NORTHROP GRUMMAN

The Enterprise Architecture – Service Oriented Architecture – Systems Engineering Relationship

July 15, 2009

Janalea M. Milford, CEA

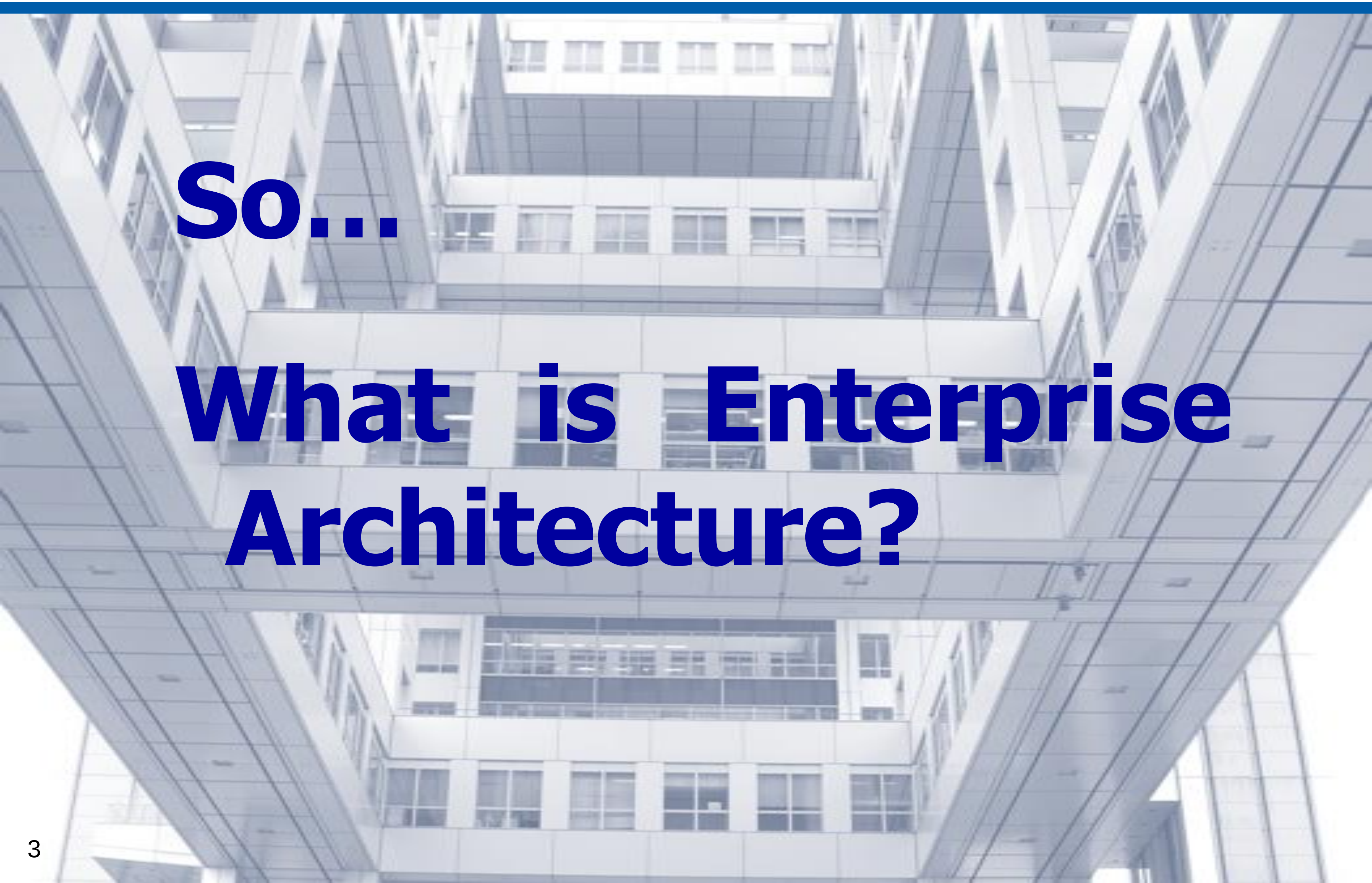
Certified Enterprise Architect
Sr. Systems Engineer

Take Away from This Presentation

A Better Understanding of:

- ❑What is Enterprise Architecture (EA)
- ❑What is Service Oriented Architecture (SOA)
- ❑What is the SOA – EA Relationship
- ❑Inherent Challenges and Risks SOA brings to The As-Is EA
- ❑Inherent Challenges and Risks SOA brings to The To-Be EA
- ❑The EA – SOA – Systems Engineering (SE) Relationship and the value SE adds.

What Is Enterprise Architecture?



So…

What is Enterprise Architecture?

What Is Enterprise Architecture?

John Zachman defines Enterprise Architecture (EA) as:

"Enterprise Architecture is the principle structural mechanism for:

- Establishing a basis for assimilating high rates of change
- Advancing the state-of-the-art in Enterprise design
- Managing the knowledge-base of the Enterprise
- Integrating the technology (automated and/or non-automated) into the fabric of the Enterprise"

What Is Enterprise Architecture?

OMB defines the Federal Enterprise Architecture as:

“A business-based foundation which provides a common framework for improvement in a variety of key areas such as:

- Budget Allocation
- Information Sharing
- Performance Measurement
- Budget / Performance Integration
- Cross-Agency Collaboration
- E-Government
- Component-Based Architectures.”

What Is Enterprise Architecture?

DOD defines Enterprise Architecture as:

“The structure of components, their relationships, and the principles and guidelines governing their design and evolution over time.”

What Is Enterprise Architecture?

Enterprise Architecture (EA)

- Is an approach to defining and describing organization-wide business (Business Architecture) and information technology (IT) (Technical Architecture)
- Incorporates Best Practices to optimize the operations of an organizational entity (commercial or government) and the Information Technology Infrastructure that enables the business entity to accomplish its mission.

What Is Enterprise Architecture?

The Purpose of Enterprise Architecture is to provide a way of:

•Organizing an operational entity’s complexities (e.g. people, processes, services, facts, relationships, technologies, et al) within the context of the entity’s mission and vision

•Documenting the organization’s human, intellectual, and technical capital

•Understanding the complexity of the interaction of the different Lines Of Businesses (LOBs) of an organization and its stakeholders and customers.

•Lowering Risk and Costs improving Return on Investment (ROI) and/or reducing Total Cost of Ownership (TCO)

•Becoming more agile as an organization

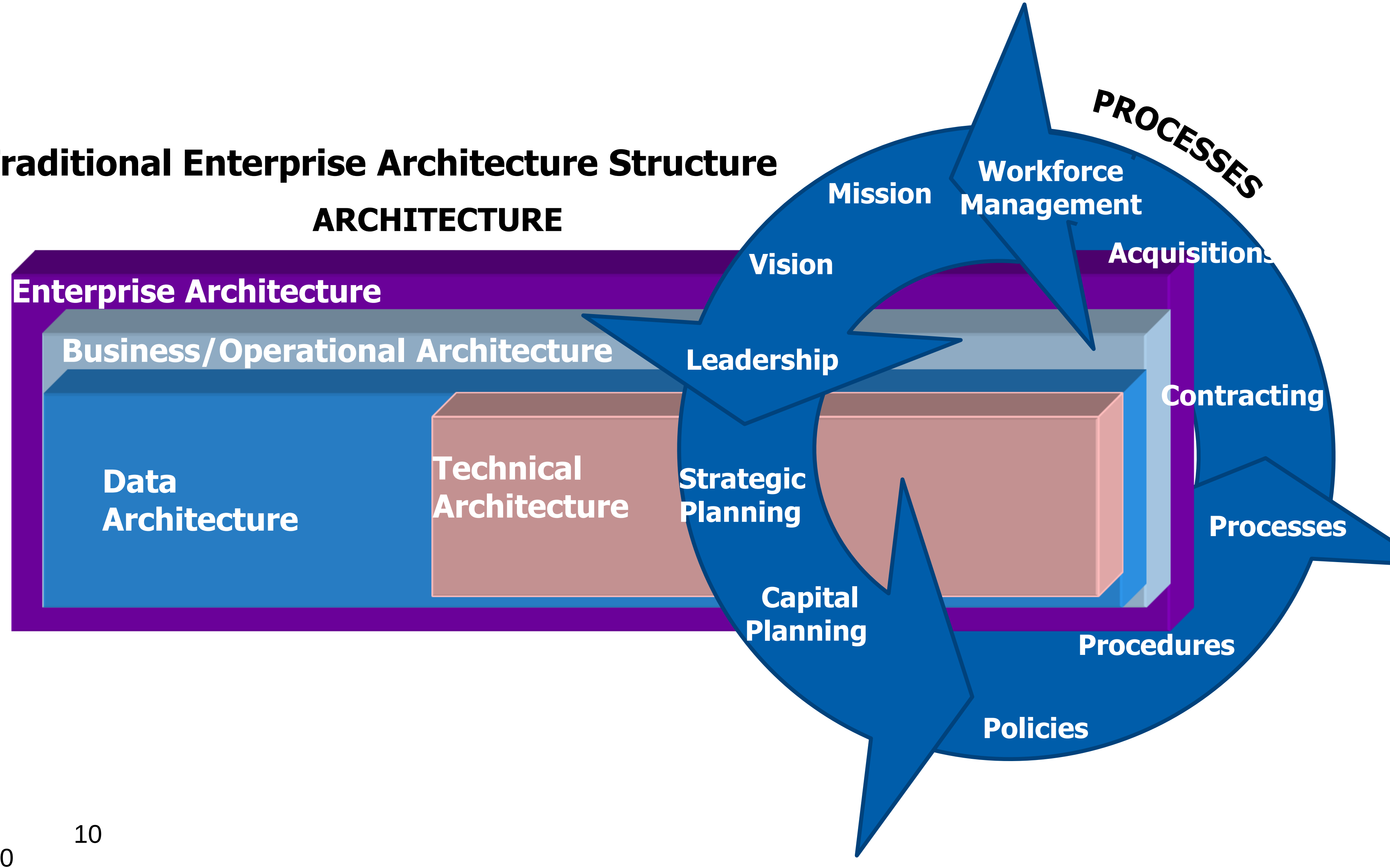
What Is Enterprise Architecture?

What are some critical *inputs* for a successful EA?

- Institutional Knowledge and Capital
  - Human (e.g. stakeholders, customers, personnel)
  - Intellectual (e.g. documentation and other forms of data)
  - Technical (Information Technology Infrastructure (ITI))
- Organizational Mission, Vision, Goals, and Objectives
- Financial Resources
- Organizational Commitment
- Processes (Capital Planning, Governance, Systems Engineering, etc.)

What Is Enterprise Architecture?
EA and the Enterprise

Traditional Enterprise Architecture Structure

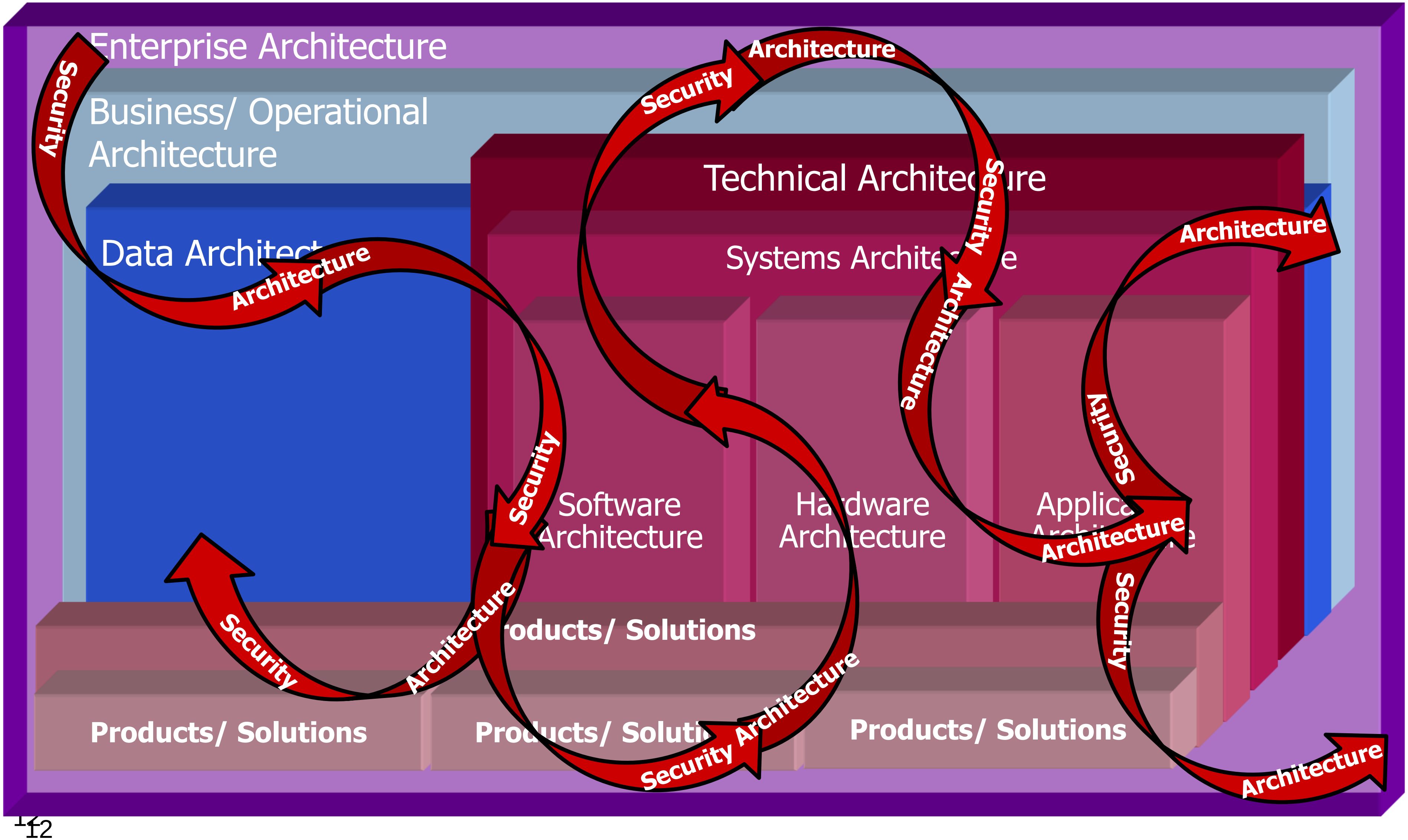


What Is Enterprise Architecture?

What are some of the traditional Architecture Components (Subject Areas) that make up the EA?

- Business Architecture
- Data Architecture
- Technical Architecture
  - Systems Architecture
  - Software Architecture
  - Hardware Architecture
  - Application Architecture
- Product/Solution Architecture
- Security Architecture

What Is Enterprise Architecture? How EA Components Relate to Each Other

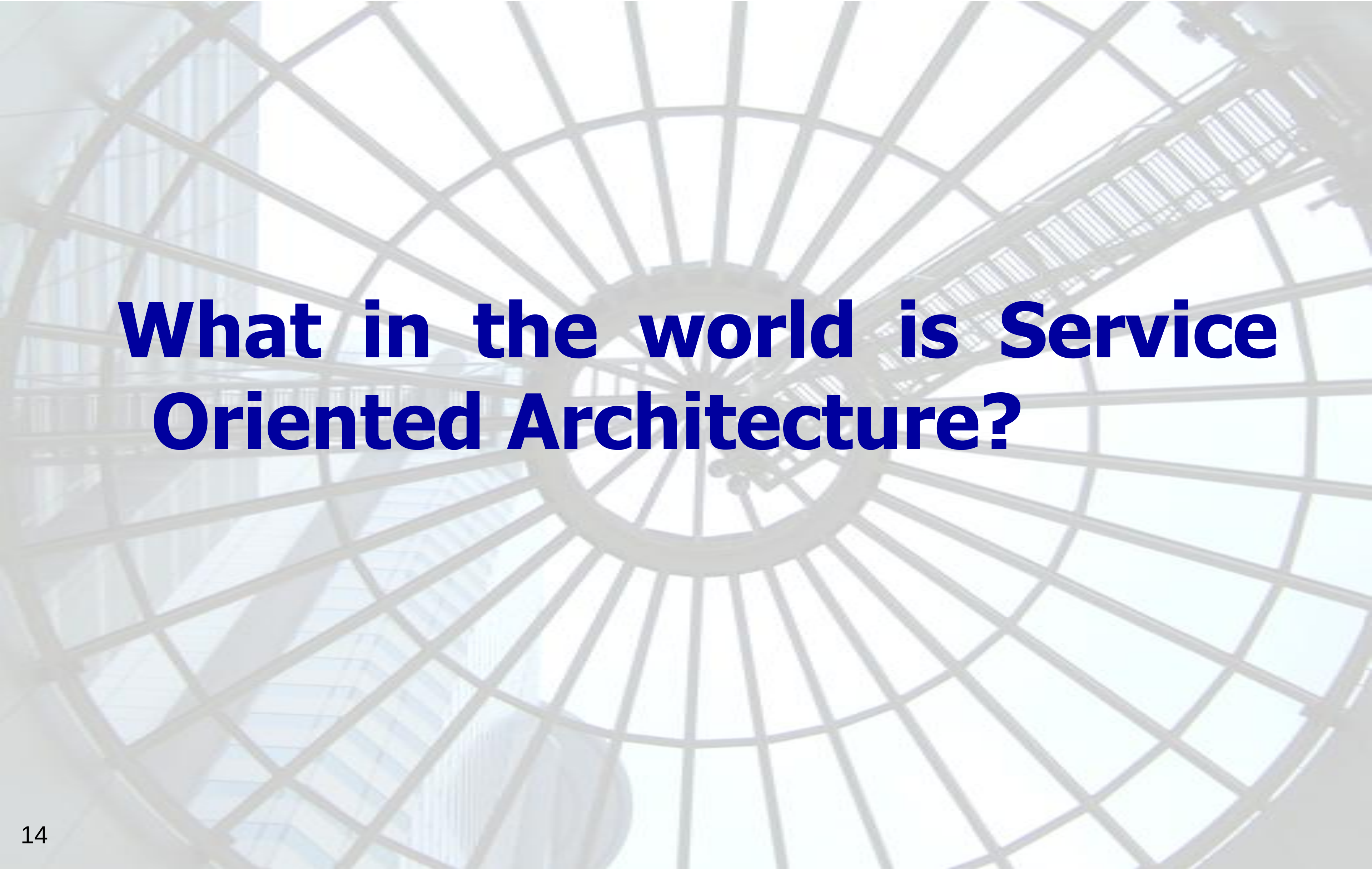


What Is Enterprise Architecture?

What are Some Critical EA Tools?

- Communications
- Governance
- Processes and Methodologies (e.g. SE, FDD, et al)
- Frameworks (e.g. Zachman, FEAF, DODAF, etc.)
- Measurements (e.g. Performance (business and technical)
- EA Applications and Repositories (System Architect, METIS, Rochade, et al)

What is Service Oriented Architecture?



What in the world is Service Oriented Architecture?

What is Service Oriented Architecture?

NORTHROP GRUMMAN

General Definition

SOA is an architectural and design discipline conceived to enable:

- Increased business & technology domain alignment
- Increased interoperability (flexibility[1], reliability[2], composability[3], reusability[4])ⓘ
- Increased federation (uniting resources & applications while maintaining their individual autonomy & self-governance)
- Increased agility and flexibility through the loose-coupling of services and technical service components enabling the business to respond and react more rapidly to a dynamic (quickly changing) environment

What is Service Oriented Architecture? SOA Evolution – The Early Days

- Service-oriented architectures are not a new thing. Some early and more immature service-oriented architecture-like approaches were attempted with the use DCOM (Distributed Component Object Model) or Object Request Brokers (ORBs) based on the CORBA (Common Object Request Broker Architectures) specification.

- Many software vendors such as SAP, Siebel, and IBM began developing integrated modules based on discrete functional capabilities designed to support and enable specific business functions before SOA became the latest technology ‘buzz-word’.

What is Service Oriented Architecture? SOA Evolution – The Early Days

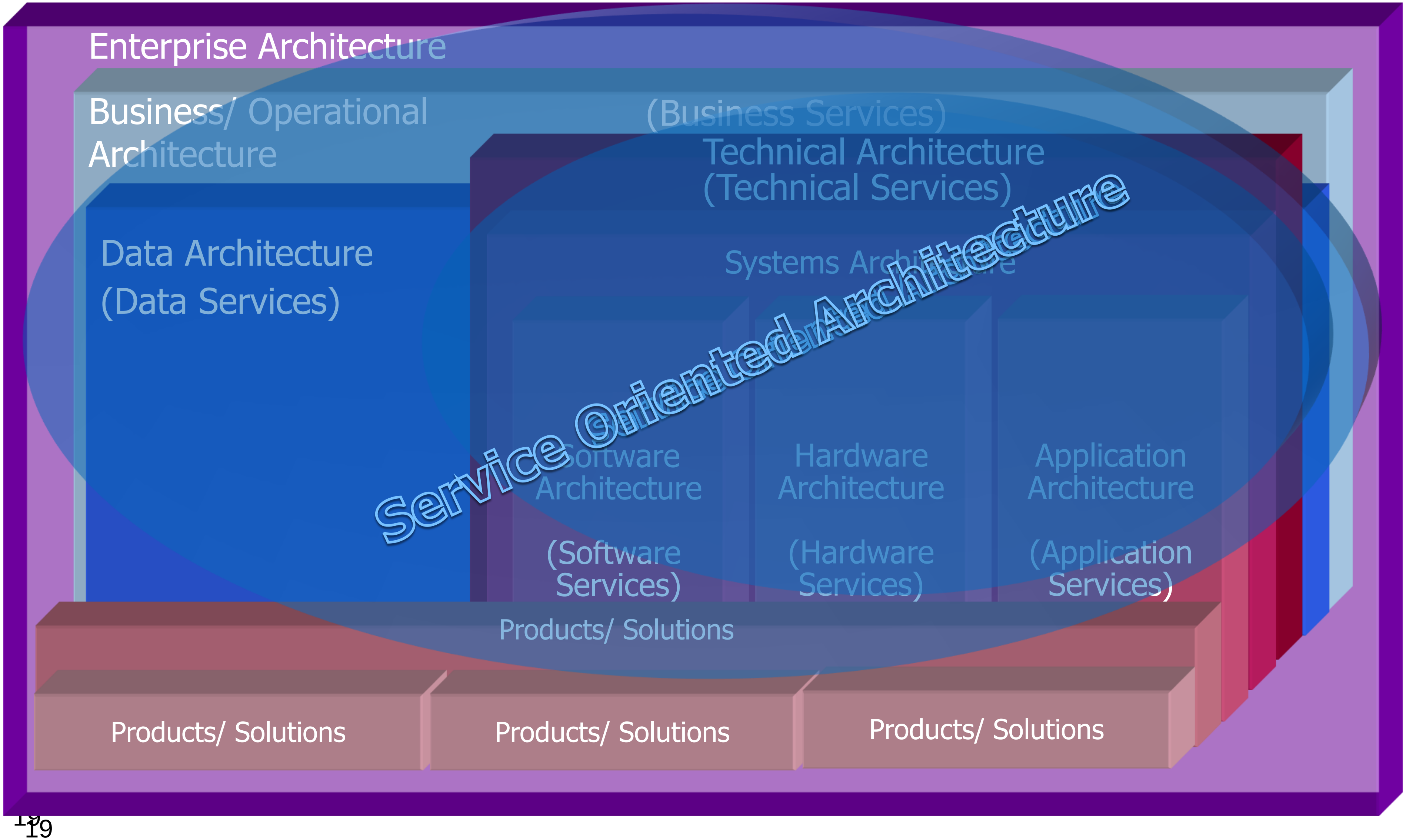
- Early Adoption of SOA was focused on IT Services, primarily software

- Major goals of SOA:
  - Realize business and IT benefits (translate this to ROI and/or TCO)
  - Create software ***services*** that
    - Adhere to principles of service-orientation
    - Are interconnected and interdependent building blocks
    - Provide an ease of connection and integration (COTS or custom) (*plug-and-play)*
    - Are grouped into well defined discrete sets of capabilities
    - Are consistent with the business activities that the software services support and enable

What is Service Oriented Architecture? SOA Evolution – Today and Future

- SOA has been maturing in the same way as almost all technology paradigm shifts. From the bottom up…
- SOA has been gaining ground as a mechanism for:
  - Defining flexible business services by categorizing like operational activities into the business services
  - Defining IT (technical) services to align with the business services and satisfy actual business requirements
  - Providing a flexible and standardized IT foundation and structure enabling the business to adapt quickly
  - Providing a standard set of business and technical services to improve ROI and/or TCO

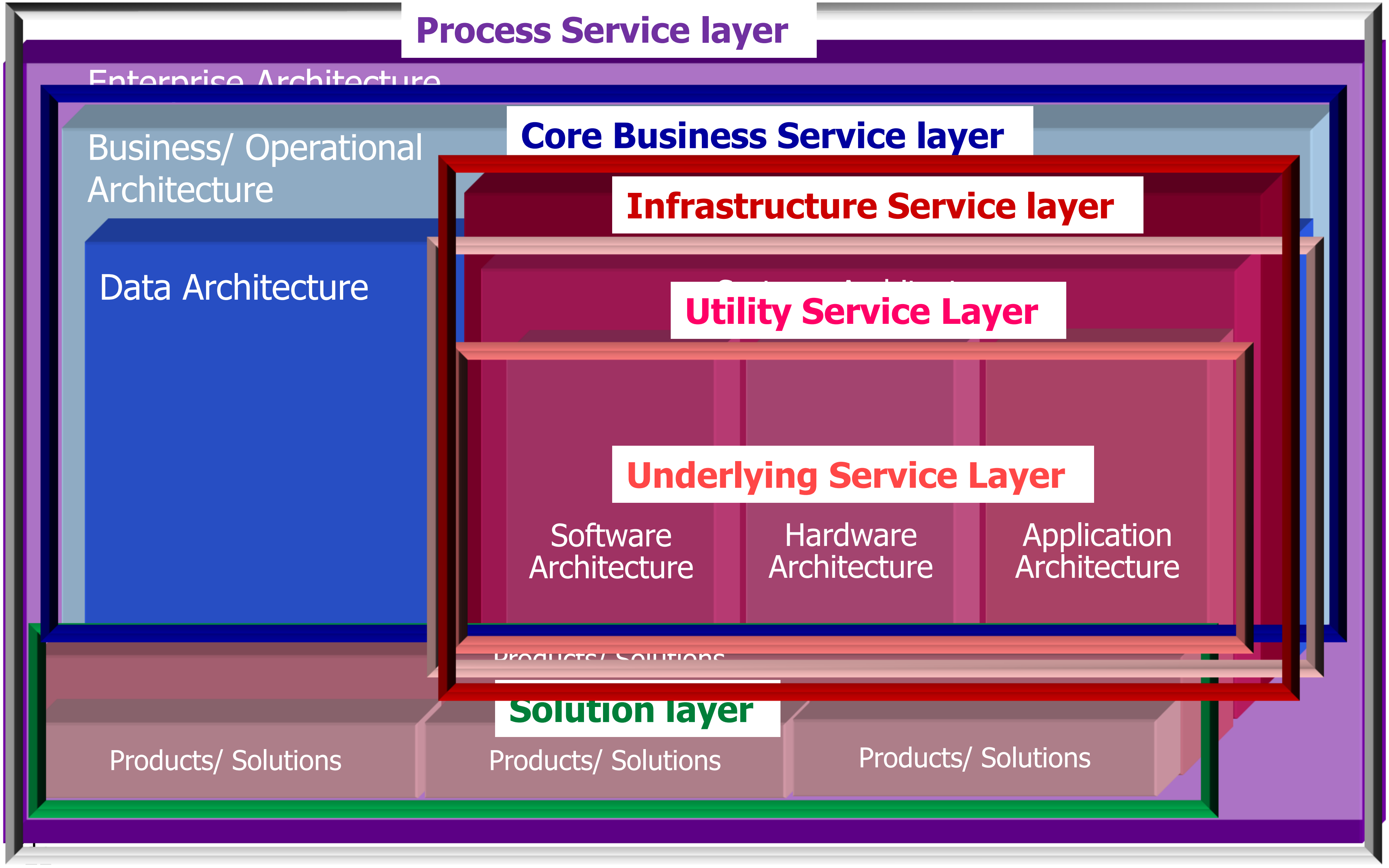
EA-SOA Relationship



The SOA – EA Relationship
How/Where SOA fits into EA?

- Process Service layer (value chain rules, coordinates other services)
- Core Business Service layer (reusable enterprise services, maintain data)
- Underlying Service layer (services that need a façade)
- Utility Service layer (commonly needed logic, highly shareable)
- Infrastructure Service layer (technical services)
- Solution layer (consumes services, does not offer any services)

The SOA – EA Relationship
How/Where SOA fits into EA?



The SOA – EA Relationship

SOA is a way of defining and implementing all or part of an operational entity’s Enterprise Architecture

The SOA – EA Relationship

However, in today's enterprise, it is impractical and unwise to fully implement the Enterprise Architecture using a SOA.

The SOA – EA Relationship

Why?

The SOA – EA Relationship

Because most enterprises today do not have the business/operational configuration or the resources (people, time, money, skills, etc.) required for an enterprise wide SOA implementation.

What are some of the SOA Myths

- SOA-in-a-Box – Vendors offer SOA out of the box solutions and have fully developed Service Architectures and solutions that embody all the elements necessary for a successful SOA implementation
- ESB spells Service Oriented Architecture
- Web Services/Web based solutions and technologies and SOA are synonymous

Inherent SOA Challenges and Risks for the AS-Is EA

What are some of the inherent challenges and risks and critical considerations when developing SOAs and integrating them into the As-Is EA?

•Significant performance degradation – Legacy environments are not designed to accept loose-coupling that SOA brings

•Inefficient and/or underdeveloped data models. Poorly defined or nonexistent metadata and metadata models

•Negative ROI due to resource requirements and limitations – Implementing any aspect of SOA in a legacy environment requires significant resources (i.e. modeling and simulation, data conversion/migration, testing, technical skill/training, etc.)

•Validation of usability of SOA, particularly in an As-Is environment - There are no sophisticated tools that provide testability of services (including message and database services along with web services) in a typical architecture

•Most legacy environments employ a horizontal security model. Introducing SOA into a legacy environment adds a level of complexity that requires both horizontally an vertically integrated security models

Inherent Challenges and Risks SOA Brings to The To-Be EA

What are some of the inherent challenges and risks and critical considerations when developing a To-Be Architecture that supports and incorporates SOA

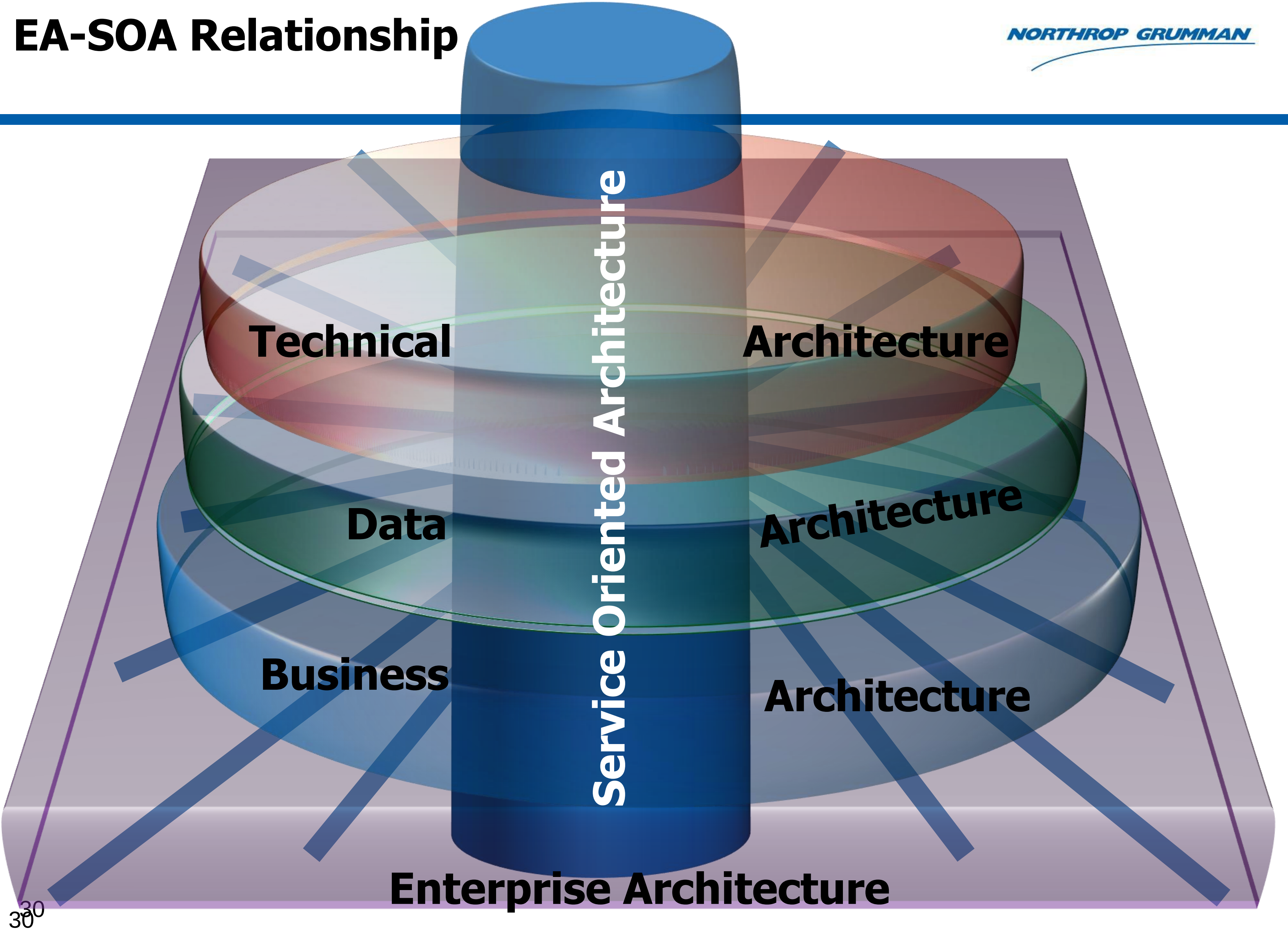
- Understanding Loose-coupling
  - Performance impacts of loose-coupling
  - Loose-coupling will 'fix' the IT and make it easier to maintain in the future once the SOA implementation is mature in our organization.
  - Web Services Management will ensure the success of loose-coupling.
- Security – Security solutions for the problems associated with SOA are still very immature. SOA requires horizontally and vertically integrated security models
- SOA is still quite immature and SOA standards are still developing

The EA – SOA Relationship

The practical and most successful use of SOA is to implement parts of the EA such that the services become the axis of the EA, and then services are slowly introduced throughout the enterprise.

This enables the enterprise to adjust, and validate the services approach as it morphs.

EA-SOA Relationship



The EA – SOA – SE Relationship

However, a successful SOA implementation, as with the successful implementation of EA or any part of it, is dependent on the effective use of Systems Engineering that includes:

- Identification and effective use of the most appropriate SE approach
- Identification and effective use of the most appropriate methodologies that support SE (i.e. Agile development/Feature Driven Development)

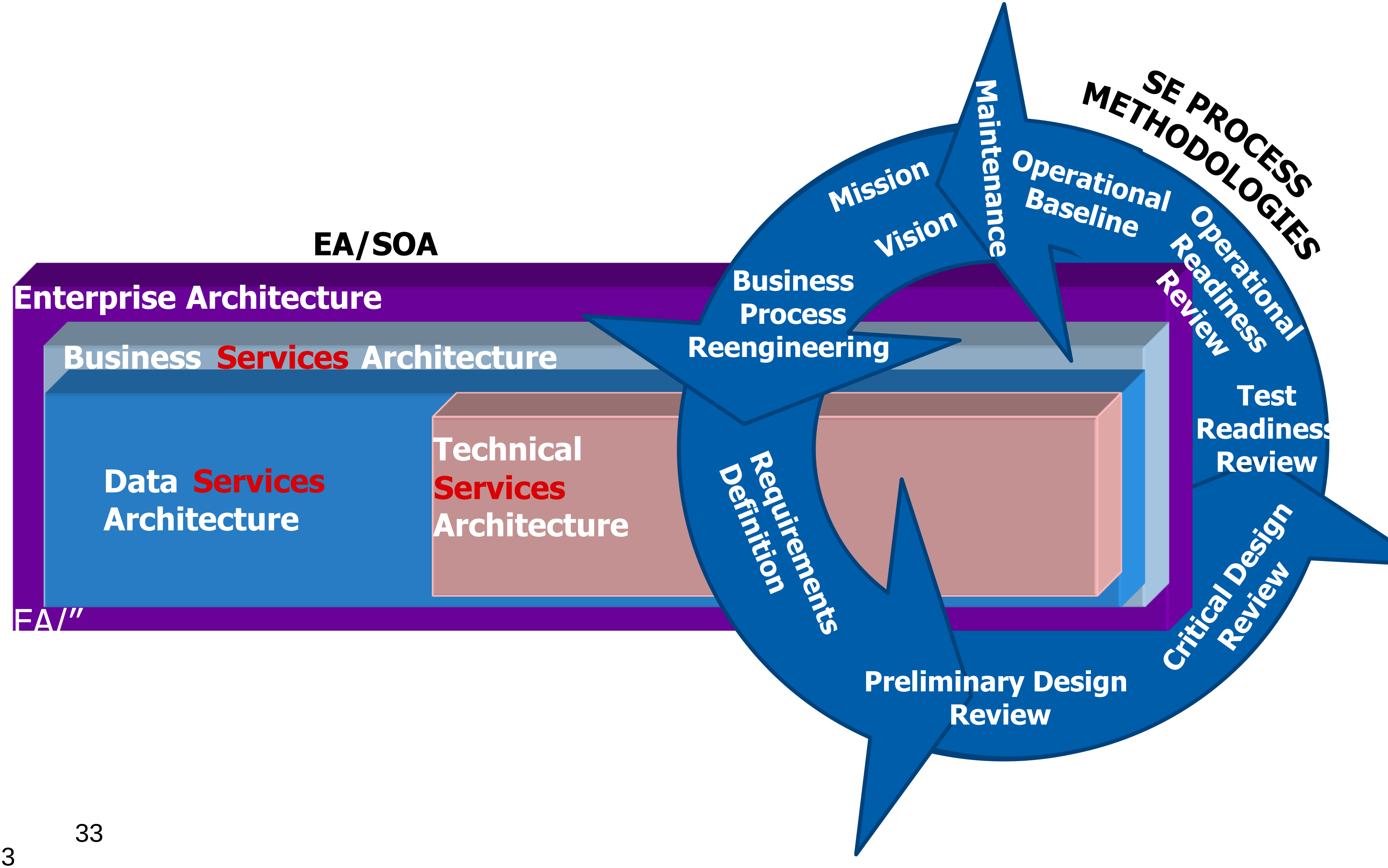
The EA – SOA – SE Relationship SOA/ Systems Engineering Tools

Like EA there are several SOA Engineering ‘tools’, which are actually consistent with standard Systems Engineering

- SOA Principles and Policies (i.e. WS Specifications)
- SOA Lexicon
- SOA Maturity Models
- SOA processes and methodologies
  - Some are standard processes that have been modified for SOA
  - Some are vendor developed (i.e. BEA Systems, AmberPoint, and IBM)
  - Governance

The EA – SOA – SE Relationship
How SE Fits into the SOA/EA Relationship

NORTHROP GRUMMAN


EA/SOA
Enterprise Architecture
Business Services Architecture
Data Services Architecture
Technical Services Architecture
EA/"
SE PROCESS METHODOLOGIES
Maintenance
Mission
Vision
Business Process Reengineering
Operational Baseline
Operational Readiness Review
Test Readiness Review
Critical Design Review
Preliminary Design Review
Requirements Definition

The EA – SOA – SE Relationship

Where is Systems Engineering in the EA-SOA Relationship?

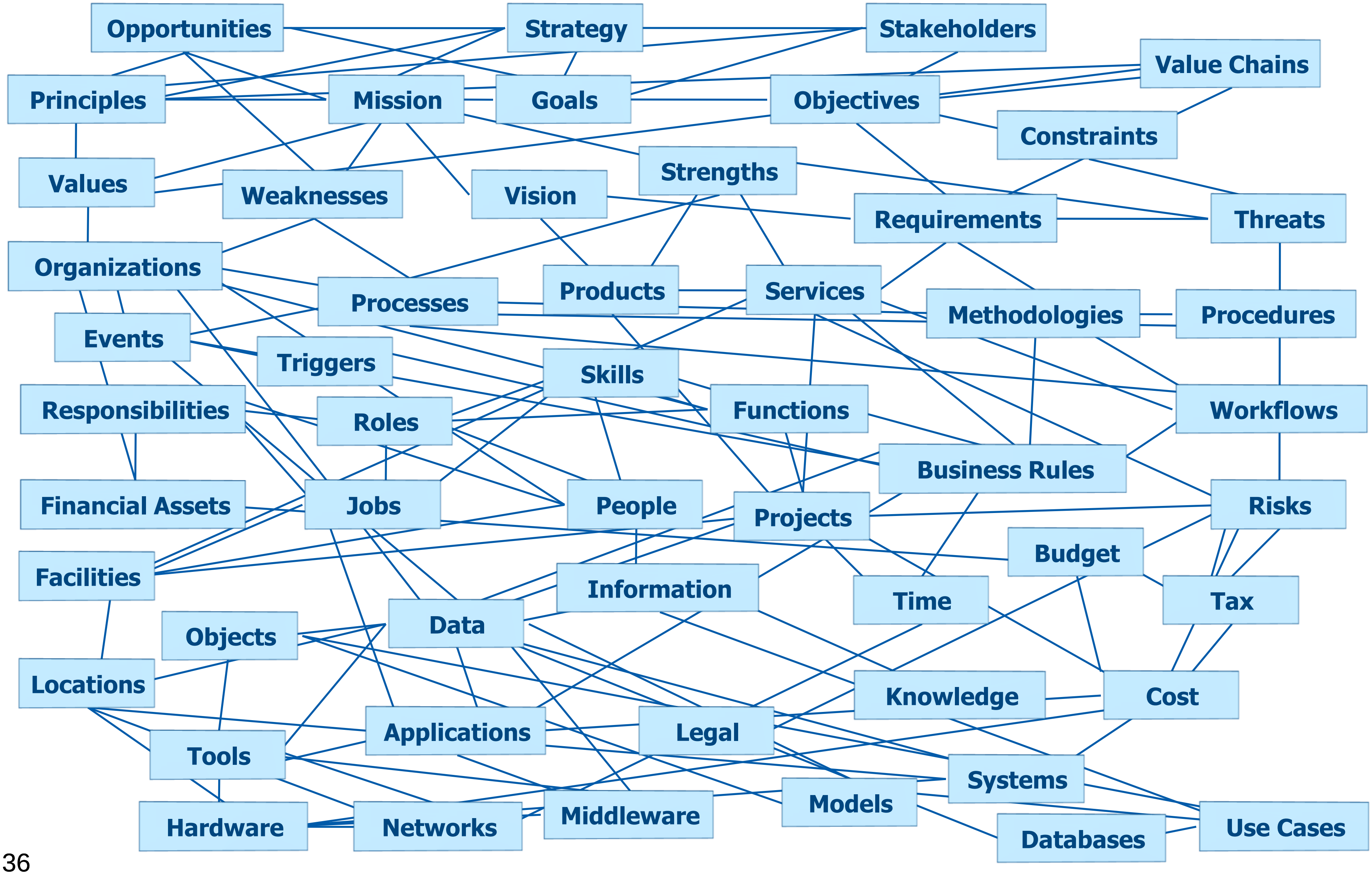
How does Systems Engineering fit into the SOA/EA Relationship?

- SOA is an approach for implementing Enterprise Architecture SOA models and incorporates processes that have been developed with the discipline of Services Engineering and Systems Engineering
- Systems Engineering is inherent in a mature EA and SOA approach
- Systems Engineering provides the processes and governance required for EA to be successful. Since SOA is an approach to implementing EA, SE and SOA have the same relationship

NORTHROP GRUMMAN

Organizational Complexity

NORTHROP GRUMMAN



‘ilities’ Definitions

1. *Flexibility* – ‘Plug-and-Play’ ‘technical service components’ to make it easier to interface and replace technical applications and components to respond to business needs.
2. *Reliability* – Standardized services that provide consistency to the business and IT
3. *Composability* – Collections of services can be coordinated and assembled to form composite services
4. *Reusability* – Common technical service components developed based on IT standards that supports multiple uses.
5. *Federation* – Associating business and technical services (service encapsulation) creating an aggregation of capabilities while maintaining the individual capabilities of the service components to be used in other service partnerships (an aspect of reusability).