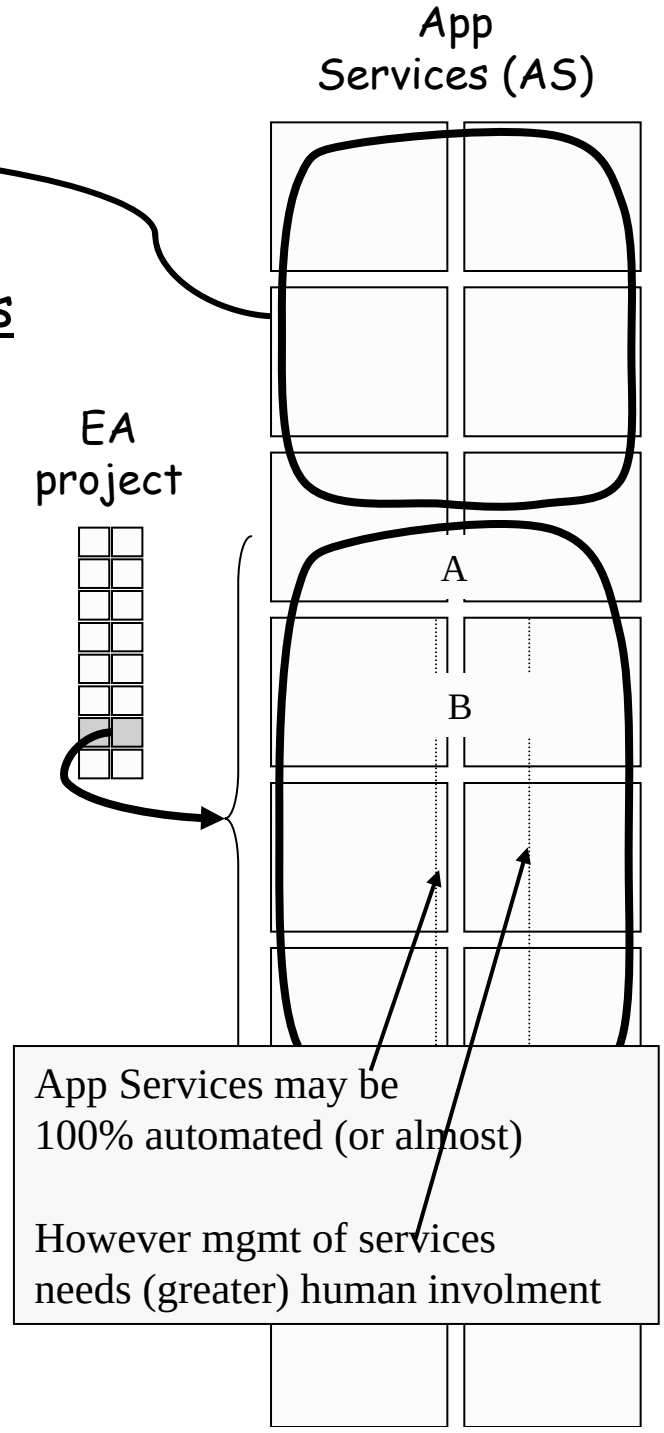
A Project *specifies* App Service Requirements

App functions, interfaces and their information content (as defined in Information Schema) & protocols to access Apps (e.g. through SOAP)

Service descriptions (e.g. using WSDL)

B Project *architecturally designs* AS

… (re) aggregates App functions into App modules *(cf* 'App Architecture')
… defines how App functions use Data Services (as defined in 'Info Architecture')
… defines human functions (roles / job descriptions) & processes, policies, procedures (mainly for mgmt roles in this case)



Architecturally designing Application Services

- As an important problem today is that application programs are tied to business processes, a recurrent strategic objective is to change this situation, in hope of
  - Implementing and separately maintaining applications as services
  - Implementing a business process design, management and execution service, so that business process implementations (*e.g.* workflows with suitably designed user interfaces) can *invoke* application services
- Usual problem: there are too many applications which support essentially the same function or there are large overlaps
- Therefore a profile cleaning is necessary
- The same is true of databases (must be done in s3!) – adoption of domain ontologies / reference models of commonly used information definitions

Cleaning application profiles

- The enterprise needs a stocktake of functions supported by applications (and associated databases)
- Decision needs to be made as to
  - what functions to aggregate in the TO BE applications
  - what functions to remove from existing applications
- A catalogue needs to be maintained of application functions for future decision making (when new business requirements suggest that new applications may be necessary)
- This decision making needs organisational roles and associated approval processes. Thus the change may end up making some organisational and policy changes across the enterprise (SBUs and including HQ) !

what to do (cont'd)…

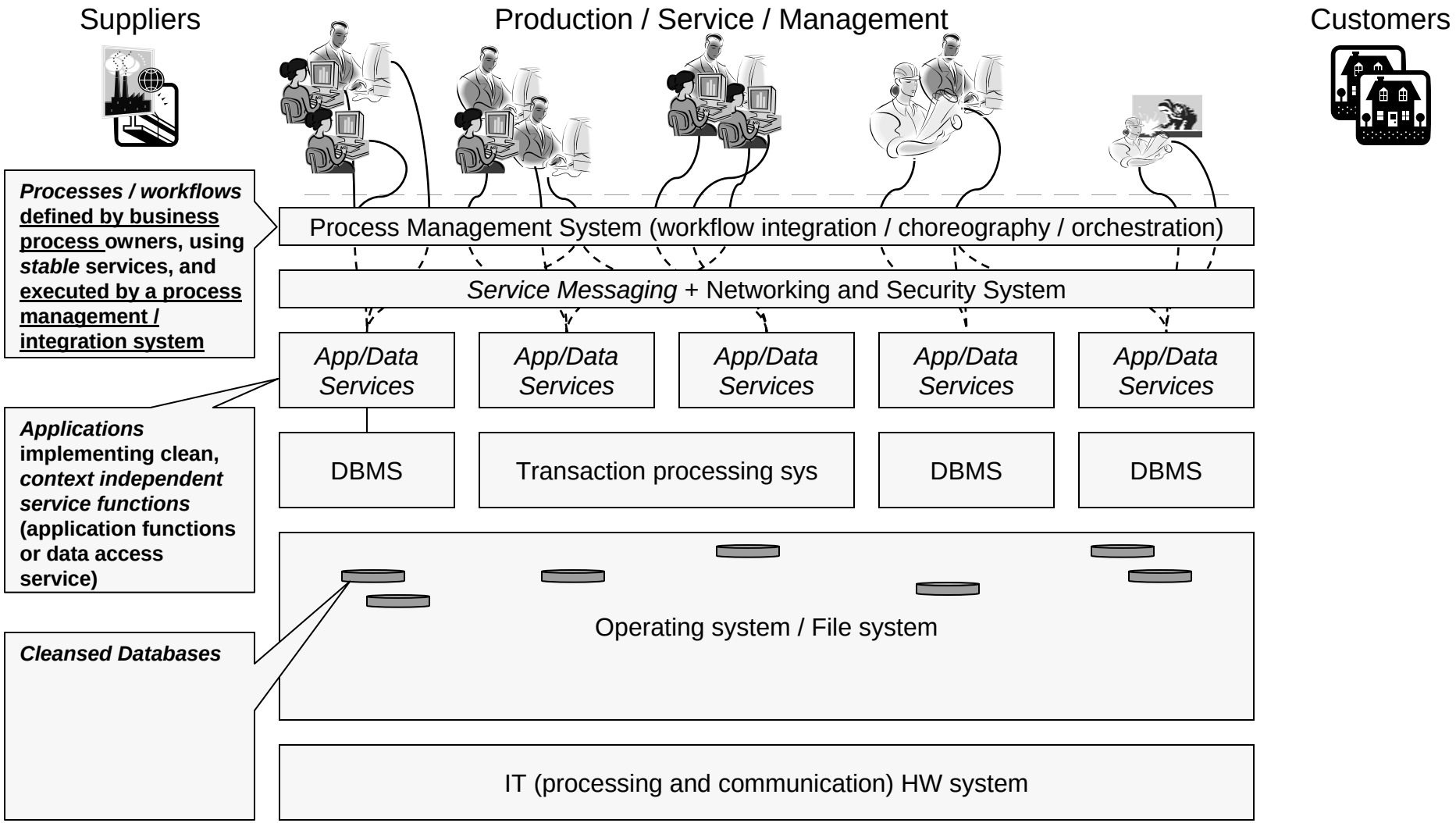
- When the above architectures are defined (in form of a Master Plan) there are several structural patterns that can be used
  - The 'Simplified 'typical' IT stack' is a pattern, which has problems as discussed
  - SOA is a different pattern, invented to solve these problems
  - This new pattern can be implemented in several ways, by making coherent detailed design decisions (Web services is one of them, but is not necessarily the best technology for the given organisation !)
  - Applying SOA as a principle, or pattern, effects all four (sub)architectures, and the way the organisation is structured overall!

SOA Strategy* must have

- Evidence of fulfilment of critical business goals
- Alignment with organizational enterprise architecture and current and future Information Technology (IT) infrastructure
- Realistic choices of technologies and infrastructures
- Realistic and gradual adoption strategy
- Adequate SOA governance structure
- *Priorities* for implementation
- Reuse strategy across internal and external organizations

* Lewis,A, Smith,D.B (September 2007) Four Pillars of Service-Oriented Architecture CROSSTALK The Journal of Defense Software Engineering

Modified IT stack



Basics (cont'd)

- Service Consumers
  - end users (through process interface), other applications or services

- Service Infrastructure
  - Service functions exposed through some description, catalogued, discoverable, accessed through [synchronous and/or asynchronous] service request / service messages
  - There are several implementation possibilities of this architecture.

3. SOA and project governance

- Policies
  - Policy Life-Cycle management (e.g. 'security policies')
- Principles
  - Use of Standards
  - Resources (Infrastructure, HR, Process)
- Decision authorities
- Decision processes, policies and procedures
  - Design decisions
    - Who (when and how) identifies the need and sponsors the creation of Services? (SOA is *part of* EA – an SOA project is created by the EA programme)
    - Decision hierarchy: CEO/Board – EA Programme cttee (involving SBU mgmt) – EA Programme Office – SOA Group – SOA Projects (a type of EA project)
    - What are the policies and procedures that determine the outcome? (including business analysis and technology evaluation)
    - Ownership of data, services and processes?
    - Responsibility for Data, Processes and Services?
    - Who absorbs the costs of managing Data, Processes and Services? (this can lead to organisational change)
  - Operational decisions
    - Access, Security / Risk
    - Availability, Performance and Cost monitoring, reporting and prediction
    - Identification of continuous improvement needs
    - Problem resolution

Lewis & Smith define four pillars of SOA*

- Develop appropriate SOA strategy
- Implement effective SOA governance
- Make sound technology assessments
- Need a 'different mindset' to use SOA

* Lewis,A, Smith,D.B (September 2007) Four Pillars of Service-Oriented Architecture CROSSTALK The Journal of Defense Software Engineering

4. The scope of the technical development (SOA Implementation Technology)

- Determine desired technical characteristics
  - Availability (24/7 or… ?)
    - SLA for both external and internal sevices
  - Security (identification, autentication, role and trust management, access control, auditing, etc)
  - Performance
    - SOA 'service abstraction' introduces overheads – not always tolerable unless infrastrcuture is adjusted !
    - Need Load Planning and Monitoring / Quality Assurance
  - Cost (investment efficiency, staggered implementation)
  - Risk (visibility for architects and governance, standards based – not vendor lock-in, lack of motivation, security risks)
  - Expected ROI

The scope of the technical development (cont'd)

- Evaluate Technology alternatives (strategically and tactically)
  - Learn (don't accept techno-babble!!!)
  - Assess (against the above characteristics [see previous slide])
- The result feeds into
  - Master planning (enterprise architecture) and
  - Business planning (master plan implementation)

Technology Evaluation

- Programme Level (strategic)
  - Decision horizon: 3-5 years
  - Decide technology architecture (this can be done as a project)
    - Define evaluation criteria for specific enterprise's needs
    - Develop process to evaluate one or more technologies (may incude prototype implementations)
    - Evaluate against criteria
    - Select preferred architecture / technology solution
    - Codify design principles and policies (p/p)
  - Create Master Plan (Architecture level Reference Model) for SOA implementations in the given enterprise
  - Periodically (or in light of significant events) re-visit / revise this decision (e.g.,1.5 - 2 years)

Technology Evaluation (cont'd)

- Project level (tactical)
  - Decision horizon 0.5 – 1 year
  - Codify p/p taking into account requirements of the deliverable
    (can restrict to a subset of Master Plan p/p!)
- Operational level monitoring and management of
  - Performance (speed, response time, volume, accuracy) – actual and trends
  - Reliability, availability, vulnerability

SOA and Business Model

- SOA (as a way of satisfying the business needs for services) must be integrated into the Business Model of the enterprise (see earlier slides)

- It is possible to adopt SOA on
  - Business Level (-> business agility)
  - Application Level (-> application reuse)
  - Infrastructure level (-> flexibility / virtualisation)

- Decisions are business driven but technology creates opportunities as well as poses constraints

Important part of SOA is Business Process Management and Execution Services

- Aim at Process Models being designed and maintained by business (process owner)
- Aim at capturing feedback on process execution from the 'trenches'
- Include continuous improvement in the tasks of the process owner and the tasks of governance
- Close the loop (Six Sigma style)

BPMN process model

BPMN is a graphical process modelling notation for

- Elicitation of Functional Requirements and
- Validation of these with process stakeholders

"The objective of BPMN is to support business process management by both technical users and business users by providing a notation that is intuitive to business users yet able to represent complex process semantics. The BPMN specification also provides a mapping between the graphics of the notation to the underlying constructs of execution languages, particularly BPEL4WS." ##

NB. the mapping to BPEL may be incomplete, and need additional detailed design decisions – in the same way as with IDEF(0,3…), EPC, … Models : BPMN is 'larger' then BPEL 2.0

BPMN Specification v1.0

What BPMN is not

- *Additional models may be needed to substantiate the value proposition of the designed process* (resource consumption, speed, sensitivity, etc)

- Simulation languages, FirstStep models, Activity Based Costing, etc. may be needed to to prove such desired characteristics. The BPMN model can be used to create a 'skeleton' of such models – with additional details added by the analyst

- Present tools *do not* have the capability to either do this translation or to maintain dependency in light of design changes

- Typical questions to prove value proposition:

  - "What cost saving is guaranteed by the new process?"

  - "Statistically what is the cost- and speed sensitivity of the process due to variations in the cost and availability of specified resources needed by the process?"

  - "How much will the risk of non-performance increase/decrease?"

BPEL workflow model vs. BPMN process model (cont'd)

BPEL is an executable process modelling language, for

- detailed design, implementation and operation

"WS-BPEL business processes represent *stateful* long-running interactions in which each interaction has a beginning, defined behavior during its lifetime, and an end.

For example, in a supply chain, a seller's business process might offer a service that begins an interaction by accepting a purchase order through an input message, and then returns an acknowledgement to the buyer if the order can be fulfilled" #

A BPEL process model can be expressed as an XML file, to be read by a workflow engine (not by humans) and executed using a workflow engine

A workflow engine creates and executes instances of BPEL workflows

WSPEL v2.0 11 April 2007. OASIS. P34

A BPEL workflow is like an procedure composed of elementary behaviours:

- **<receive>**
- **<reply>**
- **<invoke>**
- **<assign>**
- **<throw>** *[exception]*
- **<exit>**
- **<wait>**
- **<empty>**
- **<sequence>**
- **<if>**
- **<while>**
- **<repeatUntil>**
- **<forEach>**
- **<pick>**
- **<flow>**
- **<scope>**
- **<compensate>**
- **<compensateScope>**
- **<rethrow>**
- **<validate>**
- **<extensionActivity>**

5. External and internal participants in the transformation program

- EA Governance –
  - Board representatives,
  - SBU management,
  - HR & Financial leaders
- EA programme – Enterprise Architect and core team
- SOA projects –
  - End users and their representatives
  - Project team (developers), architects and project manager with combined competencies in Process, Policy and Procedures, Information, Application and Technology Infrastructure, and Organisation (!)
  - Advisors or contactors
- May include external stakeholders (e.g. enterprise Network partners)

- Adherence to open standards is important, avoid vendor lock in
- The very aim of SOA on the business level is dynamic change capability / agility
  - Mergers / acquisitions
  - Outsourcing
  - Insourcing
  - Flexible change of supply chain

6. Prioritising and staging the development effort

- Start small – e.g. choose a value added process and implement using SOA (clear ROI, buy-in)
- Determine low risk start to allow the organisation to learn
- SOA fosters reuse – which only happens if incentives are in place
- Keep a close eye on performance objectives & criteria

Determining Priorities

- Which business objectives are most pressing?
- What are the CSFs?
- Which processes would benefit most?
- Which legacy databases and applications are involved?
- EA: process and information structure (re)mapped to subsystems that will implement underlying services
- Process analysis / (re)design
- Information analysis / (re)design
- Analyse impact on Policies and Procedures and change them if necessary

Summary

We covered

- SOA as a strategic business transformation initiative
- How to build the business case and set a realistic transformation strategy
- The scope of SOA and project governance
- The scope of the technical development
- External and internal participants in the transformation program
- Prioritising and staging the development effort

The end