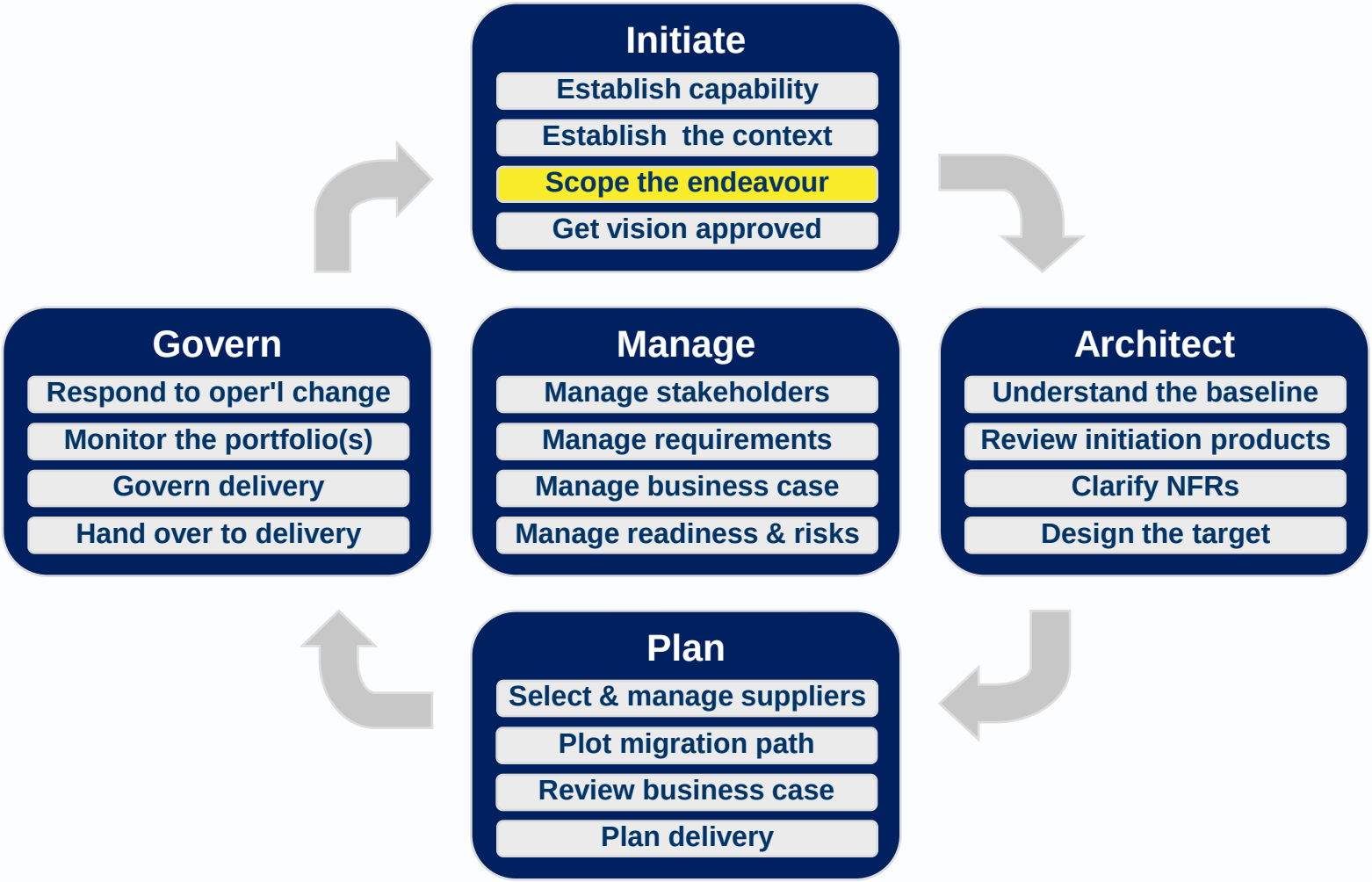
Avancier Methods (AM)

INITIATE

Scope in several ways

Scope the Endeavour (AM level 2)



Scope the Endeavour (AM level 3)

Avancier

1. Identify stakeholders
2. Identify aims
3. Identify constraints
4. Agree a solution vision
5. Scope in several ways
6. Plan the “architecture project”

Detailed in methods & training

Scope the Endeavour

Avancier

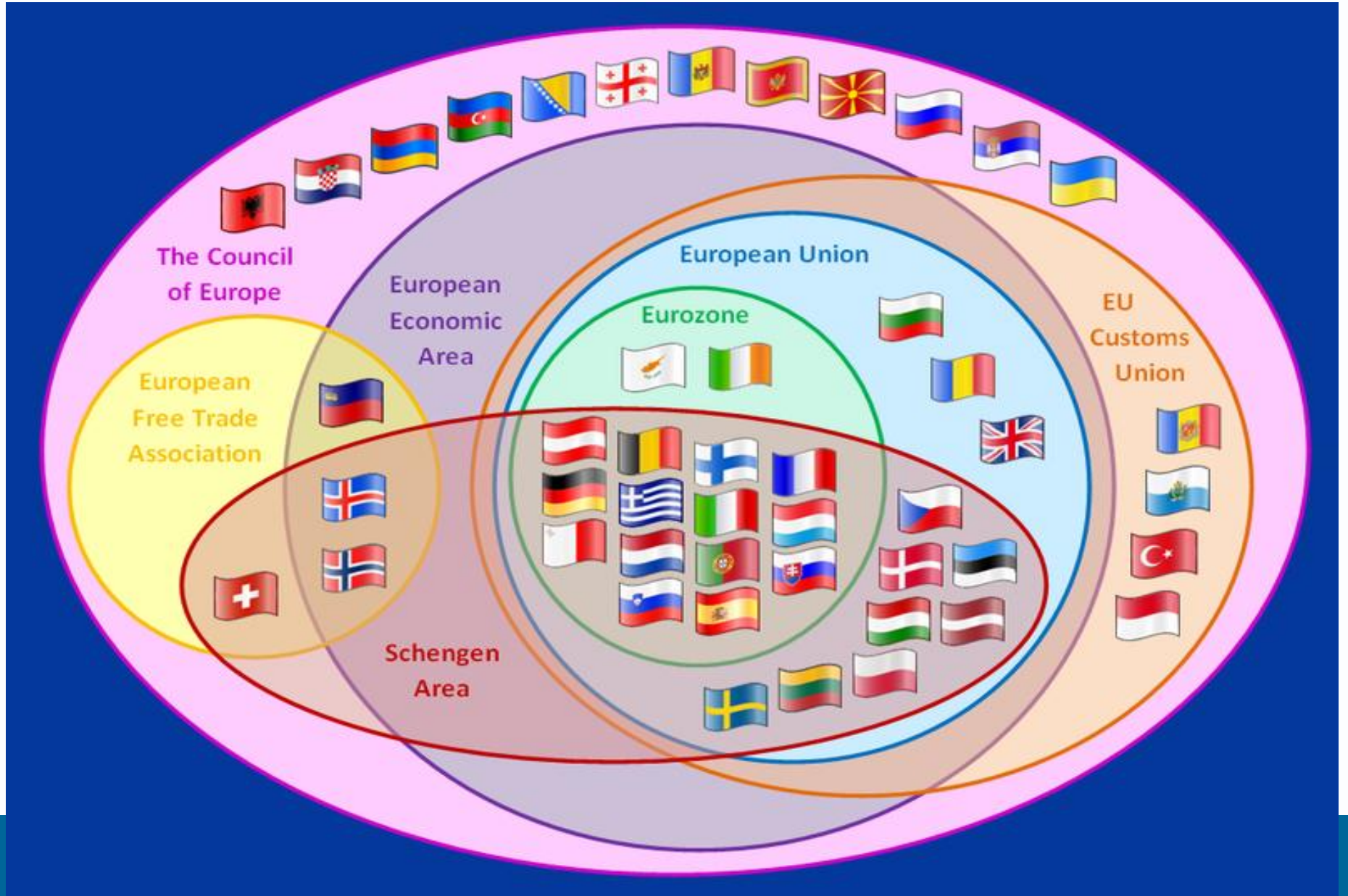
- Define three dimensions

Breadth	Constraints	Depth
Size & complexity of System/project Large / Medium / Small	**Time/resources to describe the system/project** Little / Moderate / Lots	**Level of detail reachable in descriptions/plans**
Large	**Little**	**Vacuous**
Medium	**Little**	**Sketchy**
Large	**Moderate**	**Sketchy**
Medium	**Moderate**	**Elaborate**
Small	**Little**	**Elaborate**
Large	**Lots**	**Elaborate**
Small	**Moderate**	**Fulsome**
Medium	**Lots**	**Fulsome**
Small	**Lots**	**Complete**

You can’t have scope creep until you know what the scope is

Avancier

2009?



Several ways to look at system breadth

Avancier

Requirements-oriented

What are stakeholders' aims?
Requirements catalogue

What do users want?
Services

What does it do?
Processes

	Behaviour	Structure
External	Service	Interface
Internal	Process	Component

What does it consume and produce?
Context diagram
Interface definition

What is it made of?
Components
Data Structures

Design-oriented

Aims

What are stakeholders' aims?
Requirements catalogue

What do users want?
Services

What does it do?
Processes

	Behaviour	Structure
External	Service	Interface
Internal	Process	Component

What does it consume and produce?
Context diagram
Interface definition

What is it made of?
Components
Data Structures

What are the stakeholder's aims? Requirements catalogue

- Here, a subset of a more extensive template

Goal/Objective/Requirement catalogue entry	
Identifier	Must be static throughout the aims' lifecycle
Description	Use plain English. Avoid expressing as a solution
Type	e.g. business goal, functional, non-functional, audit, legal
Owner	The sponsor of the aim
Priority	e.g. MoSCoW or High/Medium/Low
Source	Where was the aim identified
Author	Who wrote the aim
SMART = Specific, Measurable, Actionable, Realistic, Time-bound	
Measure	How will you measure success? Acceptance criteria?
Action	What actions are needed to succeed?
Timing	When must the aim be met?

Context / interface

What are stakeholders' aims?

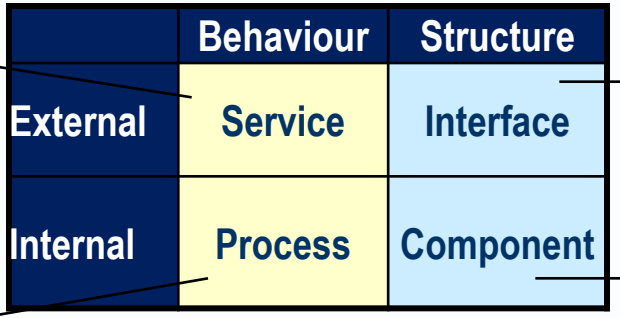
Requirements catalogue

What do users want?

Services

What does it do?

Processes



What does it consume and produce?

Context diagram

Interface definition

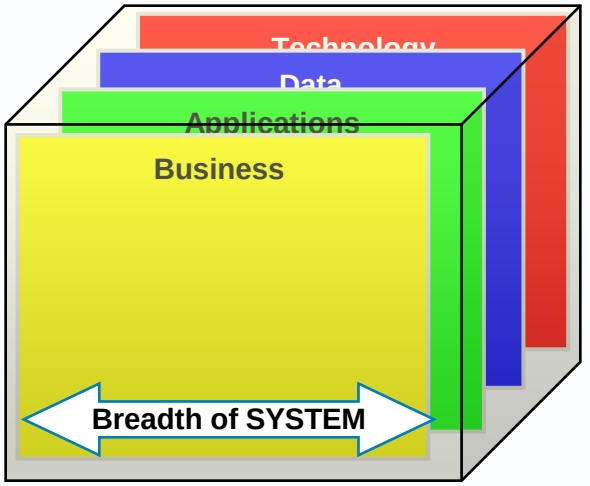
What is it made of?

Components

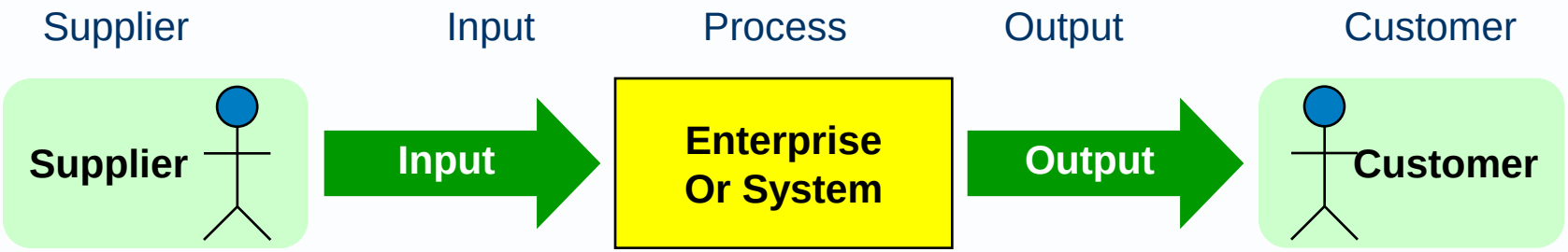
Data Structures

Context diagram

- [an artefact] that shows a system's scope in terms of
  - inputs consumed ,
  - outputs produced, and
  - the external entities (actors and/or roles) that send inputs and receive outputs.
- The system is shown as a 'black box'.

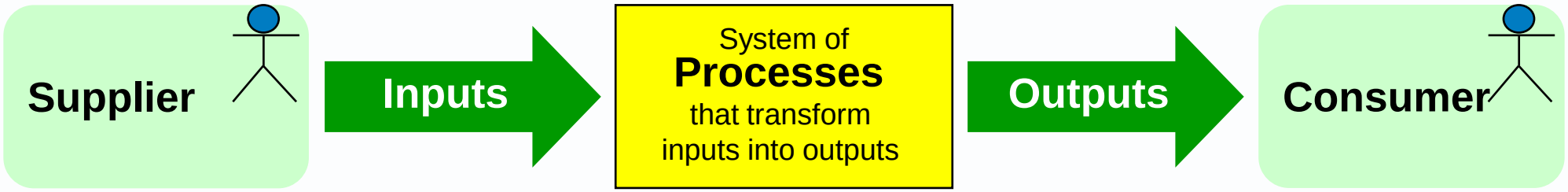


Cf."SIPOC" in Six Sigma



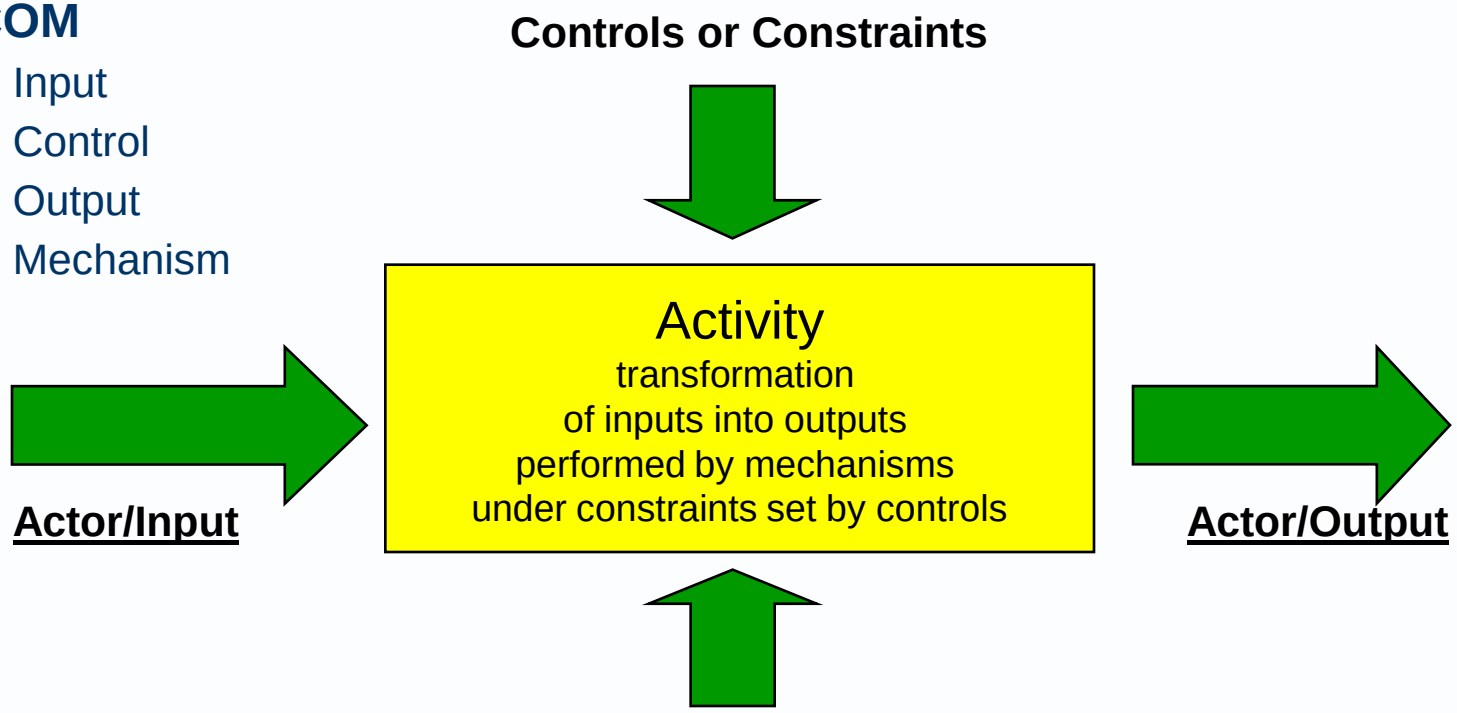
SIPOC (Six Sigma)

- Shows why the system exists
- Does not show what the system is made of
  - The external view of a system is an abstraction from
  - The internal components and processes



IDEF0 Business Modelling notation

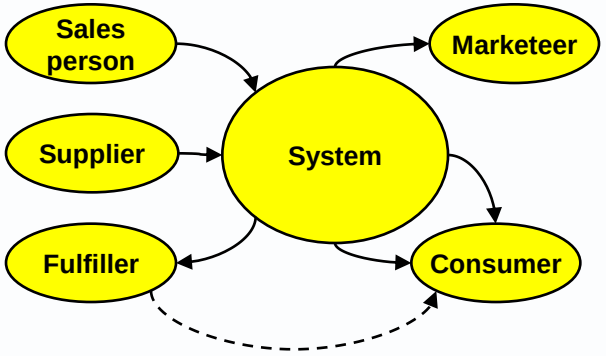
- **ICOM**
  - Input
  - Control
  - Output
  - Mechanism



- **Cf. SIPOC**
  - Supplier
  - Input
  - Process
  - Output
  - Consumer

The architect must start with the numbers!

- Capture business capacity and performance measurements

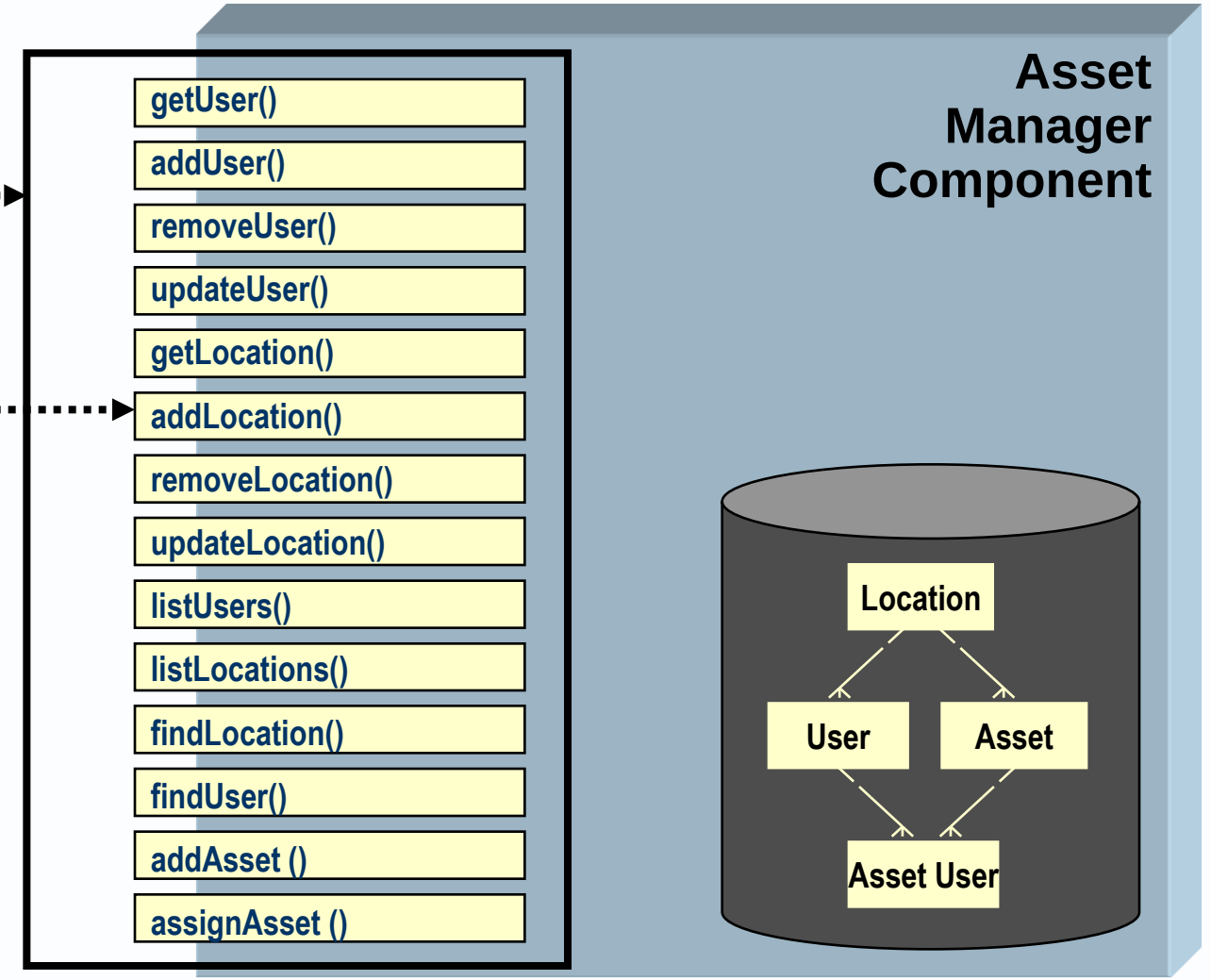


- For each input and output
  - Throughput, w peaks and troughs
  - Duration / response time
  - Availability (24*7?)
  - Time constraints (before 4.00 p.m.)
  - Resource cost
  - Etc.

- Run time service level measures
  - How many I/O per minute?
  - % I/Os completed OK
  - 5 Most common I/O
  - 5 Least common I/O
  - 5 Worst performing I/O
  - 5 Best performing I/O

What is an interface?

- Interface
- Service



A“coarse-grained” interface to an ITSM organisation

- User Management services
  - Allocate new desktop computer
  - Provide new office phone and voicemail
  - Set up conference room.
- Configuration Management (CM) services
- Performance Management services
- Availability and Fault Management services
- Accounting Management services
- Security Management services (Identity and Directory)
  - Add employee to identity management system
  - Assign employee to role
  - Enable access to SAP Financials system
- Print Management services
- Network Management services
- Backup and Restore services
- Online Disk Management services
- License Management services
- Capacity Management services (Power and Storage)
- Software Installation services
- Trouble Ticketing services

May be presented in a SLA doc

May be presented in a GUI

Very “fine-grained” interface to an Active Directory

- Return Username from Email
- Return Email from Username
- Return Domains
- Return Groups
- Does User belong to Group
- Search Usernames
- Search Display Names
- Authenticate

File Transfer Protocol (FTP) – an interface – highlighting one service

Avancier

FTP	An interface implemented by a platform component whose role is to copy files to and from computers. The services below are expressed as in the common FTP utility program on a UNIX computer.
Service name	**Summary description of service contract**
?	to request help or information about the FTP commands
ascii	set the mode of file transfer to ASCII
bye	exit the FTP environment (same as quit)
cd	change directory on the server computer
close	terminate a connection with another computer
delete	delete (remove) a file in the current remote directory (same as rm in UNIX)
get ABC DEF	copies file ABC in the current remote directory to a file named DEF in your current local directory.
get ABC	copies file ABC in the current remote directory to a file with the same name, in your current local directory.
help	request a list of all available FTP commands
mget	copy multiple files from the server computer to the client computer; you are prompted for a y/n answer before transferring each file
mput	copy multiple files from the client computer to the server computer; you are prompted for a y/n answer before transferring each file
open	open a connection with another computer
put	to copy one file from the client computer to the server computer
quit	exit the FTP environment (same as bye)
rmdir	to remove (delete) a directory in the current remote directory

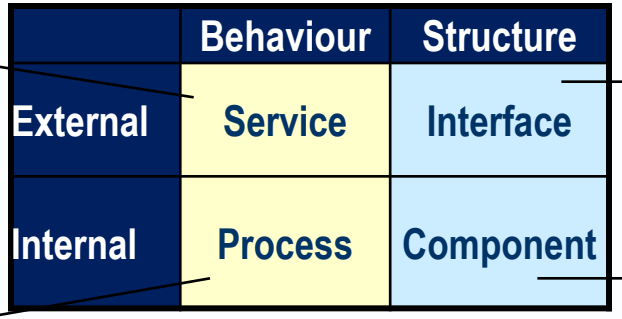
Services

What are stakeholders' aims?

Requirements catalogue

What do users want?

Services



What does it consume and produce?

Context diagram

Interface definition

What does it do?

Processes

What is it made of?

Components

Data Structures

Service contract example

Avancier

Service contract for FTP “get” operation		Values
Signature	Name	get
	Inputs	Remote file name Local file name
	Outputs or results	Reply = OK or Fail (see post conditions)
Semantics or rules	Preconditions - the state of the system in which the event is allowed	Remote computer can be reached. Remote file exists in the current remote directory.
	Post conditions - the state of the system after the event is complete	Remote file copied to (or on top of) local file current local directory.
Non-functionals	Response time	30 seconds
	Throughput	20 per minute
	Availability	99.99%
	Integrity	100% perfect file copy
	Scalability	Up to 100 per minute
	Security	No encryption
	Serviceability	
	Etc. Other non-functionals, dependencies and commercials.	

What is a service?

- Encapsulates a requestable process
- Definable in a contract without details of the process flow

Service Contract	Business Service	999
Signature	Name	Haircut
	Input	Hair length
	Output	Shorter hair (see also post conditions)
Semantics or rules	Preconditions	Barbershop open and barber ready
	Post conditions	Money received. Resource wear.
Non-Functional Requirements	Response time	45 minutes
	Throughput	6 per hour per shop
	Availability	90% waiting times less than 20 minutes from 09.00 to 17.00

What is a service contract?

Avancier

Service contract

The signature, semantics and non-functional characteristics of a service.

The **signature** is what a client needs to invoke a service - composed of a name, inputs (arguments) and outputs.

The **semantics** are what a client designer needs to know of what the service does - composed of its preconditions and post conditions.

The **non-functional characteristics** are what a client designer needs to know of the conditions under which the service works, which includes both performance and commercial conditions.

Service Contract	Business Service	999
Signature	Name	Haircut
	Input	Hair length
	Output	Shorter hair (see also post conditions)
Semantics or rules	Preconditions	Barbershop open and barber ready
	Post conditions	Money received. Resource wear.
Non-Functional Requirements	Response time	45 minutes
	Throughput	6 per hour per shop
	Availability	90% waiting times less than 20 m from 09.00 to 17.00

Processes

What are stakeholders' aims?

Requirements catalogue

What do users want?

Services

What does it do?

Processes

	Behaviour	Structure
External	Service	Interface
Internal	Process	Component

What does it consume and produce?

Context diagram

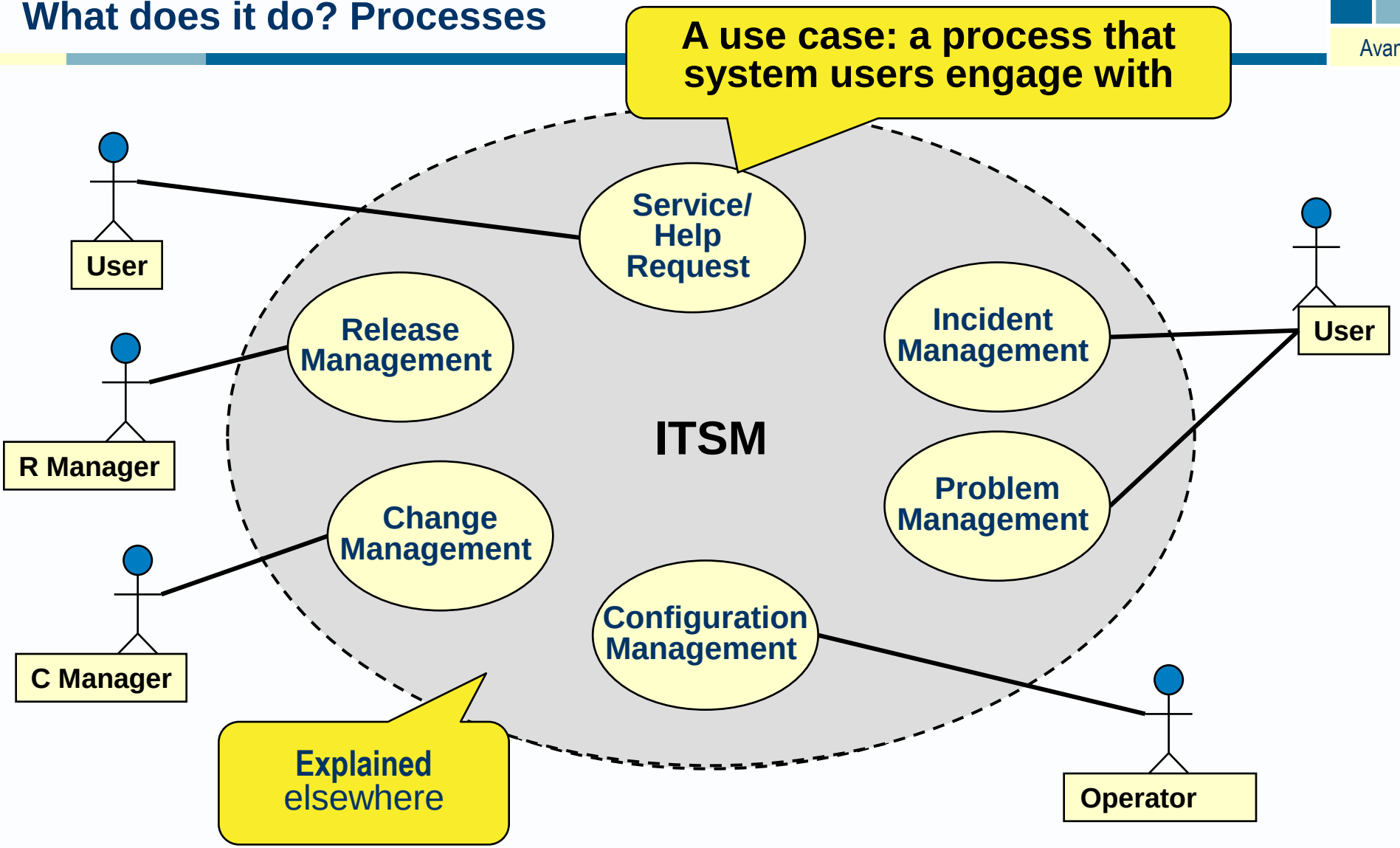
Interface definition

What is it made of?

Components

Data Structures

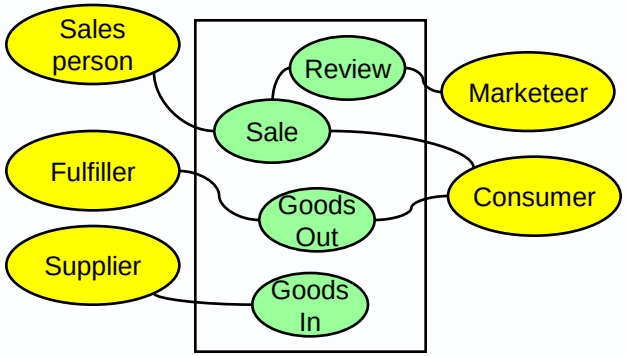
What does it do? Processes



Don’t forget the numbers – which inform NFRs

Avancier

- Capture business capacity and performance measurements



- For each use case process:
  - Throughput, w peaks and troughs
  - Duration / response time
  - Availability (24*7?)
  - Time constraints (before 4.00 p.m.)
  - Resource cost
  - Etc.

- Run time service level measures
  - How many use cases per hour
  - % use cases completed in target time
  - 5 Most popular use cases
  - 5 Least popular use cases
  - 5 Worst performing use cases
  - 5 Best performing use cases

Components

What are stakeholders' aims?

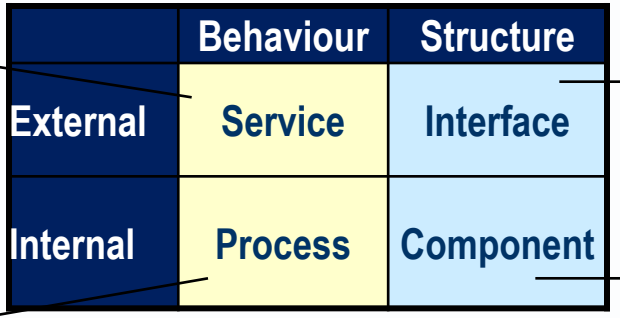
Requirements catalogue

What do users want?

Services

What does it do?

Processes



What does it consume and produce?

Context diagram

Interface definition

What is it made of?

Components

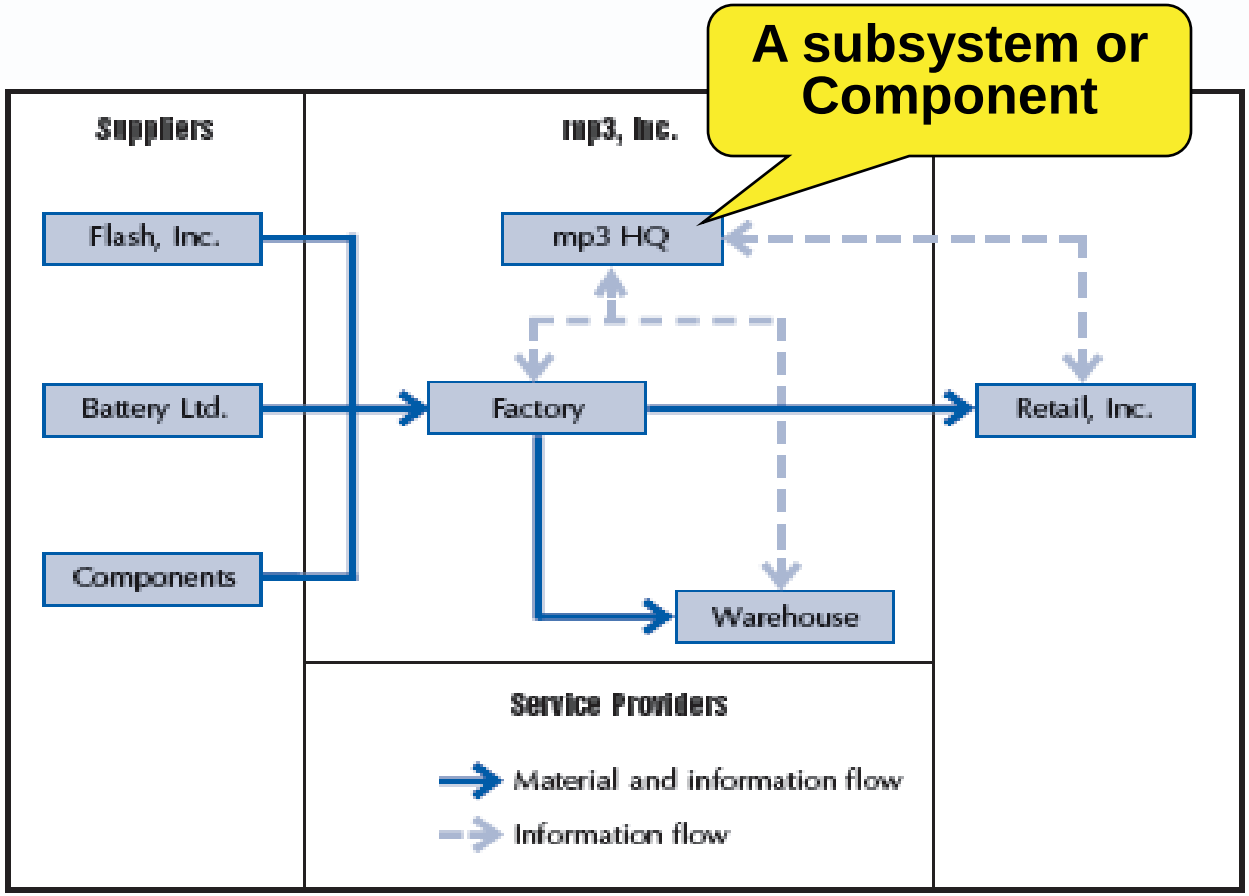
Data Structures

What is it made of? Components

Avancier

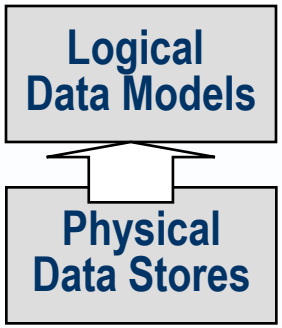
A scoping technique (as published by SCOR)

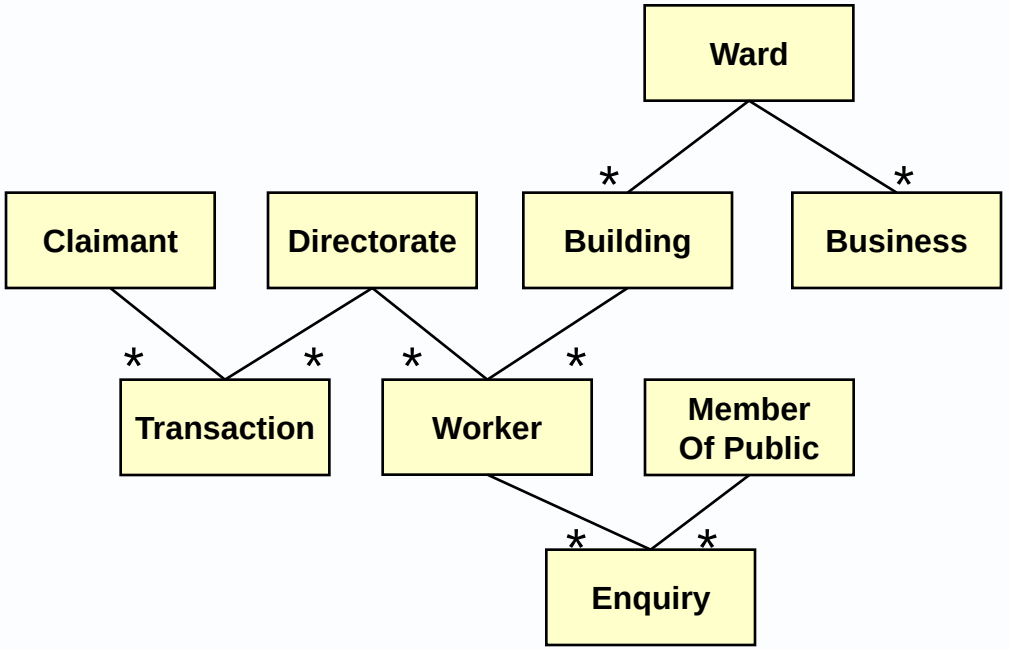
1. For your system or project
2. Identify your **Consumers**
3. Identify your **suppliers**
4. Identify the **key nodes**
   - Logical or geographic entity in supply chain:
   - Warehouse, Factory, Store, HQ etc.)
5. Link nodes using a different color and/or stroke to differentiate material and information flows.



What is it made of? Data entities (UML notation)

- Best to compare at the level of logical data model





Scope the breadth in several ways

Avancier

What are stakeholders' aims?
Requirements catalogue

What do users want?
Services

What does it do?
Processes

	Behaviour	Structure
External	Service	Interface
Internal	Process	Component

What does it consume and produce?
Context diagram
Interface definition

What is it made of?
Components
Data Structures

2nd dimension: constraints

Avancier

You can only do what you have time, money and resources to do

Three dimensions of scope		
Breadth	**Constraints**	**Depth**
Size & complexity of system or project Large / Medium / Small	**Time & resources to describe the system or project** Little / Moderate / Lots	**Level of detail reachable in descriptions or plans**
Large	Little	Vacuous
Medium	Little	Sketchy
Large	Moderate	Sketchy
Medium	Moderate	Elaborate
Small	Little	Elaborate
Large	Lots	Elaborate
Small	Moderate	Fulsome
Medium	Lots	Fulsome
Small	Lots	Complete

3rd dimension: depth

Architect's identify the major costs and risks, then to address those at whatever level of detail is necessary

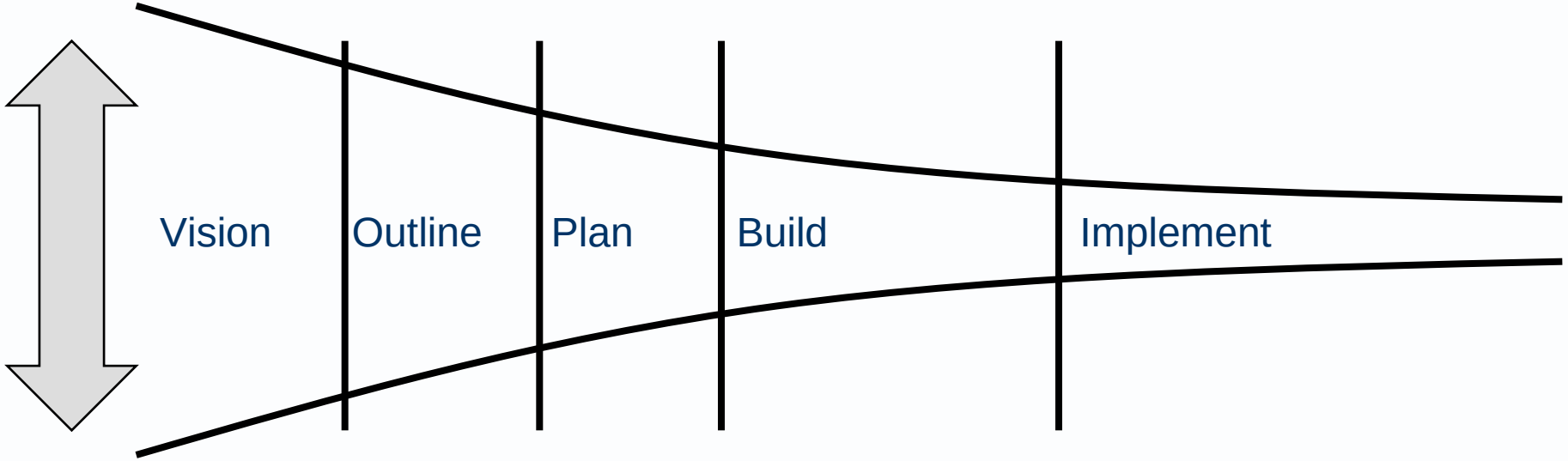
Avancier

Getting the level of detail right is a huge challenge

Three dimensions of scope		
Breadth	**Constraints**	**Depth**
Size & complexity of system or project Large / Medium / Small	**Time & resources to describe the system or project** Little / Moderate / Lots	**Level of detail reachable in descriptions or plans**
Large	Little	Vacuous
Medium	Little	Sketchy
Large	Moderate	Sketchy
Medium	Moderate	Elaborate
Small	Little	Elaborate
Large	Lots	Elaborate
Small	Moderate	Fulsome
Medium	Lots	Fulsome
Small	Lots	Complete

How far *should* an architecture description be refined?

- Until the cone of uncertainty has narrowed sufficiently that
  - stakeholders understand the benefits, costs and risks
  - a decision to invest in the next stage can be made.



- Focus early on costs and risks associated with NFRs.
- Analysts complete functional requirements incrementally

Scope completion

- Stop when the architecture
  - bounds the scope of the solution
  - helps to settle the time and costs of solution delivery
  - and the identified risks are accepted
- The cone of uncertainty is sufficiently narrowed
- There is no defined level of detail.
  - An architect may focus on 3 critical use cases
  - And leave another 30 user cases to others

Lessons learned

- Scope can only be controlled to the extent it is known and agreed.
- Scoping documents are always - necessarily - an abstraction.
- They hide things not known to Customer and/or Supplier.
- And things not agreed between Customer and/or Supplier.
- No requirements catalogue or outline solution document ever defined scope so clearly there was no room for argument over what is in and out of scope.

- Mutual trust and goodwill helps a lot.

4th dimension: Perspective or domain

You may focus on one or two perspectives, rather than all

OK if the change to be made has no impact the other architecture perspectives

Or the other perspective is to remain stable anyway

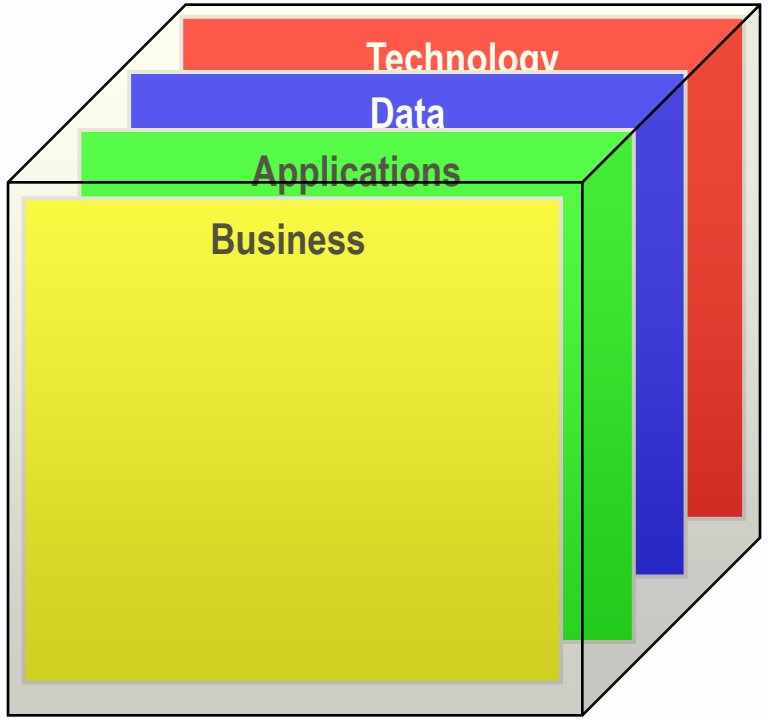
(e.g.

Re-hosting

Replace MS exchange

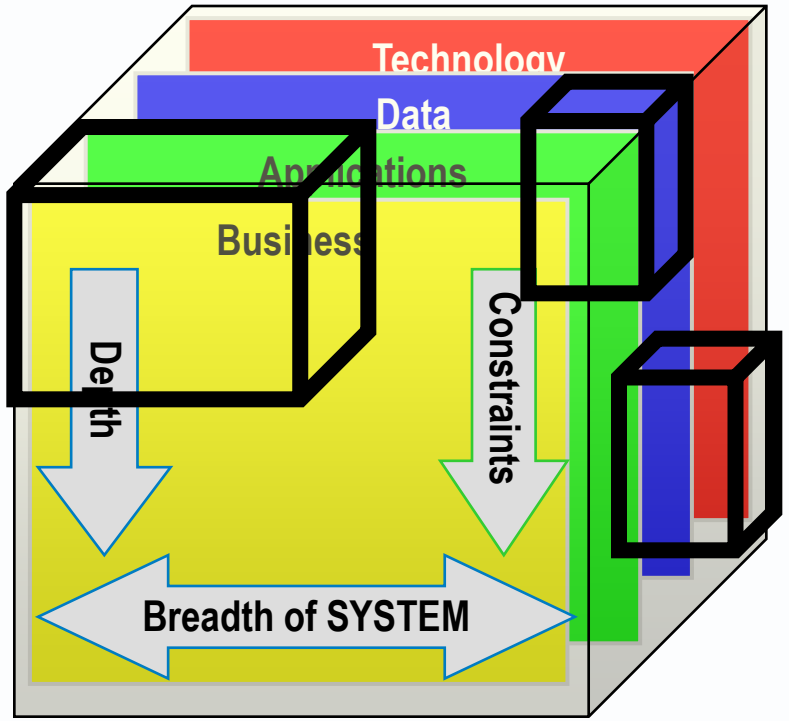
Apply patches

Server consolidation)



Two architectures can be wholly incomparable!

- High-level business architecture?



- Data quality improvement
- Or low-level technology architecture?
  - DR
  - Server consolidation

Always look for impacts across the domains!