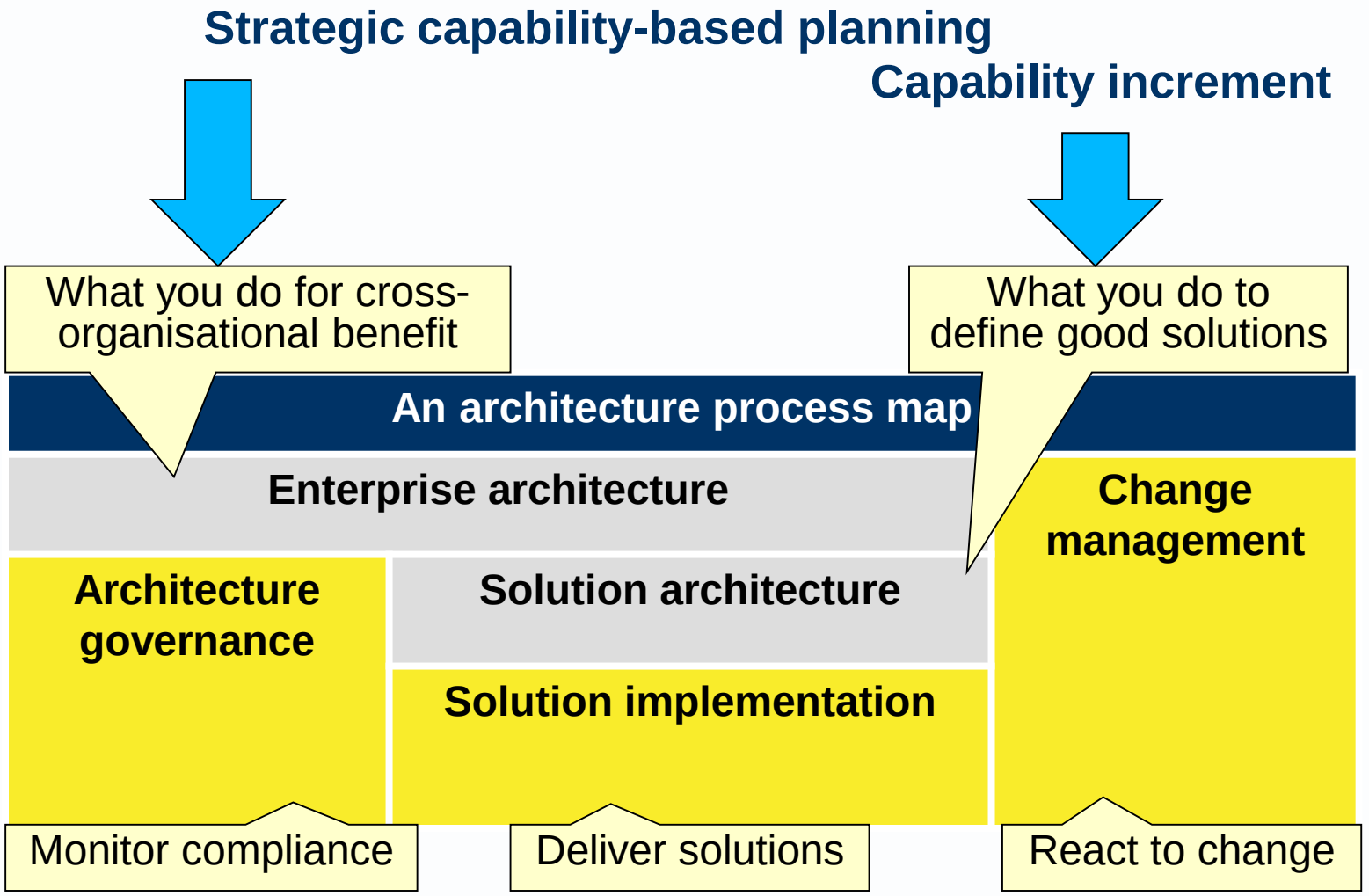
Avancier Reference Model

Architecture Management (ESA 11)

A business planning context



Architecture management

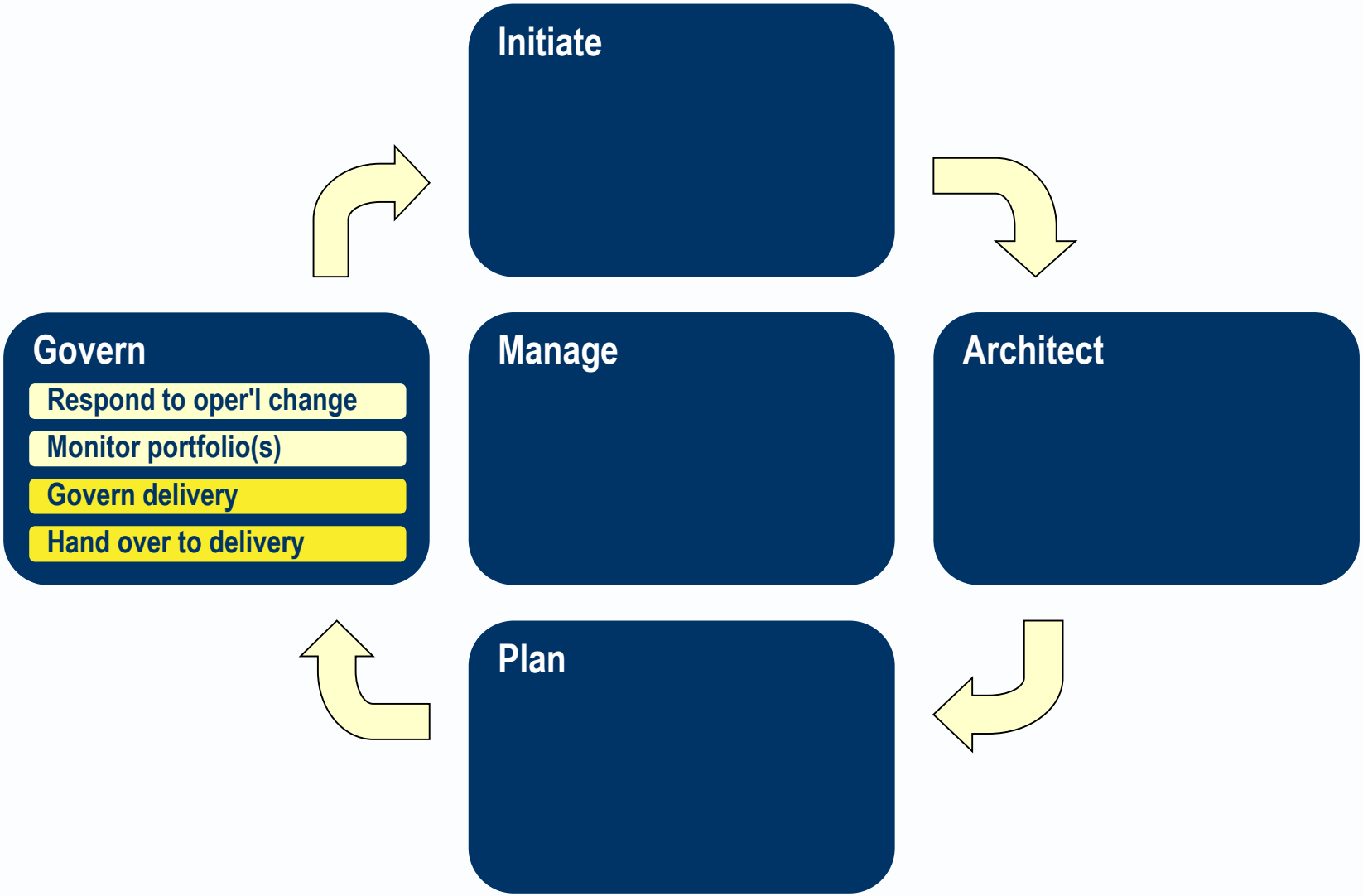
Avancier

The organisation and processes that are needed to **govern** and **implement** an architecture description, both in development and in operation, including the management of **changes**.

	People	Processes	Products
General Management	Board Manager Team	Set aims and directives Plan Start/initiate Monitor Control Stop/close	Aims and directives Plans Review criteria Progress reports Conclusions Process definitions

Architecture Implementation - in the AM process

Avancier



Architecture Implementation

- [A work process] that realise an architecture through system development and deployment.
- This requires programme and project management roles and processes.

Software Development Life Cycle (SDLC)

- [A work process] centered on analysis, design software engineering, testing and roll out.
- There are agile, iterative and waterfall variants.

	People	Processes	Products
SDLC	Steering group Project manager Project team	Plan projects Initiate projects SDLC Phase/milestone reviews Close projects	Project plans Project initiation documents Review checklists Review reports Project closure reports Process definitions

SDLC- Waterfall

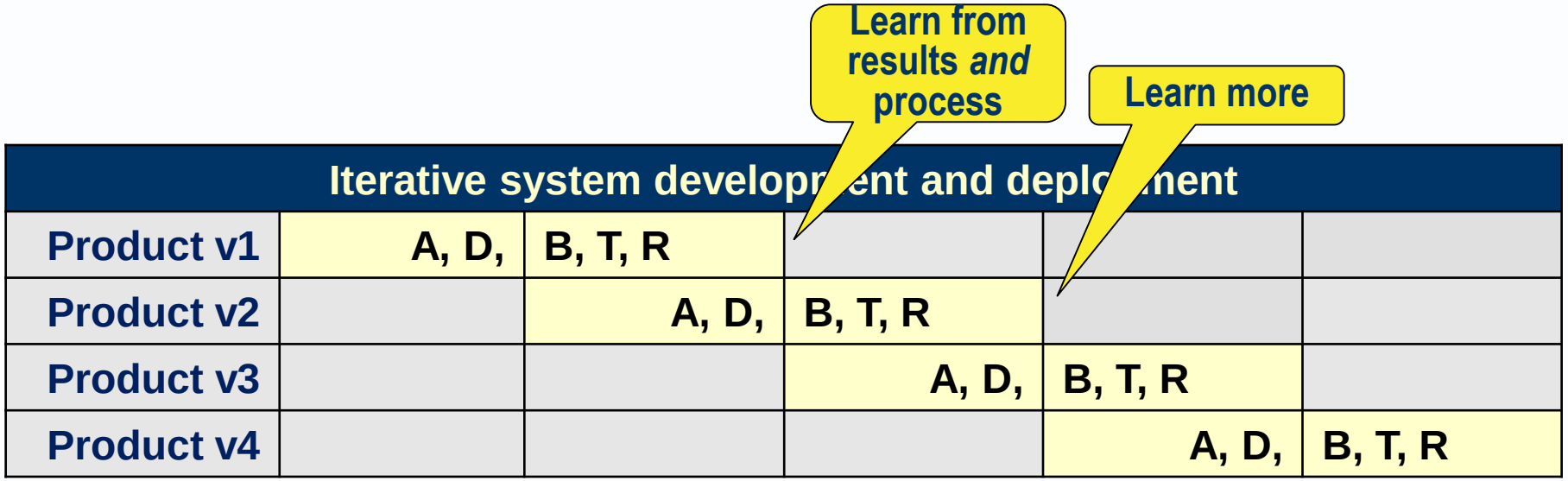
Avancier

- [A technique] a development process that places analysis, design, build, test and roll out in sequence.
- Engineers proceed from one kind of work to the next without significant iteration or parallelism between stages.

Waterfall system development and deployment					
Analyze	**A**				
Design		**D**			
Build			**B**		
Test				**T**	
Roll out					**R**

SDLC- Iterative Development

- [A technique] a development process that proceeds by increments, meaning that a working subset of the full solution is delivered as early as possible.
- It is a foundation of the Unified Method and known as Incremental Development in DSDM.
- It is an essential feature of agile methods, and may be used in non-agile projects also



SDLC: Agile development

Avancier

- [A technique] a solution development process that is not only iterative, but also flexible about the requirements, the solution and the process being followed.
- It favours
  - negotiation over planning, and flexibility about requirements.
  - early testing for usability and performance; user involvement and feedback is a prerequisite.
  - short-cycle iterative development; it looks for the minimum change that adds value to a system, and strives to deliver that change in the next sprint/release.
  - capitalising on the skills and knowledge of a small team

- *"I estimate that 75% of those organizations using Scrum will not succeed in getting the benefits that they hope for from it."*
- Ken Schwaber in an interview posted on Agile Collab

What does agile mean?

- Fail faster is good!
- Delivery early, commit late.
- Accept flexible, prioritised and ever changing requirements.
- High-level documentation of specifications and models.
- Test-driven rather than model-driven.
  - Waterfall methods suggest model > code > test.
  - Agile methods suggest test > code > model.

- "Two of the greatest [agile] rallying cries … are the slogans:
  - ‘Do the Simplest Thing that Could Possibly Work’ and
  - ‘You Aren't Going to Need It’ (known as YAGNI).
- Both are manifestations of the XP practice of Simple Design.“

Martin Fowler

BUT IN THE FIRST PLACE Does your project suit agile?

Avancier

- Score your current or most recent project using the form below
- High scores make agile methods difficult (but not a bad thing)

What kind of project?		
Time/cost-driven	0, 1, 2, 3	Mandatory requirements-driven
Users available for Joint App Dvlpmnt	0, 1, 2, 3	Users not available
Developers empowered	0, 1, 2, 3	Developers not empowered
What kind of system and work?		
Divisible into usable releases (soft target)	0, 1, 2, 3	Indivisible (hard target)
Simple UX and/or UI technology	0, 1, 2, 3	Complex UX and/or UI technologies
Output/enquiry/report dominated	0, 1, 2, 3	Input/update dominated
Simple (CRUD) data processing rules	0, 1, 2, 3	Complex data processing rules
On-line	0, 1, 2, 3	Batch
Stand-alone	0, 1, 2, 3	Highly integrated w other systems
Add up the scores for your agile potential quotient		

Agilists favour

Difficult projects

Difficult projects

Role of solution architect wrt agile software development

- Complete a solution outline in accord with EA
- Stabilise “infrastructure” as far as possible
  - platform technologies
  - persistent database structures
- Use agile development as far as possible
- Govern acceptability of changes during an agile project.

Can agile be effective with outsourcing?
12 Lessons Learned by Peter DeYoe www.it-insight-blog.com

Avancier

- **Lesson 1: [**Schedule] an overlap of at least 2 hours for your onshore and offshore teams' working day, if possible. This greatly increased the communication flow and cohesiveness of the teams.
- **Lesson 2:** Create a robust repository and collaboration site that will be the site of record for all specifications, test cases and discussions. SharePoint [recorded] all communication, collaboration and critical Artefacts.
- **Lesson 3:** Do not use email as your *primary* method of communication for topics such as requirements clarification or design decisions. Ensure communication is conducted through the repository.
- **Lesson 4:** Implement web conferencing to create a sense of proximity. We used this daily to conduct stand ups, review wireframes and specifications, walk through requirements and conduct Sprint reviews.
- **Lesson 5:** Have one central point of entry for project status. Each team member records progress via a central Scrum management tool - we used ScrumWorks to create accurate product backlogs, Sprint backlogs and burn-downs on a daily basis.
- **Lesson 6:** Ensure your offshore team has their own development and test environments, bug reporting tools (Bugzilla in our case) and source code repository.
- **Lesson 7:** Shorten your Sprints. We shortened the Sprints from 4 to 2 weeks.
- **Lesson 8:** The Scrum Master - the key arbitrator - must be top notch - perhaps co-located with Product Owner.
- **Lesson 9:** If possible, your Scrum Master should speak the languages of onshore and offshore teams.
- **Lesson 10:** The Product Owner must clearly define what "done" means for each user story. Well- defined acceptance criteria should be included in each user story.
- **Lesson 11:** A strong technical leader on the offshore team – with all of the technical skills to be self sufficient.
- **Lesson 12:** The offshore team must be properly trained in Scrum and specifically in your particular implementation of Scrum.
- Additional thoughts on running an Agile project using offshore resources can be found in an article written by Martin Fowler http://martinfowler.com/articles/agileOffshore.html

Transition

- [A work process] that, once the architecture has been realised in the form of an operational system, hands over that system to the organisations that will operate it.

Transition into business operations

- [A work process] that hands over a completed solution to a business unit for business process operation and management.

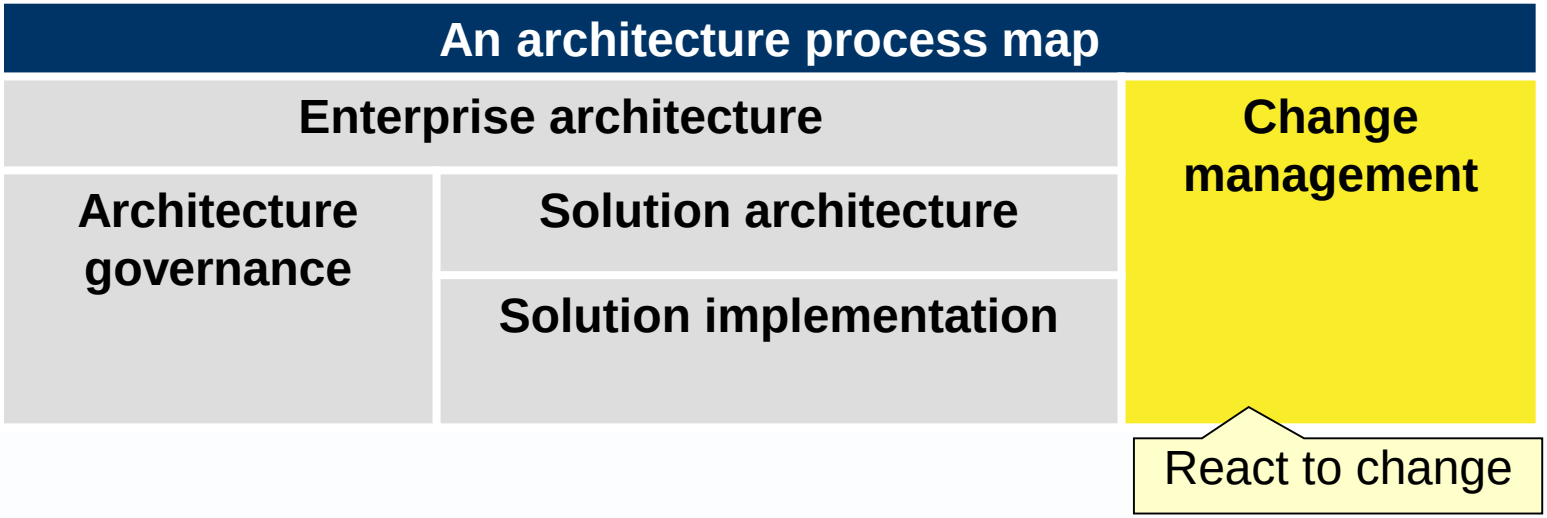
Transition into IT operations

- [A work process] the hands over a production system to be run by managed operations or ITSM organisation.

Transition into maintenance

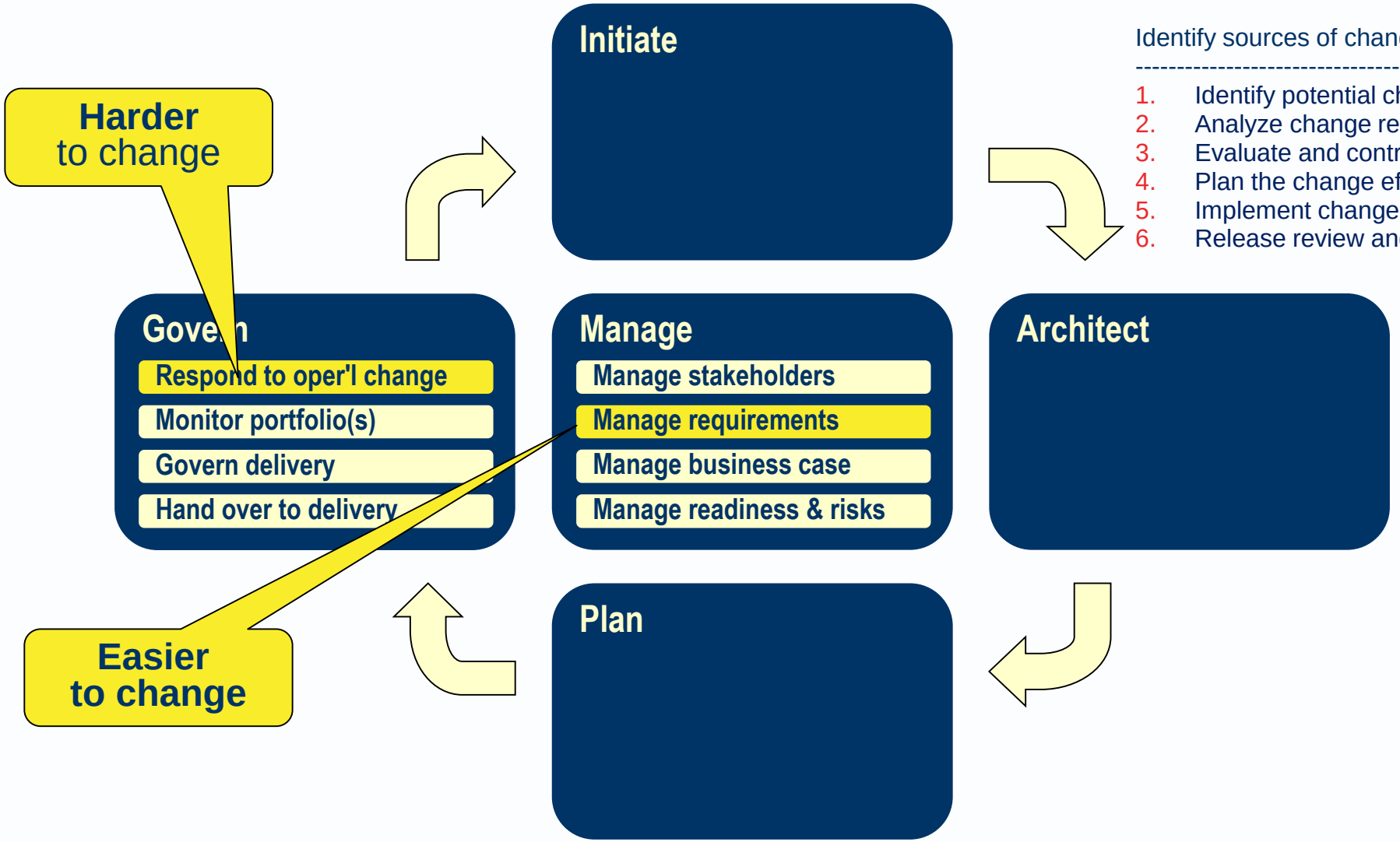
- [A work process] that hands over design and compile-time system to be maintained and perhaps enhanced by some kind of maintenance organisation.

Change management



Architecture change management - in the AM process

Avancier



Change management

- [A technique] the roles and processes needed to both
- exercise **change control** to a baseline, and
- perform **configuration management.**

	People	Processes	Products
Chanage Management	CAB Change reviewers Review administrators	**Change control,** Monitor events Record change requests Analyse impacts of change Determine actions Process changes **Configuration management**.	Baseline configurations Configuration items Requests for change (RFC), Impact statements

Change control

- **Change Control**
- [A technique] featuring the roles and processes needed within change management to
  - monitor potential sources of change
  - record change requests
  - perform impact analysis
  - deterrmine, make and release changes

  - **RFC: Request for Change**
  - [An artifact] or form used to record details of a new requirement, a problem report or a change request to any configuration item.
  - **Impact analysis**
  - [A technique] to analyse the effects of a change, determine feasibility, update the business case and produce an impact analysis report.

Configuration management in general

- [A technique] the roles and processes needed within change management to establish a baseline configuration and apply changes to that baseline configuration.
- Involves work to:
  - Identify and document the characteristics of each item.
  - Define dependencies between items.
  - Control the introduction of new versions of items.
  - Report the status of configuration items and changes to them.

Configurations in general

Avancier

Baseline configuration

- [A passive structure] a specification or product that has been formally reviewed and agreed upon.
- The basis for further development.
- Can be changed only through formal change management.
- E.g. a contract, a requirements catalogue, architecture documentation, or a hardware configuration.

Configuration Item

- [A data object] an item in a baseline configuration.
- It could be a requirement, a source code component or a hardware device.
- It can be at any level of granularity.
- A “Component of an Infrastructure under the control of configuration management.
- A configuration item can range from an entire system (hardware, software, documentation) to a single hardware component.” ITIL

Architecture change management

- [A work process] needed to manage changes to architectures, mostly stemming from changes to requirements or constraints, or operational systems.

- No different in principle from general change management (below).

Agile

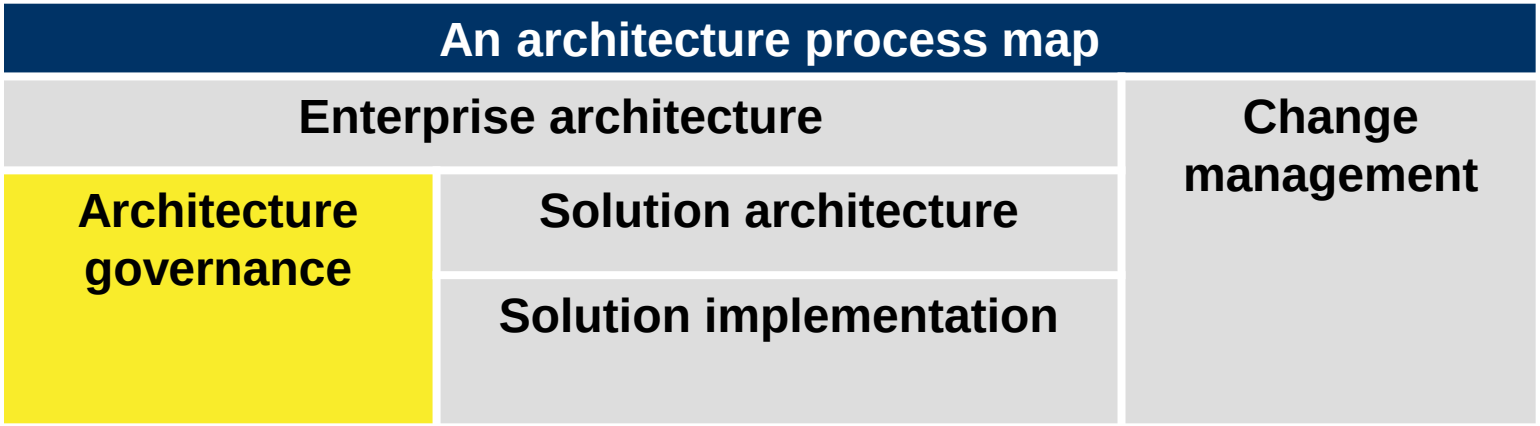
- [A property] willing and able to speedily respond to change.

- Change management can turn into a bureaucracy that not only stifles change but also stifles progress
- Change management can be heavy or light
- The trick is to be agile without abandoning change control

Architecture change control organisation

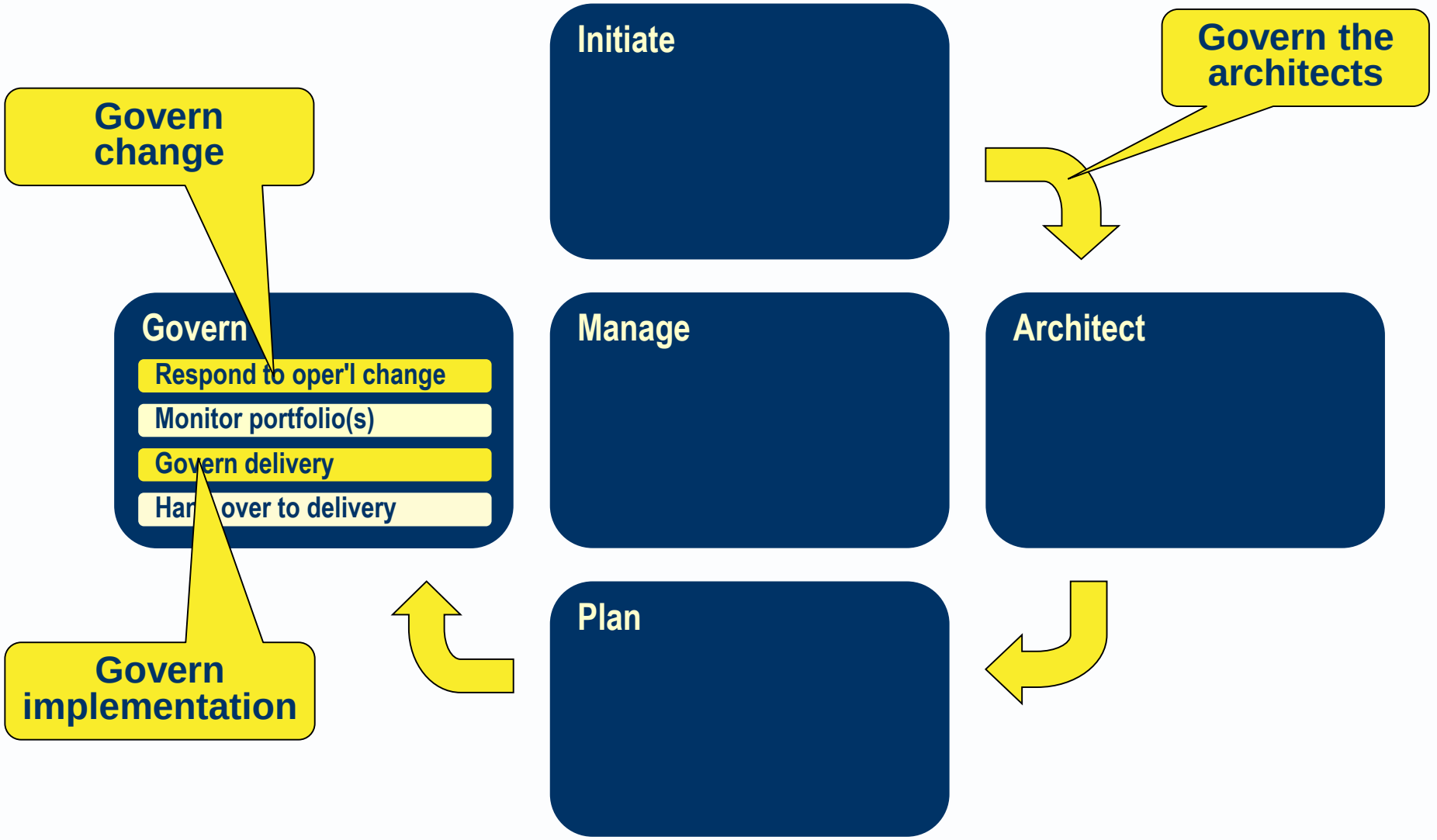
- **Strategy & Architecture Board**
- *Architecture* change control
  - CIO
  - Business rep(s)
  - PMO rep(s)
  - Enterprise architect or lead solution architect
  - Domain specialists (business, data, apps, infrastructure, security)

Governance



Architecture governance in the AM process

Avancier



What does a governor do?

Avancier

- BBC?
- School?
- Building site?

- Monitor compliance of operations against strategic principles, policies and goals

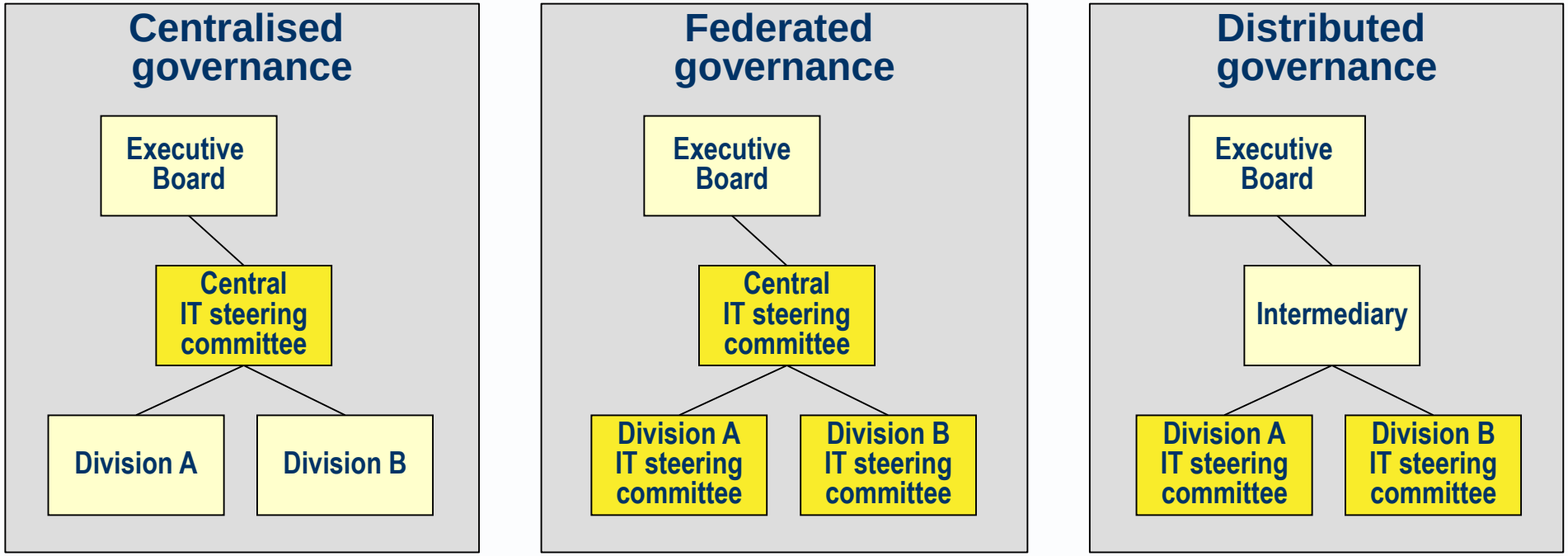
	People	Processes	Products
Governance	Board Governing architect Compliance reviewers Review administrators	Directive adoption & review Compliance review Dispensation Monitor and report Business alignment Environment management	Principles, policies, standards etc. Reference models Contracts Review checklists Review reports Dispensations Process definitions

Governance

- [A discipline] for monitoring and steering the management of an enterprise in accord with overarching drivers, goals/objectives and principles/policies.
- It may be subdivided into:
  - **Corporate governance:** the responsibility of the enterprise's executive board. (SOX, Cadbury)
  - **IT governance:** the responsibility of an IT board (COBIT)
  - **Architecture governance:** the responsibility of an architecture board.
- Different enterprise relate these governance organisations in different ways.

IT governance organisation

- IT governance responsibility of IT steering committee
- Typical IT governance organisation models are



Architecture Governance

Avancier

Architecture governance

- [Governance] of architecture, development and operations to ensure it conforms to pre-defined architectural requirements, principles, policies and models.

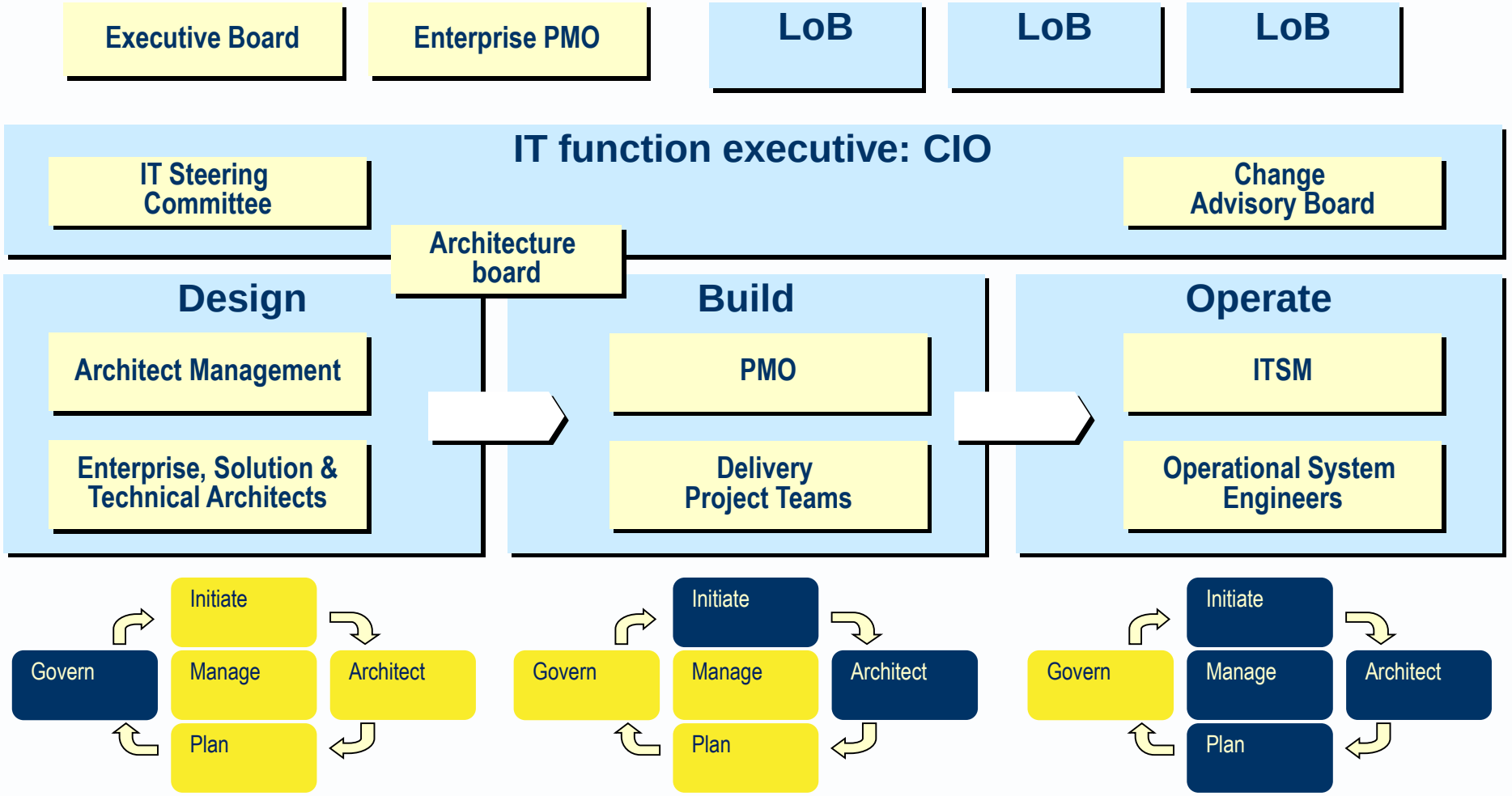
Architecture board

- [An organisation unit] that maintains architecture principles and governance processes, promotes and ensures provision of architecture resources, and reviews compliance reports.

Architecture contract

- [A document] that defines those architectural requirements, principles, policies and specifications that a system should conform to as it is built and when it runs.
- Also defines any architecture stakeholder rights and interests that must be met.

Where does Architecture Board sit in an IT organisation?



Governing architect

- [A work role] the architect who has been nominated by the governance organisation to ensure a system is built and/or run in accord with its architecture contract, to manage risk and to ensure the value of the system to its stakeholders.
- The role may be played by a chief architect or design authority, architecture domain specialist or solution architect, depending on what is to be reviewed.

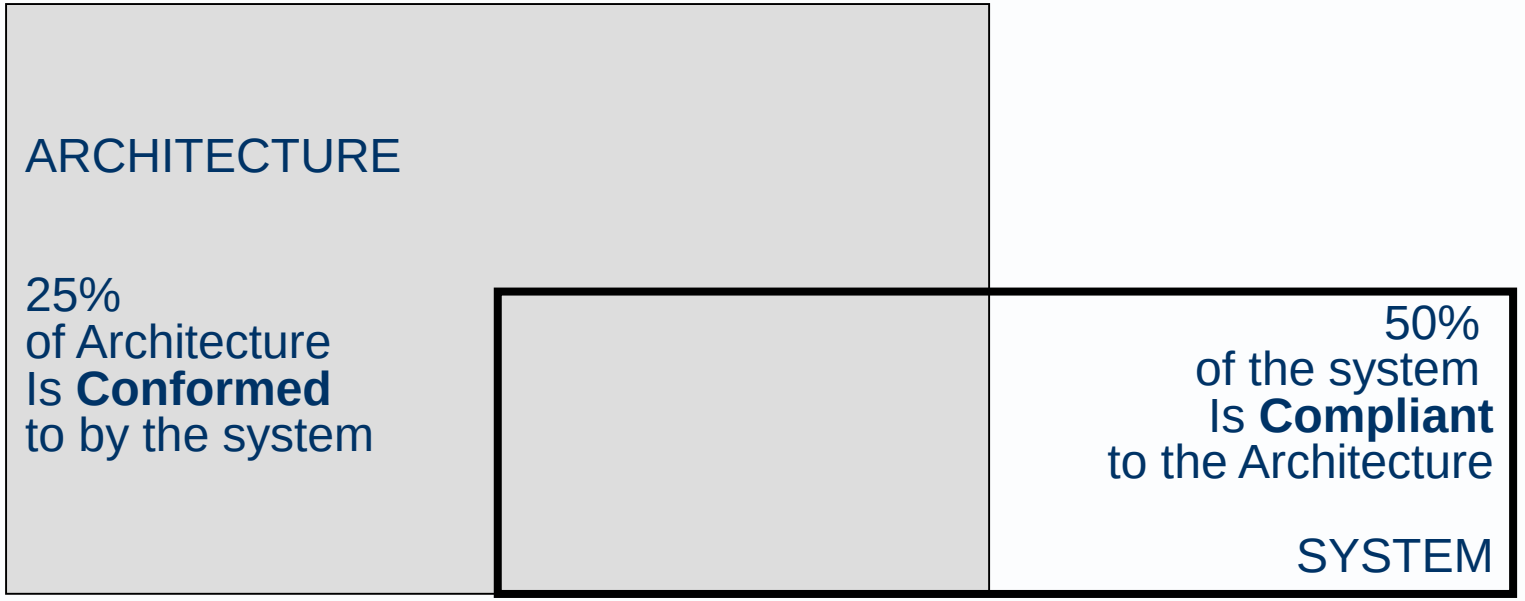
Architecture Governance - Compliance

Avancier

- **Architecture compliance review**
- [A work process] for monitoring work against architecture aims, directives and documents. Different reviews may be carried out at different points in system design and development.
- Reviews may require a governing architect and/or use an architecture review checklists.
- **Architecture review checklist**
- [An artifact] a standard checklist of questions for an architecture compliance review.
- The questions are general ones, not necessarily mentioned in the architecture contract.
- **Architecture conformance level**
- [A property] showing how well or how much of an architecture contract is met by a system, or an architecture is realised in a system.
- **Architecture compliance level**
- [A property] showing how well or how much of a system corresponds to its architecture contract and/or description.
- **Dispensation**
- [A document] a time-bound waiver from the terms of an architecture contract.
- It is granted by a governing architect, and should be reviewed after the specified time.

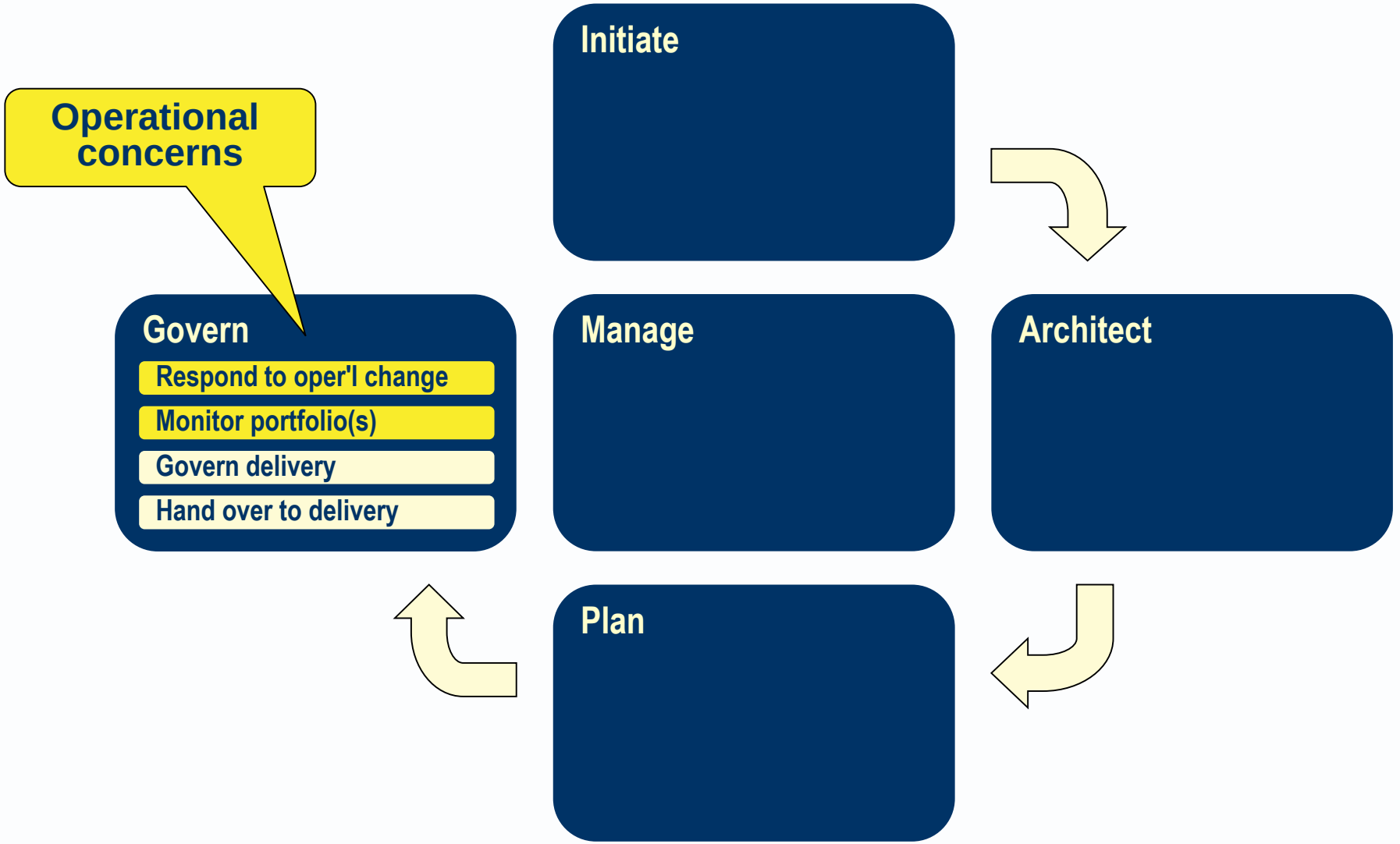
Conformance v compliance

- Your chance of remembering the difference is 50/50!



Architecture in Operations in the AM process

Avancier



Architecture in Operations

Avancier

- The organisation and processes that are needed to manage the architecture description of an operational system.

- Public companies subject to the U.S. Sarbanes-Oxley Act of 2002 are encouraged to adopt COSO and/or COBIT
- Many more appear to have adopted ITIL

	People	Processes	Products
Service Management	Board Service managers Service administrators	System administration Problem/help desk Event monitoring Service level monitoring Exception reporting Exception handling	Service level requirements Service level agreements Service level reports

IT service

Avancier

- [A service] provided an IT operations department. E.g.
  - management of user roles and identities,
  - client device configuration,
  - storage administration,
  - network provision, monitoring and analysis,
  - server provision, monitoring and analysis,
  - business activity monitoring,
  - virtualisation,
  - back up & restore,
  - incident and problem management.

newScale Service Portfolio Library™

Locate Service Offerings by Category

Application Services
Hosting, support and maintenance and enhancement of existing and new applications.

- Application Hosting
- Enterprise Application Management

Consulting Services
Advisory and hands-on functions performed by specialized IT staff at a fraction of the cost quoted by outside vendors.

- Business Process Reengineering
- Project Consulting

Workplace Services
Service offerings for the end user environment, including desktop computing, devices and remote end user experience.

- Desktop Computing
- Email and Calendaring

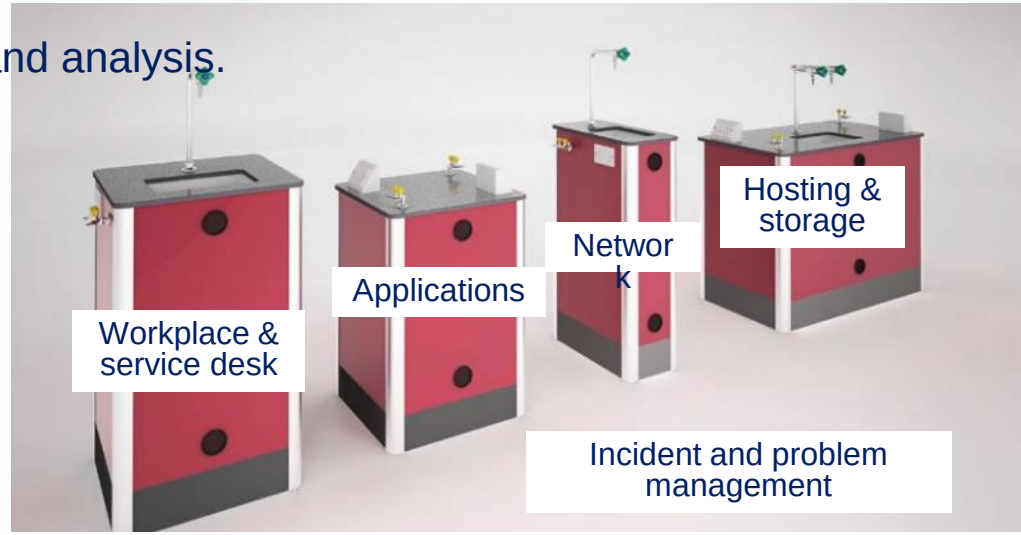
Technical Services (IT to IT)

- Mainframe
- Network
- Servers
- Storage

ITSM - Managed Operations - Service Towers

Avancier

- Workplace and service desk:
  - Desktop/laptop/mobile configuration and onsite support,
  - Management of user roles and identities
- Applications:
  - Support and maintenance
  - Projects
- Network:
  - Network and telephony provision, monitoring and analysis
- Hosting and storage:
  - Data centre, server provision, monitoring and analysis.
  - Virtualisation, back up & restore
- Incident and problem management



ITSM

Avancier

- **ITSM: IT Services Management**
- [A work process] the roles and processes for managing IT infrastructure and the services it provides.

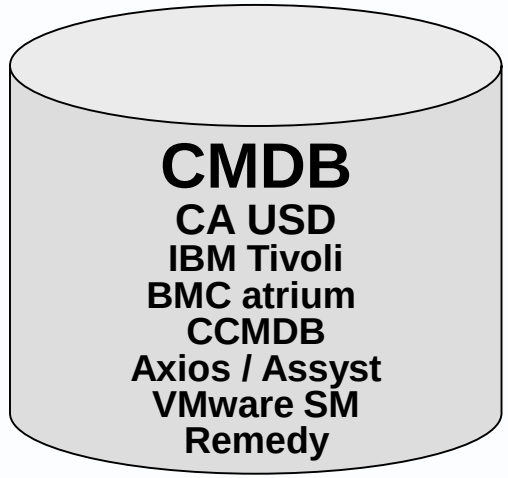
- **IT4IT**
- A product of The Open Group that applies enterprise architecture principles to ITSM.
- It defines a value chain with four primary value streams.
- It decomposes each value stream into functions and defines artifacts that are produced by and exchanged between functions.

Recording infrastructure

Avancier

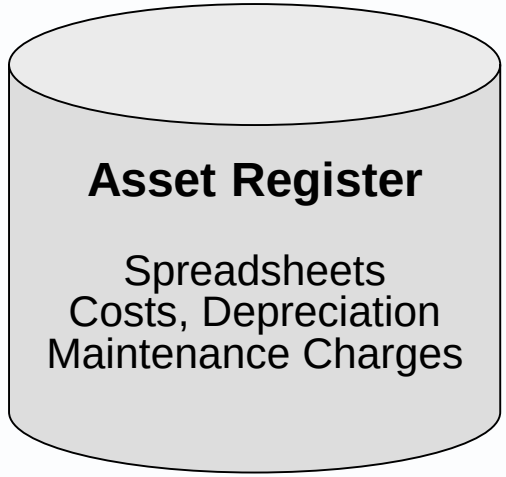
CMDB: Configuration Management Database

- [A data store] a record of configuration item specifications including relationships among configuration items, where the items are significant to ITSM.

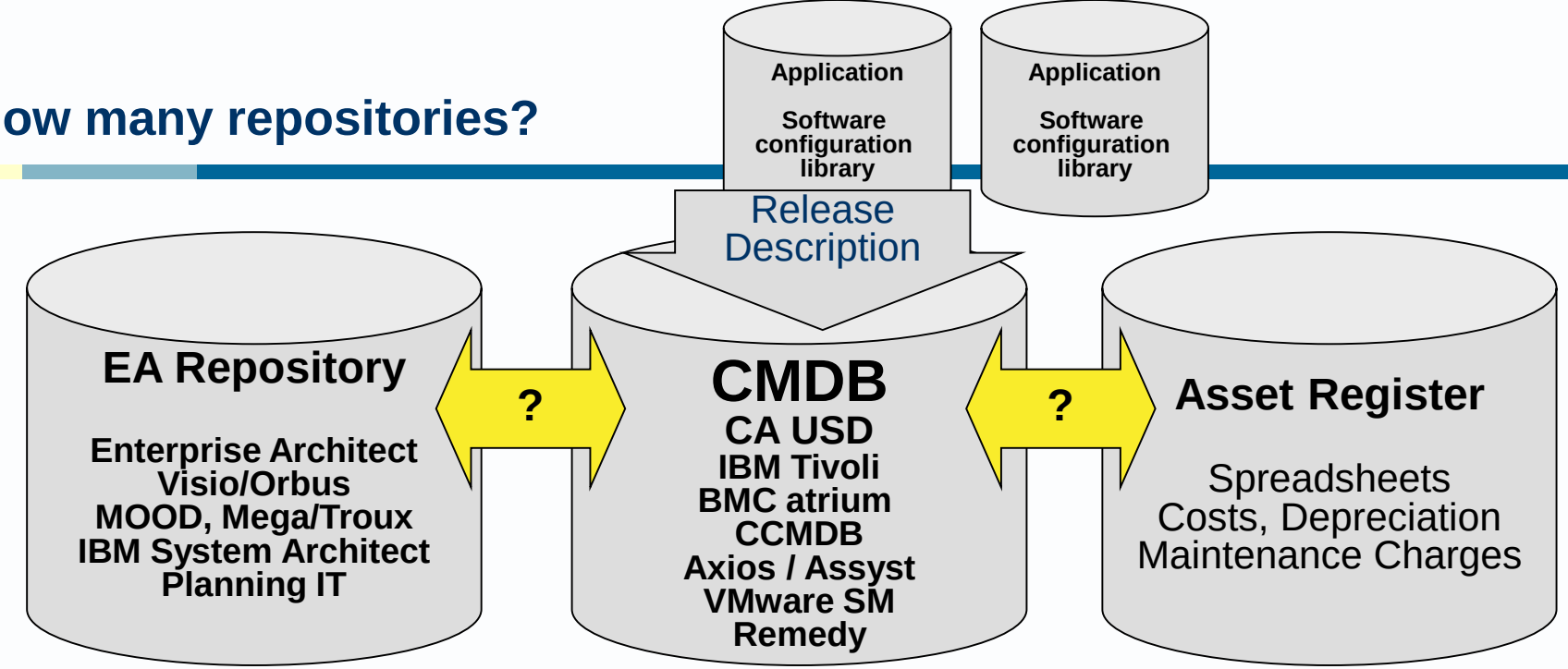


Asset management system

- [A data store] a record of IT assets.
- It is sometimes use to record end user devices, outside of the data centre.
- It may be related to a CMDB.



How many repositories?



- In theory at least, all these configurations are related

- “Integrating IBM Rational System Architect with other IBM Rational solutions and solution delivery products such as IBM Tivoli CCMDB can provide vital support for this pragmatic approach.
- It facilitates efficient creation and maintenance of the IT architecture by importing auto-discovered application and technology portfolios.”

Capability Maturity

Capability Maturity

Capability maturity model

- [A reference model] for evaluating the maturity of an organisation and its processes.
- The first was the capability maturity model (CMM) for software processes.
- The capability maturity model integration (CMMI) widened CMM to cover other processes.
- There are now various maturity models for architecture processes.

- See also the GAO EA management maturity framework.
- http://www.gao.gov/new.items/d03584g.pdf

The standard levels in a “maturity model”

Avancier

Formally

5 Optimizing: Focus on process improvement
4 Managed: Process measured and controlled
3 Defined: Process characterized, fairly well understood.
2 Repeatable: Can repeat previously mastered tasks
1 Initial: Unpredictable and poorly controlled
0 Missing: Does not exist

Processes are

5 – continuously improved
4 – measured
3 – documented & followed
2 – repeated
1 – ad hoc (hero-level)
0 - irrelevant

Department of Commerce: ACMM levels and elements

Avancier

6 maturity levels

- 1. None
- 2. Initial
- 3. Under development
- 4. Defined
- 5. Managed
- 6. Measured

9 architecture characteristics:

- 1. IT architecture process
- 2. IT architecture development
- 3. Business linkage
- 4. Senior management involvement
- 5. Operating unit participation
- 6. Architecture communication
- 7. IT security
- 8. Architecture Governance
- 9. IT investment and acquisition strategy

2 kinds of maturity rating.

- a weighted mean IT architecture maturity level.
- the percentage achieved at each maturity level for the nine architecture characteristics.