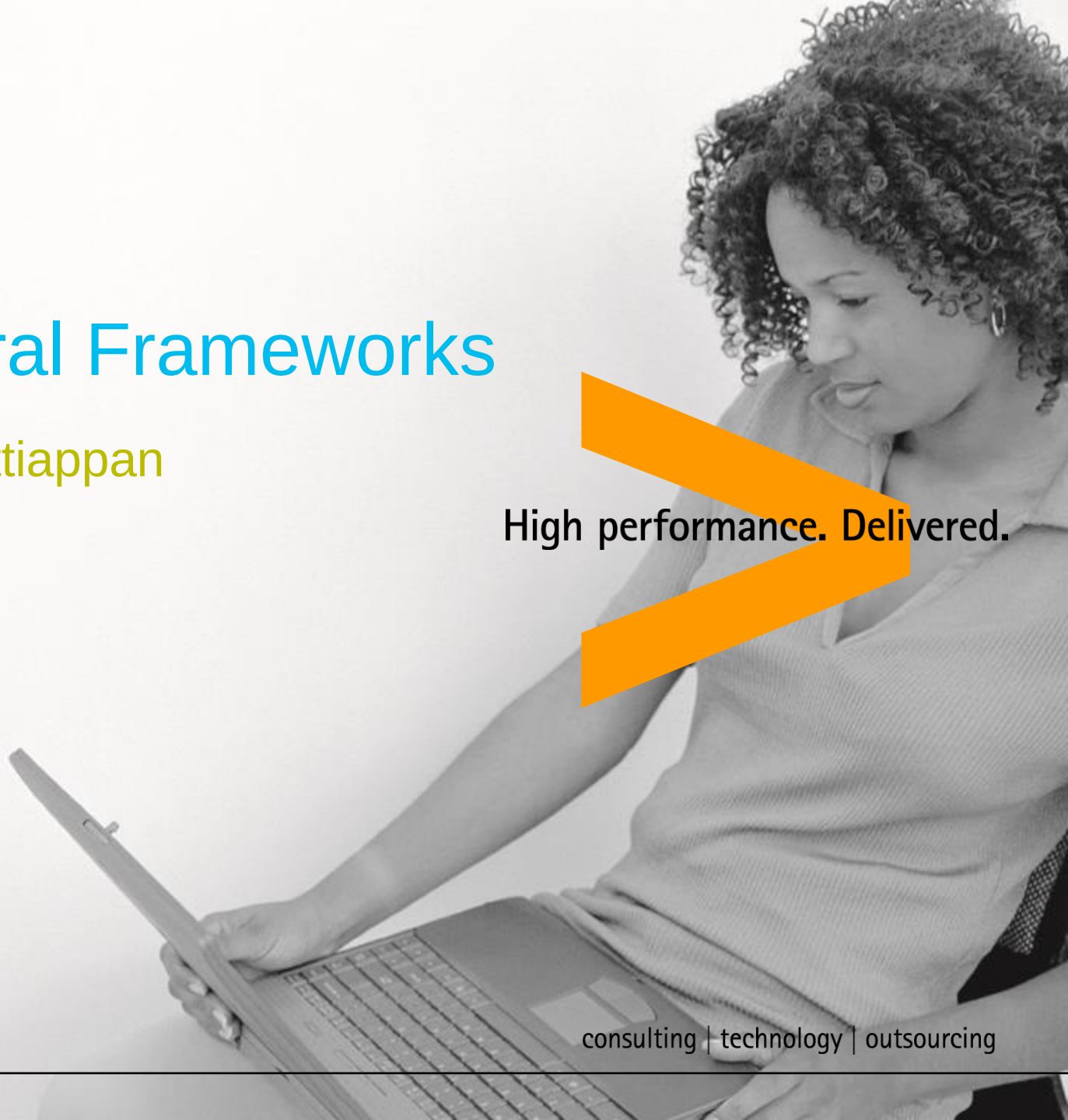
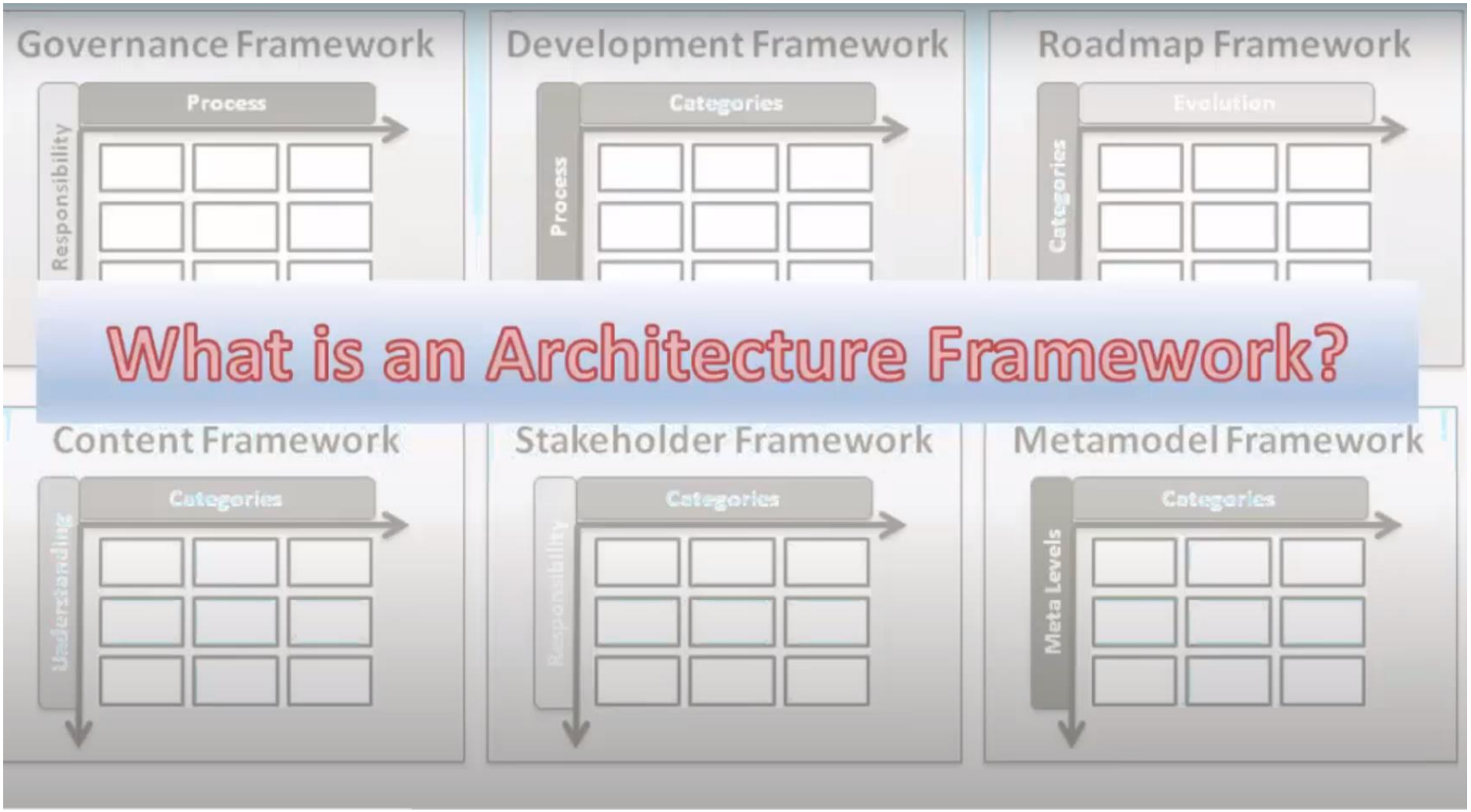
Architectural Frameworks

Parameswari Ettiappan




Governance Framework
Process
Responsibility
Development Framework
Categories
Process
Roadmap Framework
Evolution
Categories
What is an Architecture Framework?
Content Framework
Categories
Understanding
Stakeholder Framework
Categories
Responsibility
Metamodel Framework
Categories
Meta Levels

Governance Framework

- COBIT: the acronym refers Control Objectives for Information and Related Technology.
- ITIL: it is most widely adopted IT service management framework in the world.
- COSO: it is framework for corporate governance.
- CMMI: this acronym stands for Capability Maturity Model integration.
- FAIR: it refers Factor analysis of Information Risk. It is risk management frameworks help organizations analyze, understand, and measure information risk.

RoadMap Framework

- Agile
- Scrum
- Kanban
- Scaled Agile Framework® (SAFe) roadmap

Content Marketing Framework

- Simple warm acquisition(InVision)
- B2B sales and lead generation(Auto Pilot)
- Ecommerce (Simplified) (BeardBrand)
- Evergreen forest(WebMD)
- Competitive SEO (SlideBean)
- “Trifecta” strategy (Nerd Wallet)

Stakeholder Framework

- Praxis Framework
- Scaled Agile

Metamodel Framework

- UML,SysML
- Apache MetaModel

Enterprise Architecture – Definition & Drivers

- Enterprise architecture is defined as “the set of architectural concepts, principles, guidelines, blueprints, standards, and other enterprise-wide deliverables that guide an enterprise through acquiring, outsourcing,integrating, connecting, developing, modifying, operating, and retiring the elements (internal and external to the enterprise) of an IT portfolio.
- **- Cutter Consortium**

Why Enterprise Architecture?

- Need for an efficient, faster, consistent, flexible and predictable technology support to the businesses
- Need for faster time to market
- Complex integration issues due to inconsistent and unsound technology
- Chaotic technology landscape due to mergers and acquisitions
- Lack of processes (architecture definition, technology selection, EA team funding, IT procurement etc.)
- Lack of proper technology strategy planning in line with business planning

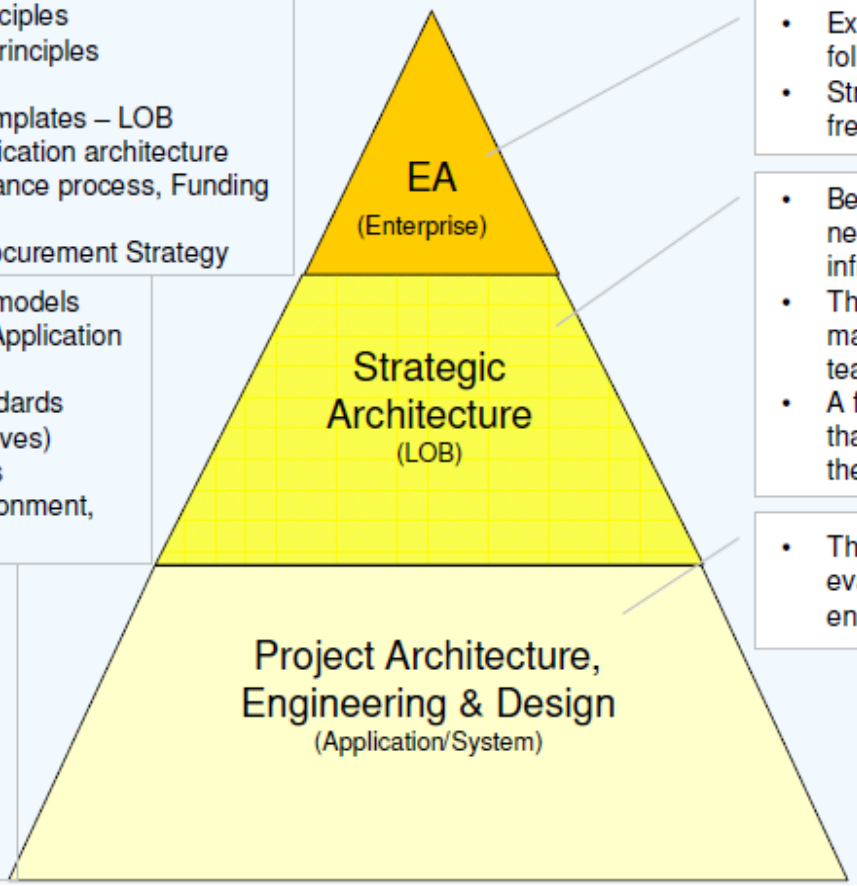
Architecture in an Enterprise can be viewed in 3 Layers

Components of Architectures

- Enterprise Business Vision and Principles
- Enterprise Technology Vision and Principles
- Technology / Product standards
- Process definition and document templates – LOB architecture definition process, Application architecture definition process, Strategic governance process, Funding process etc.
- Business and IT investment and Procurement Strategy

- As-Is and To-Be Business process models and Process-Application Mapping (Application Portfolio Rationalization)
- LOB level technology / product standards
- Technology Roadmap (List of Initiatives)
- Architecture Blueprint / Components (Software, Standard operating environment, Integration, Security, Global data)

- Application architecture based on Enterprise and LOB architecture blueprints
- Strong adherence to defined technology standards and principles
- Integration of architecture and software development (e.g. RUP) processes



Critical Success Factors

- Existence of a Governance model that is followed across the organization.
- Striking the balance between degrees of freedom and standardization

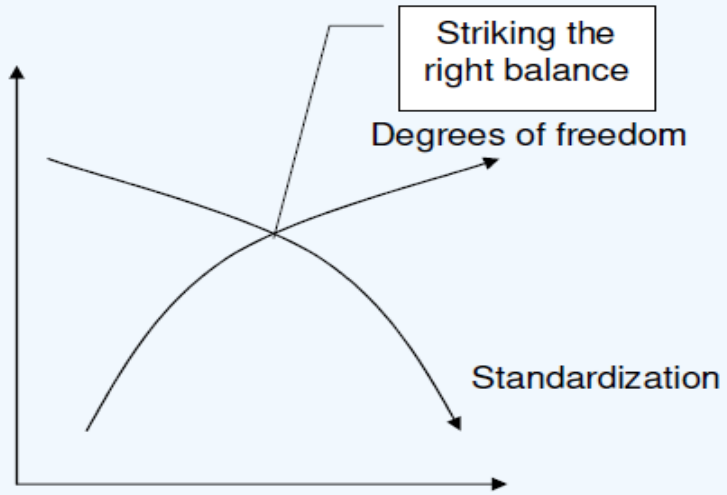
- Besides models and guidelines, there needs to be specific reusable and infrastructure components delivered
- The speed of the deliverables needs to match the expectations from the project teams that need them.
- A funding model needs to be in place that can charge-back the development of the reusable components to Projects.

- The Governance model needs to be evangelized and audited formally to ensure adherence

Acronyms used in this document

- **AD / M – Application Development / Maintenance**
- **BU – Business Units**
- **EA – Enterprise Architecture**
- **LOB – Line of Business**

Degrees of freedom Vs Standardization



- At EA level, there should be multiple but finite number of paths to be selected by the LOBs
- At LOB architecture level, the degrees of freedom should be minimal
- At project architecture level, the degree of freedom should be very much constrained

Example

- EA level
  - Different types of technology stacks with specific products in each – (ex) Microsoft stack, IBM J2EE stack, Open source stack
- LOB level
  - An LOB can adopt only one among the above stacks as its technology goal and strive to rationalize its existing systems towards the defined stack
- Project level
  - Could select one among multiple reference architectures defined at LOB level

Our Proposition on Enterprise Architecture

Key Focus

- Focus on the rationalization of the gamut of old technologies as well as the large application portfolio to a more manageable and efficient set of new technologies and applications
- Provide a future direction for the architecture and technologies to be adopted and a roadmap to achieve that from the current state.

Current State

- *A large application portfolio spread across number of business units*
- *Myriad of technologies including unsupported, stable as well as leading-edge*
- *Various business pain-points*

EA Solution

Future State

- *Continuous rationalization of approved technology / product standards*
- *Definition of business process models, IT roadmaps and architecture blueprints for lines of business*

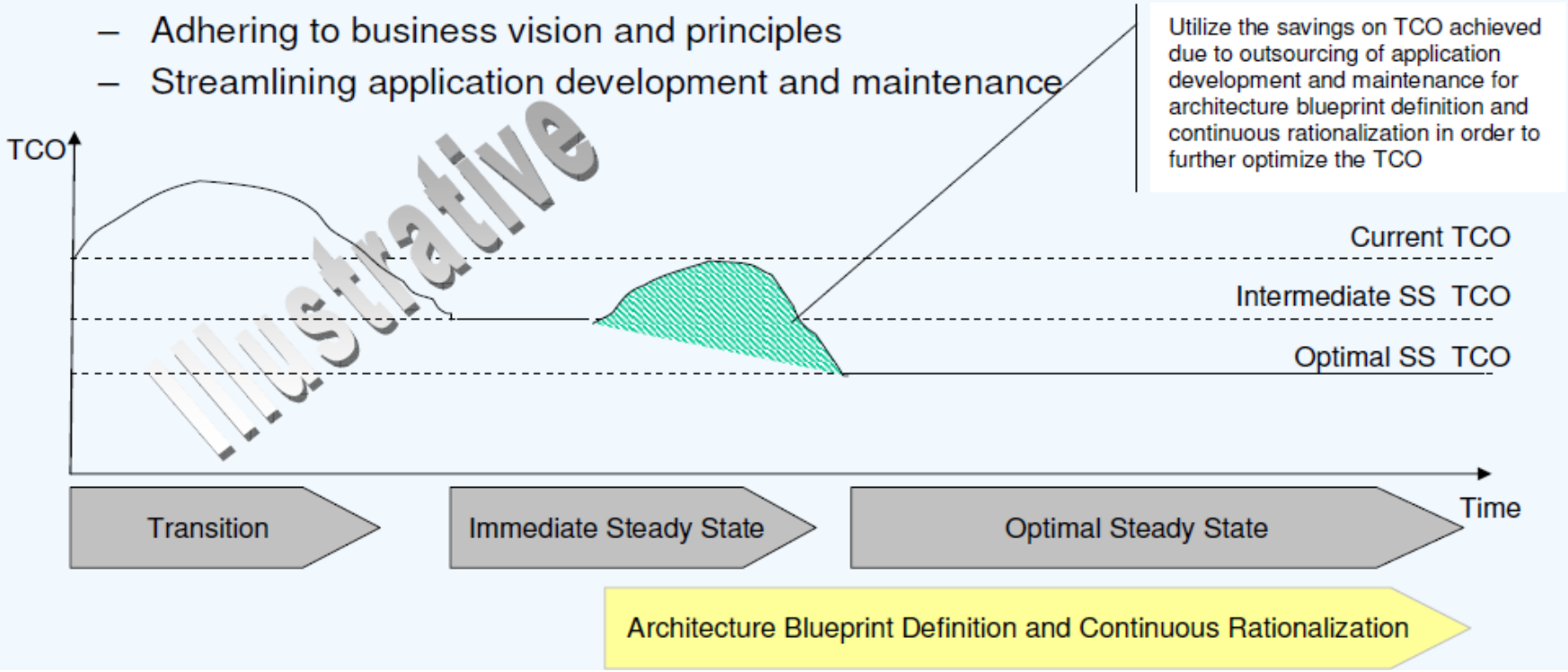
An end-to-end approach for architecture blueprint definition that is well integrated to ensure that the high-level enterprise and the Line of Business architecture blueprints get implemented in the right manner

Total Cost of Ownership (TCO)

- Gartner defines total cost of ownership (TCO) a comprehensive assessment of information technology (IT) or other costs across enterprise boundaries over time.
- For IT, TCO includes hardware and software acquisition, management and support, communications, end-user expenses and the opportunity cost of downtime, training and other productivity losses.
-

Benefits

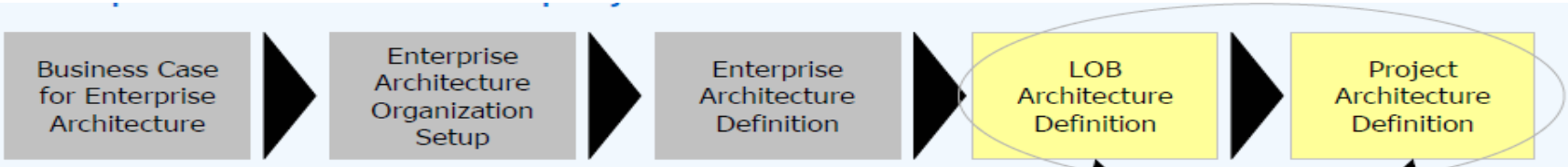
- Client would be able to optimize the TCO to a level lower than what is achieved through outsourcing of application development and maintenance
- Helps in
  - Standardizing technology / products
  - Rationalizing application portfolio
  - Adhering to business vision and principles
  - Streamlining application development and maintenance



Architecture Description Framework (ADF) for Enterprise Applications

ADF

- Among the various steps in EA definition, the focus is to prescribe the right description framework for project architecture and LOB architecture



- Business Case preparation involves carrying out a cost-benefit analysis of the architecture blueprint definition initiative and showing how it can help further optimize the TCO for the enterprise
- The organization set-up for Enterprise Architecture and LOB architecture needs to have the agreement from all stakeholders without which the success of initiative is under question
- This is followed by the Enterprise Architecture and LOB architecture blueprint definitions. The project architecture is defined in adherence with the defined enterprise / LOB architecture blueprints.

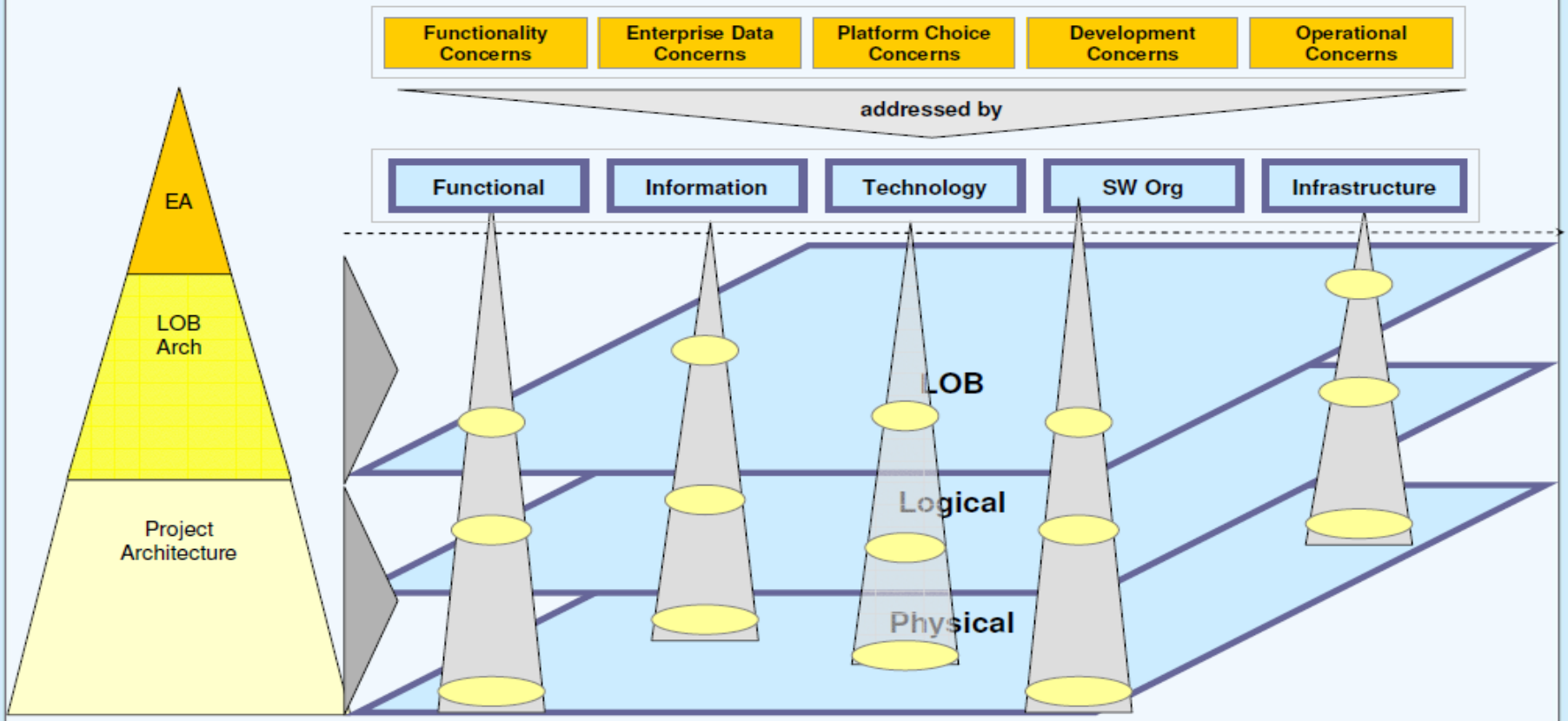
Focus of this presentation

- **Layered description for project architecture of enterprise applications**
- **Synergy between project architecture and enterprise /LOB level architectures**

Need for the “right” description of architecture of enterprise applications

- The architecture should describe the whole system at different levels of granularity depending on the hierarchy level of stakeholder.
- The architecture should describe different architectural aspects of the system focusing one at a time so as to help separate concerns of the stakeholders
- The software architecture of the system should take into account the enterprise and LOB level standards as well as architectural principles.
- To achieve the above, the Architecture Description Framework (ADF) should
  - **Provide guidelines and best practices for easy adoption**
  - **Provide tight coupling between the LOB target architecture and project architectures**

Architecture Description Framework



- The project architecture is
  - represented in 2 layers of abstraction (Logical and Physical)
  - traceable to different LOB architecture models

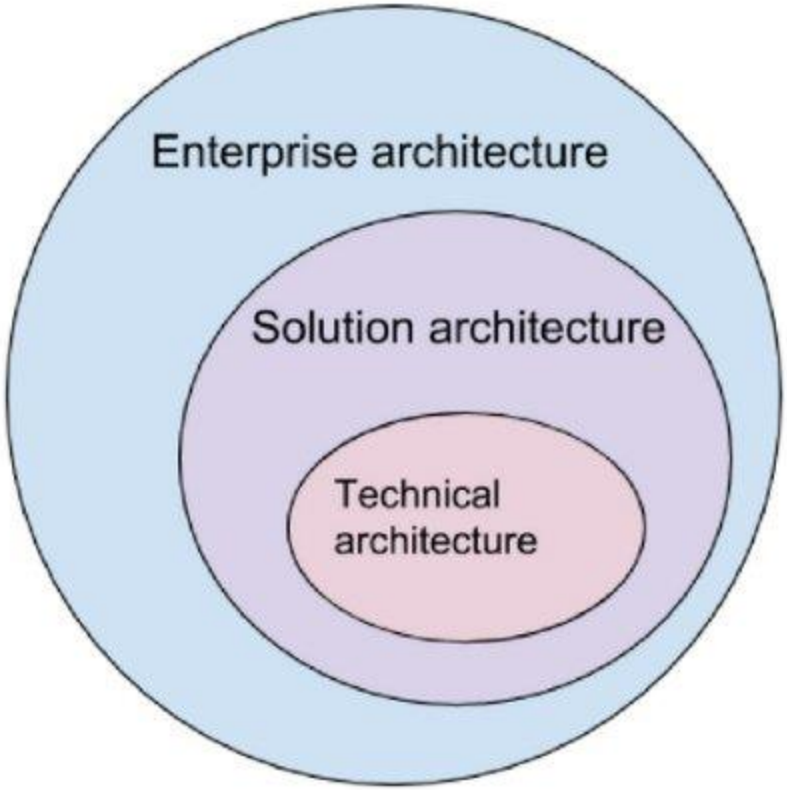
LOB Architecture Models

Viewpoint	Models
Functional	Process Application Map (PAM), High level business process models
Information	Standard entity definition, Entity to Application CRUD (Create, Read, Update, Delete) matrix
Technology	Technology / product blueprint
Software organization	N/A
Infrastructure	Standard operating environment for different types of applications (intranet, internet, extranet)

Definitions

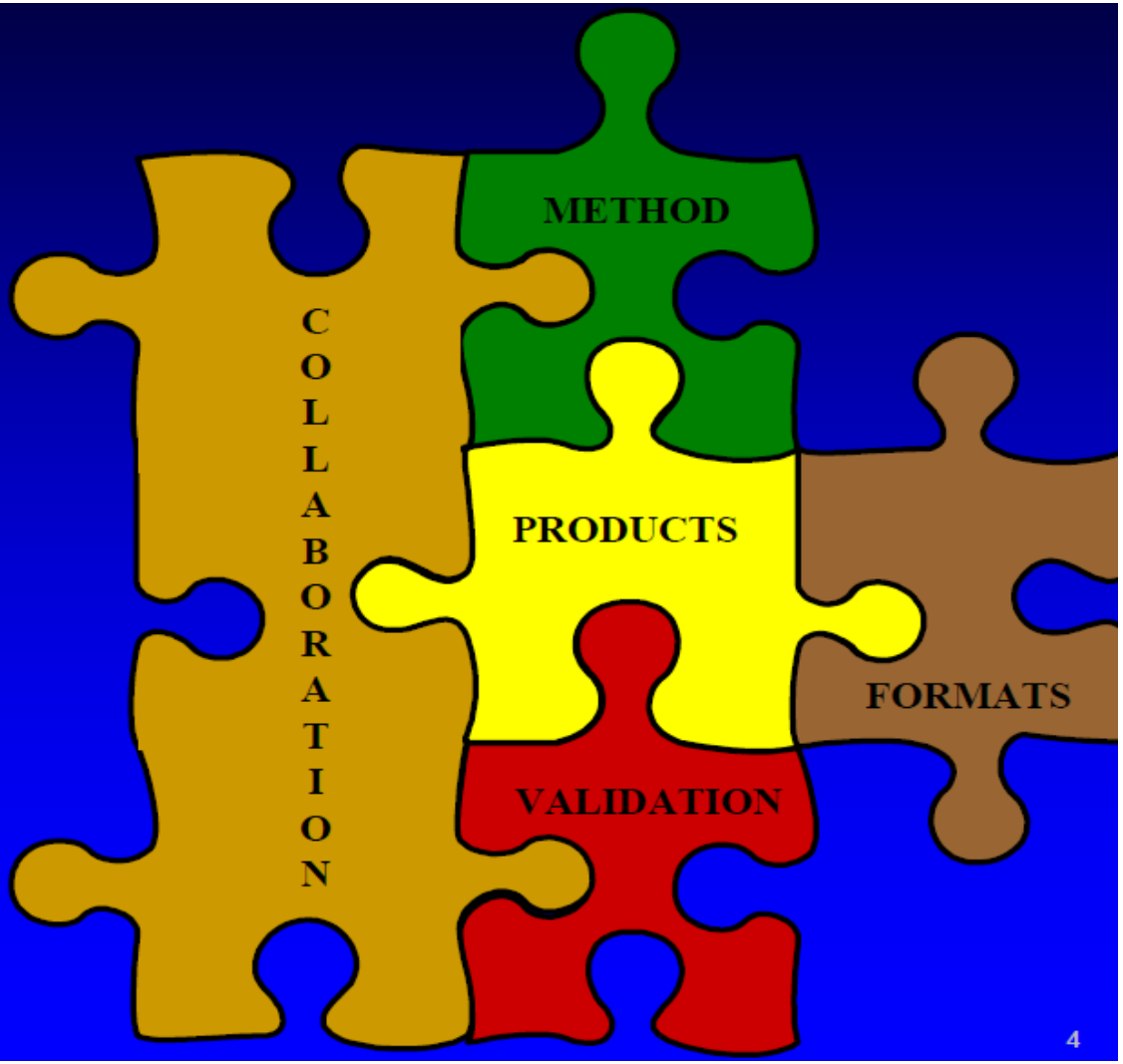
- **Architecture**
  - *"The fundamental organization of a system embodied in its components, their relationships to each other and to the environment, and the principles guiding its design and evolution." (IEEE 1471-2000)*
- **Architecture Framework**
  - *A resource that guides the development or description of an architecture*
- **Business Architecture**
  - *A perspective of the overall architecture reflecting enterprise mission, strategies, goals, business drivers, business processes, information flows, and the supporting organizational structure*
- **Technical Architecture**
  - *Perspectives of the overall architecture reflecting the enterprise's data, applications and technical components*
- **Enterprise Architecture**
  - *A blueprint (set of models) that depicts how various business and technical elements work together as a whole*
- **Enterprise**
  - *"e" : the highest level of a system or system of systems*
  - *"E" : a Department or Agency of the government*

Solution architecture in the context of the enterprise and technical architecture



Piecing The Puzzle Together: What's Needed In An "Architecting Process"?

- Architecting Method
- Architectural Products
- Product Formats
- Architecture Validation
- Collaboration

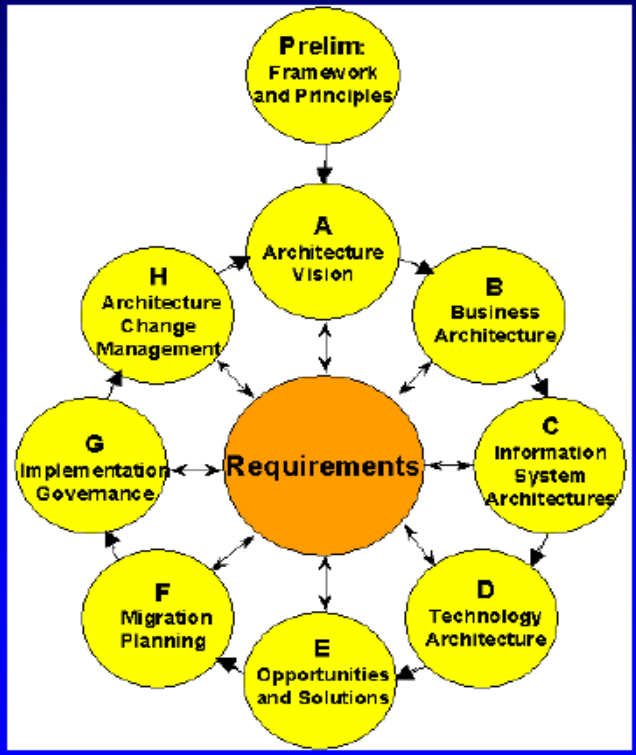


4

Building Blocks

- **Architecting Method**
- Architectural Products
- Product Formats
- Architecture Validation
- Collaboration

The Open Group Architecture Framework (TOGAF)
Version 8.0
Enterprise Edition
Architecture Development Method (ADM)

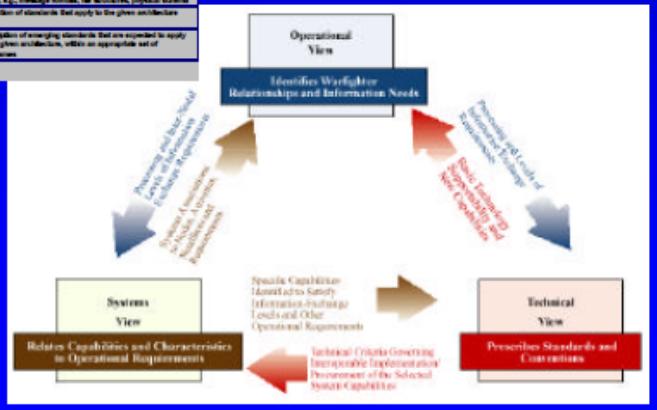


Building Blocks

- Architecting Method
- **Architectural Products**
- Product Formats
- Architecture Validation
- Collaboration

Applicable View	Framework Product	Framework Product Name	General Description
All Views	AV-1	Overview and Summary Information	Scope, purpose, intended users, environment depicted, analytical findings
All Views	AV-2	Integrated Dictionary	Data repository with definitions of all terms used in all products
Operational	OV-1	High-Level Operational Concept Graphic	High-level graphical/ textual description of operational concept
Operational	OV-2	Operational Node Connectivity Description	Operational nodes, operational activities performed at each node, connectivity and information exchange needlines between nodes
Operational	OV-3	Operational Information Exchange Matrix	Information exchanged between nodes and the relevant attributes of that exchange
Operational	OV-4	Organizational Relationships Chart	Organizational, role, or other relationships among organizations
Operational	OV-5	Operational Activity Model	Operational Activities, relationships among activities, inputs and outputs. Overlays can show cost, performing nodes, or other pertinent information
Operational	OV-6a	Operational Rules Model	One of the three products used to describe operational activity sequence and timing - identifies business rules that constrain operation
Operational	OV-6b	Operational State Transition Description	One of three products used to describe operational activity sequence and timing - identifies business process responses to events
Operational	OV-6c	Operational Event-Trace Description	One of three products used to describe operational activity sequence and timing - traces actions in a scenario or sequence of events and specifies timing of events
Operational	OV-7	Logical Data Model	Documentation of the data requirements and structural business process rules of the Operational View
Systems	SV-1	Systems Interface Description	Identification of systems and system components and their interconnections, within and between nodes
Systems	SV-2	Systems Communications Description	Systems nodes and their related communications laydowns
Systems	SV-3	Systems-Systems Matrix	Relationships among systems in a given architecture; can be designed to show relationships of interest, e.g., system-type interfaces, planned vs. existing interfaces, etc.
Systems	SV-4	Systems Functionality Description	Functions performed by systems and the information flow among system functions
Systems	SV-5	Operational Activity to Systems Function Traceability Matrix	Mapping of systems back to operational capabilities or of system functions back to operational activities
Systems	SV-6	Systems Data Exchange Matrix	Provides details of systems data being exchanged between systems
Systems	SV-7	Systems Performance Parameters Matrix	Performance characteristics of each system(s) hardware and software elements, for the appropriate timeframe(s)
Systems	SV-8	Systems Evolution Description	Planned incremental steps toward migrating a suite of systems to a more efficient suite, or toward evolving a current system to a future implementation
Systems	SV-9	Systems Technology Forecast	Emerging technologies and software/hardware products that are expected to be available in a given set of timeframes, and that will affect future development of the architecture
Systems	SV-10a	Systems Rules Model	One of three products used to describe systems activity sequence and timing - Constraints that are imposed on systems functionality due to some aspect of systems design or implementation
Systems	SV-10b	Systems State Transition Description	One of three products used to describe systems activity sequence and timing - Responses of a system to events
Systems	SV-10c	Systems Event-Trace Description	One of three products used to describe systems activity sequence and timing - System-specific refinements of critical sequences of events and the timing of these events
Systems	SV-11	Physical Schema	Physical implementation of the information of the Logical Data Model, e.g., message formats, file structures, physical schema
Technical	TV-1	Technical Standards Profile	Extraction of standards that apply to the given architecture
Technical	TV-2	Technical Standards Forecast	Description of emerging standards that are expected to apply to the given architecture, within an appropriate set of timeframes

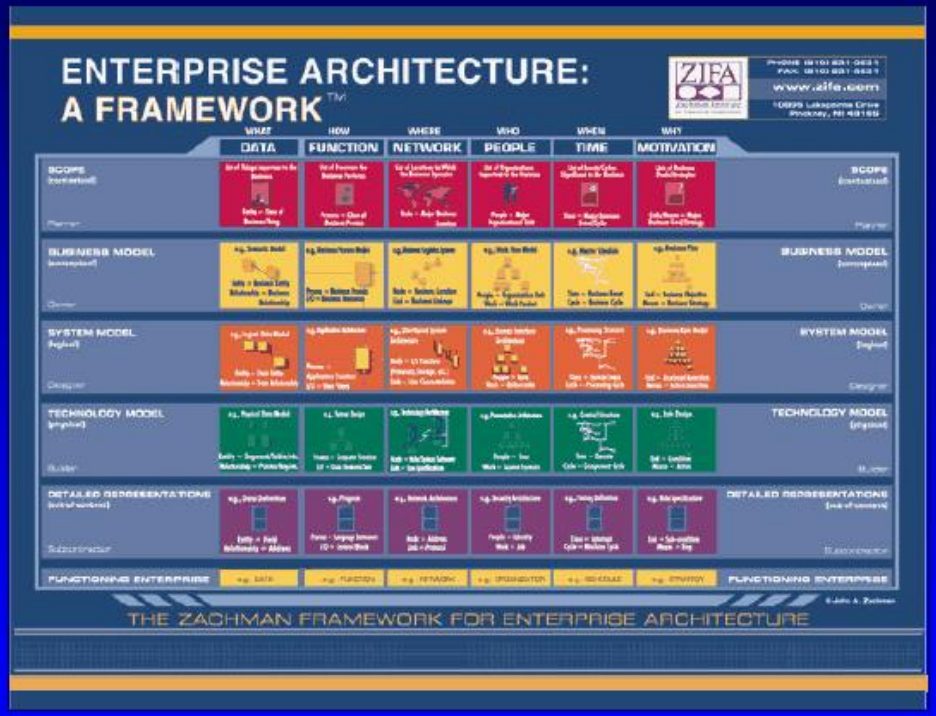
The Department of Defense Architecture Framework (DoDAF) Final Draft Version 1.0



Building Blocks

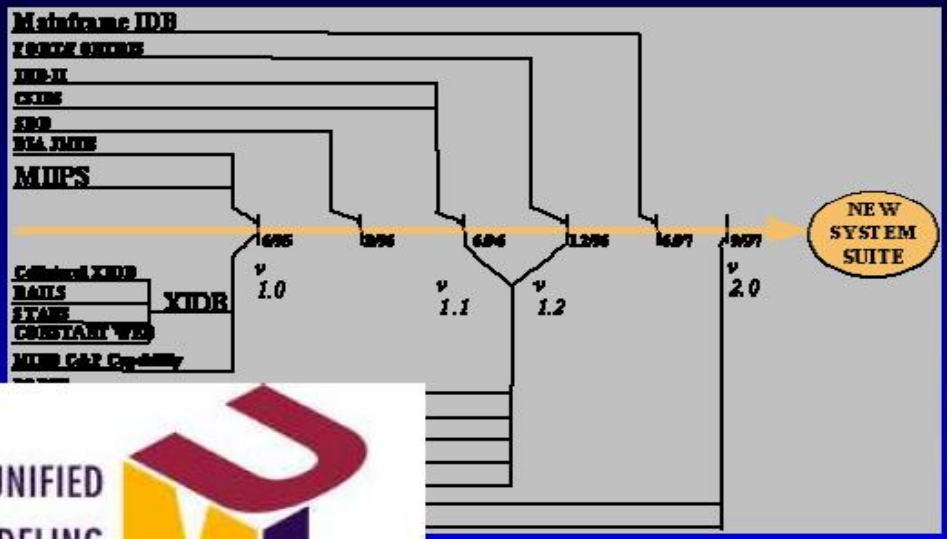
- Architecting Method
- **Architectural Products**
  - **Supplementing the DoDAF**
- Product Formats
- Architecture Validation
- Collaboration

The Zachman Framework For Enterprise Architecture

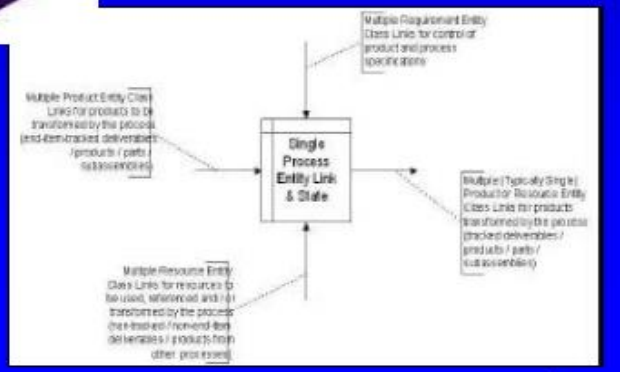


Building Blocks

- Architecting Method
- Architectural Products
- **Product Formats**
- Architecture Validation
- Collaboration

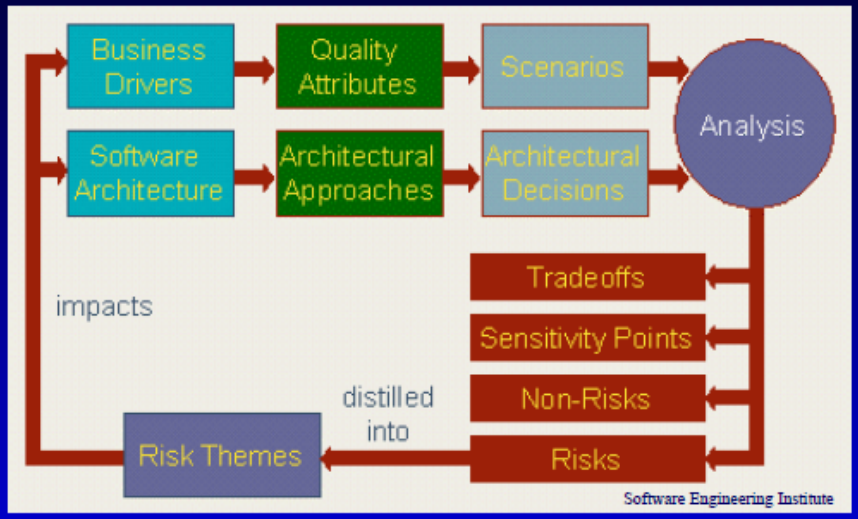


- **DoDAF Templates**
- **Unified Modeling Language (UML)**
- **Integrated Computer-Aided Manufacturing (ICAM) DEFinition (IDEF)**

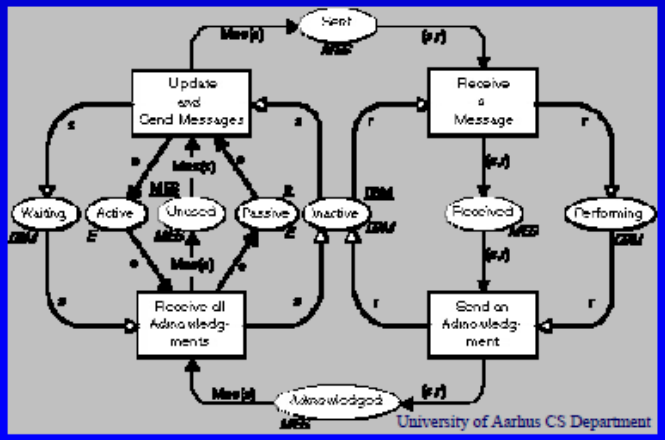


Building Blocks

- Architecting Method
- Architectural Products
- Product Formats
- **Architecture Validation**
- Collaboration



- **Software Engineering Institute's Architecture Tradeoff Analysis Method℠**
- **Quality Attribute Assessment Techniques (e.g., Colored Petri Nets)**

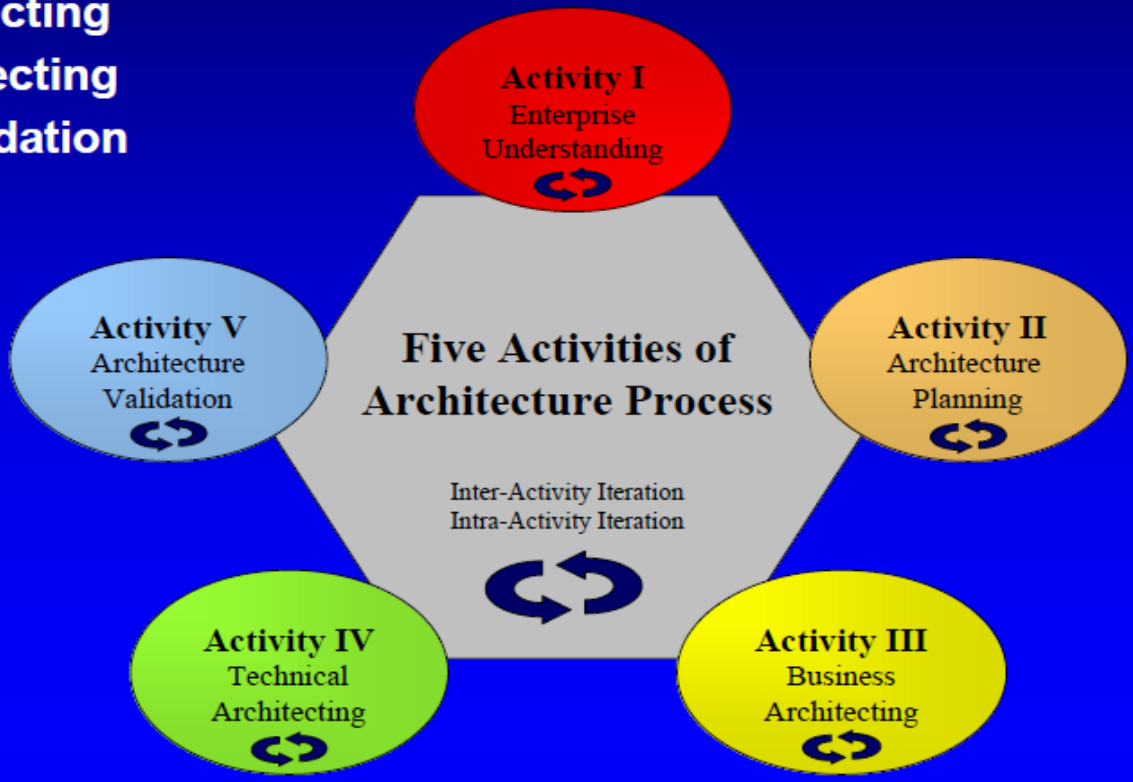


Building Blocks

- Architecting Method
- Architectural Products
- Product Formats
- Architecture Validation
- **Collaboration**

Building Blocks

- Raytheon Enterprise Architecture Process (REAP)
  - I Enterprise Understanding
  - II Architecture Planning
  - III Business Architecting
  - IV Technical Architecting
  - V Architecture Validation



Enterprise Understanding

- **Goals**
  - Set context for architecture and architecting activities
  - Common understanding with customer on the [E/e]nterprise, the architecting initiative, and the problem space
- **TOGAF Relationship**
  - ADM: Phase A

- **Subprocesses**
  - Customer-focused architecting
  - Requirements analysis
  - Operational/Business analysis
  - Quality attribute analysis

Activity I
Enterprise Understanding

- **Inputs**
  - Customer vision, needs, & requirements documents
  - Domain expertise
  - Industry & government standards

- **Outputs**
  - DoDAF AV-1, *Overview & Summary Information*
  - DoDAF AV-2, *Integrated Data Dictionary*
  - DoDAF OV-1, *High Level Operational Concept Graphic*
  - DoDAF TV-1, *Technical Standards Profile*

Architecture Planning

- **Goal**
  - Establish a plan for the upcoming architecting activities, the goals of the architecture and the architectural outputs
- **TOGAF Relationship**
  - ADM: Preliminary Phase

- **Subprocesses**
  - Identify stakeholders
  - Define architecture principles
  - Identify architectural products, formats and the supporting Zachman cells
  - Define product relationships / dependencies
  - Define schedule
  - Select tool(s)
  - Plan concordance, configuration & consolidation of architectural products
  - Form/train Architecture Team

Activity II
Architecture Planning

- **Inputs**
  - Customer vision, needs, & requirements documents
  - DoDAF AV-1, AV-2, OV-1, TV-1
  - Quality attribute-based requirements

- **Outputs**
  - Architecture principles
  - Architecture schedule
  - Enhanced DoDAF AV-1, *Overview & Summary Information*
  - Architecture engineering environment

Business Architecting

- **Goal**
  - Model the customer's view
- **TOGAF Relationship**
  - ADM: Phase B

- **Subprocesses**
  - Collect Zachman Framework "primitives" for Row 2
    - Produce mapping matrices as needed
  - Model Business/Mission Scenarios

Activity III
Business
Architecting

- **Inputs**
  - Customer vision, needs, & requirements documents
  - Domain expertise
  - Architecture principles
  - DoDAF AV-1, AV-2, OV-1
  - Architecture engineering environment

- **Outputs**
  - Business/Mission Scenarios within DoDAF OV-5, *Operational Activity Model*
  - Catalogued information from Zachman Framework Row 2 Cells

Technical Architecting

- **Goal**
  - Produce the remaining architectural descriptions of the enterprise from a variety of views
- **TOGAF Relationship**
  - ADM: Phases C, D

- **Subprocesses**
  - Develop/mature the defined DoDAF view products
  - Develop the defined additional architectural products
  - Ensure concordance between architectural products
  - Iteratively evolve an executable model

Activity IV
Technical
Architecting

- **Inputs**
  - Business Architecture
  - Customer vision, needs, & requirements documents
  - Domain expertise
  - Architecture principles
  - DoDAF AV-1, AV-2, OV-1, OV-5, TV-1 (and its referenced standards)

- **Outputs**
  - Architecture Baseline Package
  - Executable model

Architecture Validation

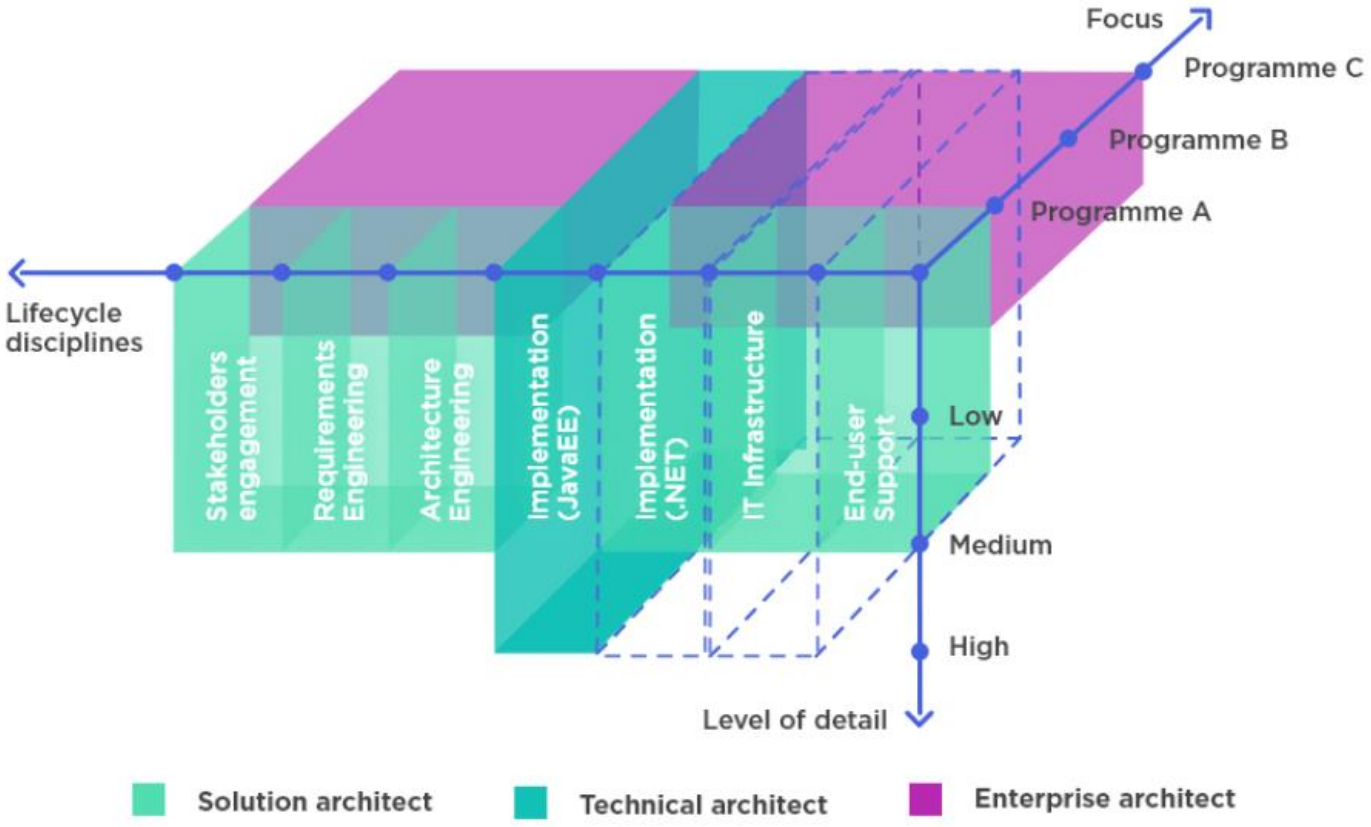
- **Goal**
  - Ensure the architecture is ready to be implemented
- **Subprocesses**
  - Architecture checklist
  - ATAM[SM]
  - Quality attribute assessments

Activity V
Architecture
Validation

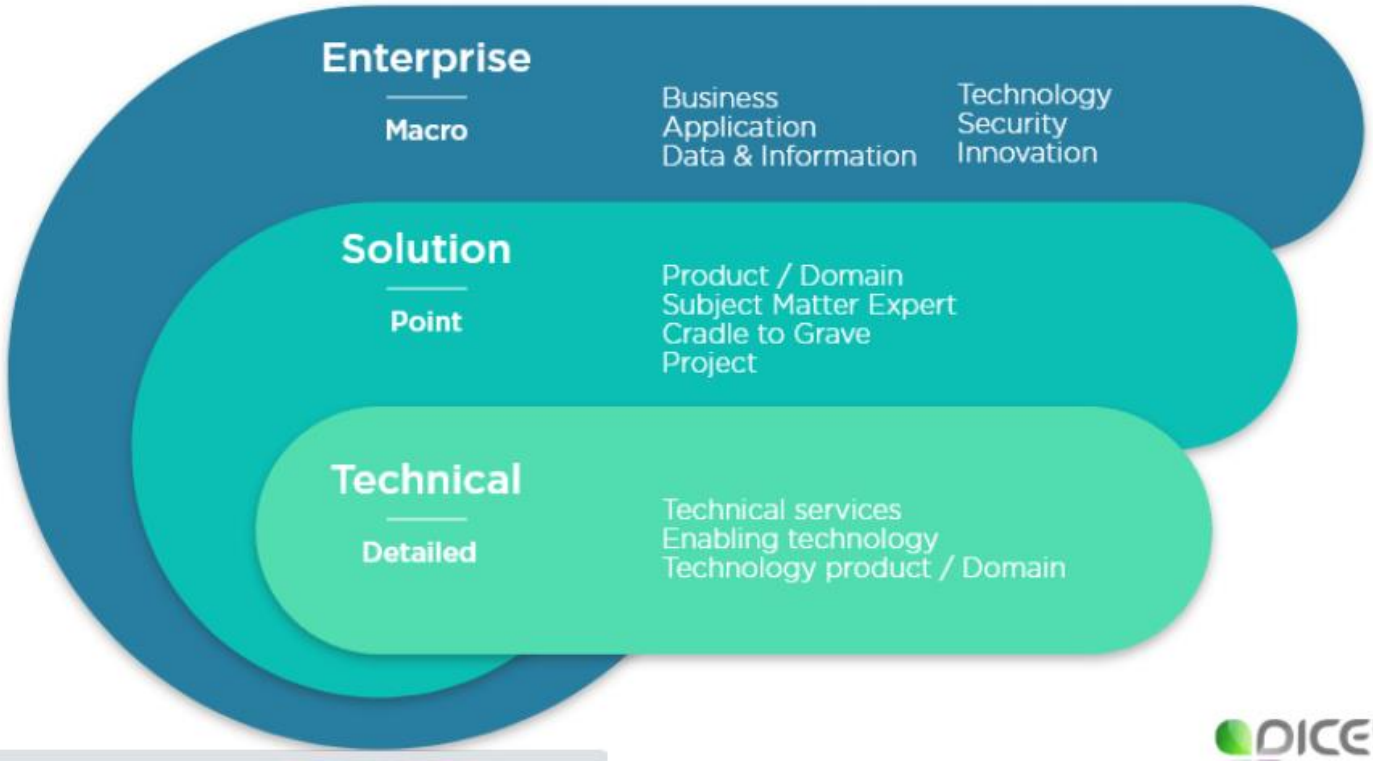
- **Inputs**
  - Architecture Baseline Package
  - Executable model
- **Outputs**
  - Completed architecture checklist
  - Simulation results
  - SEI's Architecture Tradeoff Analysis Method[SM] results
  - Validated architecture

Solution architecture in the context of the enterprise and technical architecture

Technical architect vs solution architect vs enterprise architect



Solution architecture in the context of the enterprise and technical architecture



The software architecture role

Architectural Drivers
Understanding requirements and constraints

Technology Selection
Choosing and evaluating technology

Architecting
Designing software

Architecture Evaluation
Understanding that the architecture works

Coding
Involvement in the hands-on elements of software delivery

Architecture Evolution
Ownership of the architecture throughout the delivery

Quality Assurance
Introduction and adherence to standards and principles

Coaching and Mentoring
Guidance and assistance

Types of software architects

Architect type	Strategic thinking	System interactions	Communication	Design
enterprise architect	across projects	highly abstracted	across organization	minimal, high level
solutions architect	focused on solution	very detailed	multiple teams	detailed
application architect	component re-use, maintainability	centered on single application	single project	very detail

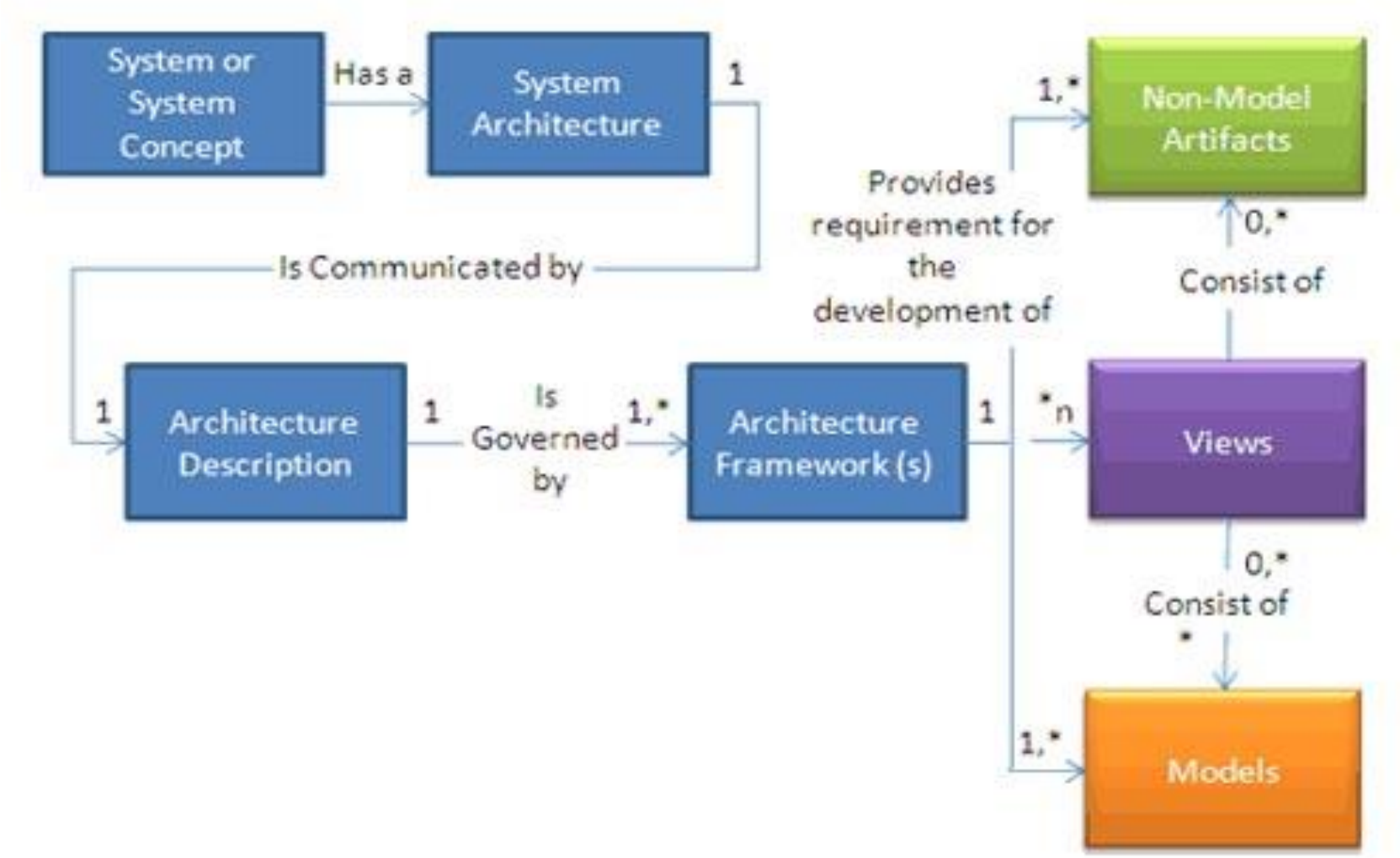
Architecture Framework

- An architecture framework is an encapsulation of a minimum set of practices and requirements for artifacts that describe a system's architecture.

Architecture Framework

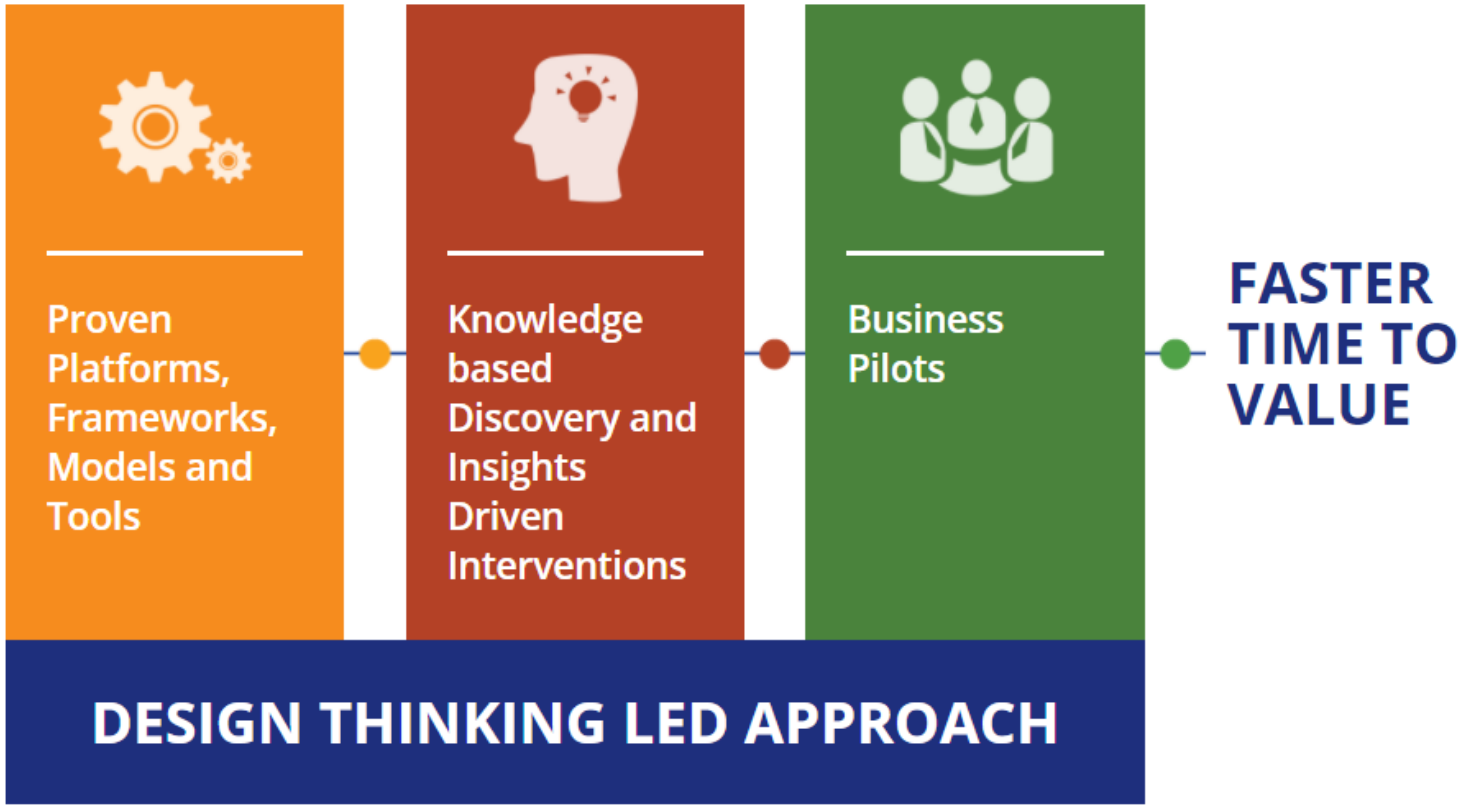
- An architecture Framework contains both the tools and the methods for creating and managing enterprise architectures.
- This includes
  - the processes and tools that are used to create the architectures.
  - the repository that stores the architecture content.
  - the organization of the team, including guidance on how to create the architectures and the governance of the teams that implement the architectures.

Architecture Framework



Architecture Framework

How we deliver value



Getting Started

- Systems are multidimensional in nature and have ‘n’ stakeholders with different concerns.
- Architecture frameworks enable the creation of system views that are directly relevant to stakeholders' concerns.
- Often, multiple models and non-model artifacts are generated to capture and track the concerns of all stakeholders.

Getting Started

- By interacting with intra- and extra-program stakeholders, including **users, experimenters, acquirers, developers, integrators, and testers**, key architectural aspects that need to be captured and communicated in a program are determined.
- These architecture needs then should be consolidated.
- It is then taken as recommendation to develop and use specific models and views that directly support the program's key decisions and activities.

Getting Started

- Concurrently, **an architecture content and development governance** structure should be developed to manage and satisfy the collective needs.
-

Architecture planning and implementation activities

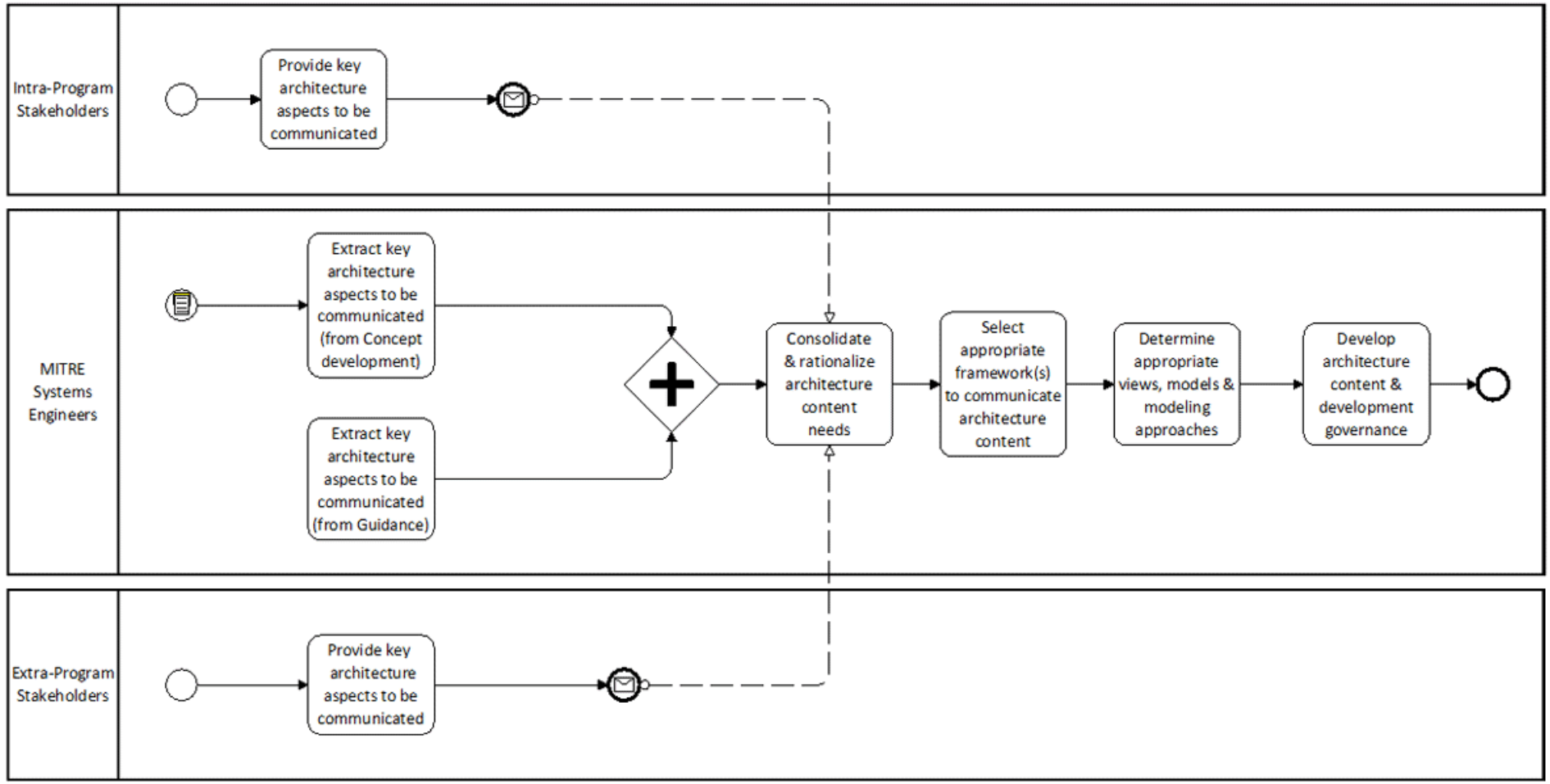


Figure 2. Architecture Planning and Implementation Activities

APPROACHES TO ARCHITECTURE DEVELOPMENT

- Architecture development can be thought of as both a **process and a discipline** that aids the development of mission-effective systems.
- Multiple complementary approaches and methodologies are used to develop enterprise and system architectures.

APPROACHES TO ARCHITECTURE DEVELOPMENT

- Some of the most popular approaches used in government departments and agencies are:
  - **U.S. Department of Defense Architecture Framework (DoDAF)**
  - **The Open Group Architecture Framework (TOGAF)**
  - **Object-oriented with Unified Modeling Language**
  - **Spewak architecture process and Zachman Framework**

Key Steps

- Define the architecture purpose, value, and decisions it will support.
- Get information needed to define the architecture from stakeholders as early as possible.
- Create, refine, and update the architecture in an iterative way throughout the acquisition life cycle.
- Validate the architecture will meet expectations when implemented.
- Define roles for team members to guide and coordinate their efforts.

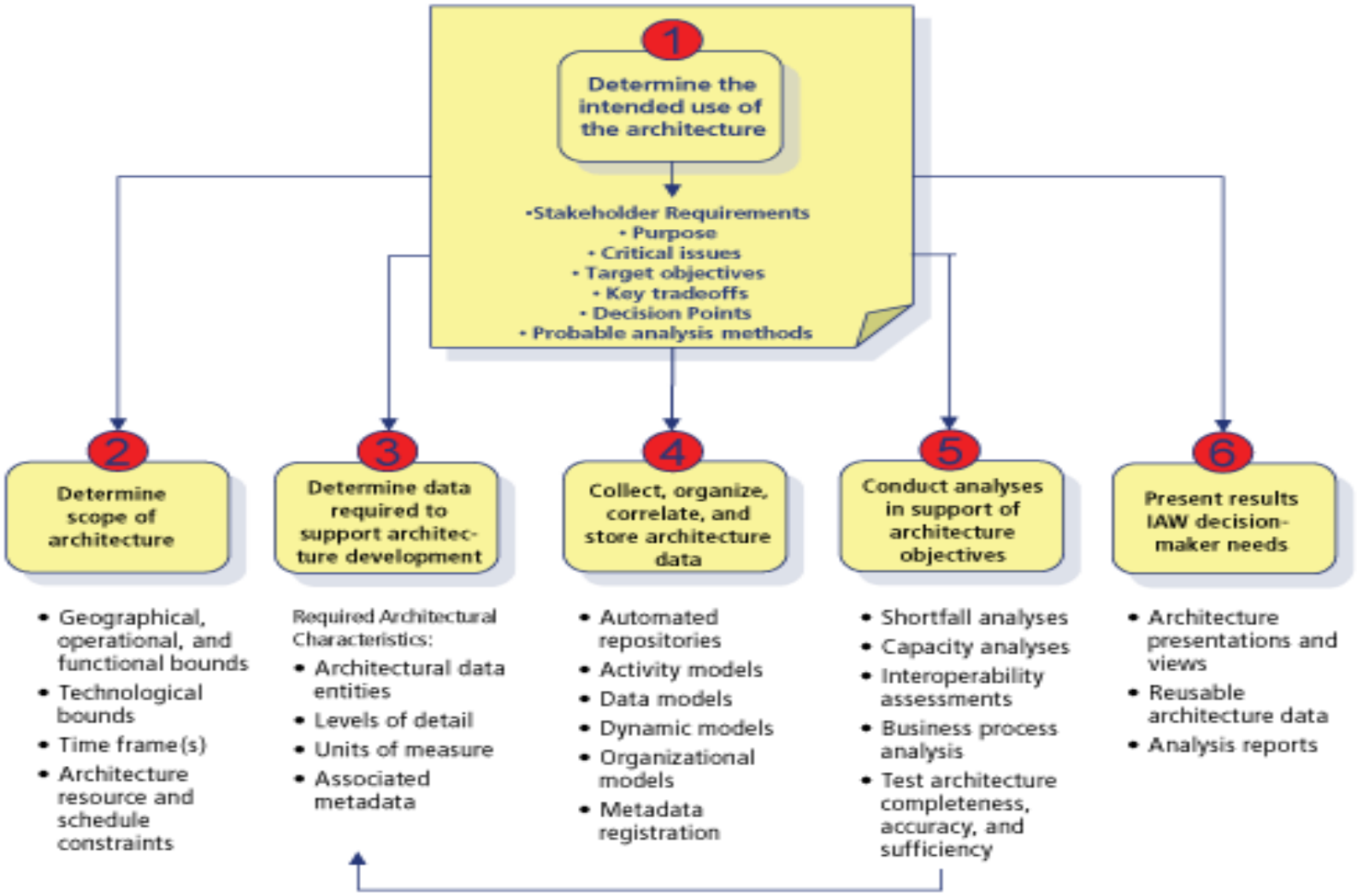
Key Steps

- Create estimates and schedules based on the architectural blueprint.
- Use the architecture to gain insight into project performance.
- Establish a lightweight, scalable, tailorable, repeatable process framework.

Determining the Right Process/Method

- Many systems engineers believe there is an "either-or" decision to be made regarding different architectural frameworks (e.g., DoDAF or TOGAF).
- But this is not necessarily the case.
- Some architectural standards address completely different elements of the architecting process.
- For example, TOGAF has a primary focus on architecture methodology—the "how to" aspect of architecting, without prescribing architecture description constructs.
- DoDAF has a primary focus on architecture description via a set of viewpoints, without a detailed specification of methodology [

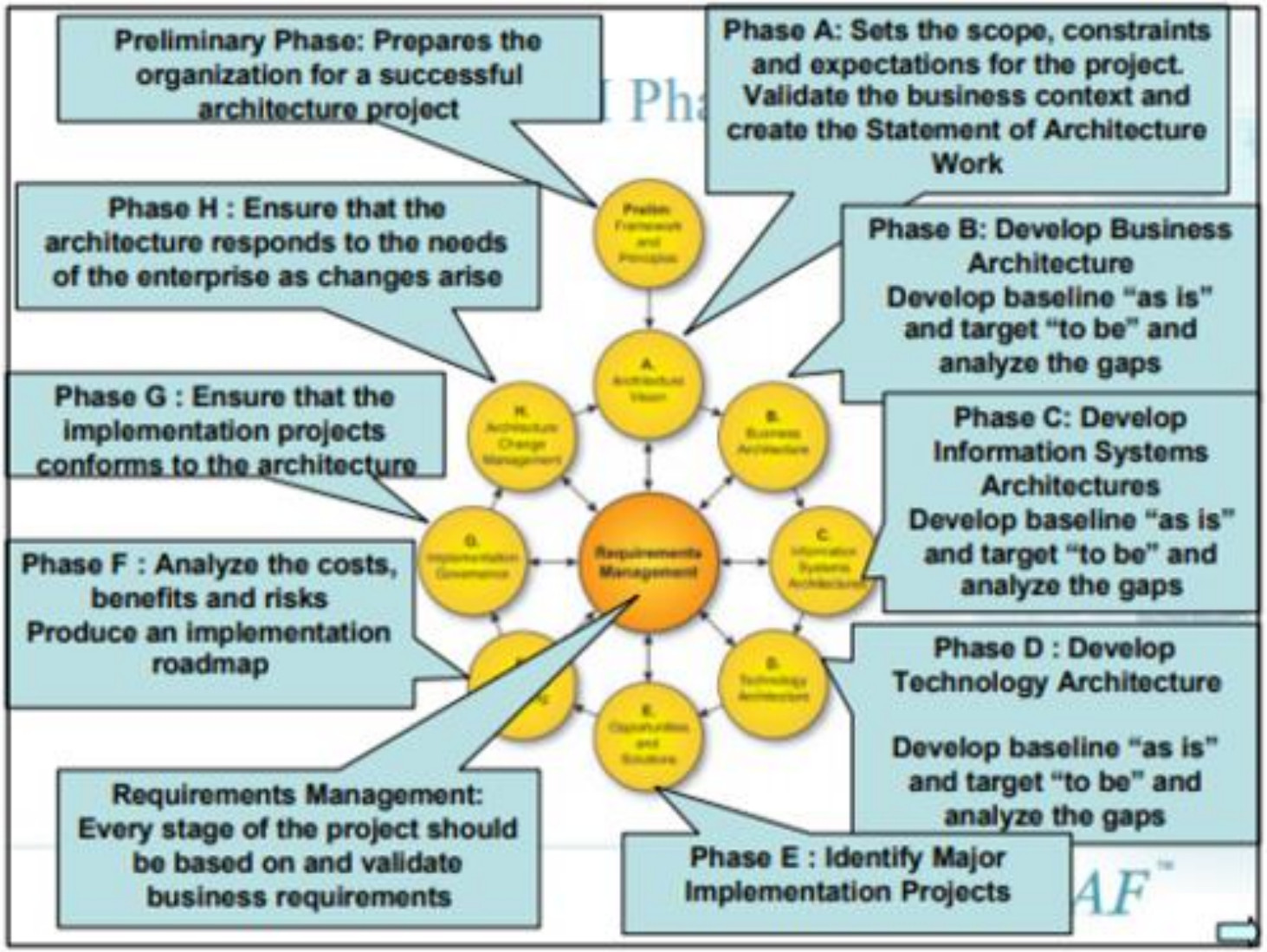
DoDAF 6-Step Architecture Process



TOGAF Architecture Development Method (ADM)

- The TOGAF Architecture Development Method (ADM) provides a tested and repeatable process for developing architectures.
- It is a generic method for architecture development that is designed to deal with most systems.
- However, it will often be necessary to modify or extend the ADM to suit specific needs.
- One of the tasks before applying the ADM is to review its components for applicability, and then tailor them as appropriate.

TOGAF Architecture Development Method (ADM)



TOGAF Architecture Development Method (ADM)

- **The Preliminary Phase** describes the preparation and initiation activities required to prepare to meet the operational directive for a new architecture, including the definition of an organization-specific architecture framework and the definition of principles.
- **Phase A:** Architecture Vision describes the initial phase of an architecture development cycle.
- It includes information about defining the scope, identifying the stakeholders, creating the architecture vision, and obtaining approvals.

TOGAF Architecture Development Method (ADM)

- **Phase B:** Business Architecture describes the development of a business architecture to support an agreed architecture vision.

- **Phase C**: Information Systems Architectures describes the development of information systems architectures for an architecture project, including the development of data and application architectures.

TOGAF Architecture Development Method (ADM)

- **Phase D:** Technology Architecture describes the development of the technology architecture for an architecture project.
- **Phase E:** Opportunities and Solutions conducts initial implementation planning and identifies delivery vehicles for the architecture defined in the previous phases.
- **Phase F:** Migration Planning addresses the formulation of a set of detailed sequences of transition architectures with a supporting implementation and migration plan.

TOGAF Architecture Development Method (ADM)

- **Phase G:** Implementation Governance provides an architectural oversight of the implementation**.**
- **Phase H:** Architecture Change Management establishes procedures for managing change to the new architecture**.**
- **Requirements Management examines** the process of managing architecture requirements throughout the ADM

Zachmen Framework

ZFI Zachman Framework

The Zachman Framework	DATA What	FUNCTION How	NETWORK Where	PEOPLE Who	TIME When	MOTIVATION Why
SCOPE (Contextual) Planner	Things Important to the Business	Processes the Business Performs	Locations in which the Business Operates	Organizations Important to the Business	Events/Cycles Significant to the Business	Business Goals/Strategies
BUSINESS MODEL (Conceptual) Owner	Conceptual Data Model	Business Process Model	Business Logistics	Work Flow Model	Master Schedule	Business Plan
SYSTEM MODEL (Logical) Designer	Logical Data Model	Application Architecture	Distributed System Architecture	Human Interface Architecture	Processing Structure	Business Rule Model
TECHNOLOGY MODEL (Physical) Builder	Physical Data Model	System Design	Technology Architecture	Presentation Architecture	Control Structure	Rule Design
DETAILED REPRESENTATIONS Sub-Contractor	Data Definition	Program	Network Architecture	Security Architecture	Timing Definition	Rule Specification
FUNCTIONING ENTERPRISE	Data	Function	Network	Organization Units	Schedule	Strategy

Best Practices and Lessons Learned

- Purpose is paramount
- A plan is a point of departure
- Know the relationships
- Be the early bird
- No one trusts a skinny cook
- Which way is right and how do I get there from here?
- Try before you buy.
- Taming the complexity beast

Best Practices and Lessons Learned

- Keep it simple
- Determining the right models and views
- But it looked so pretty in the window
- How do I create the right views?
- Bringing dolls to life
- How much architecture is enough?

Questions

Module Summary

- Determining the Right Framework
- Best Practices and Lessons Learned