Dr. Hans-Peter Hoidn
Distinguished IT Architect (Open Group certified)

Enterprise IT Architectures

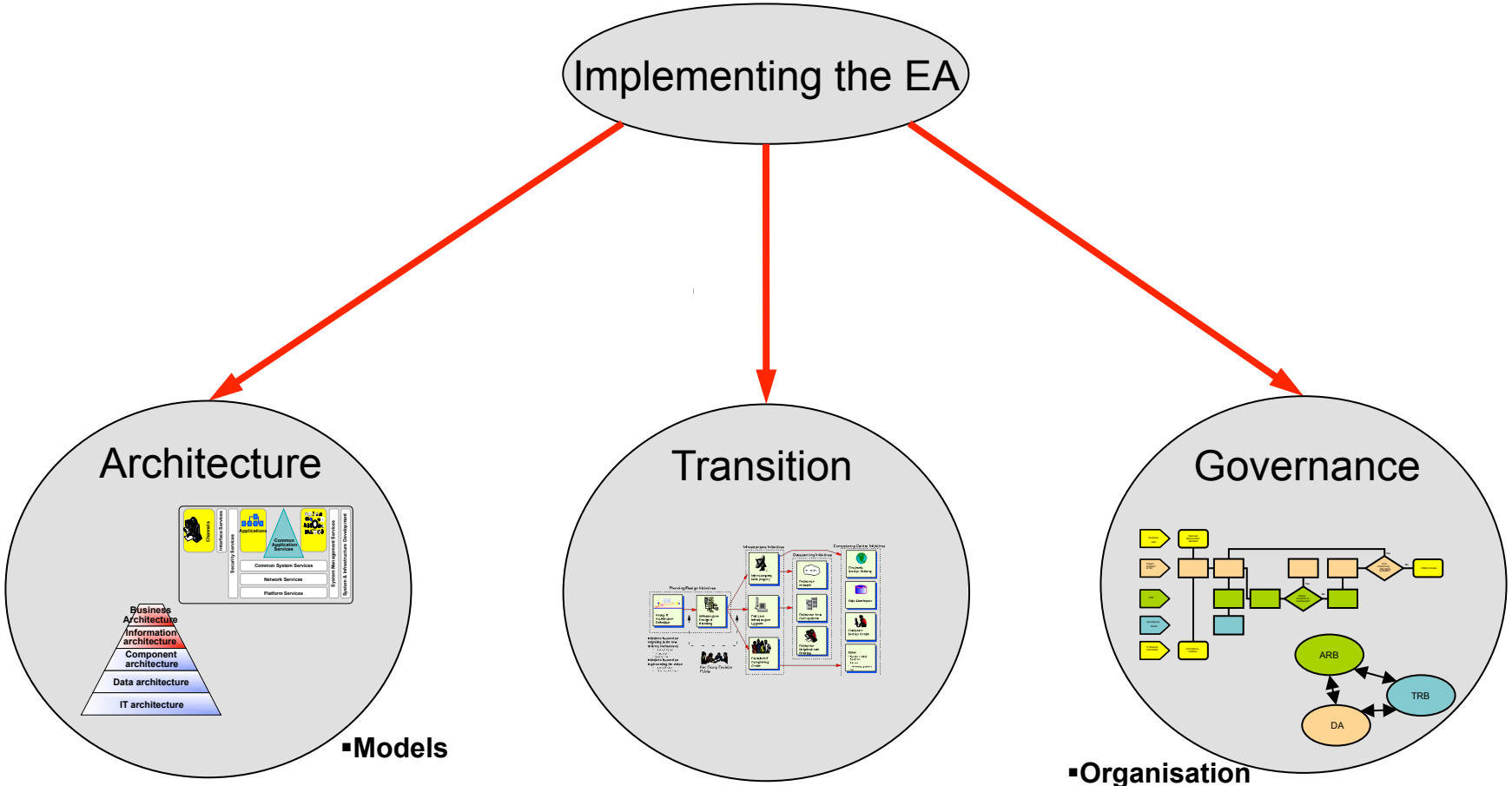
Governance, Policies / Principles
Specific Topic: ESB versus Broker

Agenda of Governance (50 Minutes)

- Governance
- SOA Governance
- Policies / Principles
- Followed by Specific Topic : ESB versus Broker

Governance

Governance – One of the 3 aspects of Enterprise Architecture



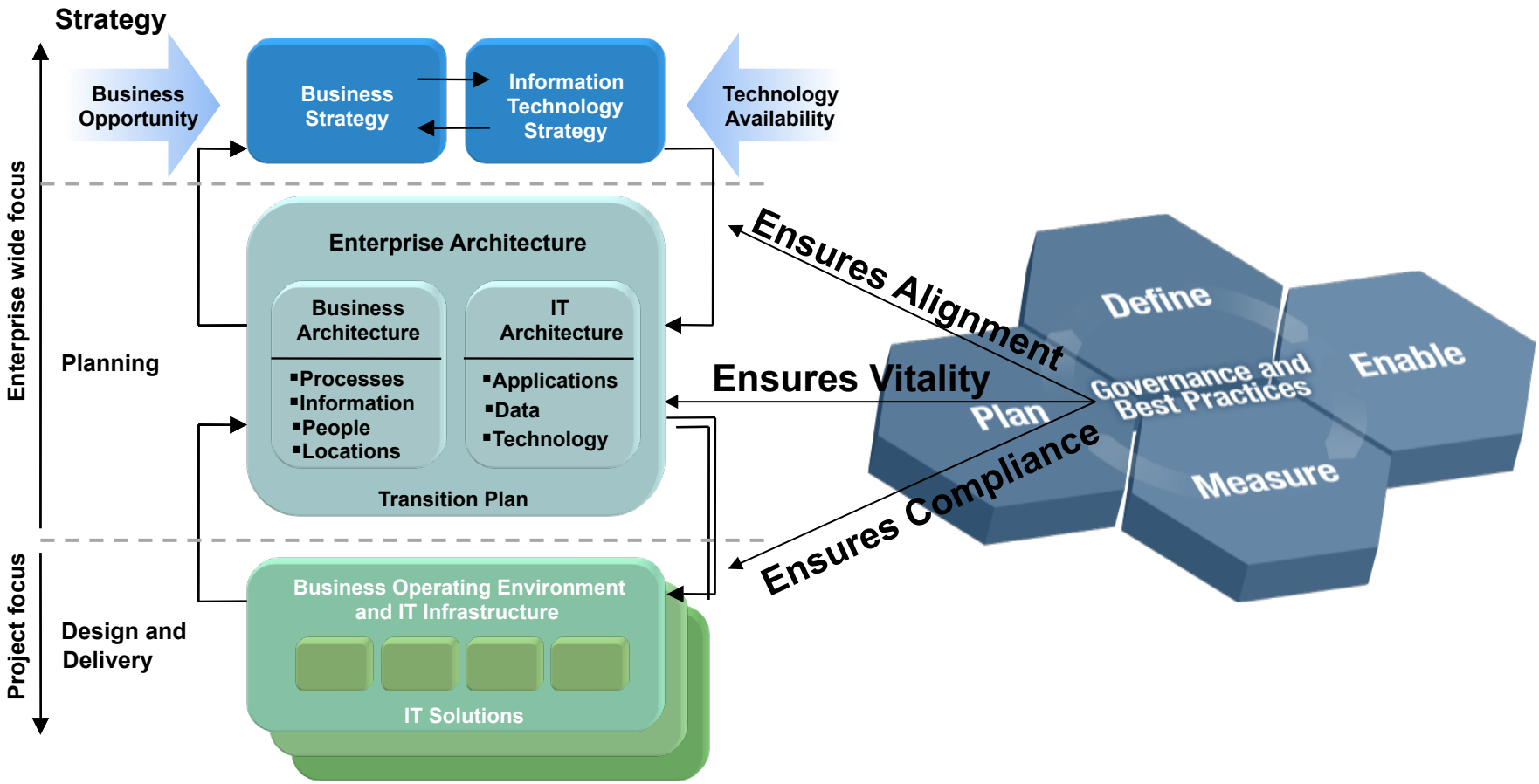
- Models
- Principles
- Enterprise Tech Framework

- Transition Plan
- Management Action Plan

- Organisation
- Processes
- Evaluation & Selection

Source: IBM

Governance – Keeps Enterprise Architecture alive



Source: IBM

Governance: Introduction Comment

- Governance is
  - Roles and responsibilities, Processes
- Governance matters
  - Realizing business benefits, Business process flexibility
  - Improving “time to market”
  - Mitigating business risk
  - Improving team effectiveness by measuring the right things and communicating clearly between business and IT
- Hint:
  - Align Governance to “C”-management procedures
  - Act as Enabler – not Inhibitor

What is *Governance*?

Establishing **chains of responsibility, authority and communication to** empower **people (decision rights)**

Establishing **measurement, policy and control mechanisms to** enable **people to carry out their roles and responsibilities**

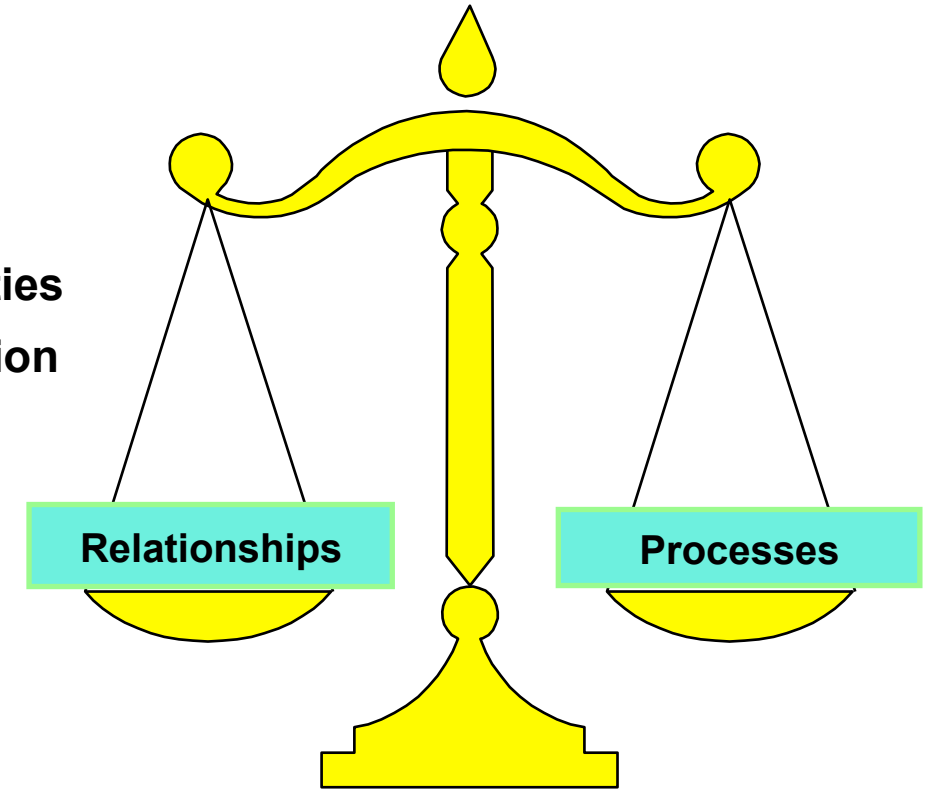
- Corporate Governance
- IT Governance
- EA Governance
- SOA Governance

Source: IBM

Governance involves balancing the key aspects of relationships and processes on a foundation of communication

THE WILL

- Trust
- Alignment
- Responsibilities
- Communication
- Behavior
- Culture



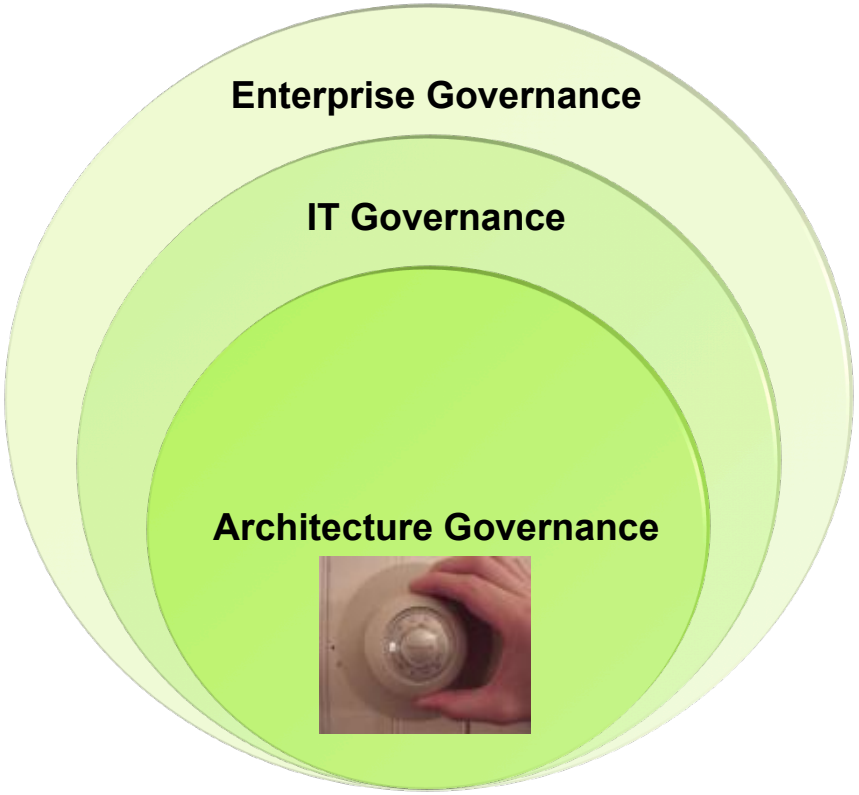
THE WAY

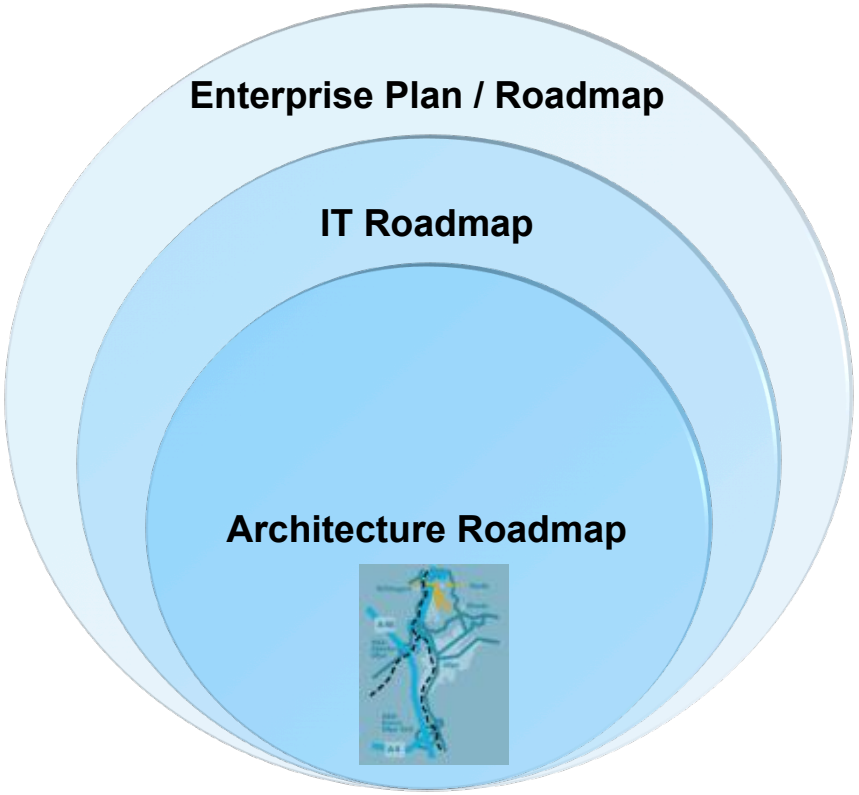
- Charter
- Organization
- Principles
- Processes
- Roles & Respon.
- Metrics
- Tools

Communication

Source: IBM

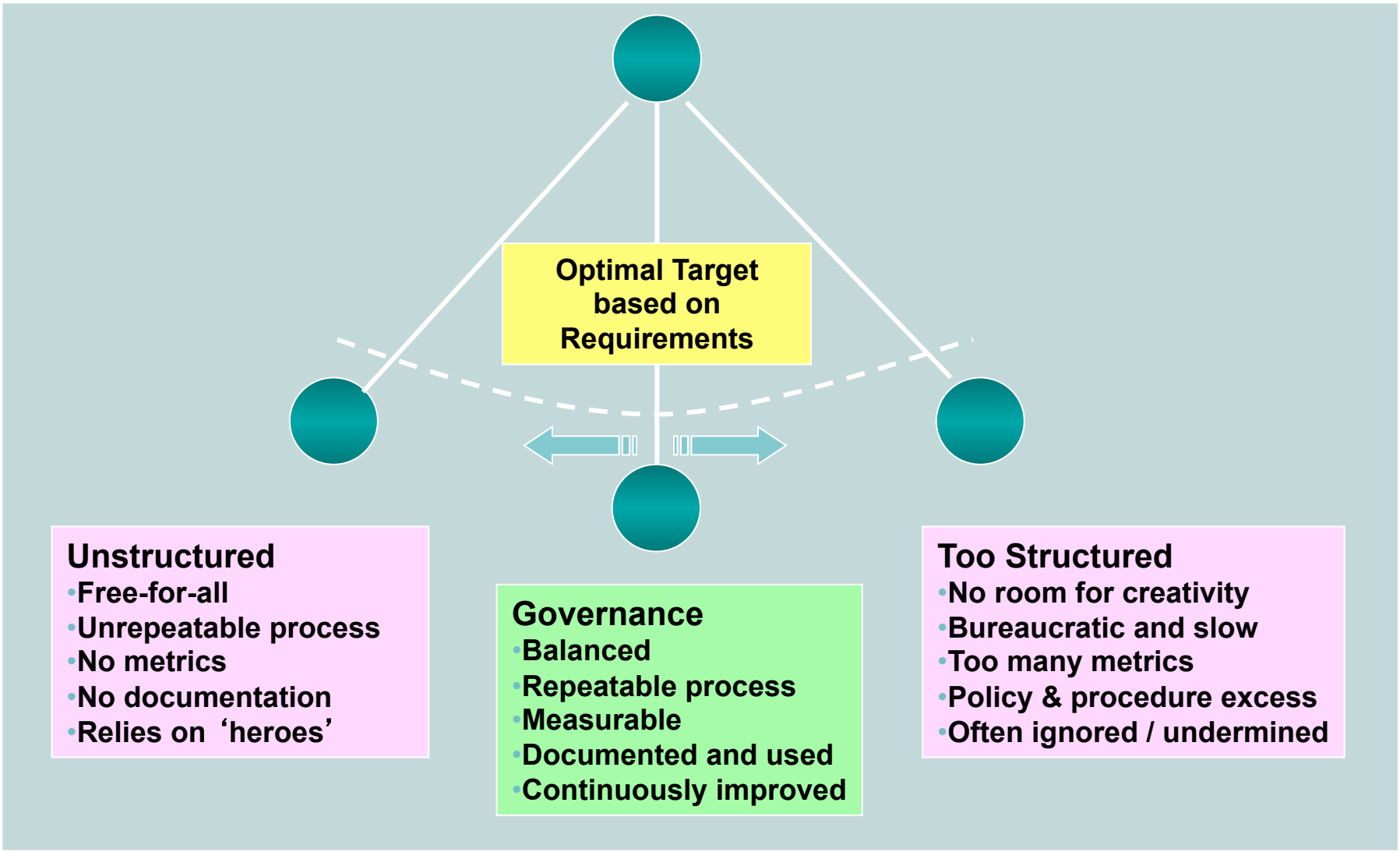
The *Governance* and *Roadmap* areas are nested in IT and Enterprise level work





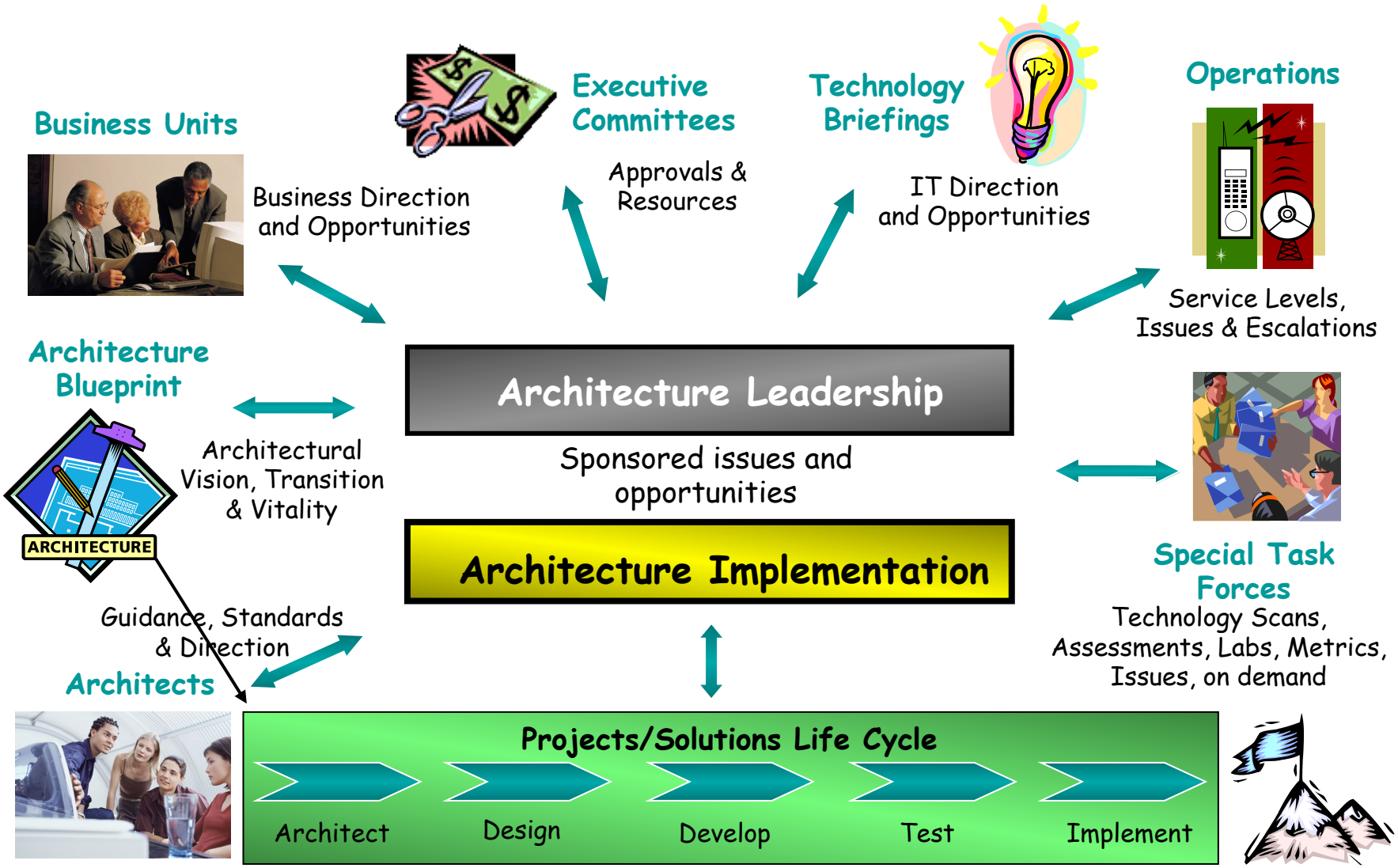
Source: IBM

Governance – A balanced approach



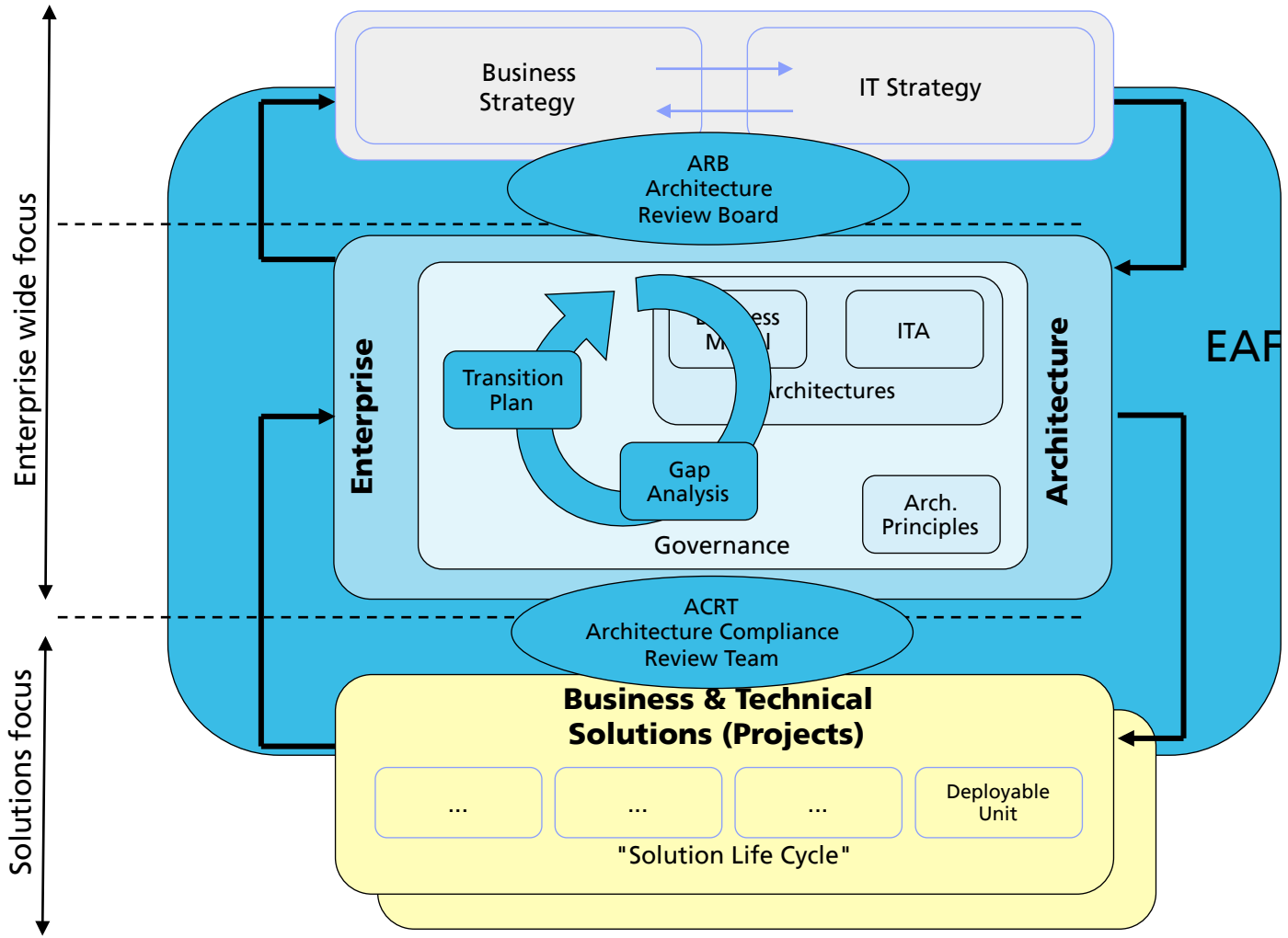
Source: IBM

Governance affects many decision-makers



Source: IBM

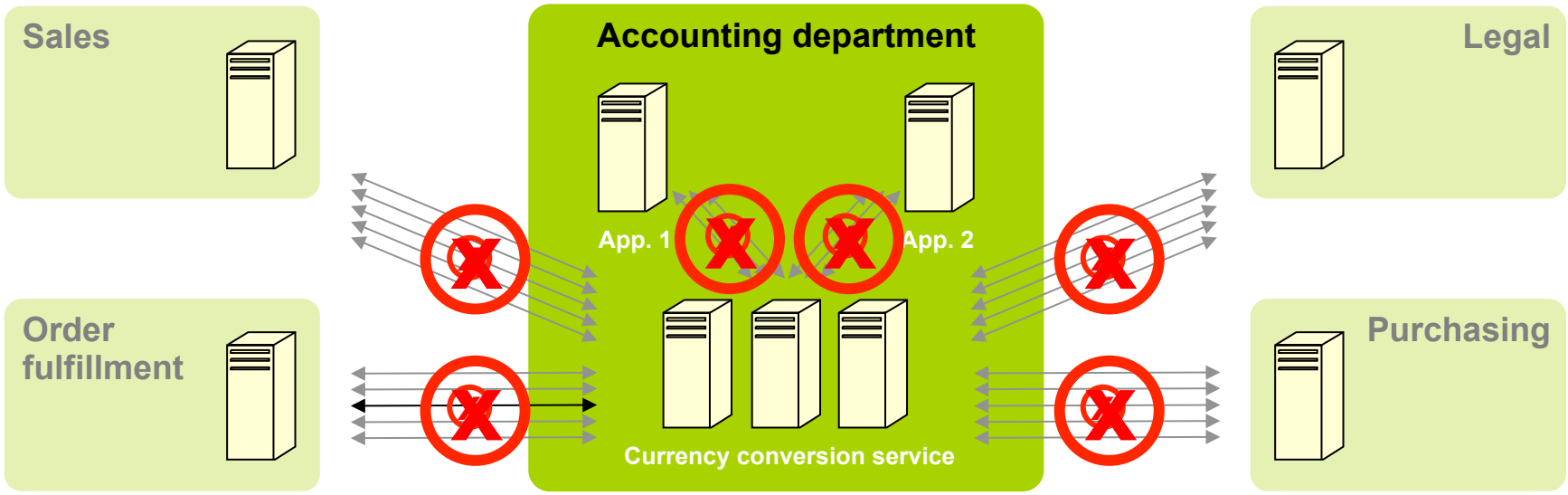
Governance – Organizational Context

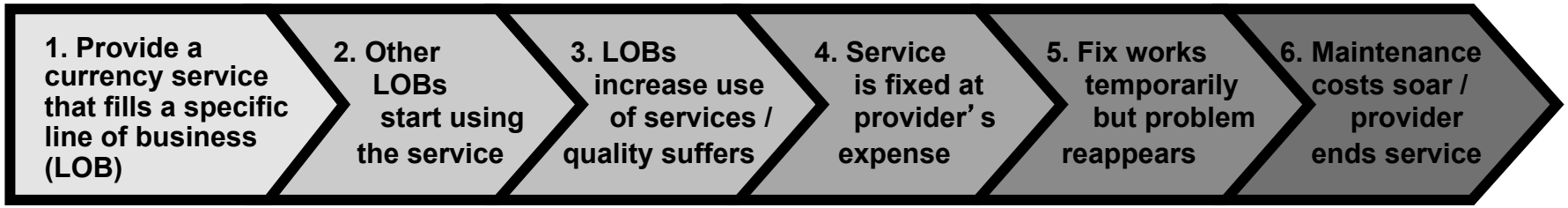


SOA Governance

Governance !

Importance by Illustrating Service Control



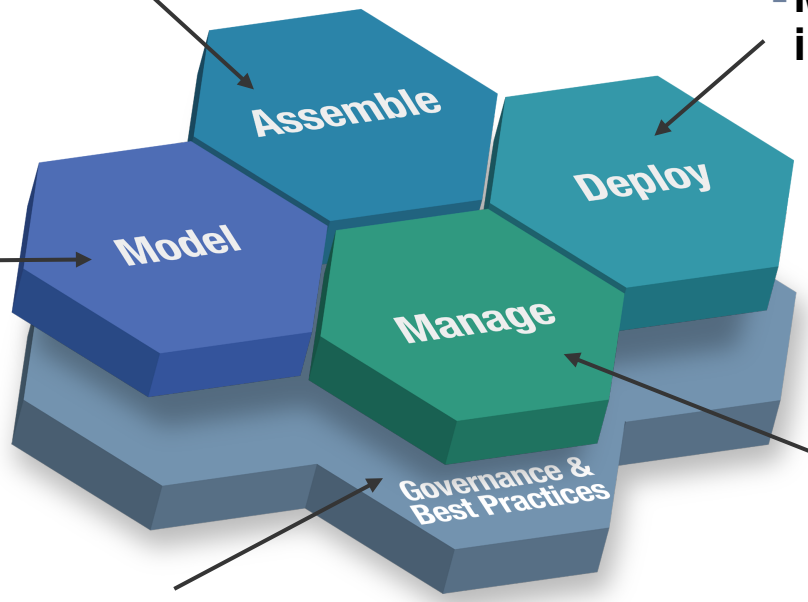


Source: IBM

Governance within the SOA Lifecycle

- Discover
- Construct & Test
- Compose

- Integrate people
- Integrate processes
- Manage and integrate information



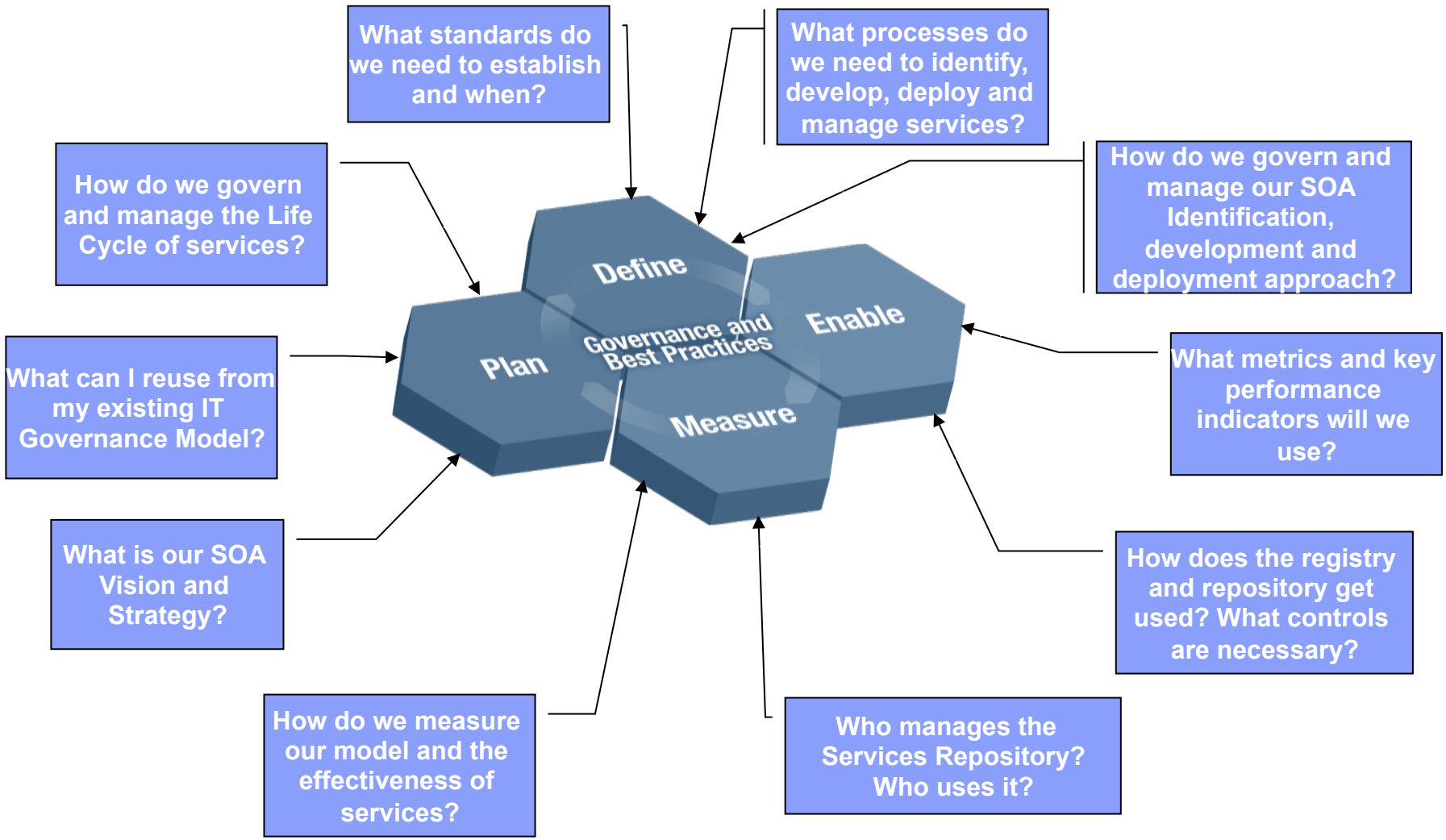
- Gather requirements
- Model & Simulate
- Design

- Manage applications & services
- Manage identity & compliance
- Monitor business metrics

- Financial transparency
- Business/IT alignment
- Process control

Source: IBM

SOA Governance Life Cycle Addresses Key Questions



Source: IBM

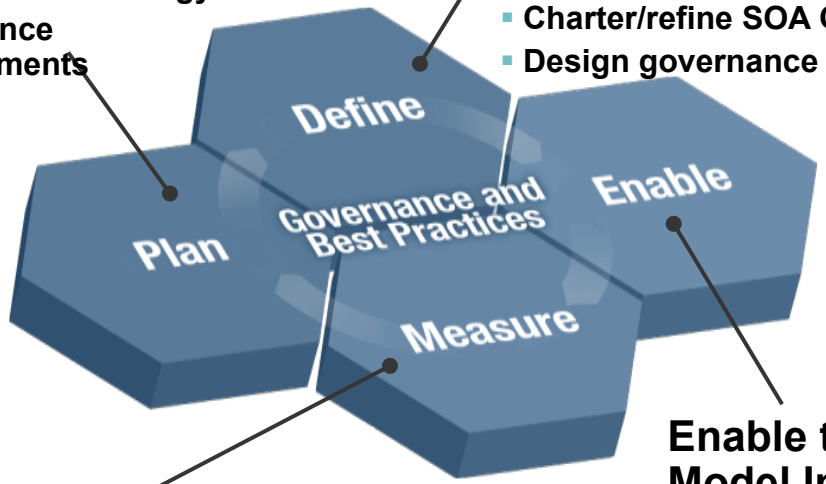
SOA Governance Lifecycle – How to establish?

Plan the Governance Need

- Document and validate business strategy for SOA and IT
- Assess current IT and SOA capabilities
- Define/Refine SOA vision and strategy
- Review current Governance capabilities and arrangements
- Layout governance plan

Define the Governance Approach

- Define/modify governance processes
- Design policies and enforcement mechanisms
- Identify success factors, metrics
- Identify owners and funding model
- Charter/refine SOA Center of Excellence
- Design governance IT infrastructure



Monitor and Manage the Governance Processes

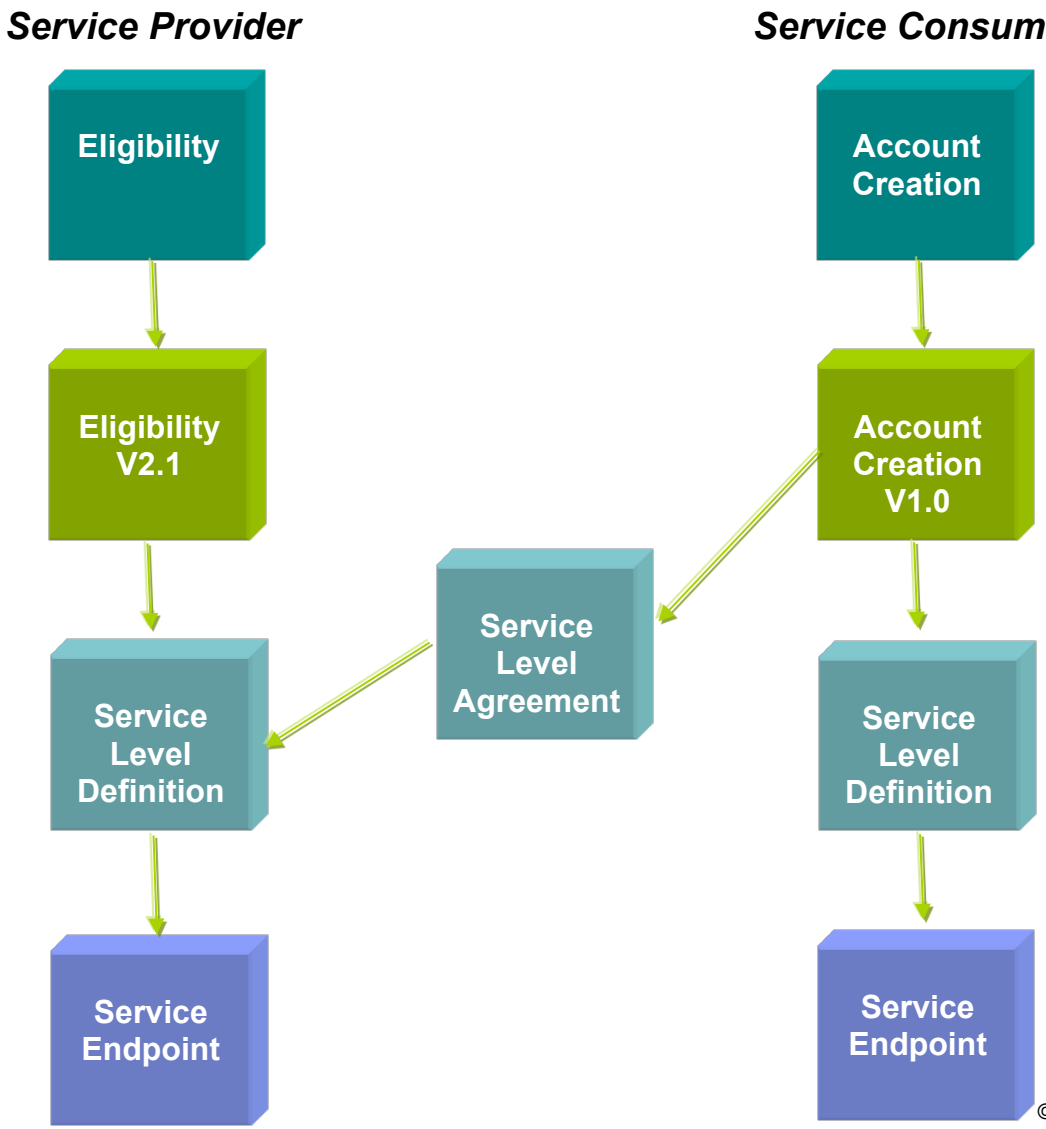
- Monitor compliance with policies
- Monitor compliance with governance arrangements
- Monitor IT effectiveness metrics

Enable the Governance Model Incrementally

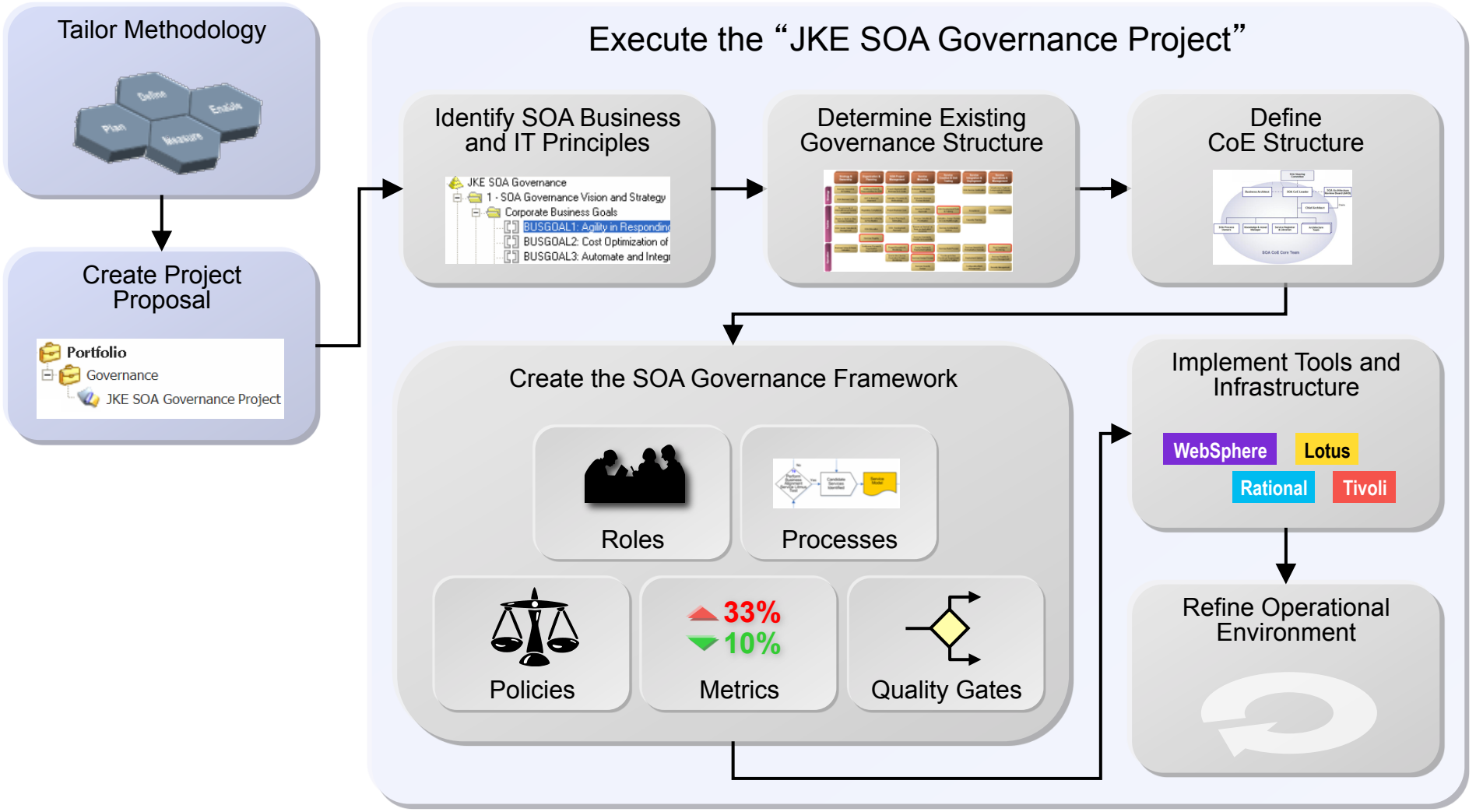
- Deploy governance mechanisms
- Deploy governance IT infrastructure
- Educate and deploy on expected behaviors and practices
- Deploy policies

Source: IBM

Governs the Service Consumers as well as the Service Provider



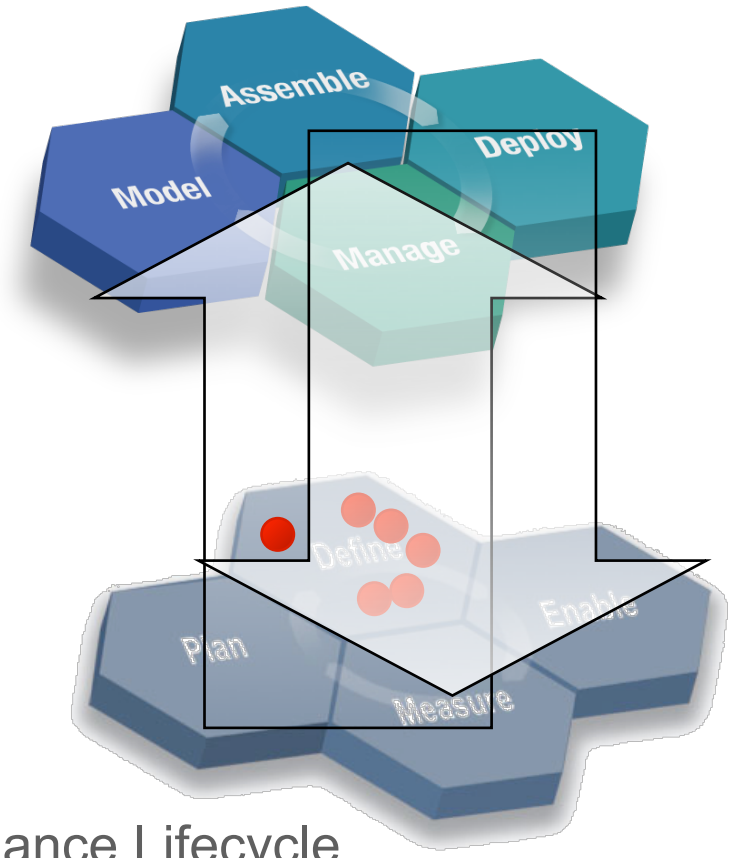
Example: Defining the Governance Solution



Source: IBM

Interaction Between the Lifecycles

Service Lifecycle



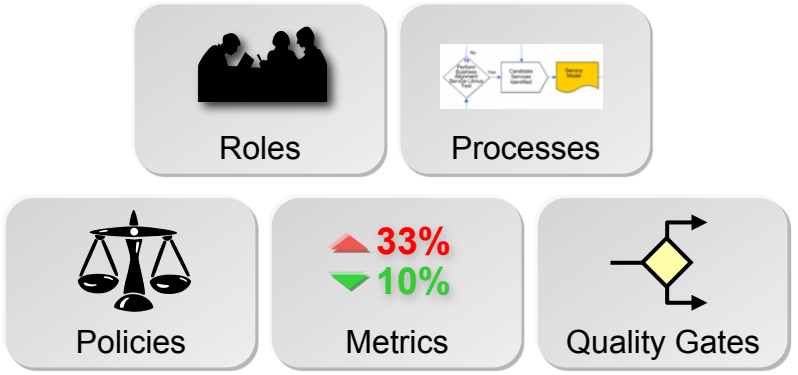
Governance Lifecycle

- Policies
  - quality gates
  - controls
  - metrics
  - standards
- are *defined* in the Governance lifecycle (for different aspects of Governance)…
- …and they are *enforced* in the service lifecycle
- metrics are captured to improve governance process

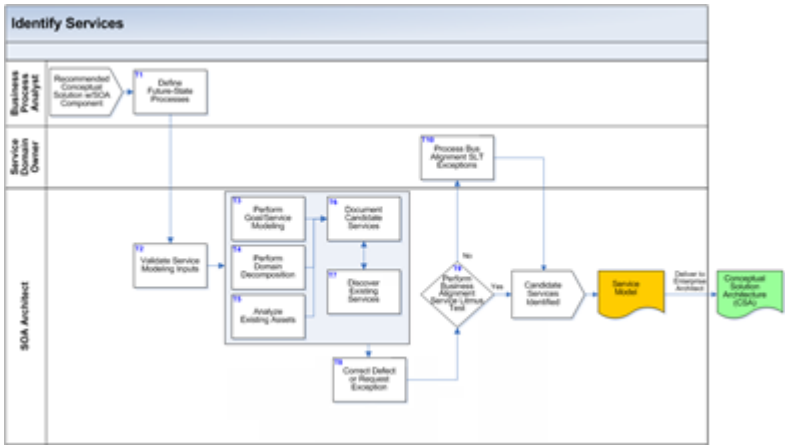
Source: IBM

The Governance Framework (Extensions to Development Processes)

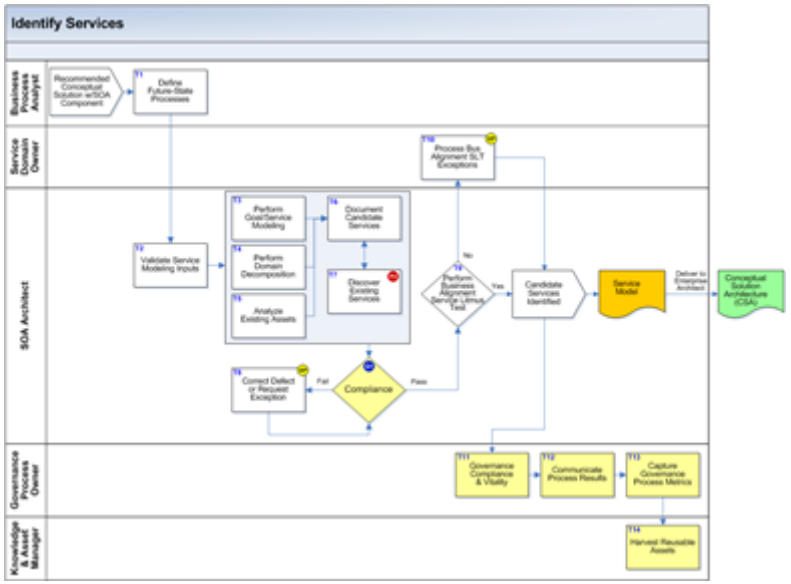
- **All the "elements" that we need to add to make a process well-governed**



non-governed process

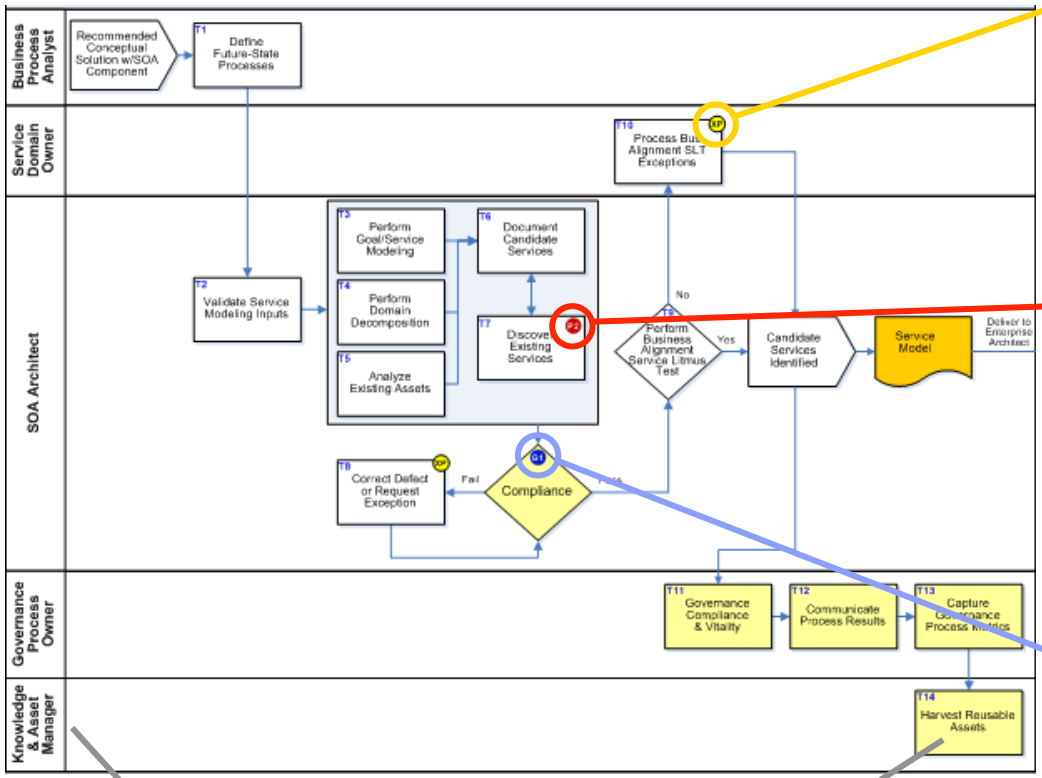


well-governed process



Source: IBM

Example – Enforcement at Development Time



exception procedure

Implement Service that failed Litmus test

human decision

policy

Services should be reused instead of created whenever possible

registry lookup

quality gate

Services must be compliant with the existing reference architecture

review

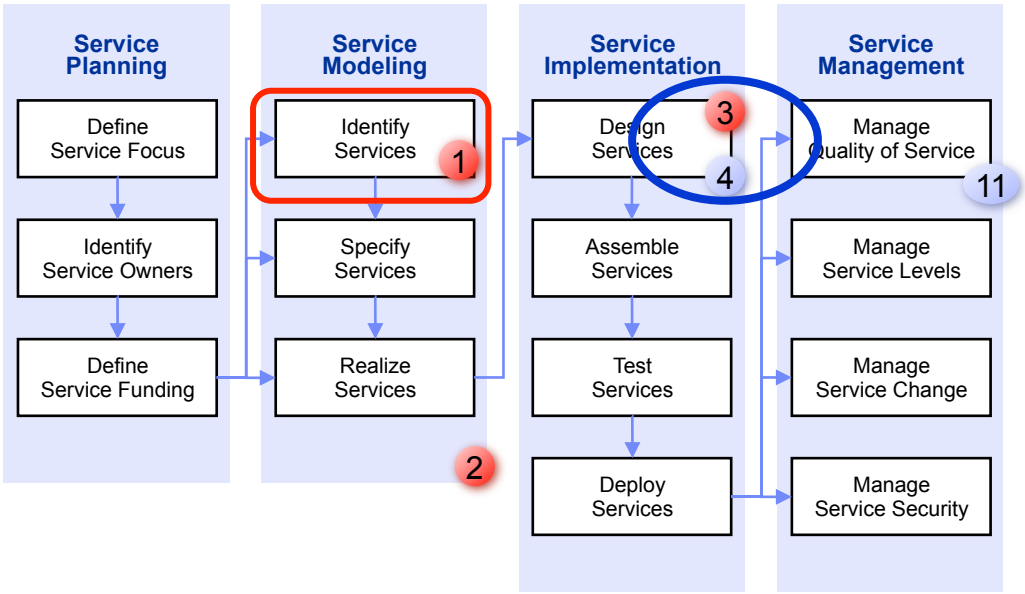
new roles

new activities

Source: IBM

Example – Enforcing Service Reuse Policy

- During the "Identify Services" activities, the SOA Architect implements the Service Reuse policy searching for existing services
- At the Validate Service Design quality gate the policy is enforced



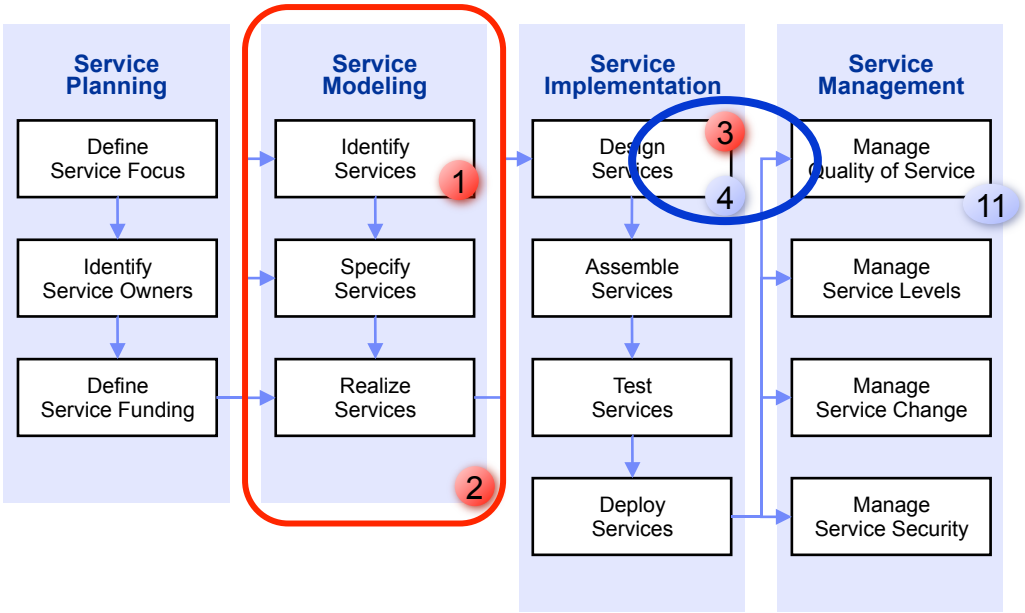
Policy 1 Services should be reused instead of created whenever possible

Quality Gate 4 Validate Service Design, semi-automatic enforcement during development

Source: IBM

Example – Enforcing Architecture Compliance Policy

- **The SOA Architect implements the Compliance with the Reference Architecture policy during all the activities in the Service Modeling phase**
- **At the Validate Service Design quality gate the policy is enforced with a manual review of the service model**



Policy 2 Services must be compliant with the existing reference architecture

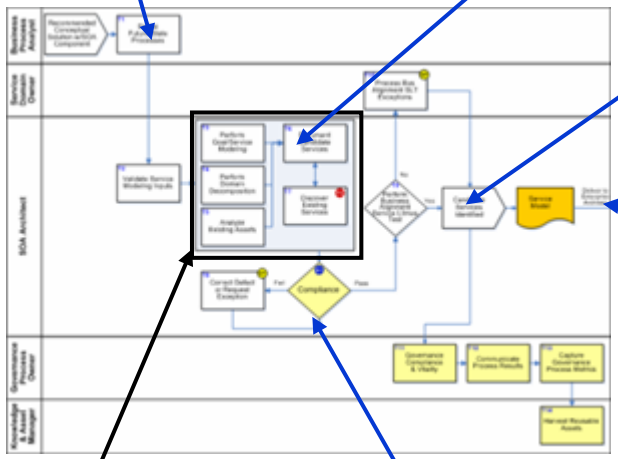
Quality Gate 4 Validate Service Design, manual enforcement during development

Source: IBM

Governance at *Development* Time – Enforcing Policies of Services Life Cycle

Ensure Compliance and Define Policies

1 **Service state transitions are defined based on governance solution**



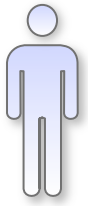
2 **Validate Service Ownership**

3 **Registry Lookup to enforce reuse of existing services**

5 **Publish Service to repository after Litmus test is successful**

6 **Add a Policy for Dynamic Binding**

Service Analysis and Design



4 **Validate Service Design for Standards Compliance**

Source: IBM

Policies / Principles

Terms (again) – Definitions (Wikipedia, Oxford Dictionary)

- **Policy:**
  - **A policy is a principle to guide decisions and achieve rational outcomes. A policy is a statement of intent, and is implemented as a procedure or protocol.**
  - **Presidential executive orders, corporate privacy policies, and parliamentary rules of order are all examples of policy. Policy differs from rules or law.**
- **Principle**
  - **Defines the underlying general rules which an organization will use to make decisions about the selection, utilization and deployment of all business and IT resources and assets, across the enterprise.**
  - **"A fundamental truth or proposition that serves as the foundation for a system of belief or behavior or for a chain of reasoning."**

Policies / Principles: Introduction Comment

- Policies / Principles
  - The terms not used consistently, Policies often legally binding, Principles internally.
  - Both defining underlying general rules which an enterprise will use to make decisions“
  - They should be well expressed (Name, Description, Motivation, Implication)
  - Hierarchy: Guiding Ps down to Detailed, Company down to Coding Ps.

- Hint: let “C”-management express Guiding Principles / Policies and explain how well they are handled (see Example)

Example *Guiding Principles / Policies* – Early Iteration

Guiding Principle	Description
Investigate TCO, ROI, and Risks	One of the element of risk management is that small suppliers should provide access to the code.
Reuse standard before buy standard before reuse non-standard solution before build internal solution	Standard is very important, reuse is the second choice but reuse of non-standard could be very costly and risky. Market applications (packet solutions) as seen as standards. This also applies to processes etc.
Simplify the portfolio of applications and reduce the complexity of the environment	Complexity of environment includes challenging the business requirements in order not to implement unnecessary functions.
"Do not reinvent the wheel" – use assets	
Manage Information/Data as a corporate wide asset	

Policies / Principles must be at a consistent level, provide guidance and not dictate how things should be

Defining Principles - Rules, Guidelines

A good principle:

- States a fundamental belief of the enterprise in one or two clearly written sentences.
- Recommends an action against which some arguments could be made.
- Has relevance to a technical architecture.
- Is worded directly and simply in terms understandable by both business and IT managers.
- Has business wide applicability.
- Is durable; will not be outdated quickly by advancing technology.
- Has objective reasons for advancing it instead of the alternatives which were considered.
- Has impacts which need to be documented.
- Represents change

A poor principle:

- Makes a statement which no one would dispute.
- Is a general business or financial statement.
- Has little or no general applicability. It may actually select a standard or a technology.
- Is stated at too low a level of detail and may not endure.
- May be included "because I say so".

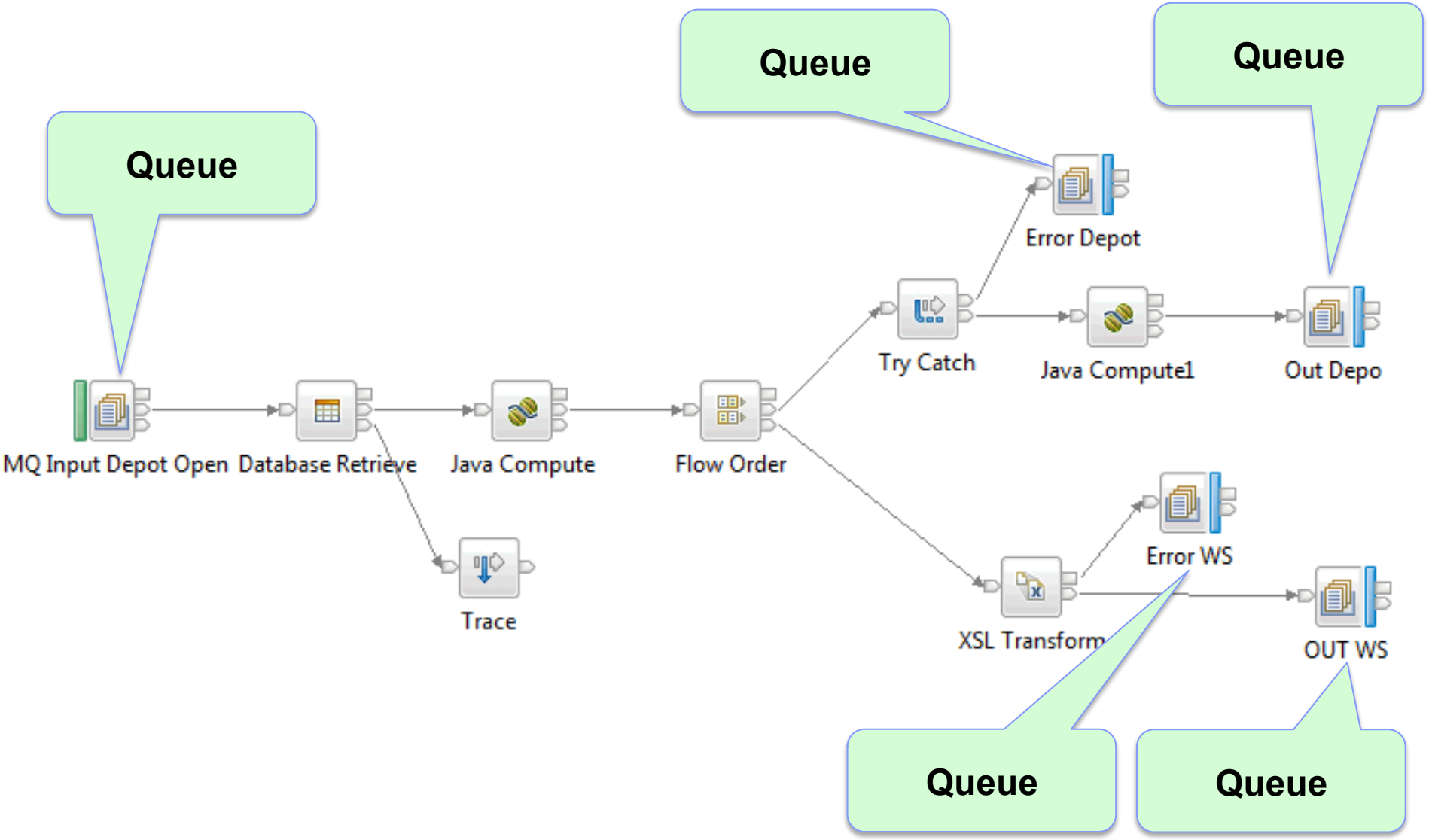
Source: IBM

Special Topic:
ESB versus Broker

ESB versus (Message) Broker

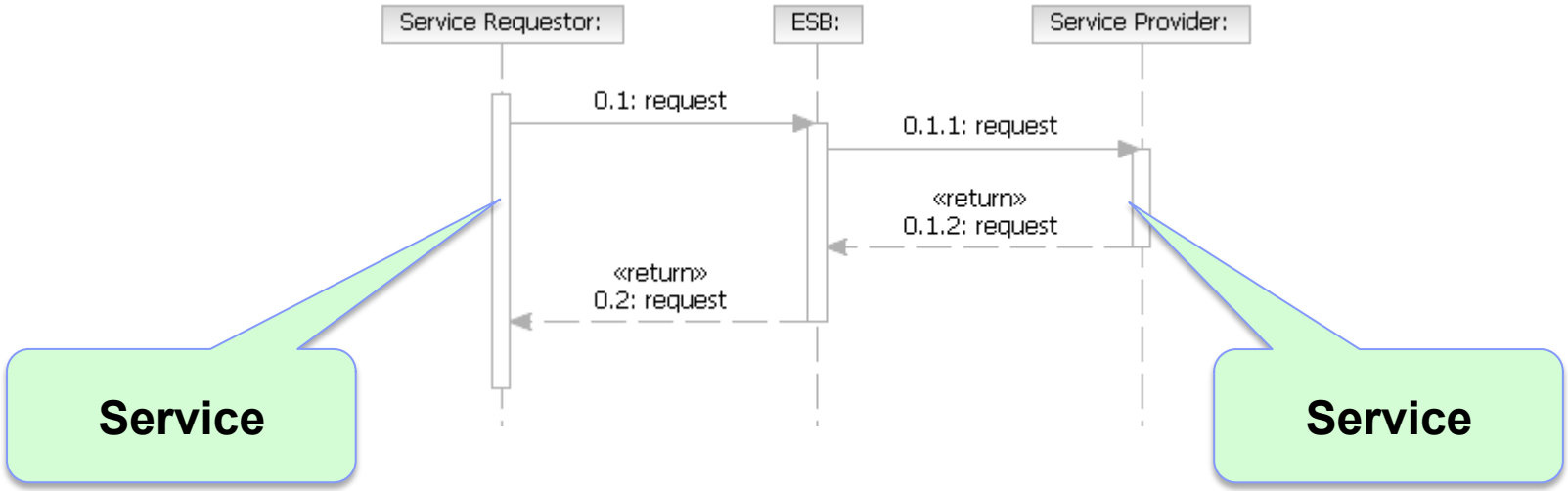
- Both
  - Intermediary, Transformation of Data
  - Often both are introduced as concepts
  - For both (named) products are availabl
- ESB (Enterprise Service Bus)
  - Transfer of Protocols, Routing
- (Message) Broker
  - Managing of Queues, Emphasis on lows
- Closer view:
  - ESB intermediary for messages – this is a higher abstraction than a broker concept of queuing
  - ESB may include a Broker, a Broker can be used as ESB
  - Broker needs on both sides the “same” installed software (aka EAI – Enterprise Application Integration)

Illustration of (Message) Broker Information Flow



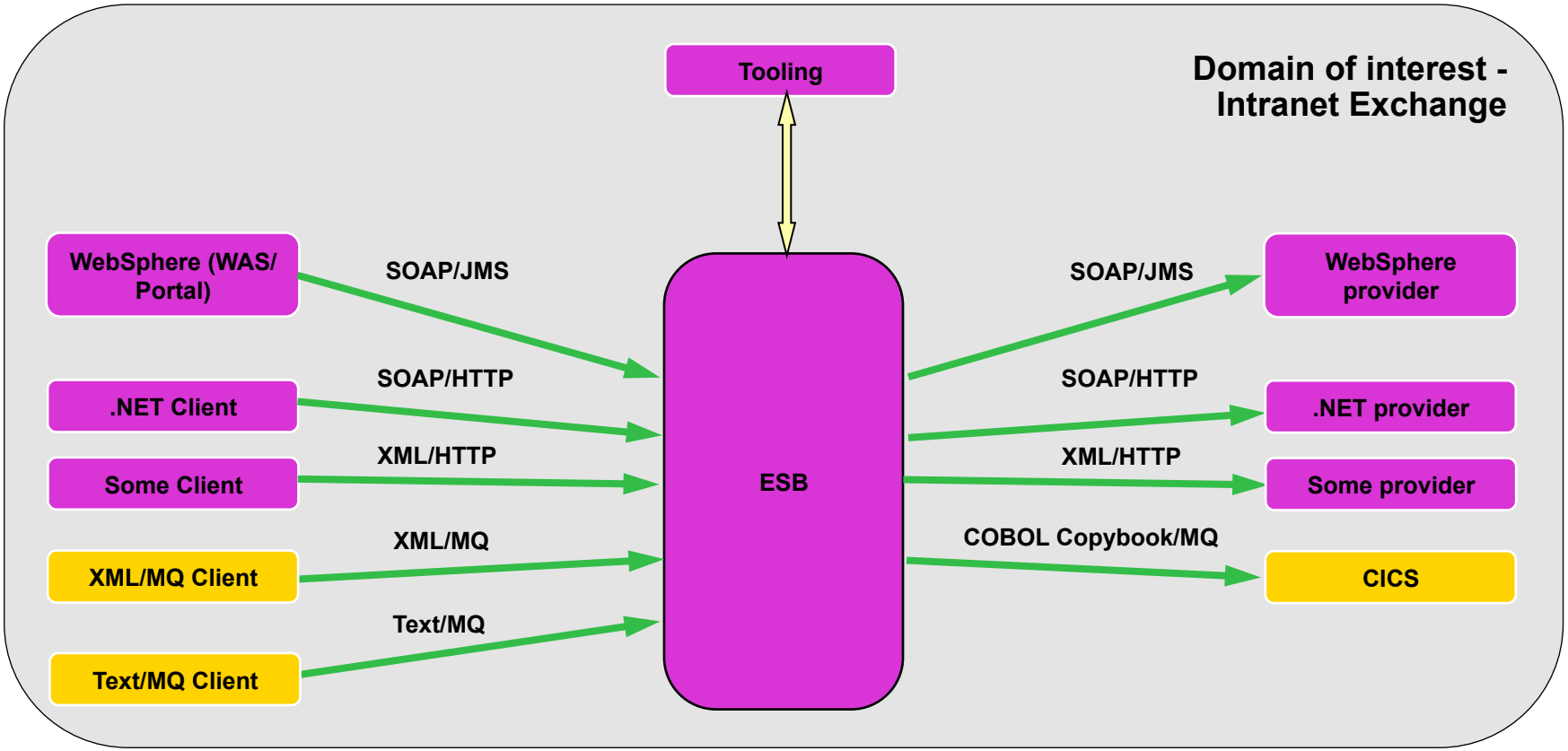
Recap: ESB (Enterprise Service Bus) – Service Virtualization

- **ESB acts as an intermediary (proxy) between requestor and provider**



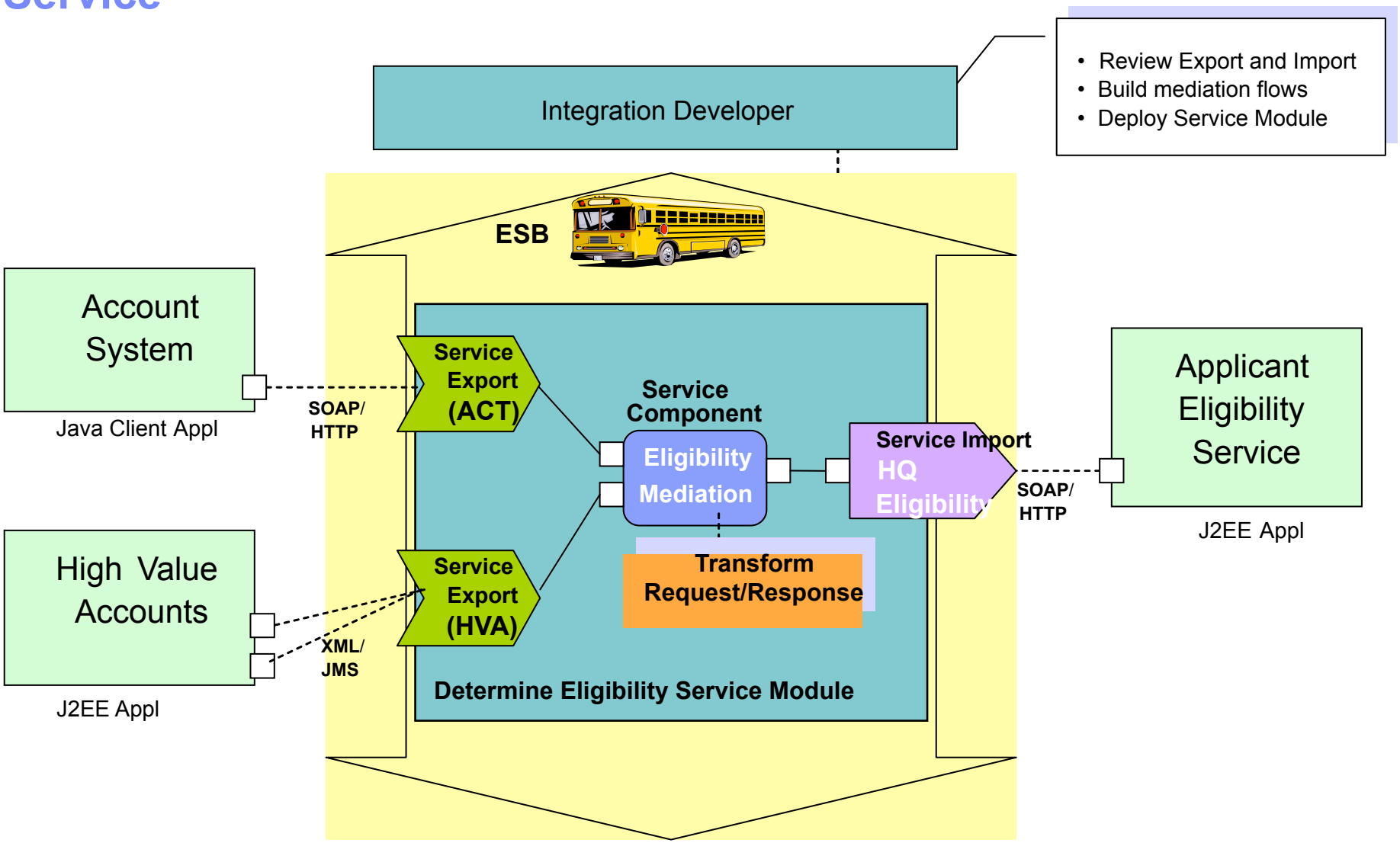
- **ESB provides *service virtualization* of**
  - ***Location and identity***
  - ***Interaction protocol***
  - ***Interface***
- **Interactions are *decoupled*, supporting *separation of concerns***

ESB – Multi-protocol Exchange – Intermediary decoupling heterogeneous consumers and suppliers

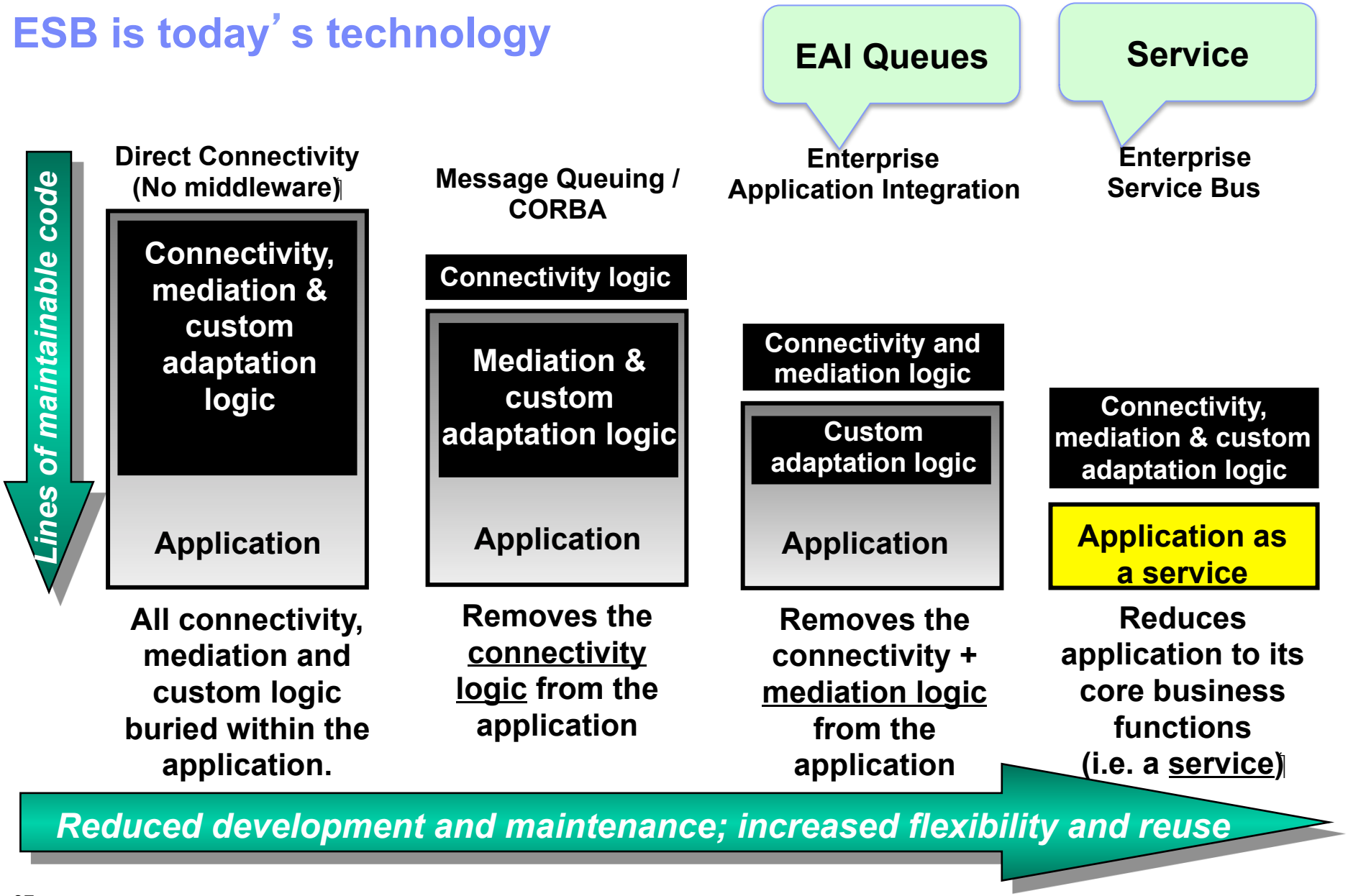


Source: IBM

Example of ESB use: Multiple Channel Access to Backend Service



ESB is today's technology



Source: IBM