# Understanding SOA

hungdn@ptit.edu.vn

Outlines

1. Introduction SOA
2. The Four Characteristics of SOA
3. The Four Common Types of SOA
4. The End Result of Service-Orientation and SOA
5. SOA Project and Lifecycle Stages

Introduction SOA
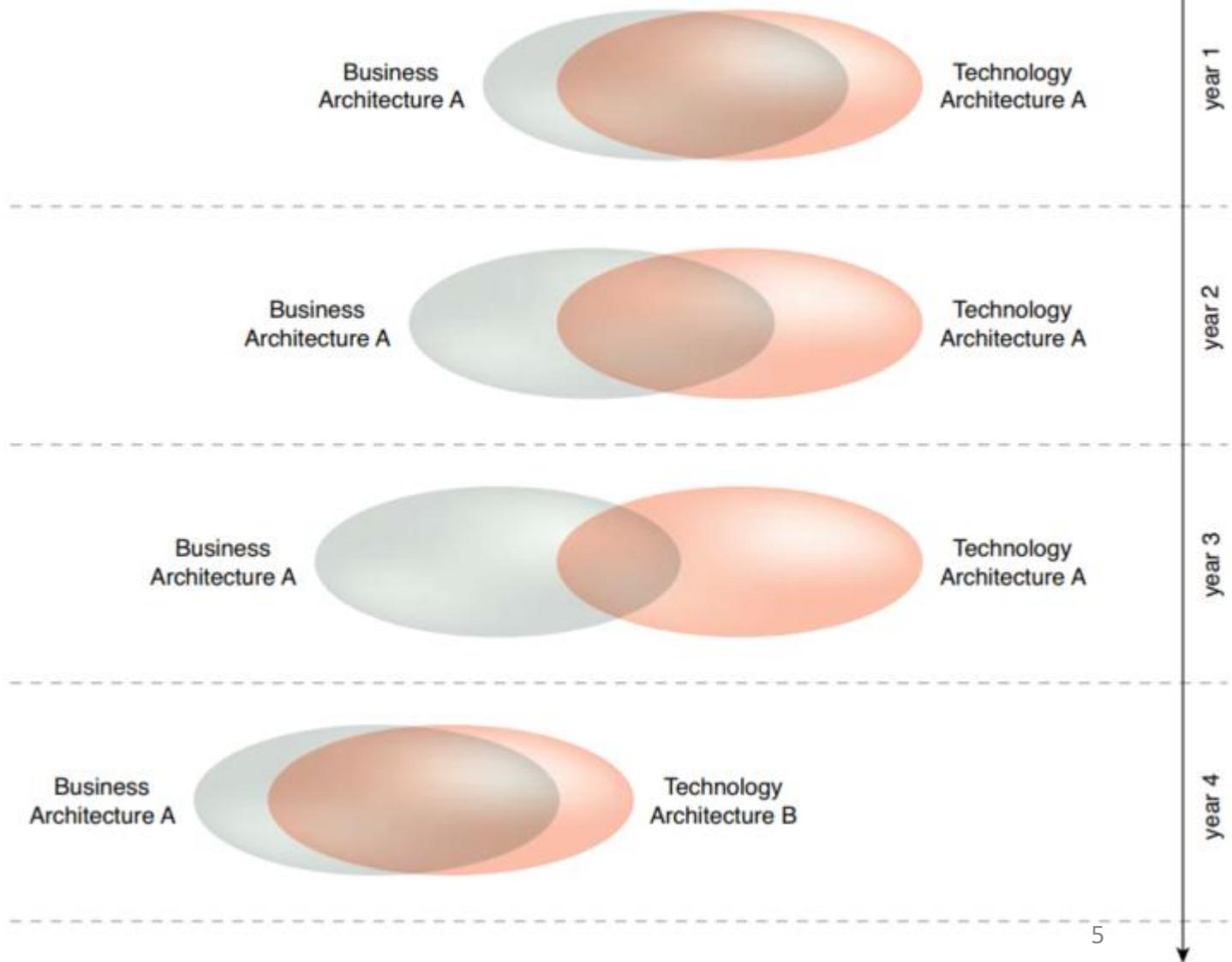
- A solution is service-oriented when service-orientation is applied significantly.

- To build successful service-oriented solutions
  - Mere understanding of the design paradigm is insufficient
  - Need a distributed technology architecture with specific characteristics

- These characteristics distinguish the technology architecture as being service-oriented. This is SOA.
  - Important during initial service delivery and when forming complex compositions.

4.1 The Four Characteristics of SOA
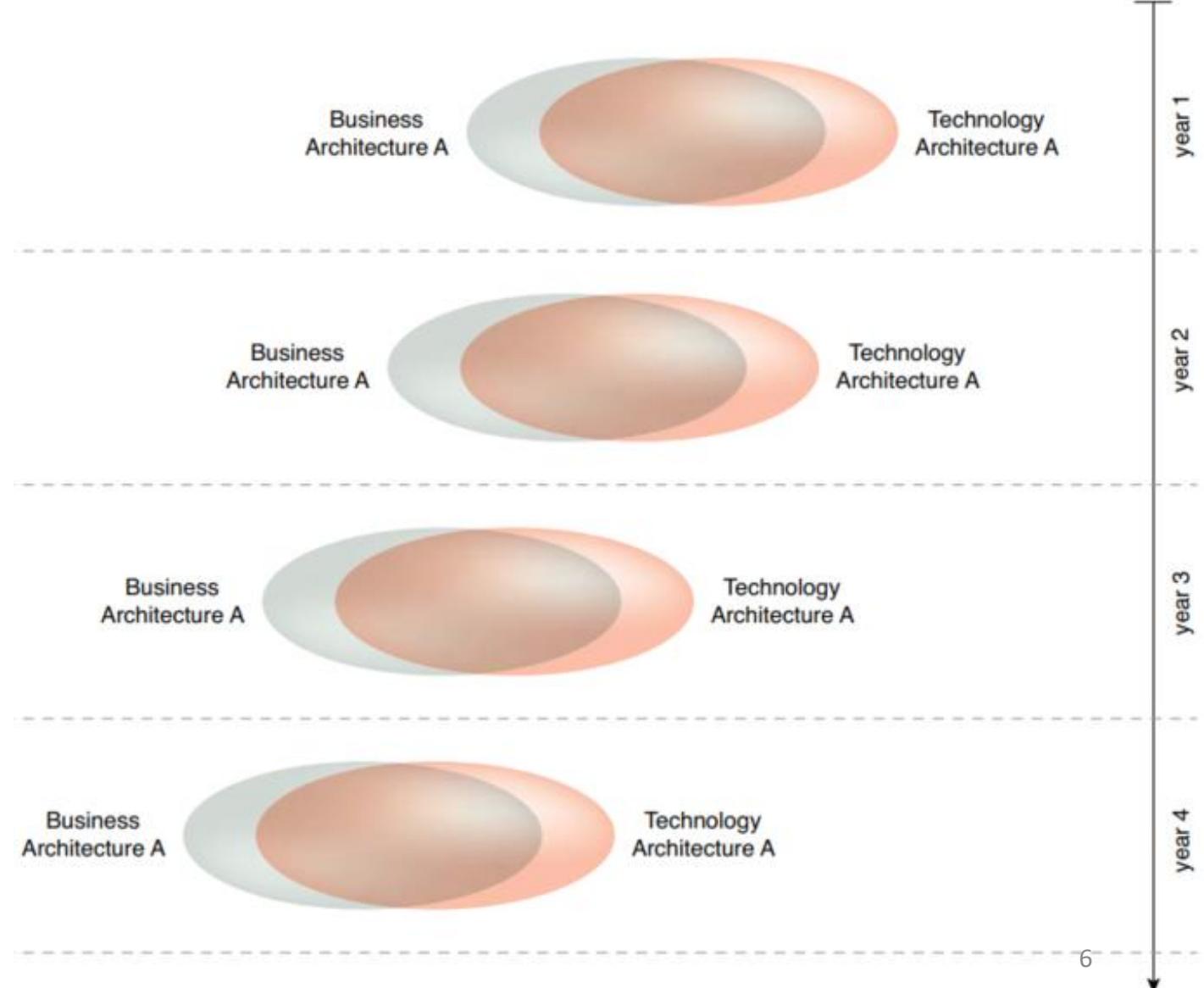
- **Traditional technology architectures focus on short-term business requirements.**
  - Lack of consideration for long-term strategic business goals can lead to misalignment.


Business Architecture A — Technology Architecture A (year 1)

Business Architecture A — Technology Architecture A (year 2)

Business Architecture A — Technology Architecture A (year 3)

Business Architecture A — Technology Architecture B (year 4)
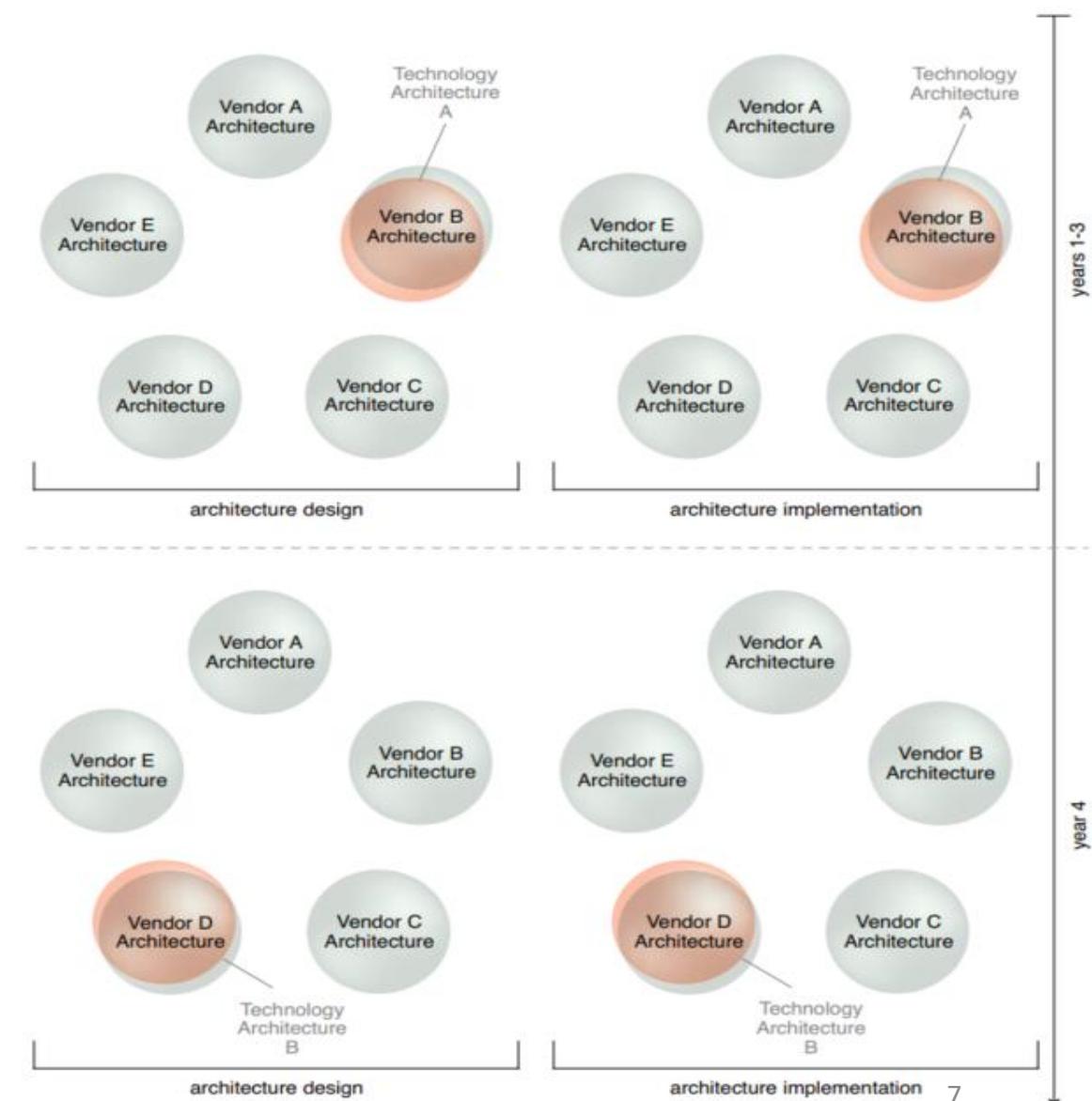
- **Business-Driven approach ensures alignment of technology architecture with strategic business vision.**
 - This alignment facilitates continuous adaptation to changing business needs.
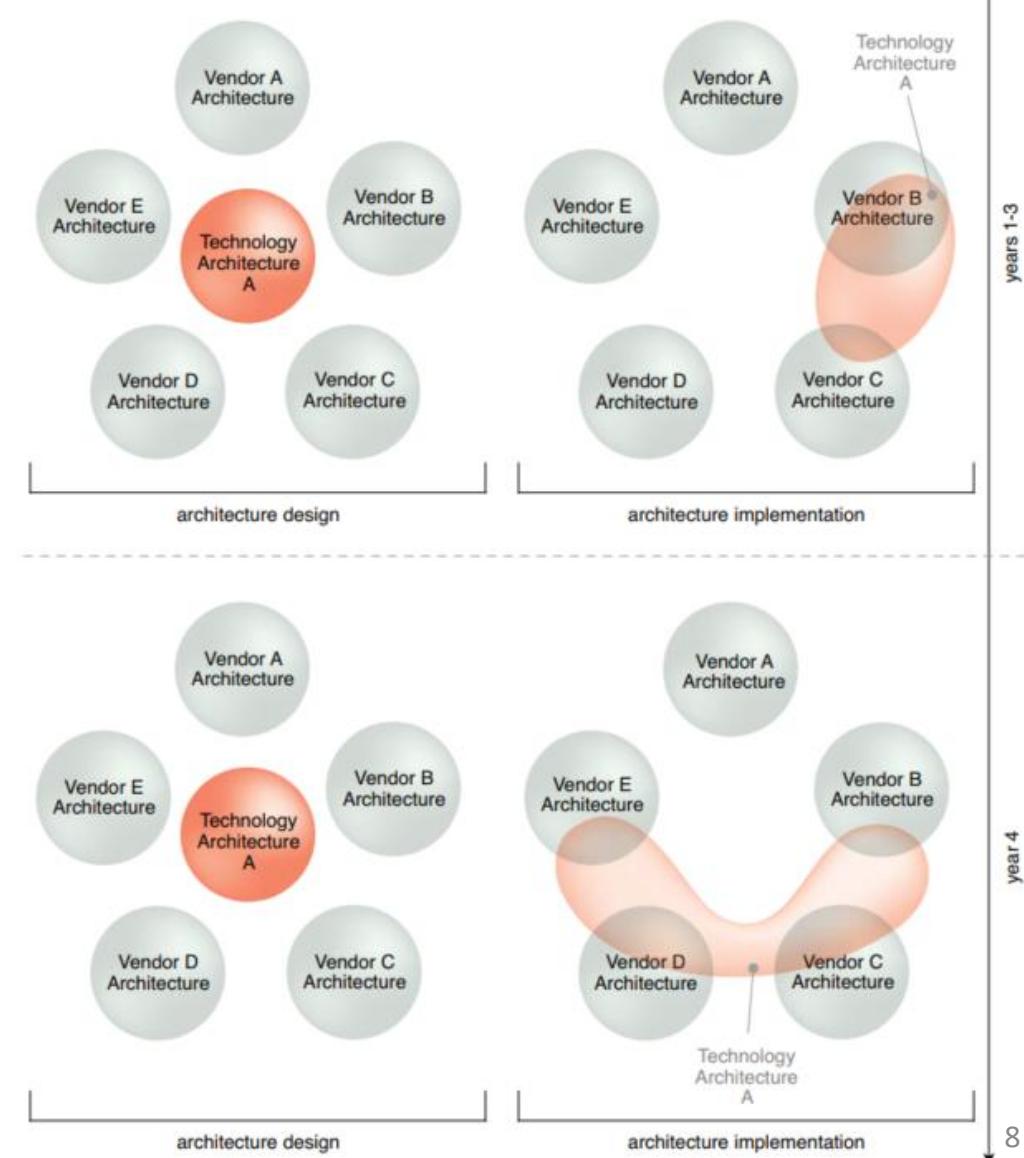
- **Designing a service-oriented architecture around a specific vendor platform (Vendor-Centric) can result in inheriting proprietary characteristics.**
  - This inhibits the architecture's future evolution in response to innovations from other vendors.
 - Limited to adapt and lifespan

7
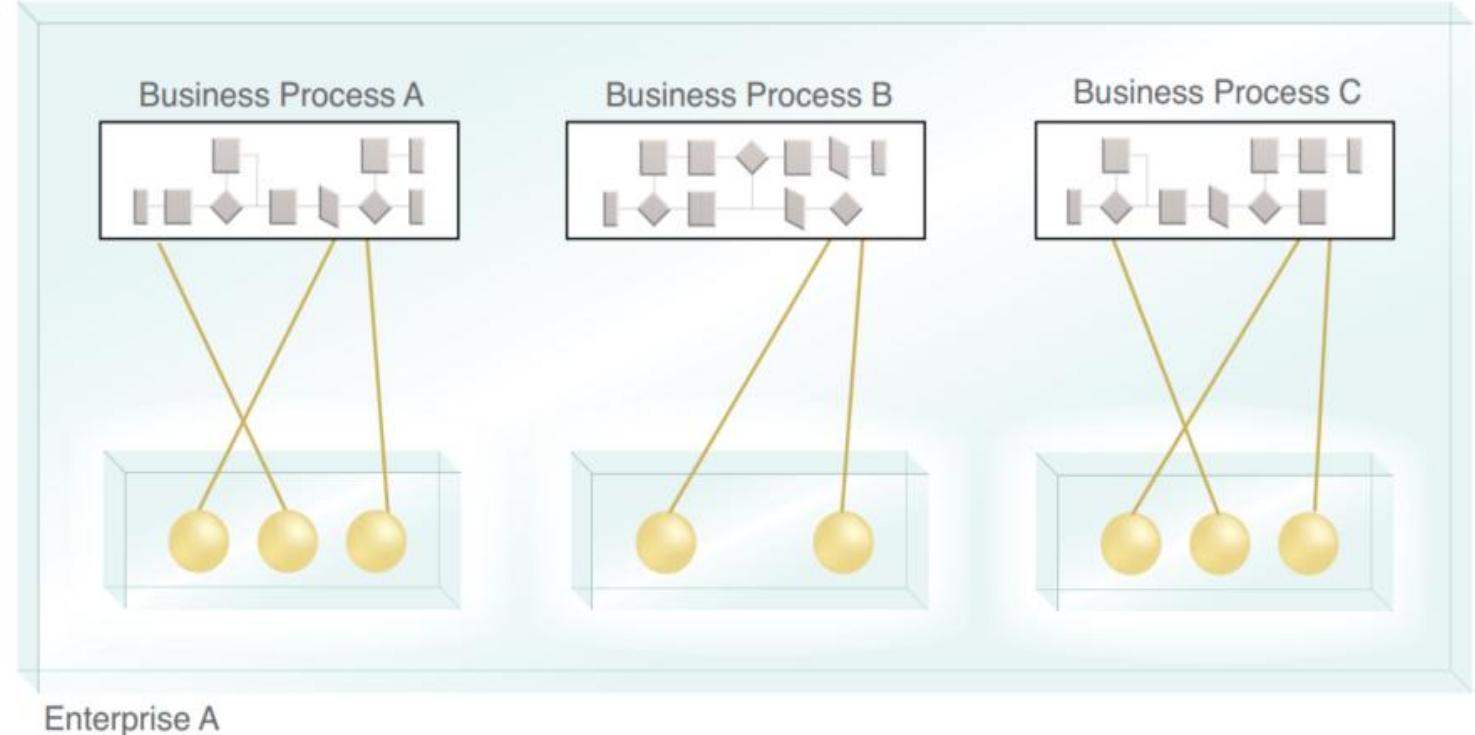
- **A vendor-neutral approach allows the architecture to evolve independently of specific vendors, ensuring long-term effectiveness.**
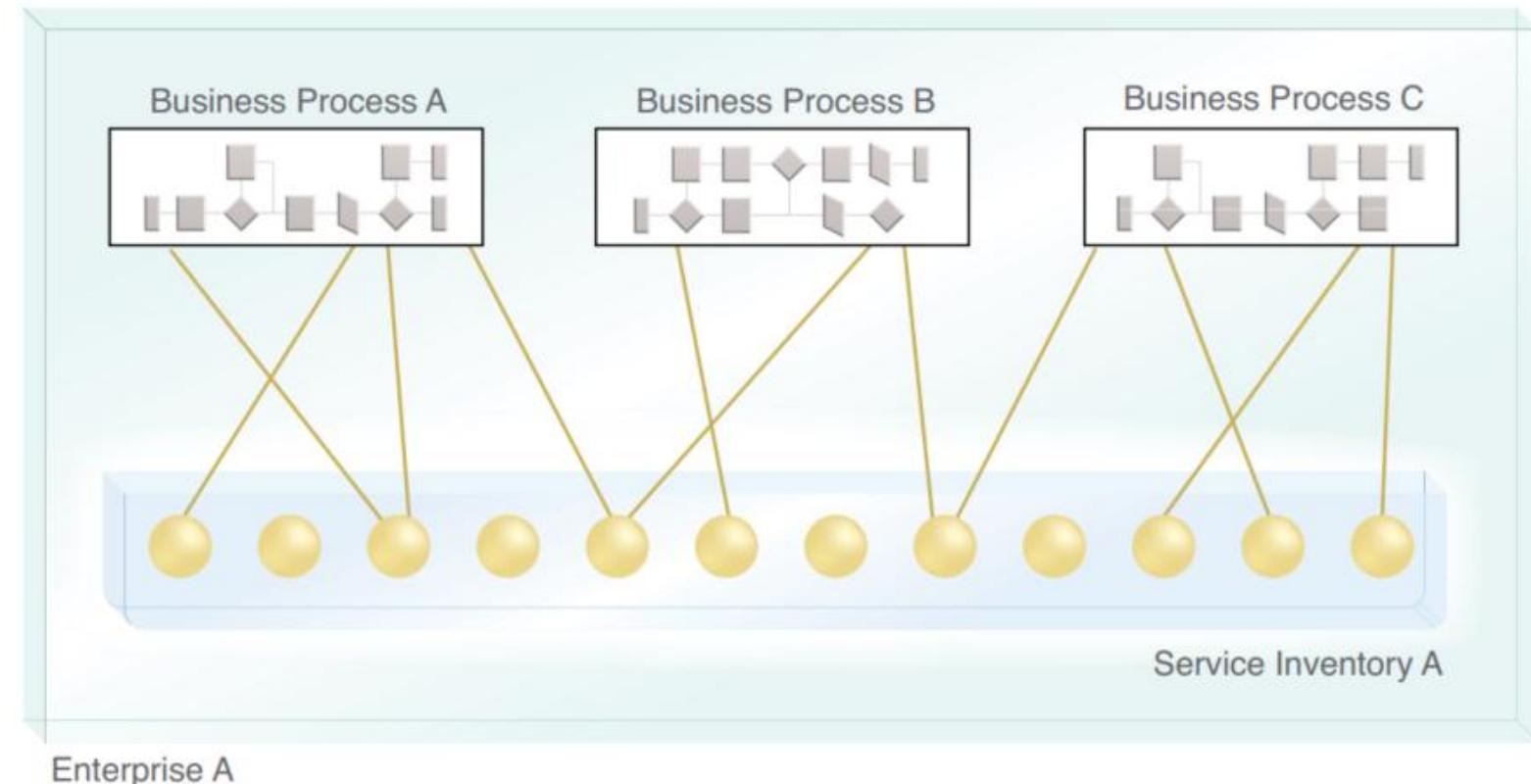
- **Service-oriented solutions, despite being distributed, can lead to the creation of new silos within an enterprise.**

Single-purpose Services
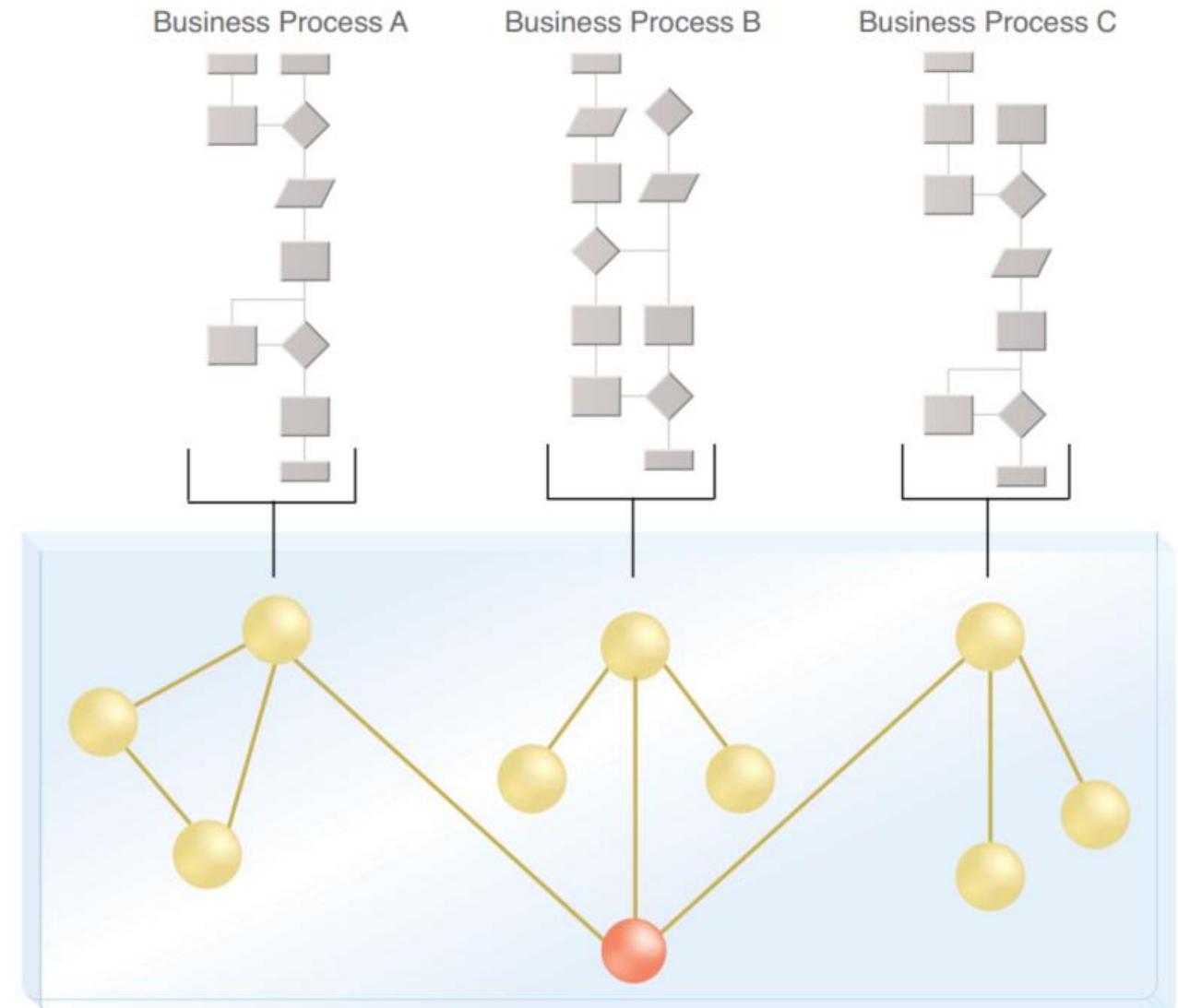
- **Services are positioned as enterprise resources, requiring specific design characteristics**
 - Available beyond a specific implementation boundary
  - Designed according to established design principles and enterprise standards.

Services as Enterprise Resource

- **Service-orientation emphasizes designing software programs not just as reusable but also as flexible resources that can be plugged into various service-oriented solutions.**

- **To accomplish this, services must be composable. As advocated by the Service Composability (302) principle**

Business Process A   Business Process B   Business Process C

Service Inventory A
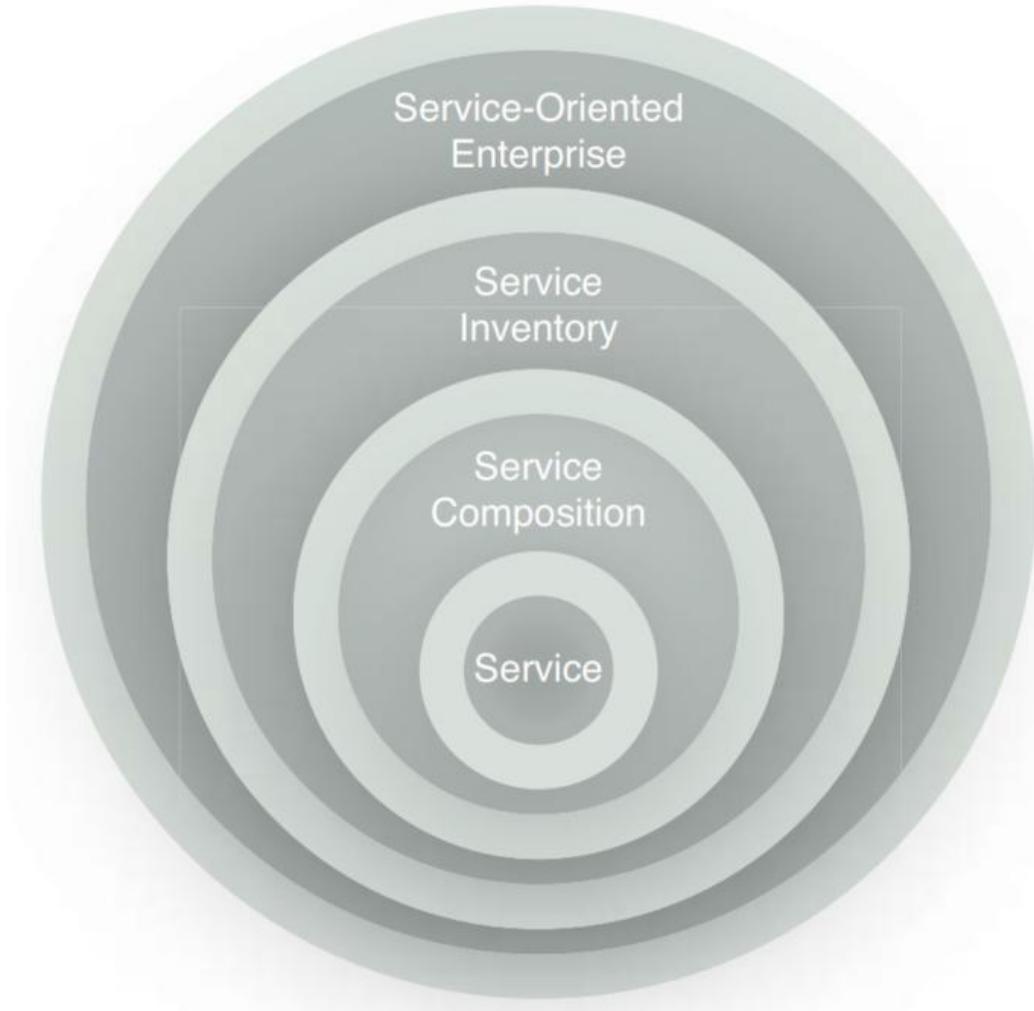
Design Priorities

- Service orientation is a paradigm that frames what you do. Service-oriented architecture (SOA) is a type of architecture that results from applying service orientation.

- Design Priorities:
  - Business value over technical strategy
  - Strategic goals over project-specific benefits
  - Intrinsic interoperability over custom integration
  - Shared services over specific-purpose implementations
  - Flexibility over optimization
  - Evolutionary refinement over initial perfection

4.2 The Four Common Types of SOA
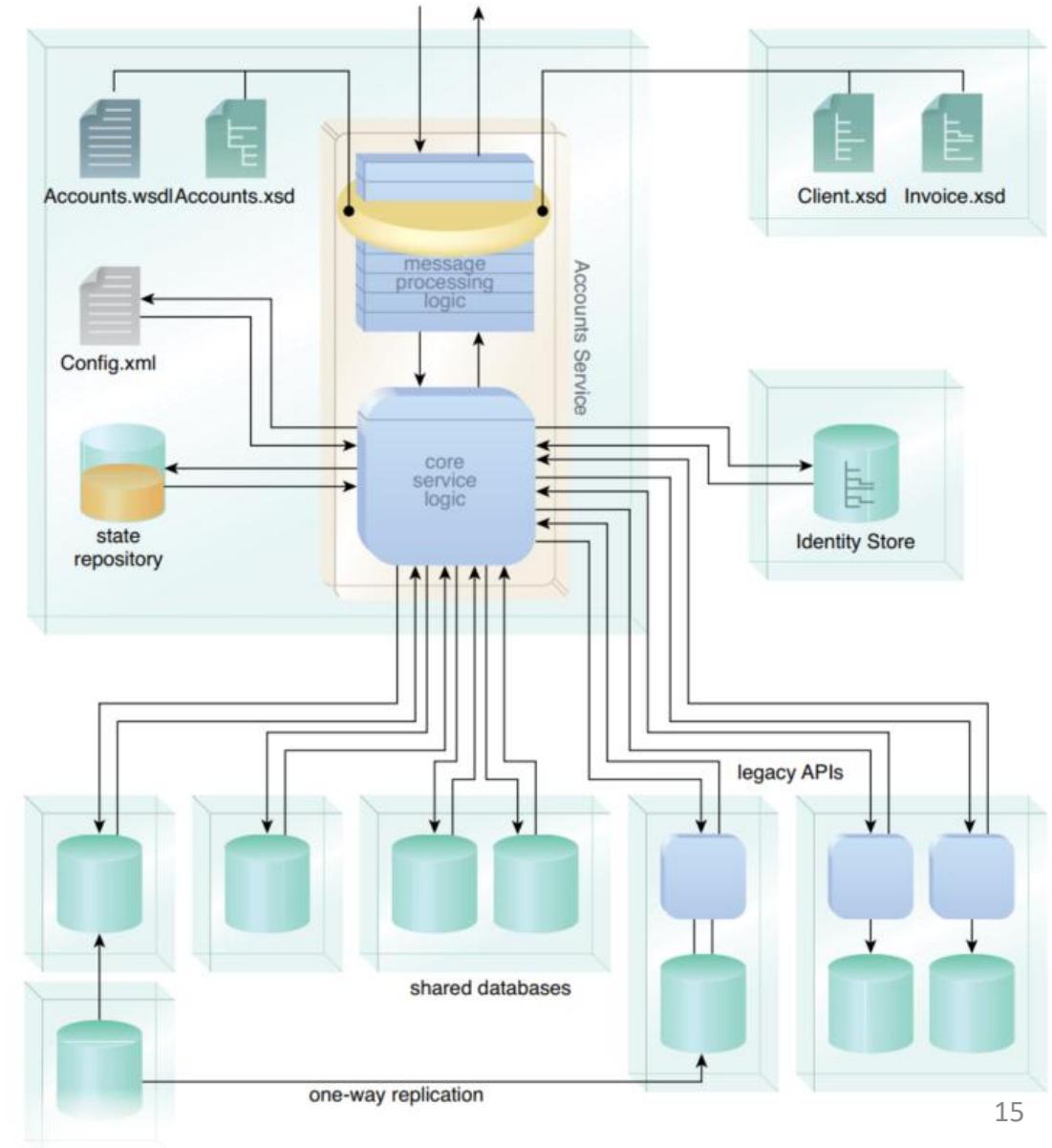
Architecture types - SOA patterns

- **To understand SOA mechanics, explore common types of technology architectures in a service-oriented environment:**
  - Service Architecture: Architecture of a single service.
  - Service Composition Architecture: Architecture of services assembled into a composition.
  - Service Inventory Architecture: Supports a collection of independently standardized and governed services.
  - Service-Oriented Enterprise Architecture: Architecture of the entire enterprise to the extent it is service-oriented.
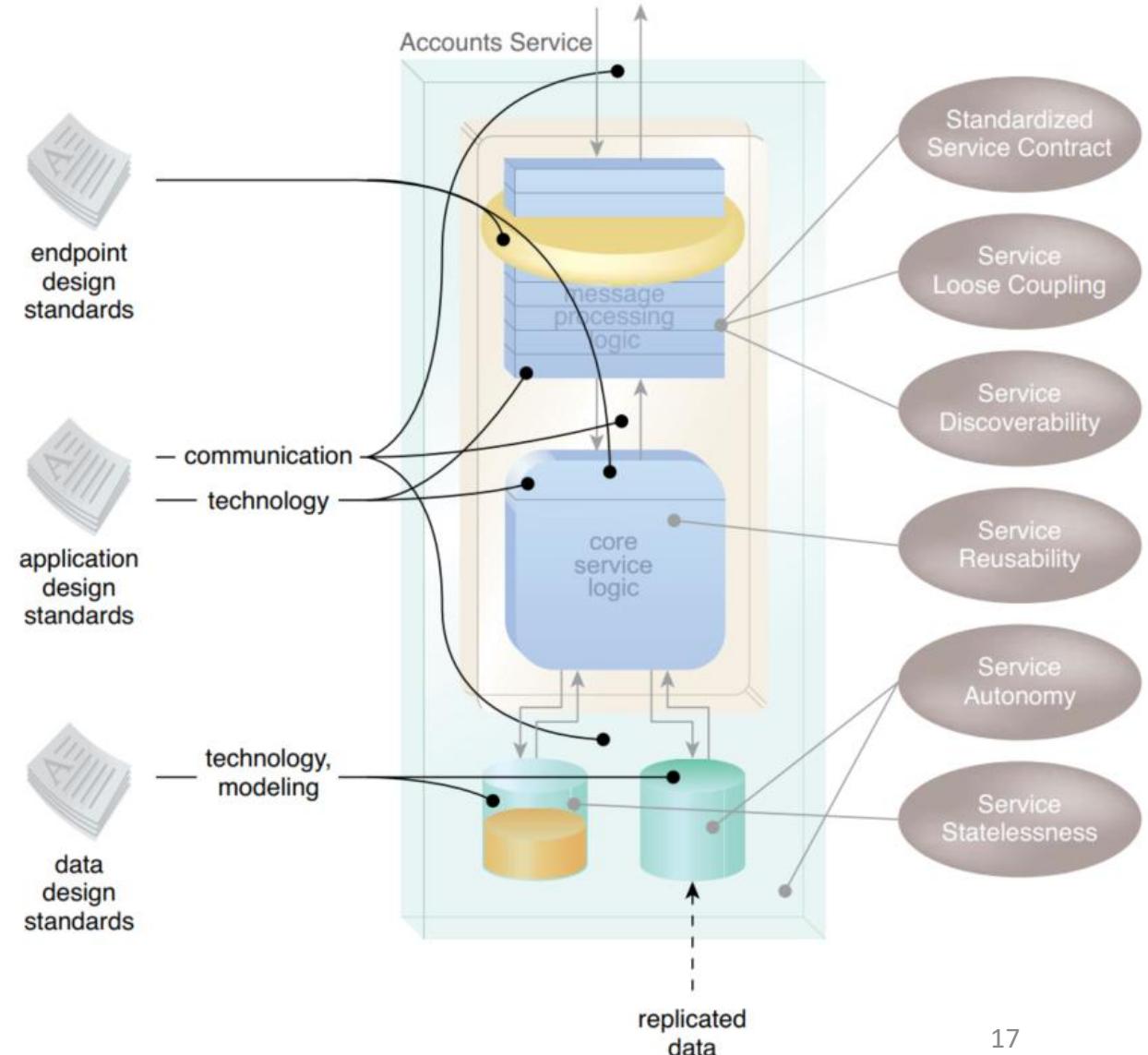
Service Architecture

- **The technology architecture focusing on the physical design of a software program as a service is termed the service architecture**

- **Comparable to a component architecture**
  - but extends to a greater degree for increased reliability, performance, scalability, behavioral predictability, and autonomy
  - Scope is larger as a service can encompass multiple components


15
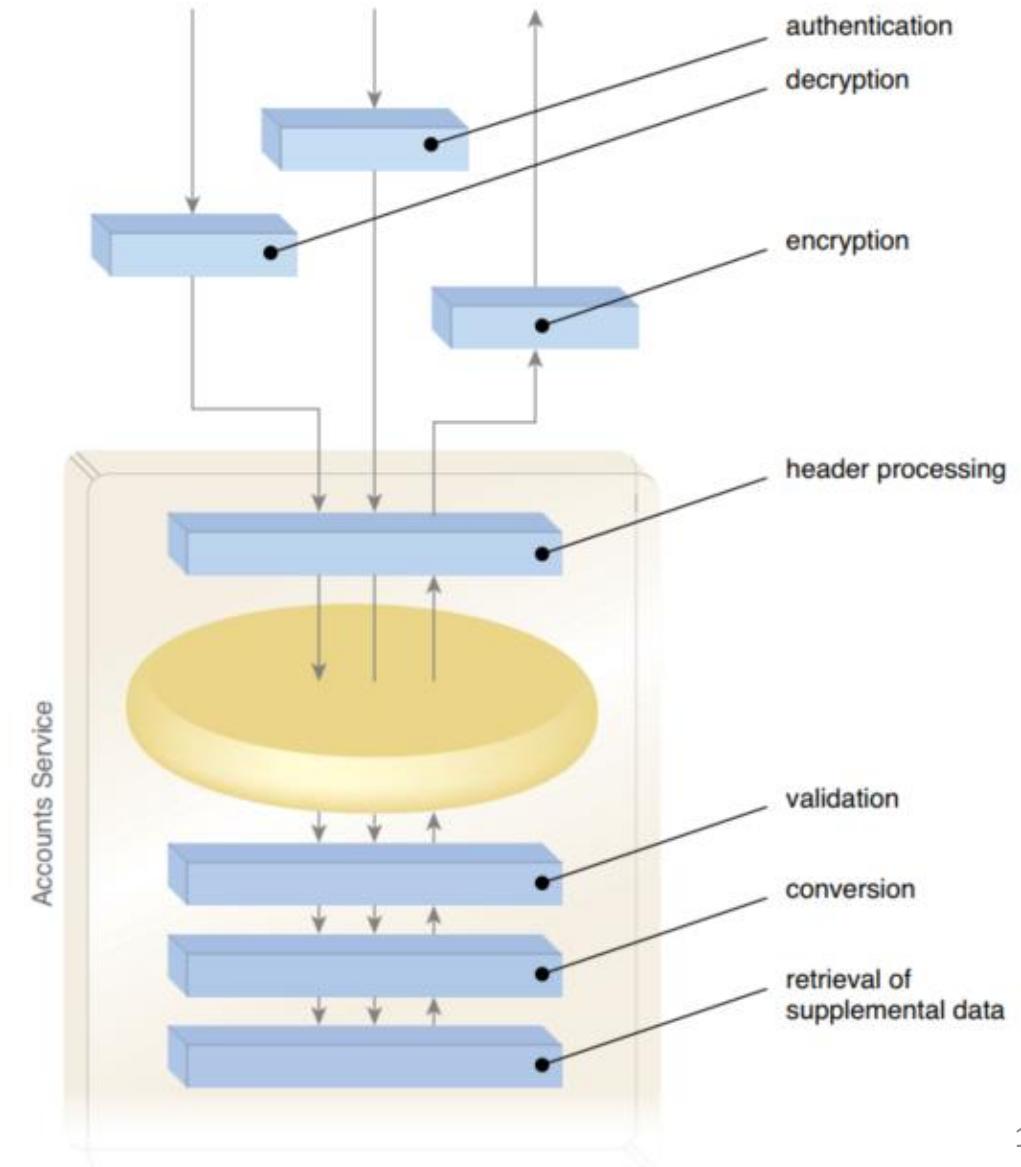
- **The application of design standards and other service-orientation design principles further affects the depth and detail to which a service's technology architecture may need to be defined (Figure 4.11)**
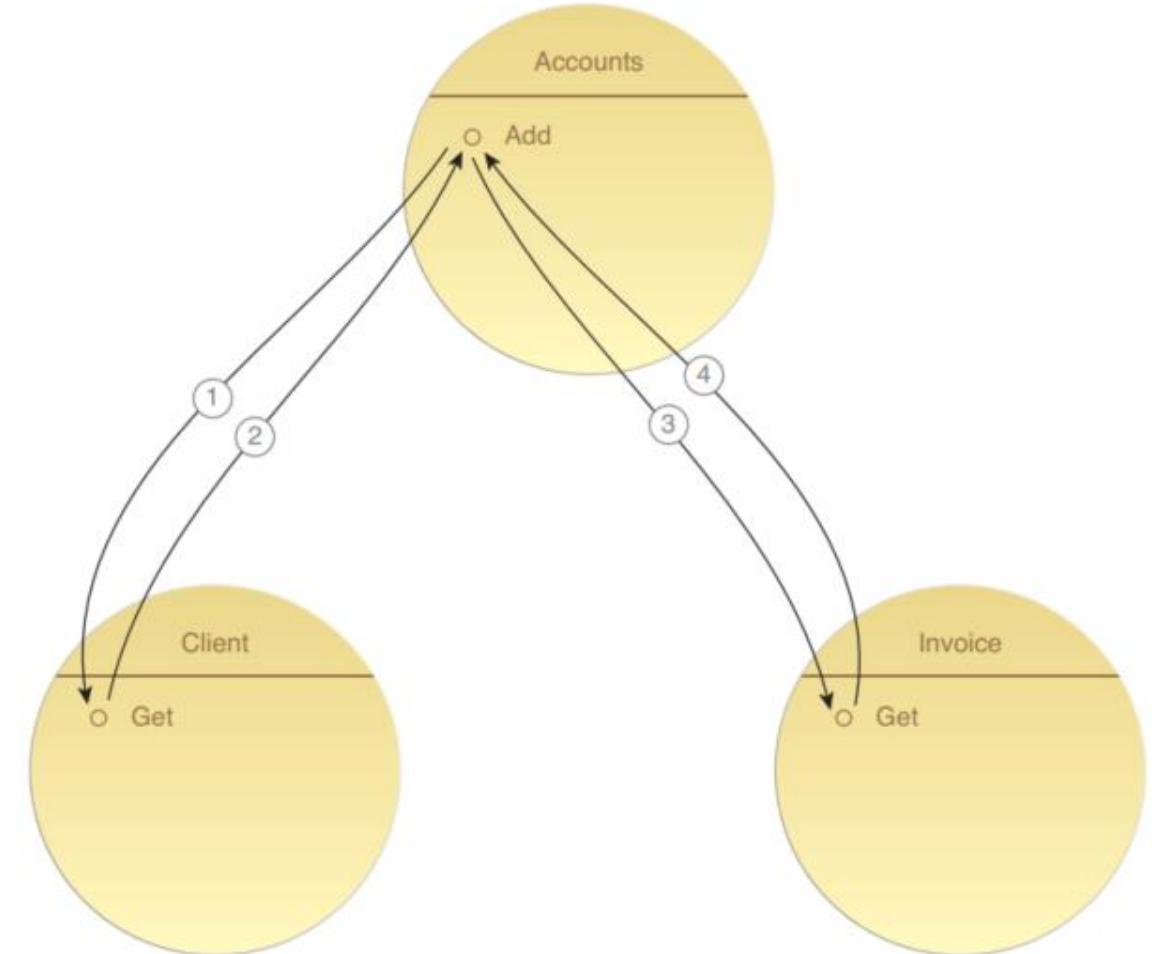
- **An important aspect related to service architecture is the Service Agent**

- **A variety of service agents are part of the service architecture**
  - Some perform general processing of all data whereas others are specific to input or output data flow


19
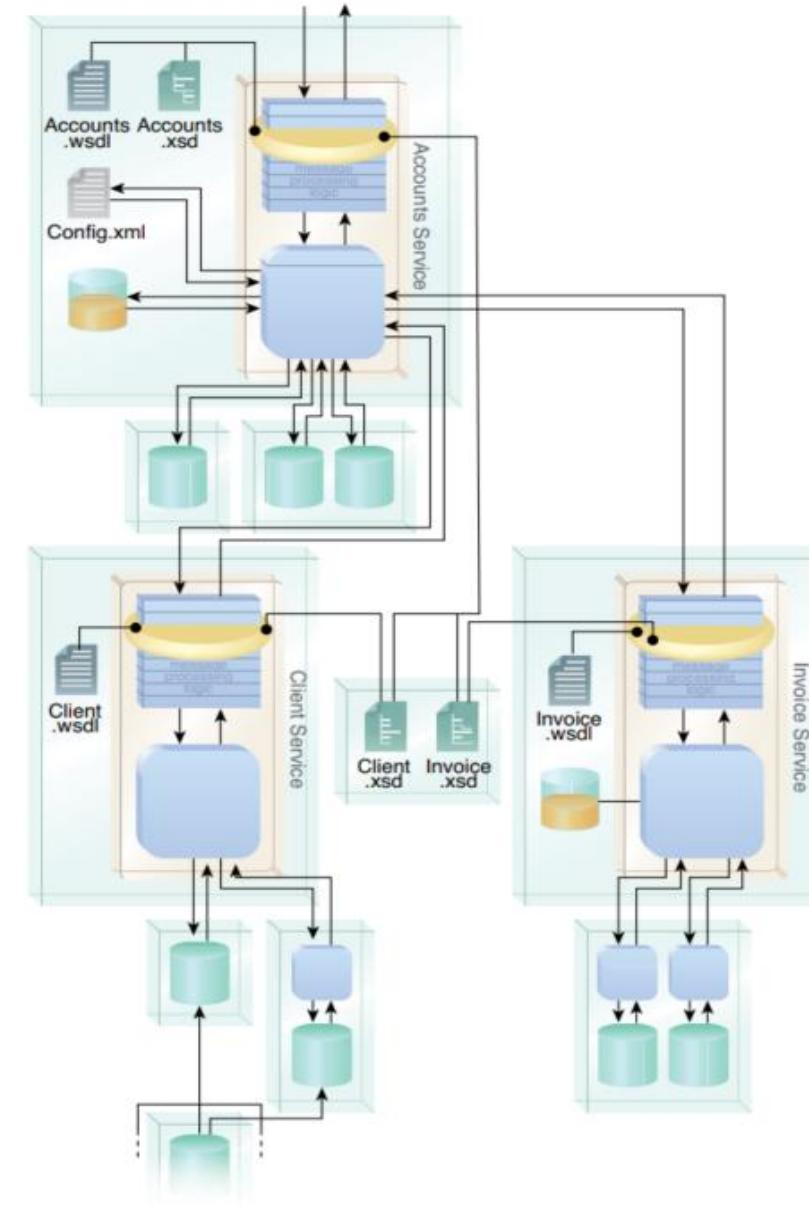
- **The fundamental purpose of delivering a series of independent services is so they can be combined into service compositions, fully functional solutions capable of automating larger, more complex business tasks.**

- **The Accounts service composition from a modeling perspective (Figure 4.14)**
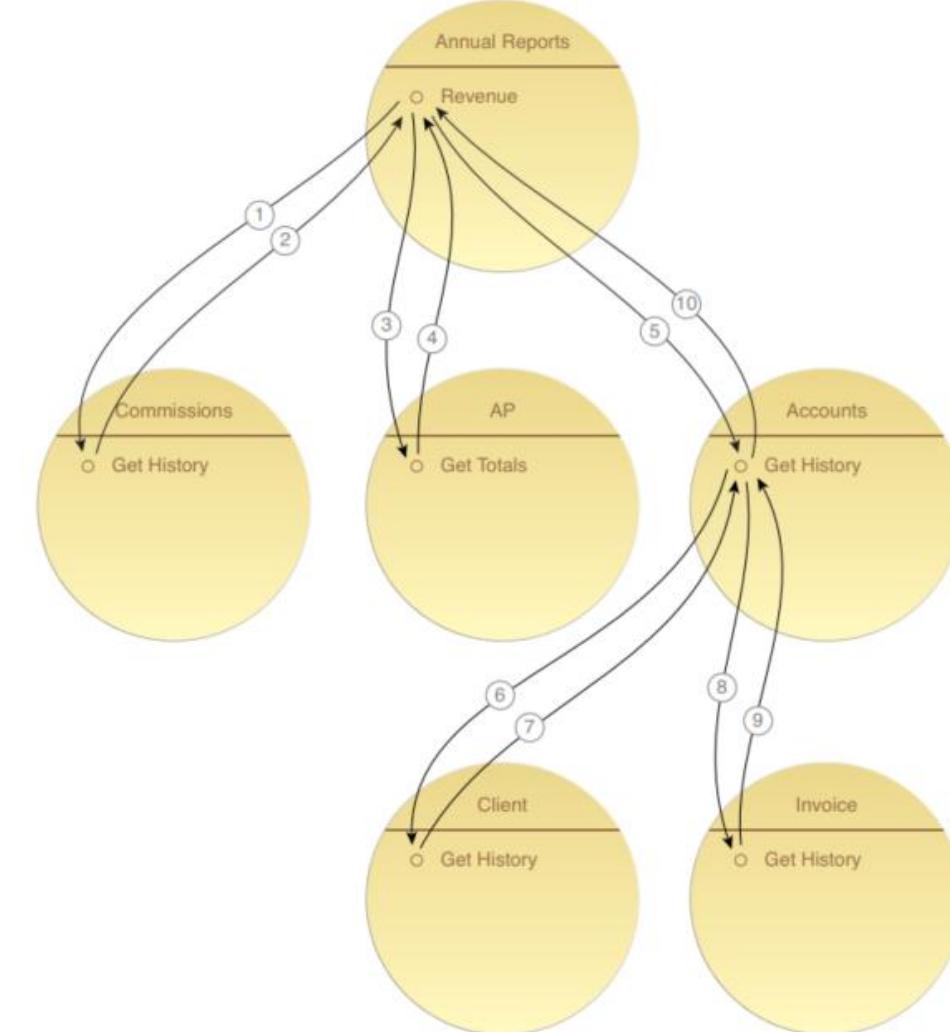
- **The design considerations emphasized by service-orientation ensure that the design of a service composition is much different than that of integrated applications**

- **The same Accounts service composition from Figure 4.14 viewed from a physical architecture perspective (Figure 4.15)**
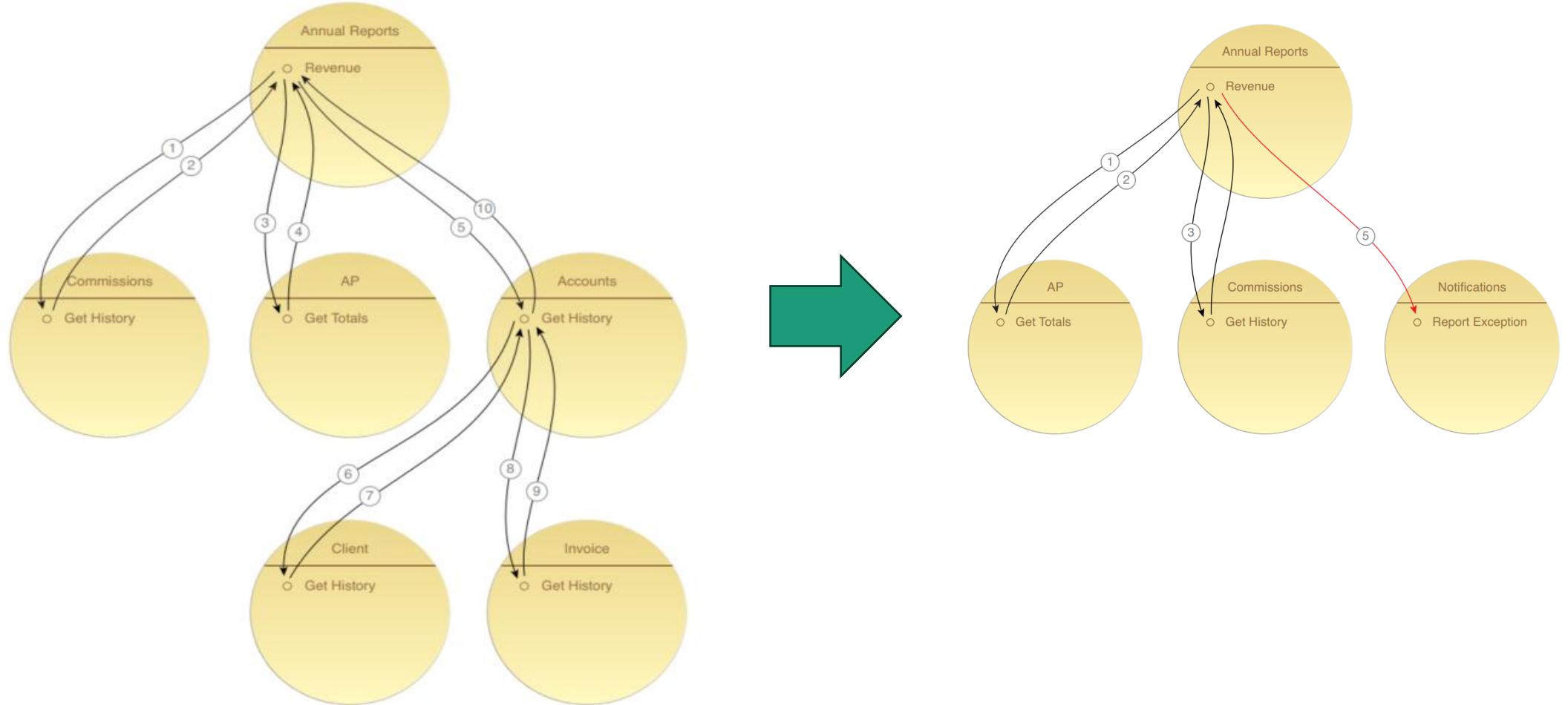
- **One composition architecture may contain or reference another, nested.**

- **Figure 4.17, the Accounts service finds itself nested within the larger Annual Reports composition**

- The business process the service is required to automate may involve the need for composition logic capable of dealing with multiple runtime scenarios

- **Ideally, the service inventory is first conceptually modeled, leading to the creation of a service inventory blueprint**

- **This blueprint is often defining the required scope of the architecture type (Figure 4.19)**

Figure 4.19

25
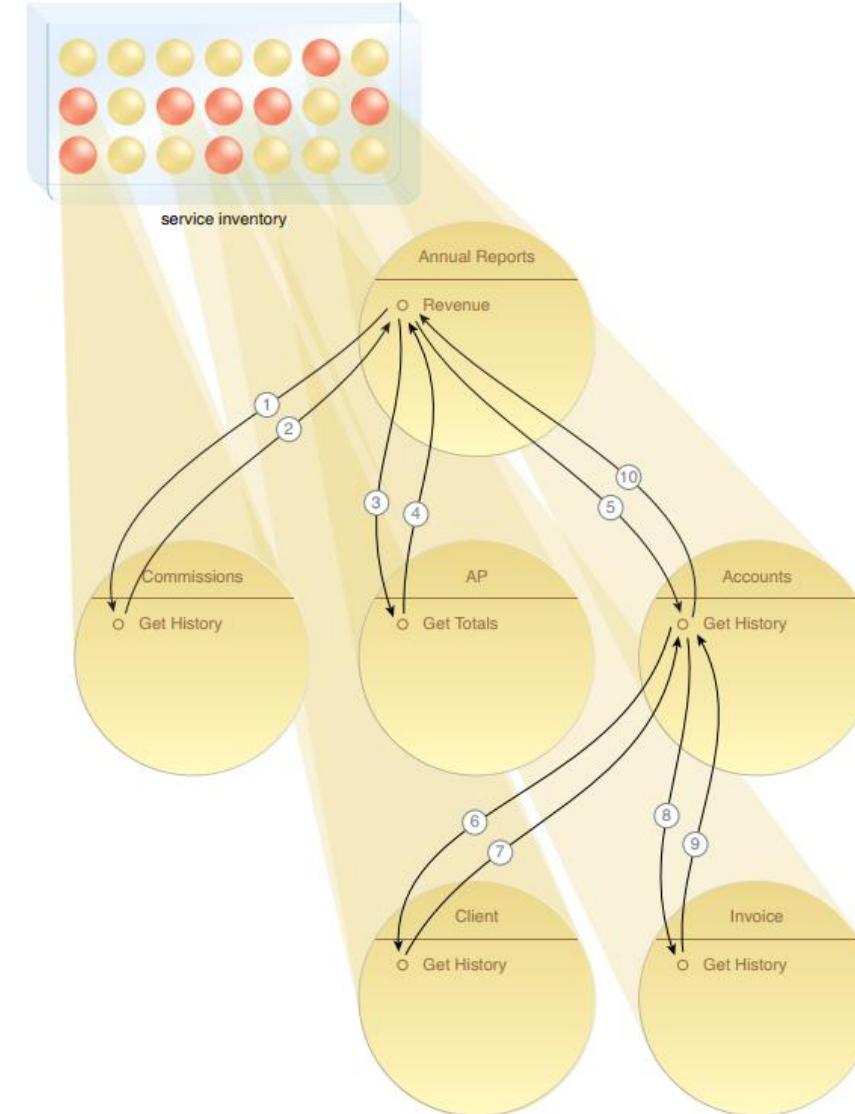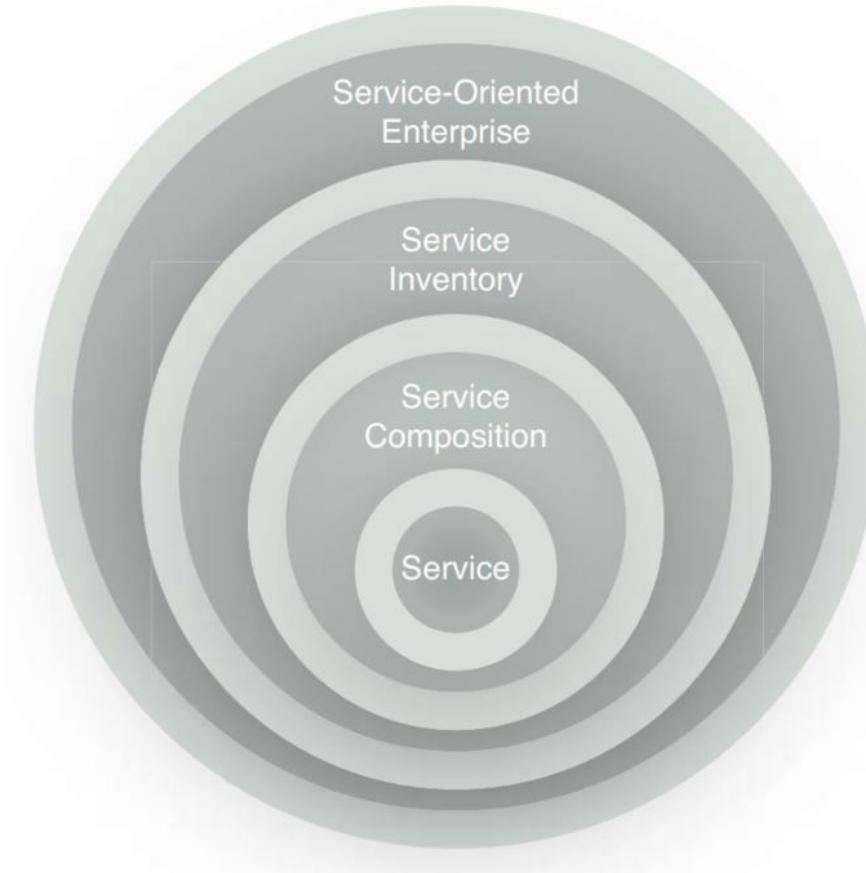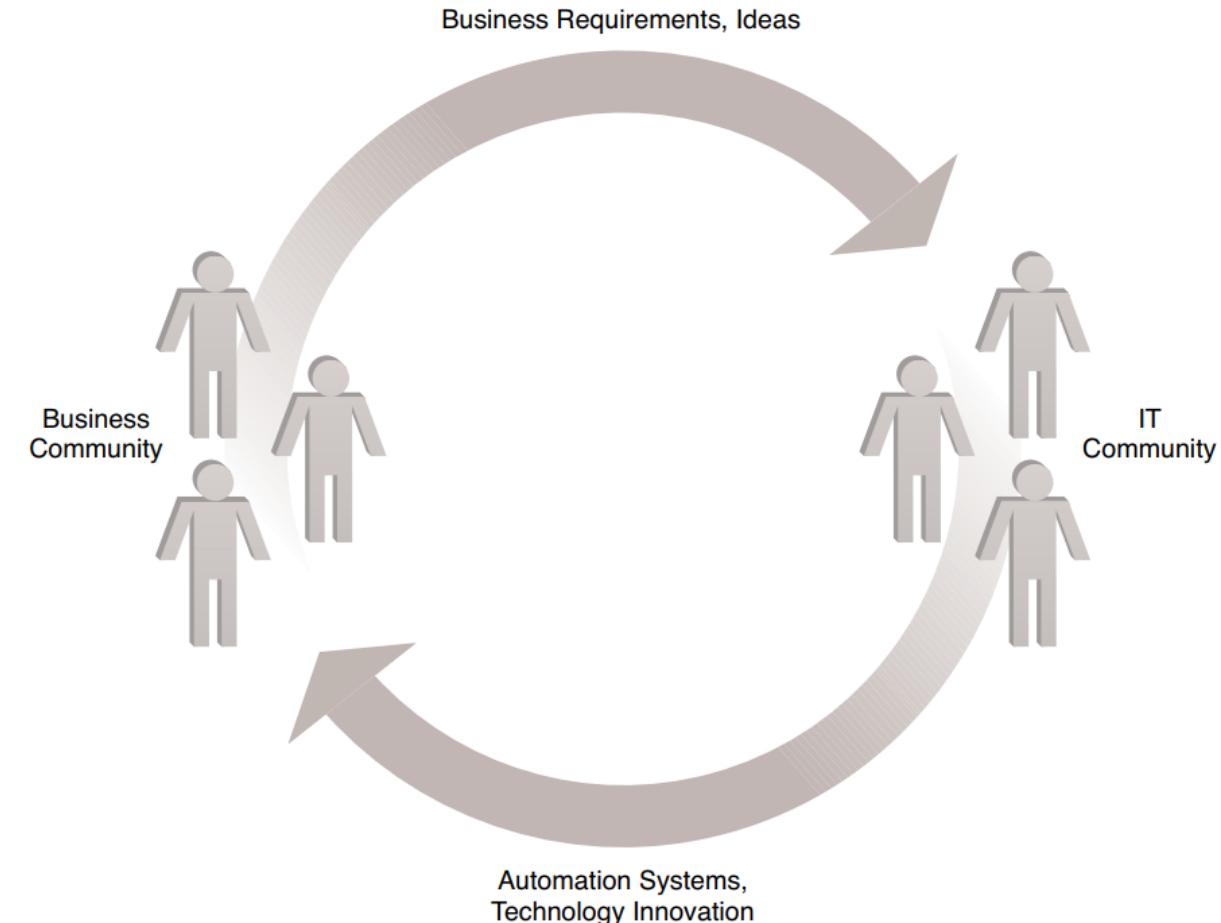
- **This form of technology architecture essentially represents all service, service composition, and service inventory architectures that reside within a specific enterprise**

- **This section is focused on technology architecture. However, a "complete" service-oriented enterprise architecture would encompass both the technology and business architecture of an enterprise**

4.3. The End Result of Service-Orientation and SOA
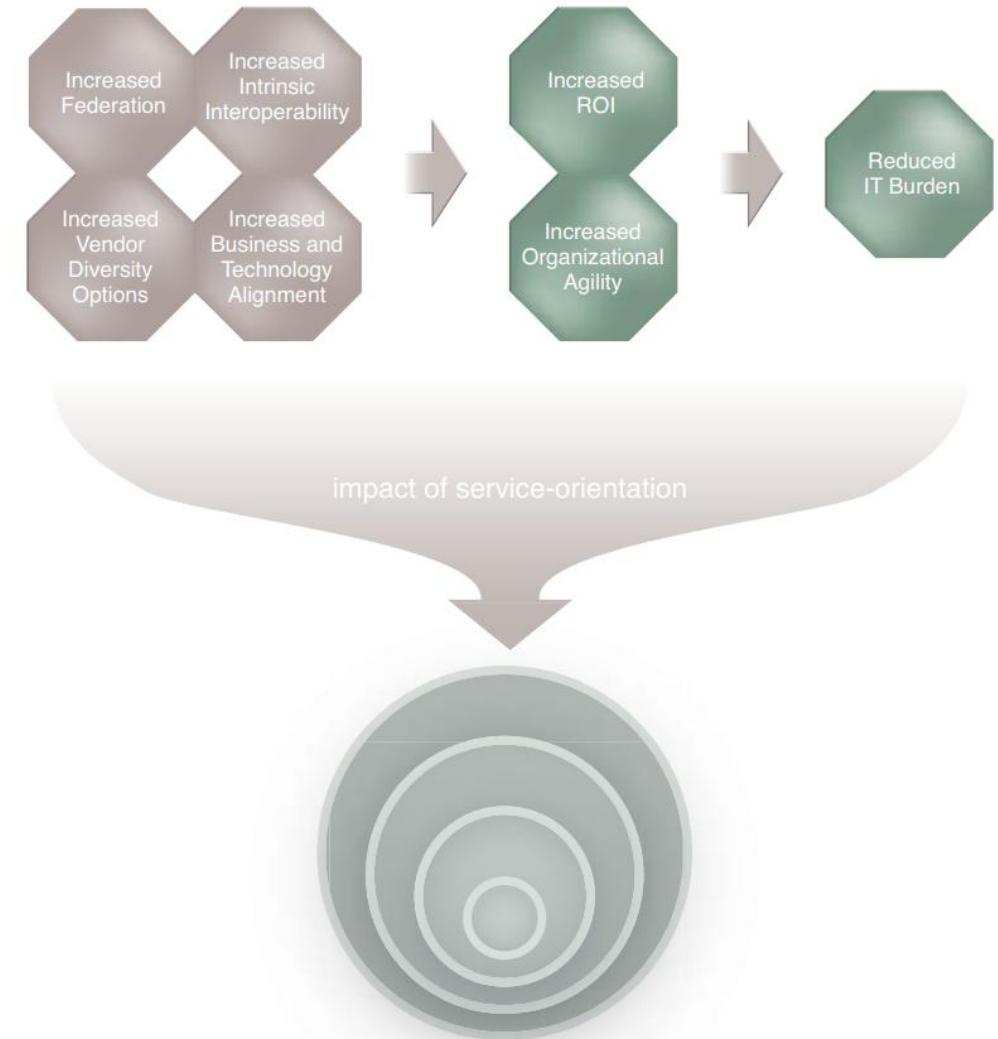
- **The endless progress cycle establishes the dynamics between the business and IT communities**

- **The IT industry has been through the cycle depicted in Figure 4.20 many times.**

- **Service-oriented computing is no exception.**

Business Requirements, Ideas

Business Community

IT Community

Automation Systems, Technology Innovation
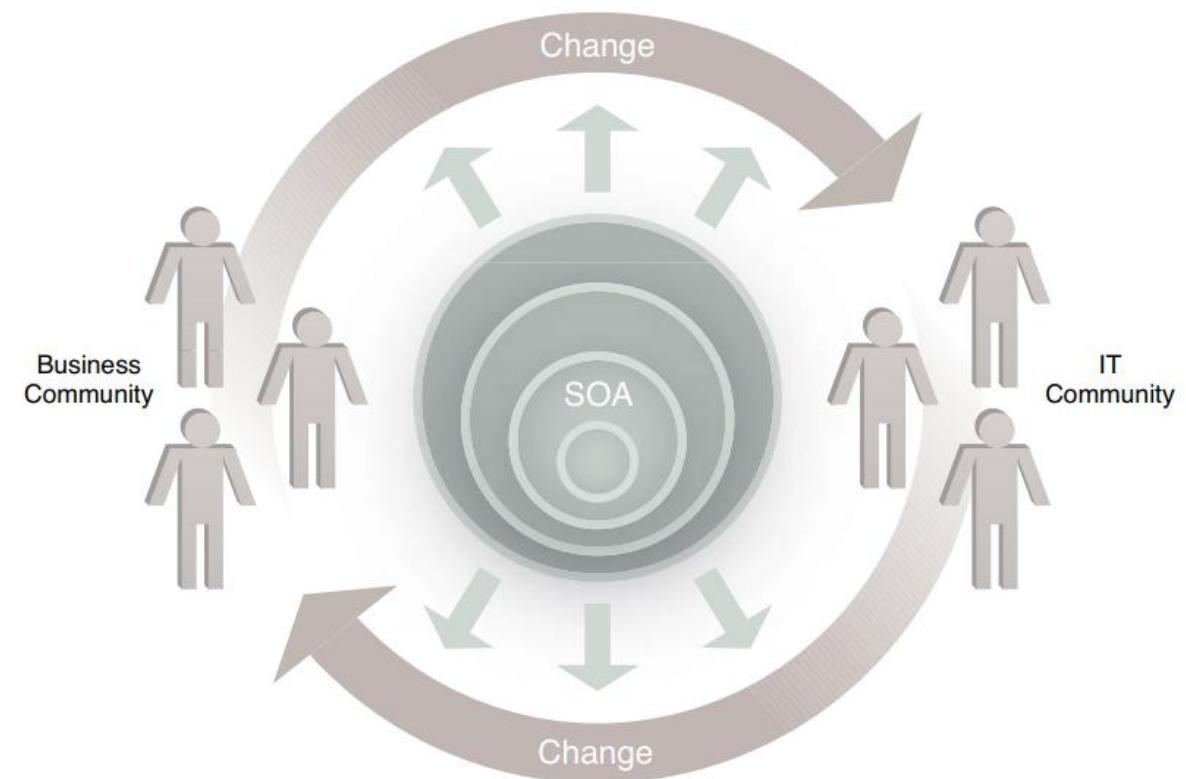
- **How then does this relate to service-oriented technology architecture?**

- **-> The common strategic goals and benefits of service-oriented computing are realized through the application of service-orientation (Figure 4.21)**
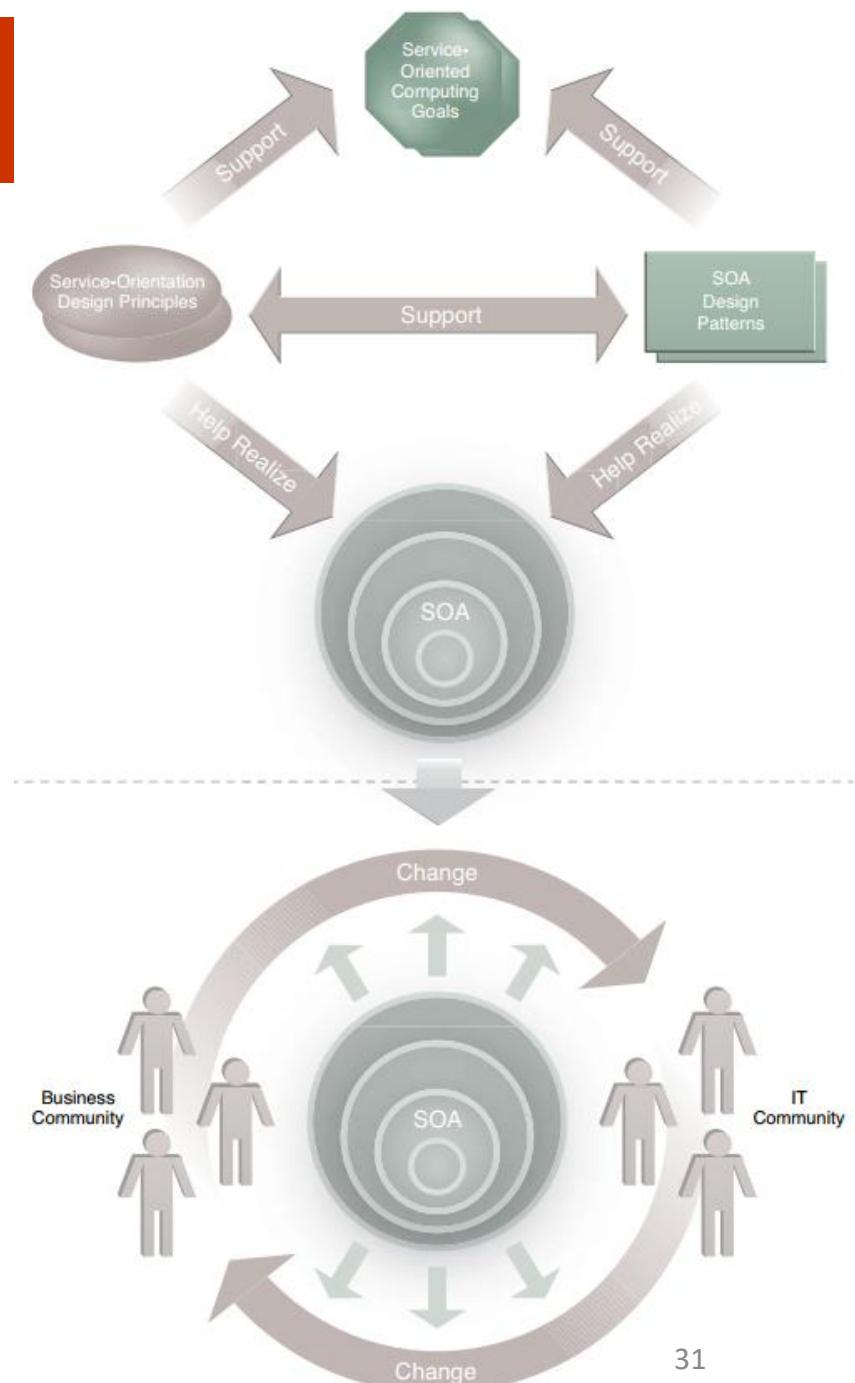
- **Service-oriented technology architecture supports the two-way dynamic between business and IT communities**
  - allowing each to introduce or change throughout an endless cycle.

30

The End Result of Service-Orientation and SOA
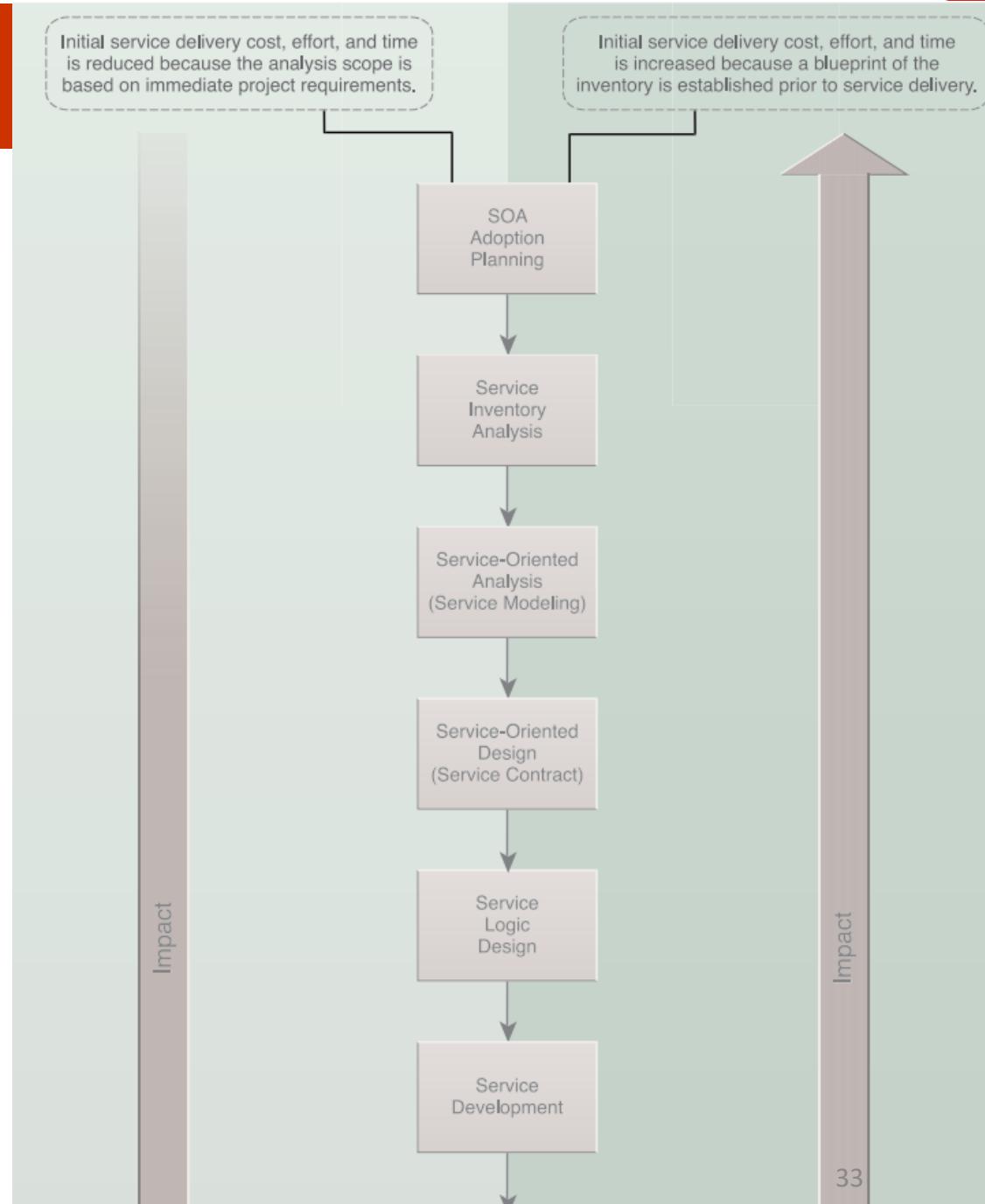
- The strategic goals of service-oriented computing can be achieved through by application service-orientation principles and SOA design patterns

- The successful application of service-orientation principles and supporting SOA design patterns helps to shape and define requirements for different types of service oriented architectures

-> Resulting in an IT automation model that is designed to fully support the two-way cycle of change through which business and IT communities continually transition

4.4. SOA Project and Lifecycle Stages

Methodology and Project Delivery Strategies
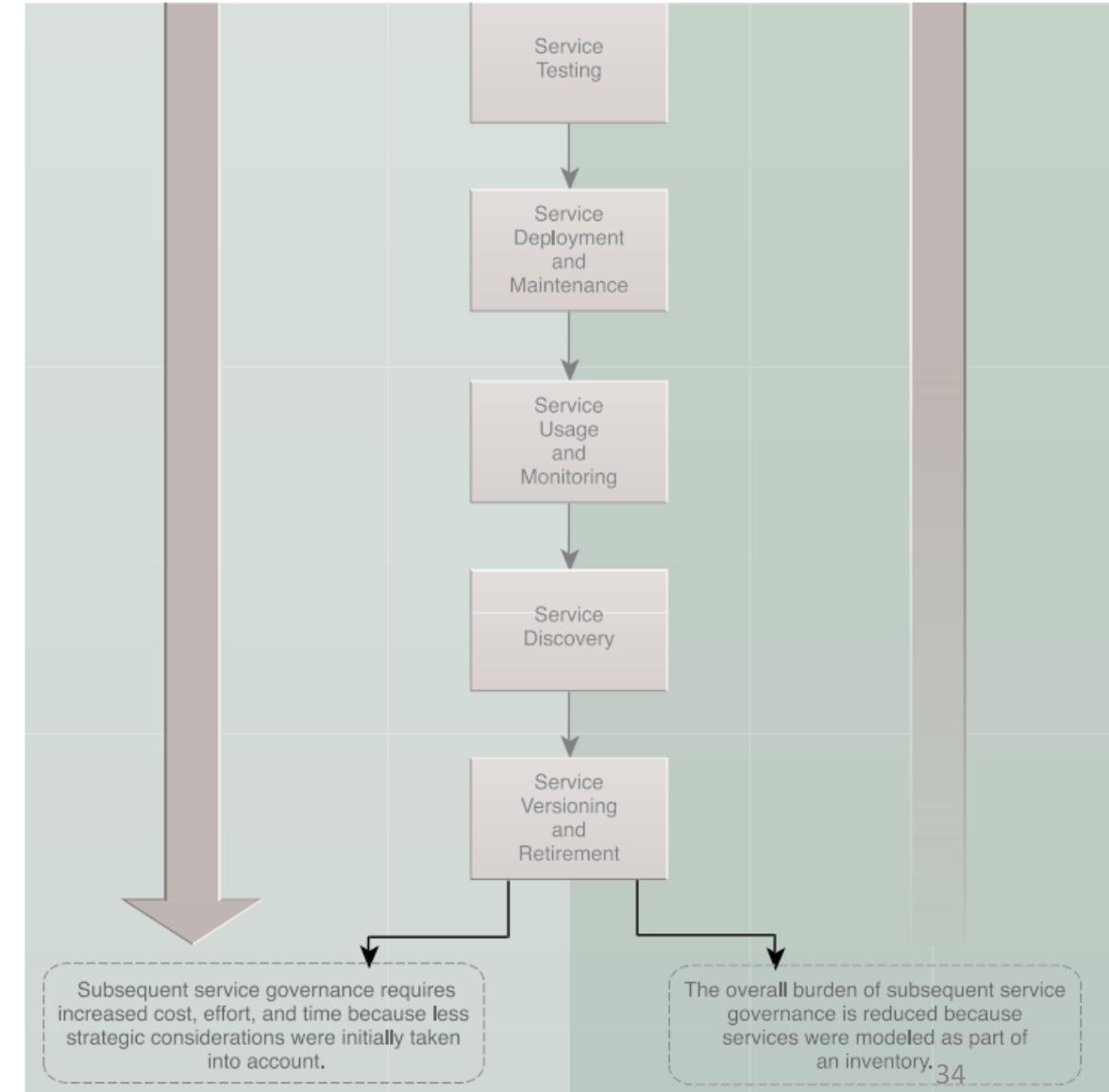
- **Several project delivery approaches can be employed.**

- **Bottom-up strategy**
  - Focus on implementing immediate business requirements as the main priority and goal of the project

- **Top-down strategy**
  - Complete an inventory analysis before the actual design, development, and delivery of services.

Initial service delivery cost, effort, and time is reduced because the analysis scope is based on immediate project requirements.

Initial service delivery cost, effort, and time is increased because a blueprint of the inventory is established prior to service delivery.

SOA Adoption Planning

Service Inventory Analysis

Service-Oriented Analysis (Service Modeling)

Service-Oriented Design (Service Contract)

Service Logic Design

Service Development

Impact

Impact

33

- **Generally, the less time and effort spent on the service analysis -> the greater post-deployment governance burden.**

Service
Testing

Service
Deployment
and
Maintenance

Service
Usage
and
Monitoring

Service
Discovery

Service
Versioning
and
Retirement

Subsequent service governance requires
increased cost, effort, and time because less
strategic considerations were initially taken
into account.

The overall burden of subsequent service
governance is reduced because
services were modeled as part of
an inventory.

34

SOA Project Stages

- **The common and primary stages related to SOA project delivery and the overall service delivery lifecycle**
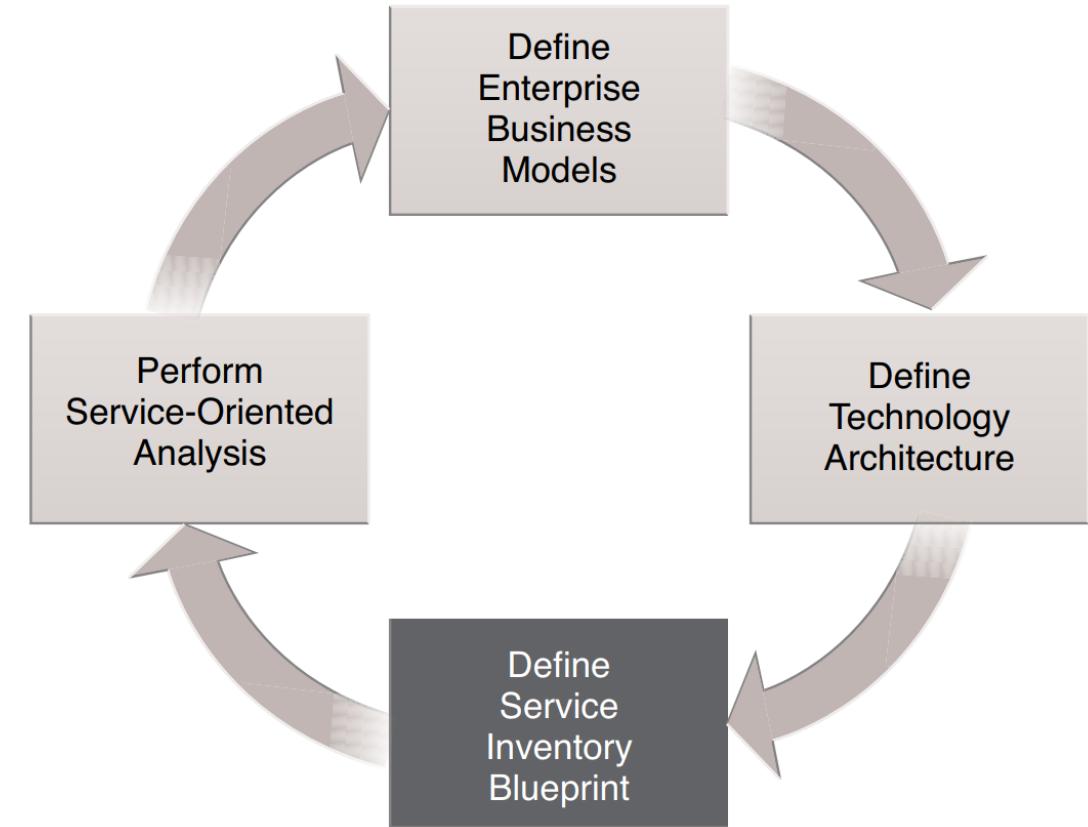
- **Terms**
  - SOA project stages
  - Service delivery project stages (except deploy, maintenance, and monitoring)
  - Service lifecycle stages (except planning and inventory analysis)

SOA Adoption Planning → Service Inventory Analysis → Service-Oriented Analysis (Service Modeling) → Service-Oriented Design (Service Contract)

Service Logic Design → Service Development → Service Testing → Service Deployment and Maintenance

Service Usage and Monitoring → Service Discovery → Service Versioning and Retirement

Service Inventory Analysis
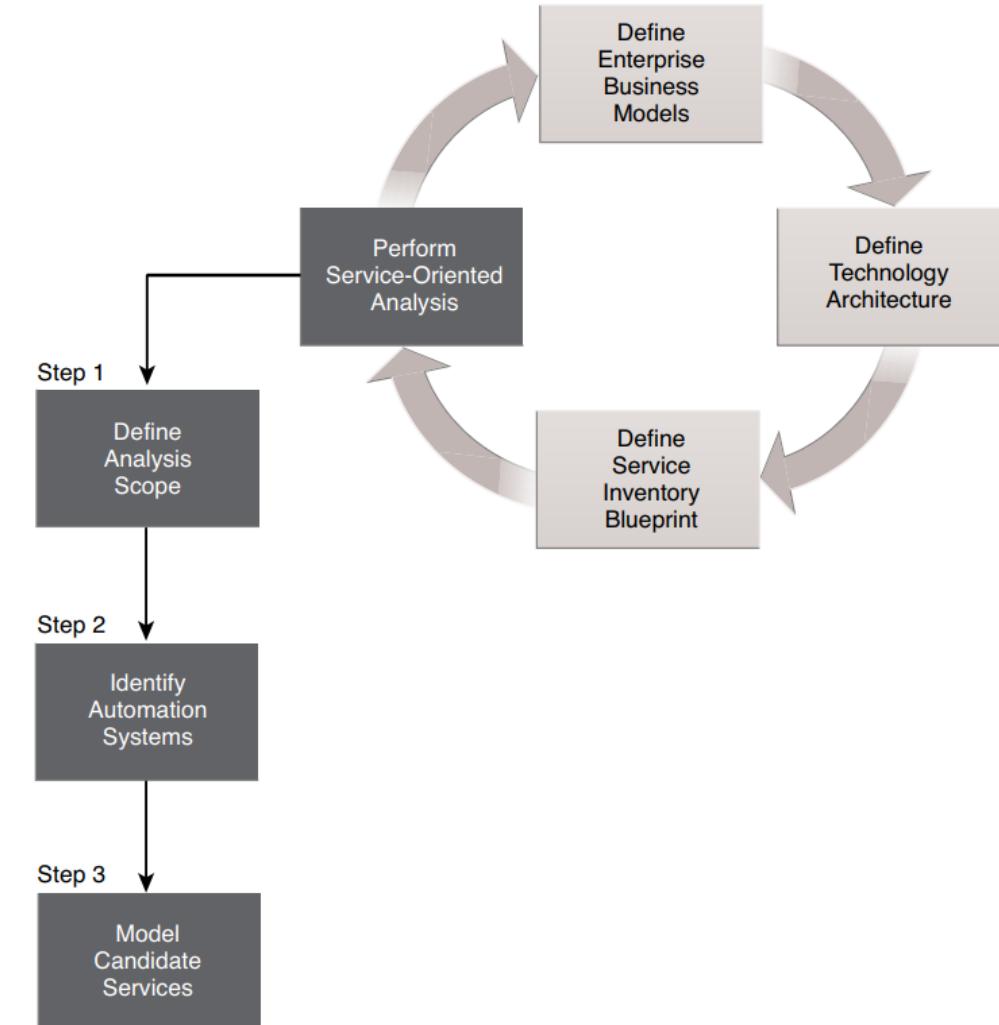
- **This service inventory analysis stage aims to define an inventory of services at the concept level**

- **The service inventory analysis cycle**
  - Define Enterprise Business Models
  - Define Technology Architecture
  - Define Service Inventory Blueprint
  - Perform Service-Oriented Analysis

- **The service inventory blueprint is incrementally defined as a result of repeated iterations of steps that include the service-oriented analysis**
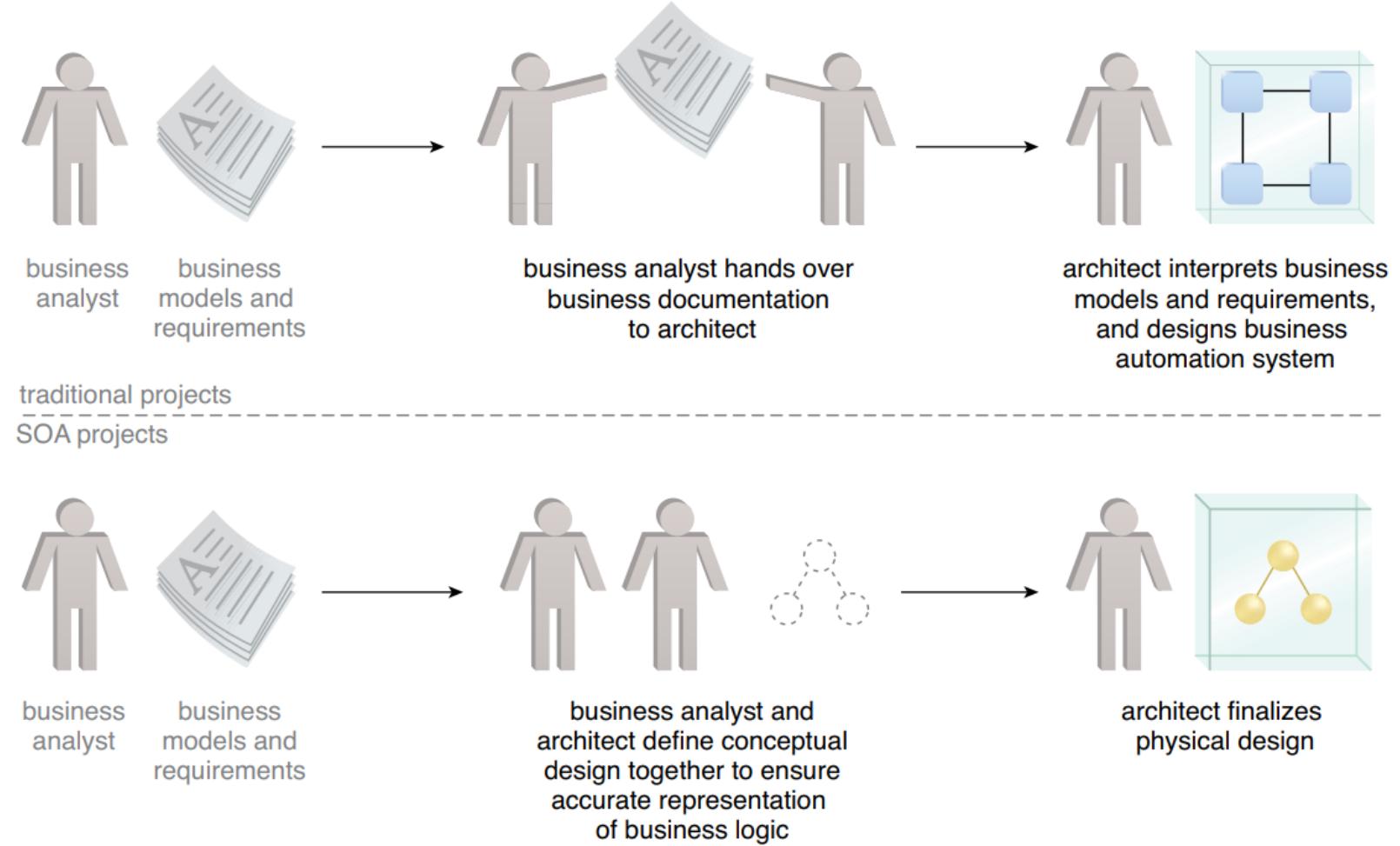
- **The service-oriented analysis process is generally performed iteratively**
 - Steps 1 and 2: Information-gathering tasks, that are performed in preparation for the modeling process performed in Step 3.

- **Step 1:** Define Business Automation Requirements

- **Step 2:** Identify Existing Automation Systems

- **Step 3:** Model Candidate Services

Service-Oriented Analysis (Service Modeling)
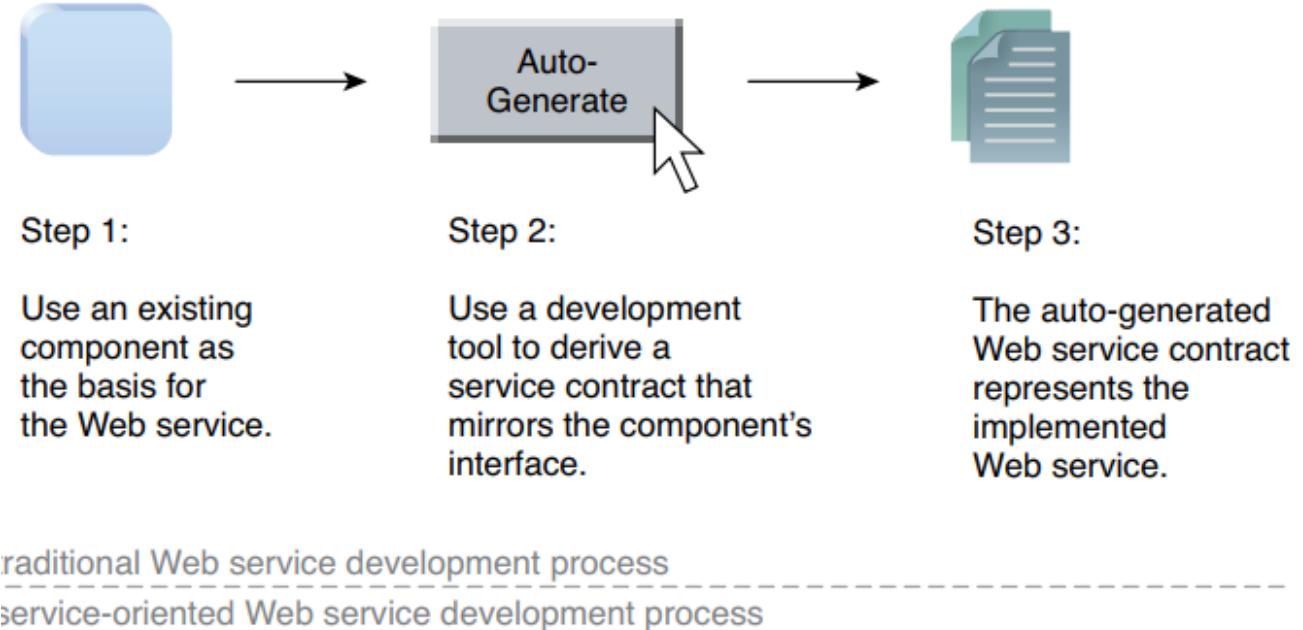
- **A key success factor of the service-oriented analysis process is the hands-on collaboration of both business analysts and technology architects**

traditional projects

business analyst → business models and requirements → business analyst hands over business documentation to architect → architect interprets business models and requirements, and designs business automation system

SOA projects

business analyst → business models and requirements → business analyst and architect define conceptual design together to ensure accurate representation of business logic → architect finalizes physical design

39

- **The service-oriented design phase represents a service delivery lifecycle stage dedicated to producing service contracts**

Step 1:

Use an existing component as the basis for the Web service.

Step 2:
Auto-Generate

Use a development tool to derive a service contract that mirrors the component's interface.

Step 3:

The auto-generated Web service contract represents the implemented Web service.

raditional Web service development process

service-oriented Web service development process
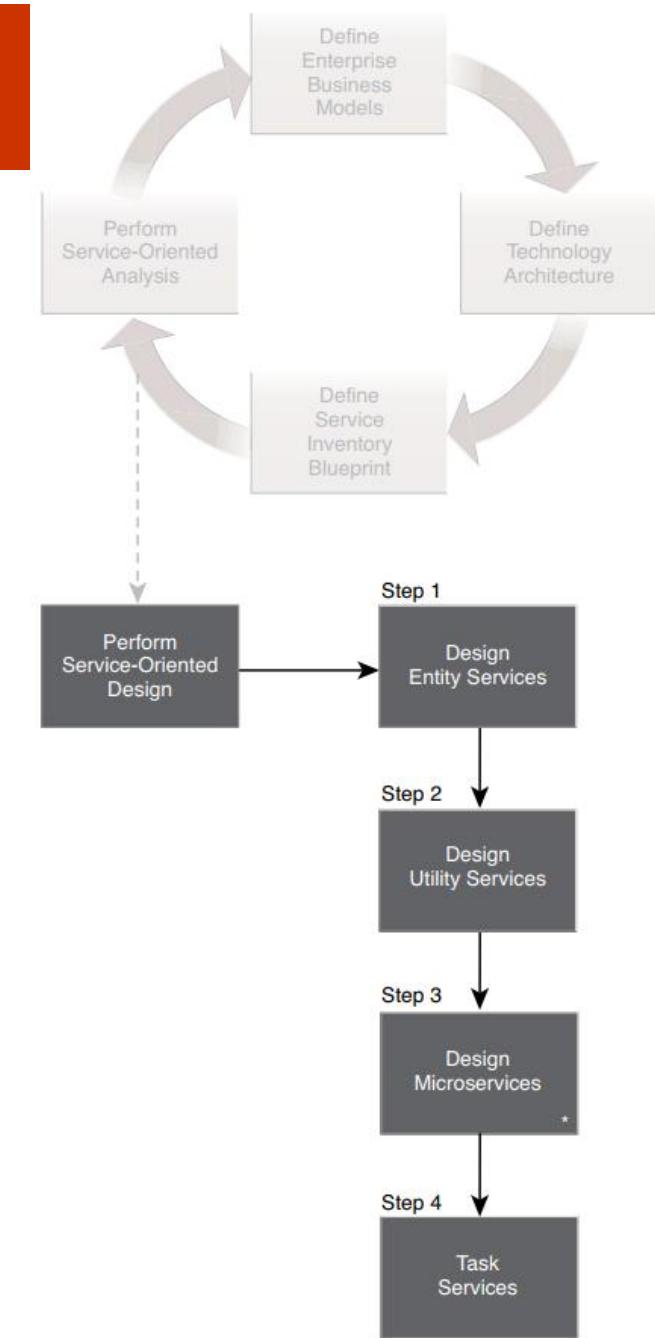
- **"Contract-first" approach to software development (Figure 4.29)**

Step 1:

Custom design the Web service contract.

Step 2:
Import

Import the Web service contract into a development environment.

Step 3:

Build the underlying solution logic in support of the pre-defined Web service contract.

40

- **The input for the service-oriented design process is list a service candidate that was produced by the service-oriented analysis process**

Define
Enterprise
Business
Models

Perform
Service-Oriented
Analysis

Define
Technology
Architecture

Define
Service
Inventory
Blueprint

Perform
Service-Oriented
Design

Step 1
Design
Entity Services

Step 2
Design
Utility Services

Step 3
Design
Microservices

Step 4
Task
Services

41

Service Deployment and Maintenance
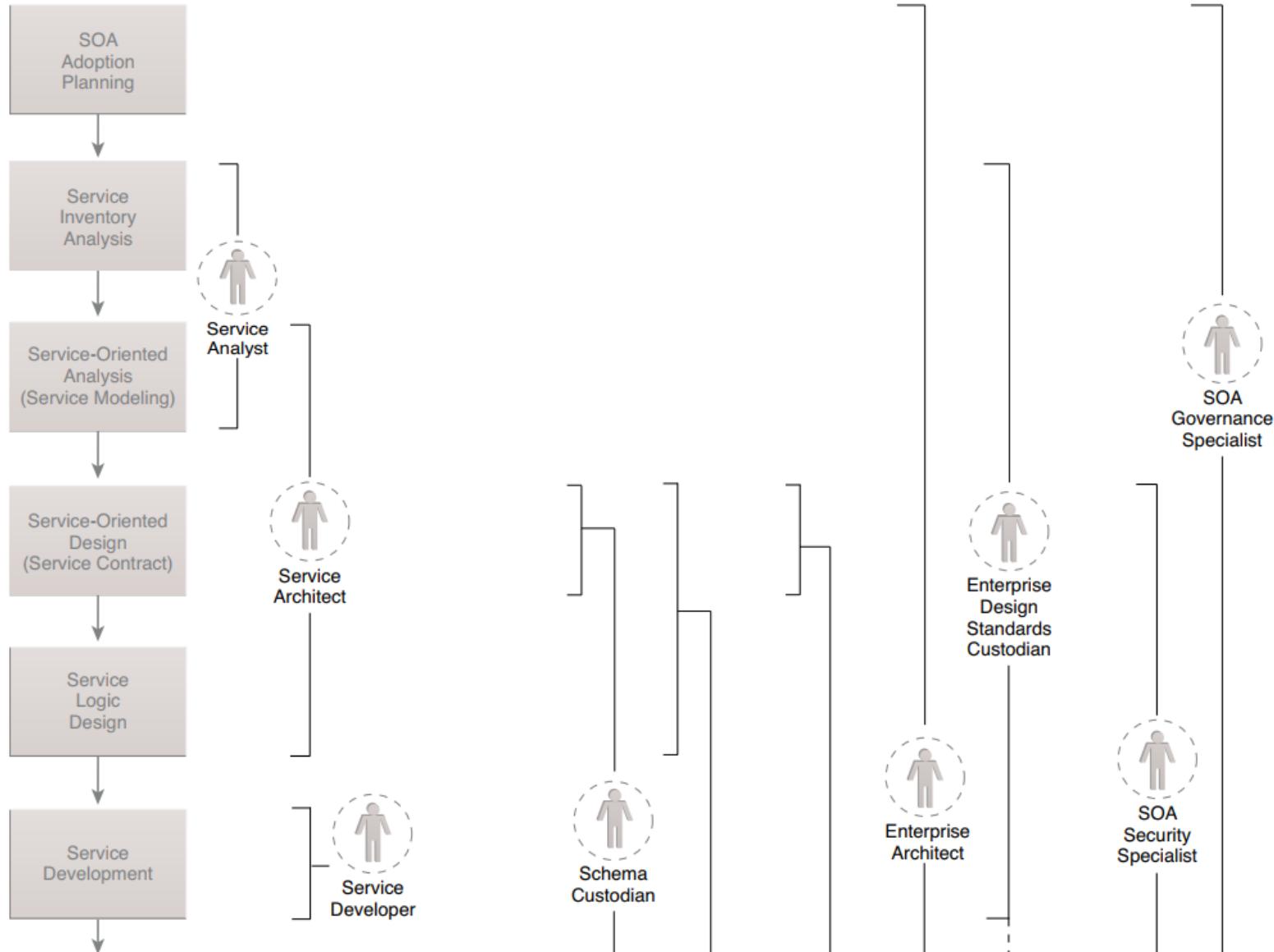
- **Service deployment represents the actual implementation of a service into the production environment**
  - Middleware (such as ESB and orchestration platforms)
  - Cloud service implementation considerations
  - Cloud-based IT resources encompassed by an on-premise or cloud-based service
  - On-demand and dynamic scaling and billing configurations
  - Proprietary runtime platform extensions
  - Administration and monitoring products

- **Service maintenance refers to upgrades or changes that need to be made to the deployment environment.**

45

- To ensure that the versioning of a service can be done with minimal impact and disruption to service consumers

-> Need a formal service versioning process

- There are different versioning strategies, each of which introduces its own set of rules and priorities when it comes to managing the backward and forward compatibilities of services

# Project Stages and Organizational Roles
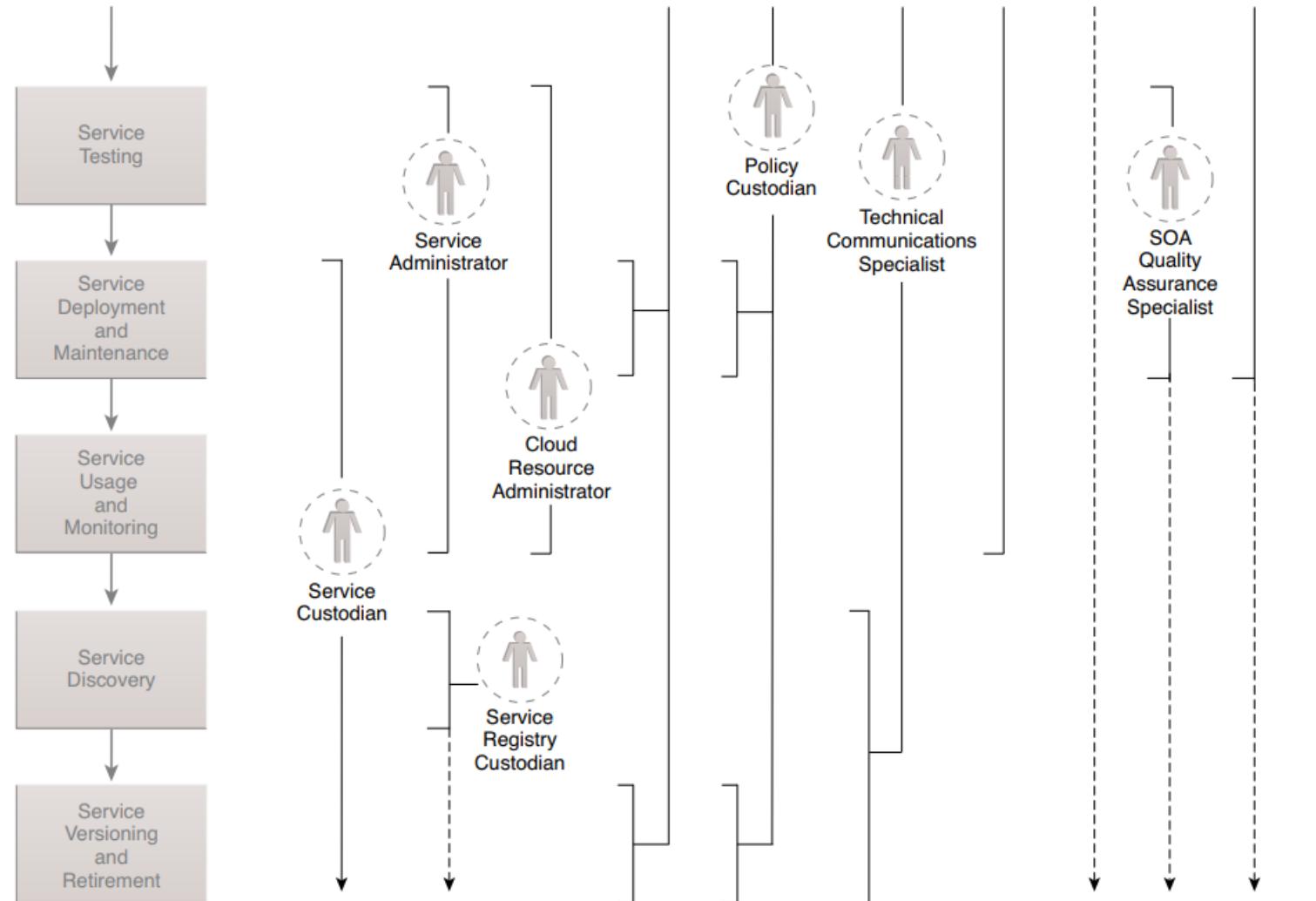
Figure 4.31

Shown here are common associations of organizational roles with different SOA project stages.

Q & A