Enterprise Architecture Best Practice Handbook

Building, Running and Managing Effective Enterprise Architecture Programs - Ready to use supporting documents bringing Enterprise Architecture Theory into Practice

Jeff Handley

Enterprise Architecture Best Practice Handbook:

Building, Running and Managing Effective Enterprise Architecture Programs - Ready to use supporting documents bringing Enterprise Architecture Theory into Practice

***Notice of Rights**: Copyright © The Art of Service. All rights reserved. No part of this book may be reproduced or transmitted in any form by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher.*

***Notice of Liability**: The information in this book is distributed on an "As Is" basis without warranty. While every precaution has been taken in the preparation of the book, neither the author nor the publisher shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the instructions contained in this book or by the products described in it.*

***Trademarks**: Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and the publisher was aware of a trademark claim, the designations appear as requested by the owner of the trademark. All other product names and services identified throughout this book are used in editorial fashion only and for the benefit of such companies with no intention of infringement of the trademark. No such use, or the use of any trade name, is intended to convey endorsement or other affiliation with this book.*

Write a Review and Receive a Bonus Emereo eBook of Your Choice

Up to $99 RRP – Absolutely Free

If you recently bought this book we would love to hear from you – submit a review of this title and you'll receive an additional free ebook of your choice from our catalog at http://www.emereo.org.

How Does it Work?

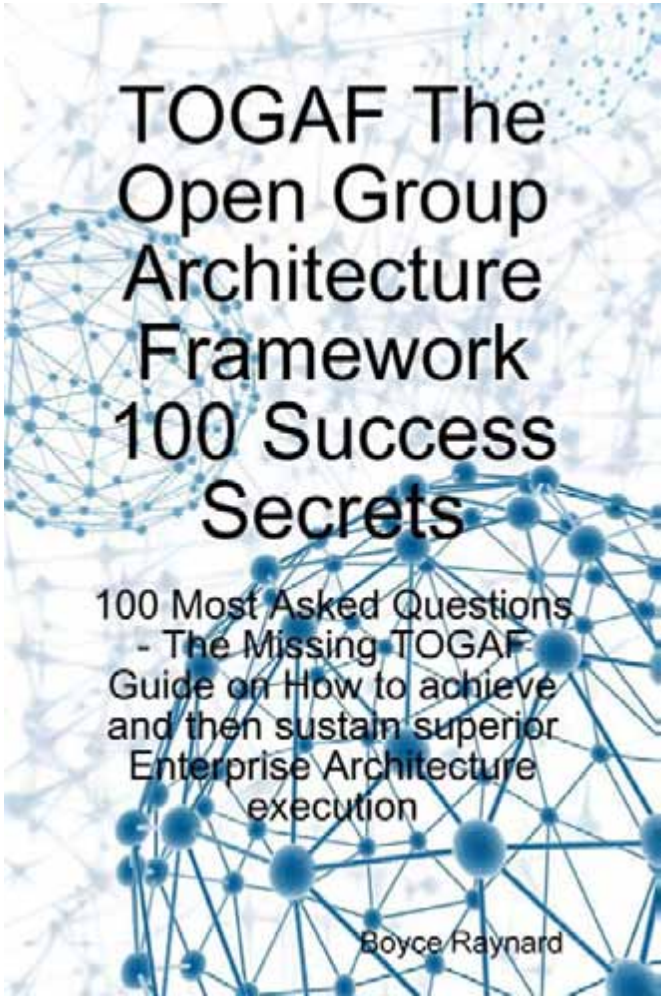
Submit your review of this title via the online store where you purchased it. For example, to post a review on Amazon, just log in to your account and click on the 'Create Your Own Review' button (under 'Customer Reviews') on the relevant product page (you'll find plenty of example product reviews on Amazon). If you purchased from a different online store, simply follow their procedures.

What Happens When I Submit my Review?

Once you have submitted your review, send us an email via review@emereo.org, and include a link to your review and a link to the free eBook you'd like as our thank-you (from http://www.emereo.org – choose any book you like from the catalog, up to $99 RRP). You will then receive a reply email back from us, complete with your bonus ebook download link. It's that simple!

TABLE OF CONTENTS

Also from Emereo Publishing:

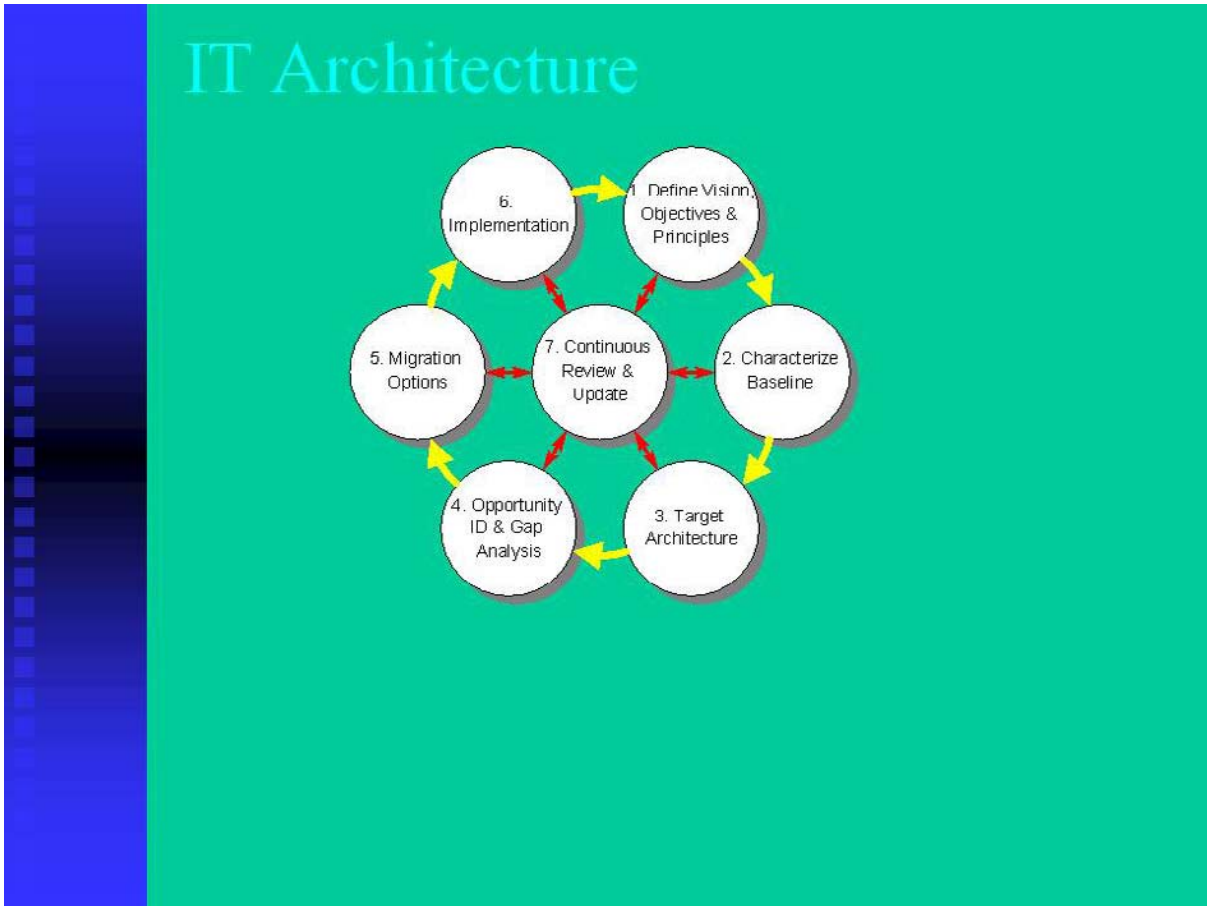


TOGAF The Open Group Architecture Framework 100 Success Secrets:

100 Most Asked Questions - The Missing TOGAF Guide on How to Achieve and Sustain Superior Enterprise Architecture Execution

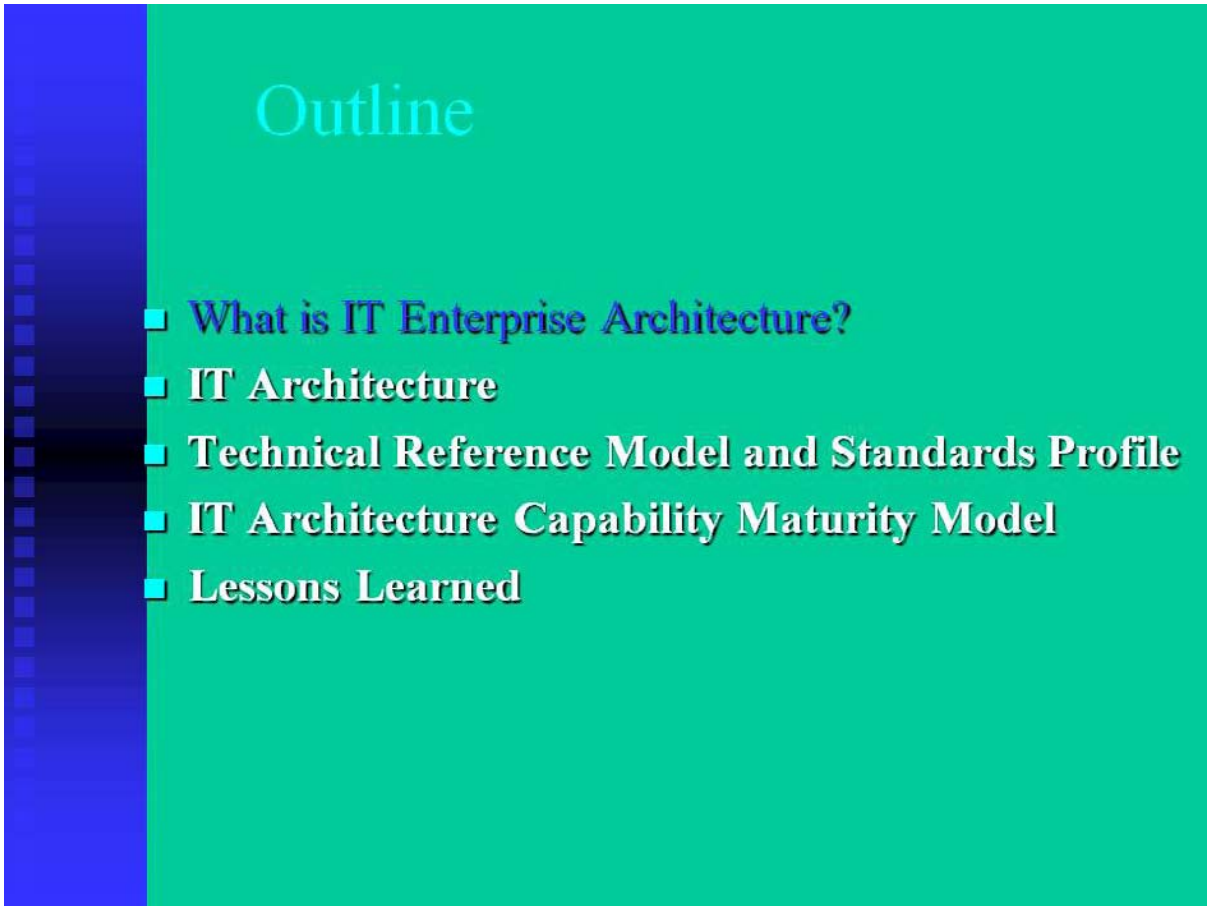
Addresses the top 100 consultancy and education forum questions, with tips and success factors on TOGAF design, planning, implementation, and governance of enterprise IT architecture.

ENTERPRISE ARCHITECTURE



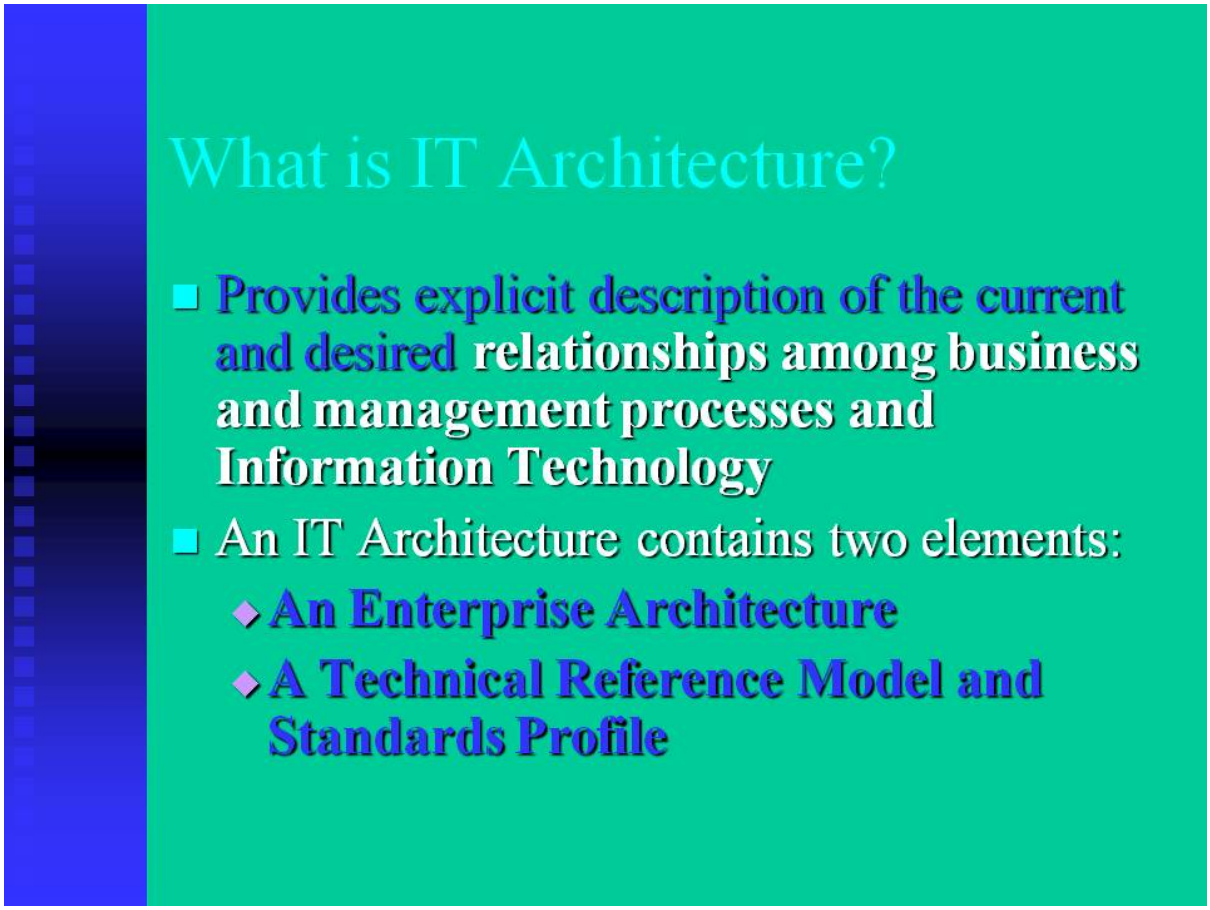
Objectives

- How to establish a Federated IT Architecture hierarchal structure
- Learn how to establish a Federated IT Architecture governance process
- The role of a Technical Reference Model and Standards Profile in a Federated IT Architecture
- The role of IT Architecture Capability Maturity Model in a Federated IT Architecture
- Apply lessons learned from two Federated IT Architecture efforts



What is IT Architecture?

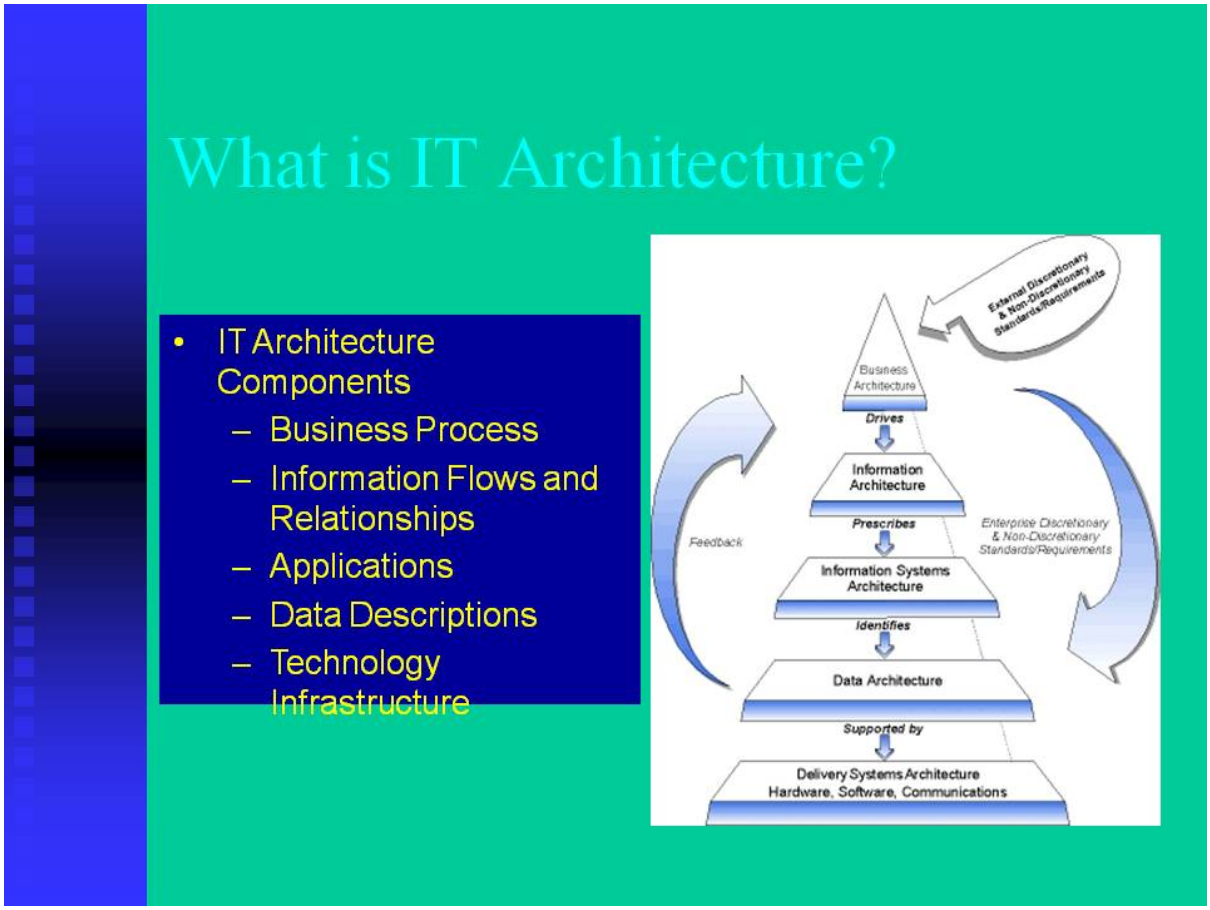
- A blueprint that explains how all the Information Technology and Management elements work together as a whole
- Provides explicit description of the current and desired relationships among business and management processes and Information Technology
- IT Architecture contains two elements:
  - An Enterprise Architecture
  - A Technical Reference Model and Standards Profile



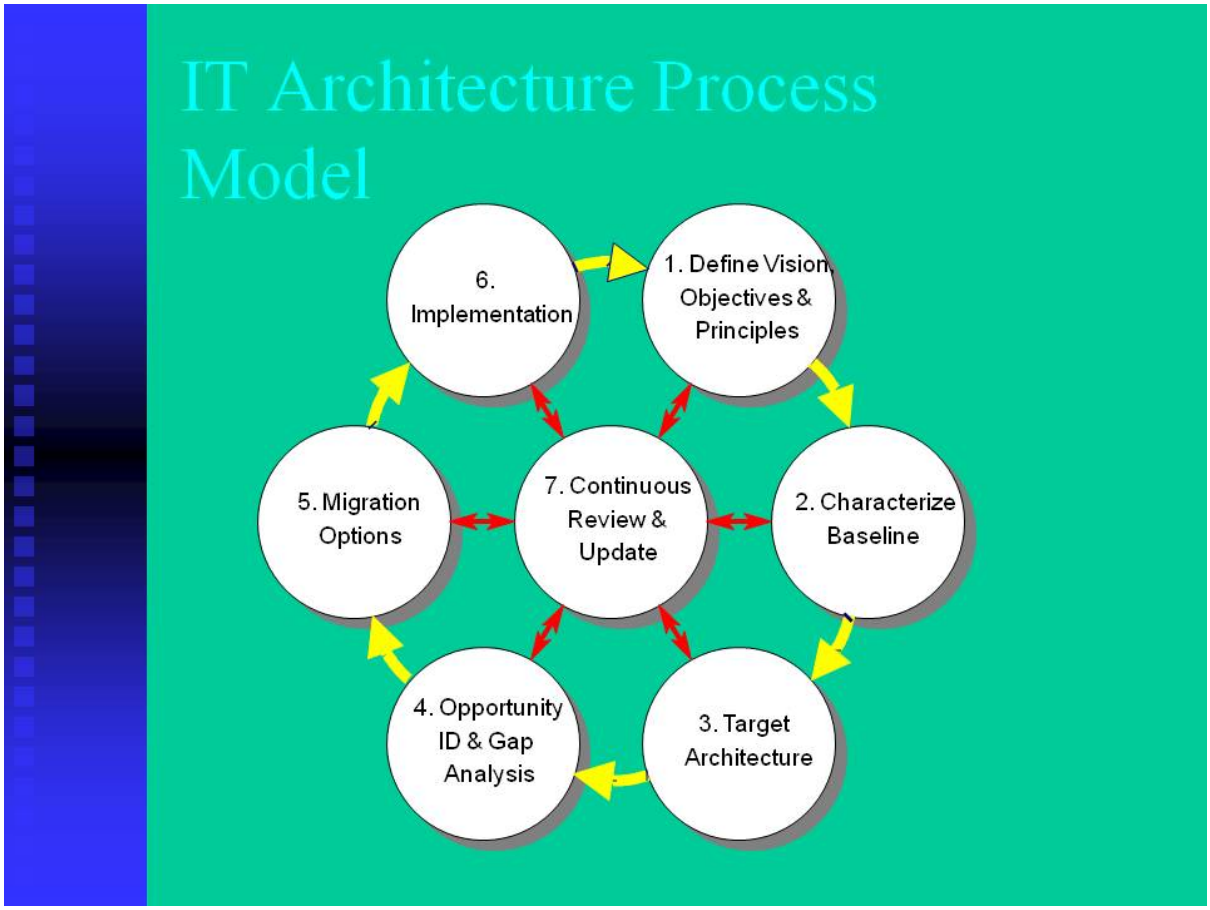
(a) What is the Enterprise Architecture?

An EA is the explicit description and documentation of the current and desired relationships among business and management processes and Information Technology. It describes the "current architecture" and "target architecture" to include the rules and standards and systems life cycle information to optimize and maintain the environment which the agency wishes to create and maintain by managing its IT portfolio.

The EA must also provide a strategy that will enable the agency to support its current state and also act as the roadmap for transition to its target environment. These transition processes will include an agency's capital planning and investment control processes, agency EA planning processes, and agency systems life cycle methodologies. The EA will define principles and goals and set direction on such issues as the promotion of interoperability, open systems, public access, compliance with GPEA, end user satisfaction, and IT security.



Please see The Overview of Enterprise Architecture on page 69 within this workbook.



Implementing an IT Enterprise Architecture

Create IT Architecture team

Get management's support and commitment to the IT Architecture process

Take a snapshot (baseline) of what currently exists

Determine where the Organization wants to be in the next three to five years

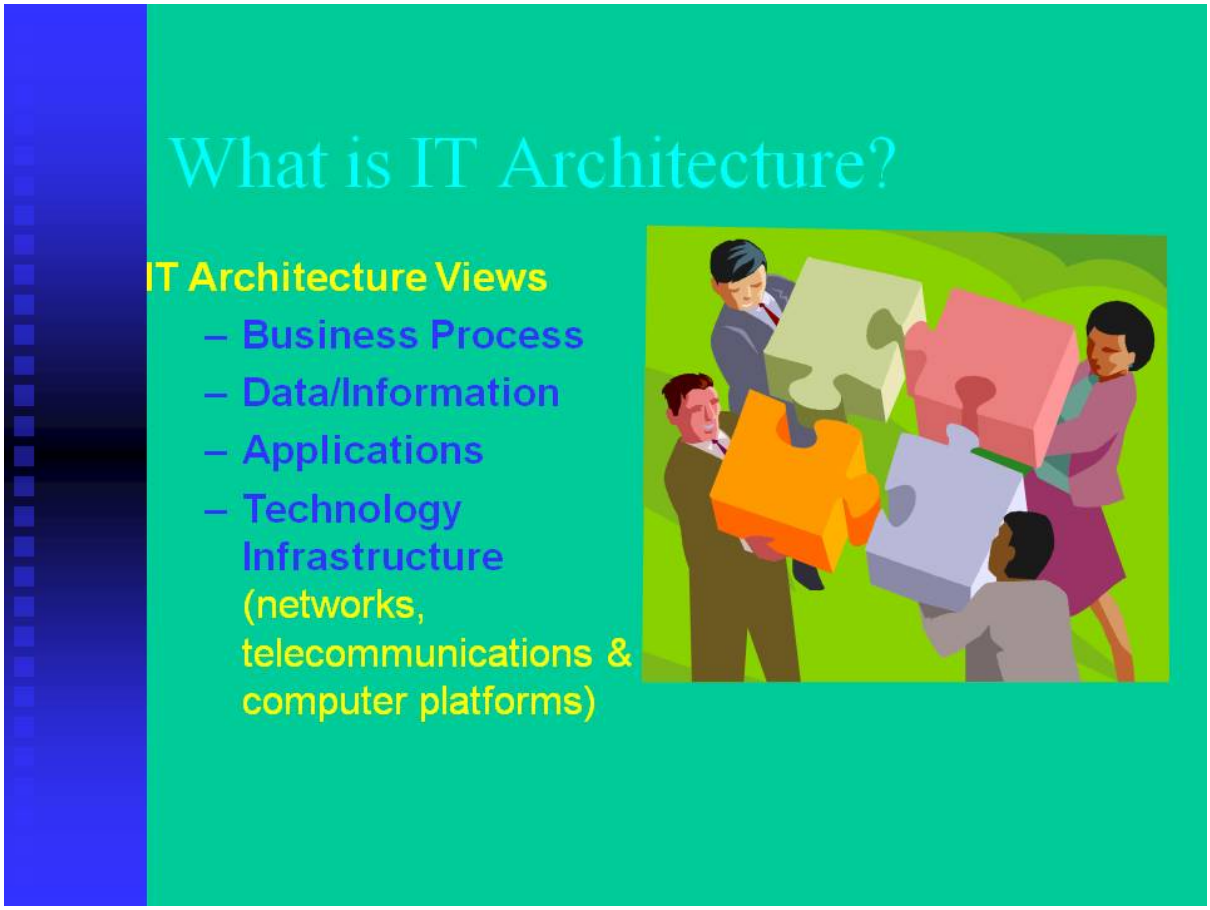
Develop a blueprint that explains how all the IT elements work together as a whole.

Determine the gaps between the current baseline and the future vision

Develop a plan and budget for bridging the gaps and reaching the IT goals

Identify and accomplish early success projects that will create buy-in

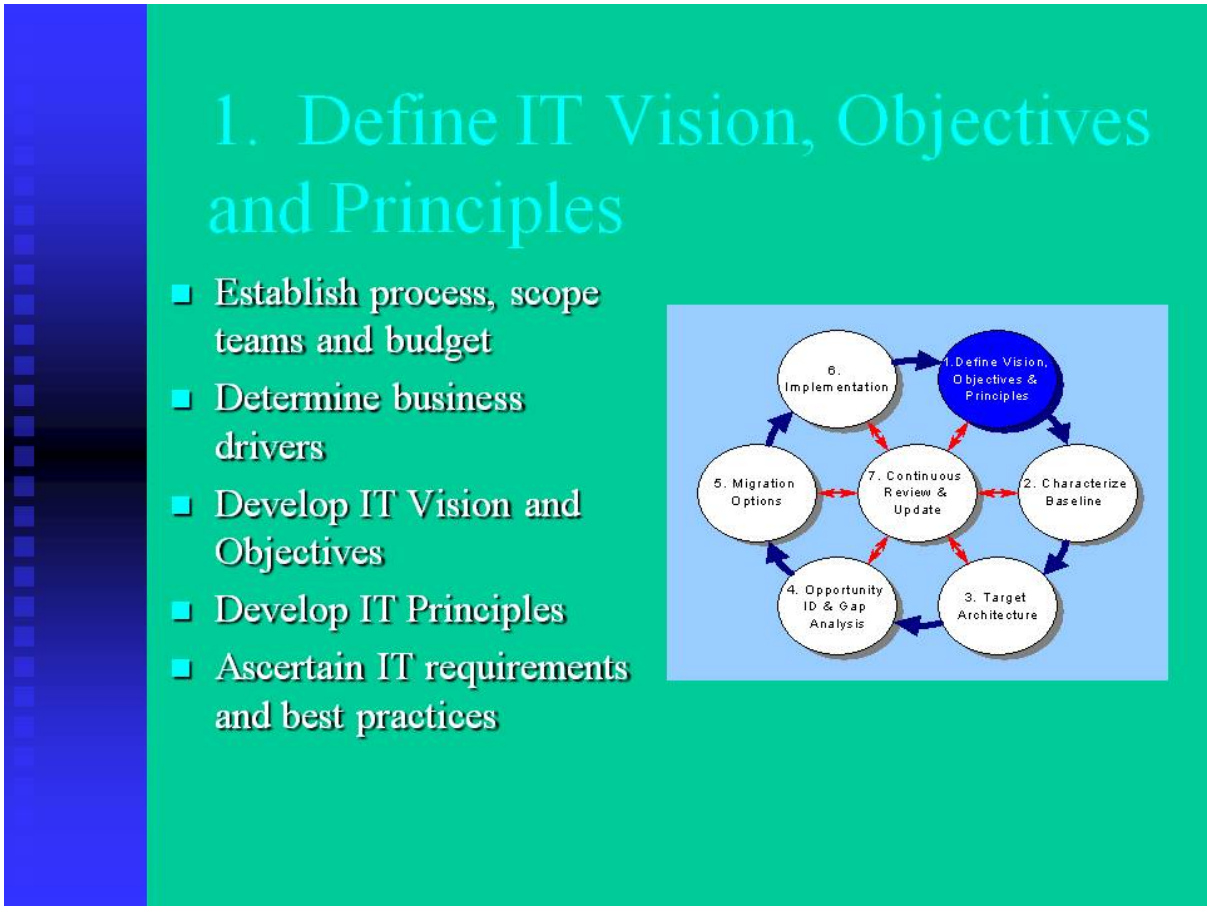
Update annually to accommodate strategic and budget plans.



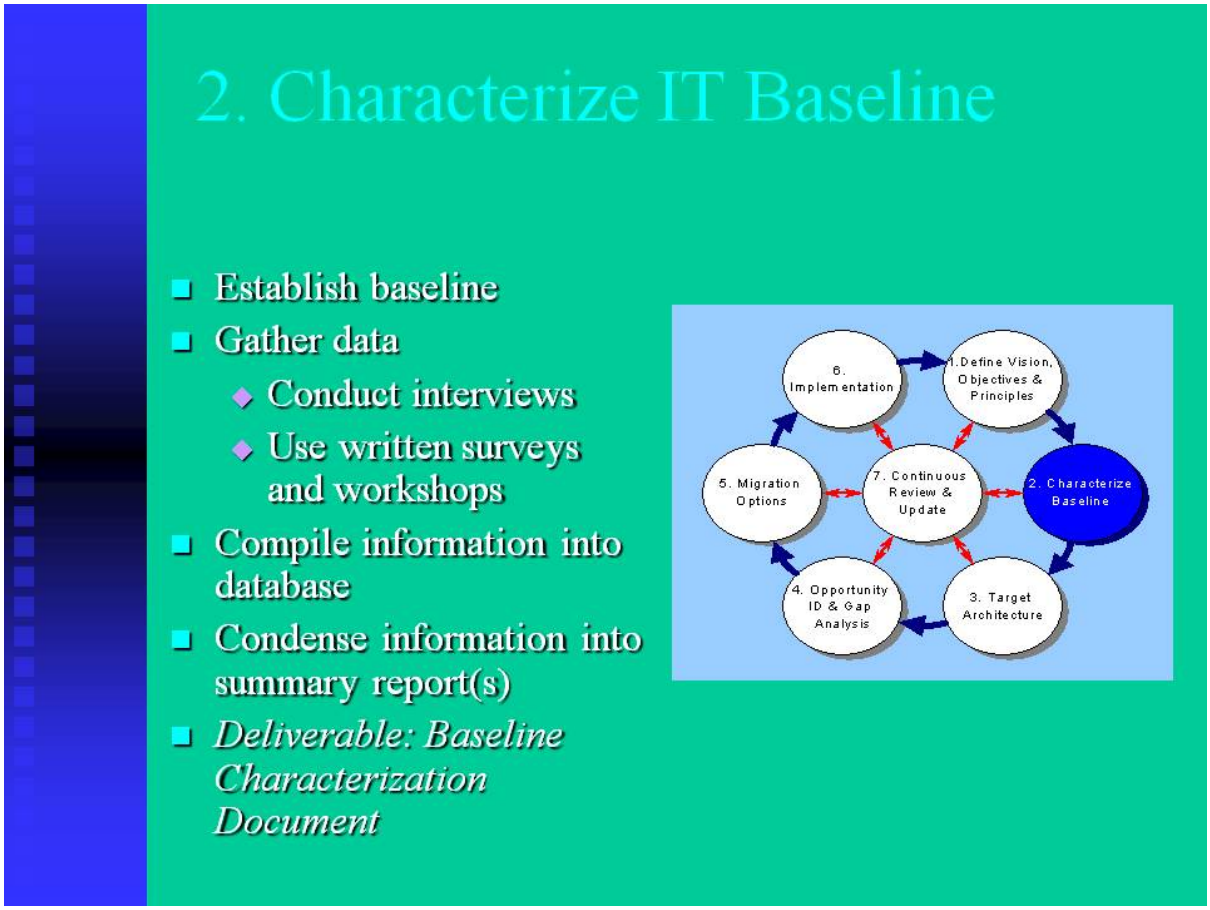
What is IT Architecture?

IT Architecture Describes:

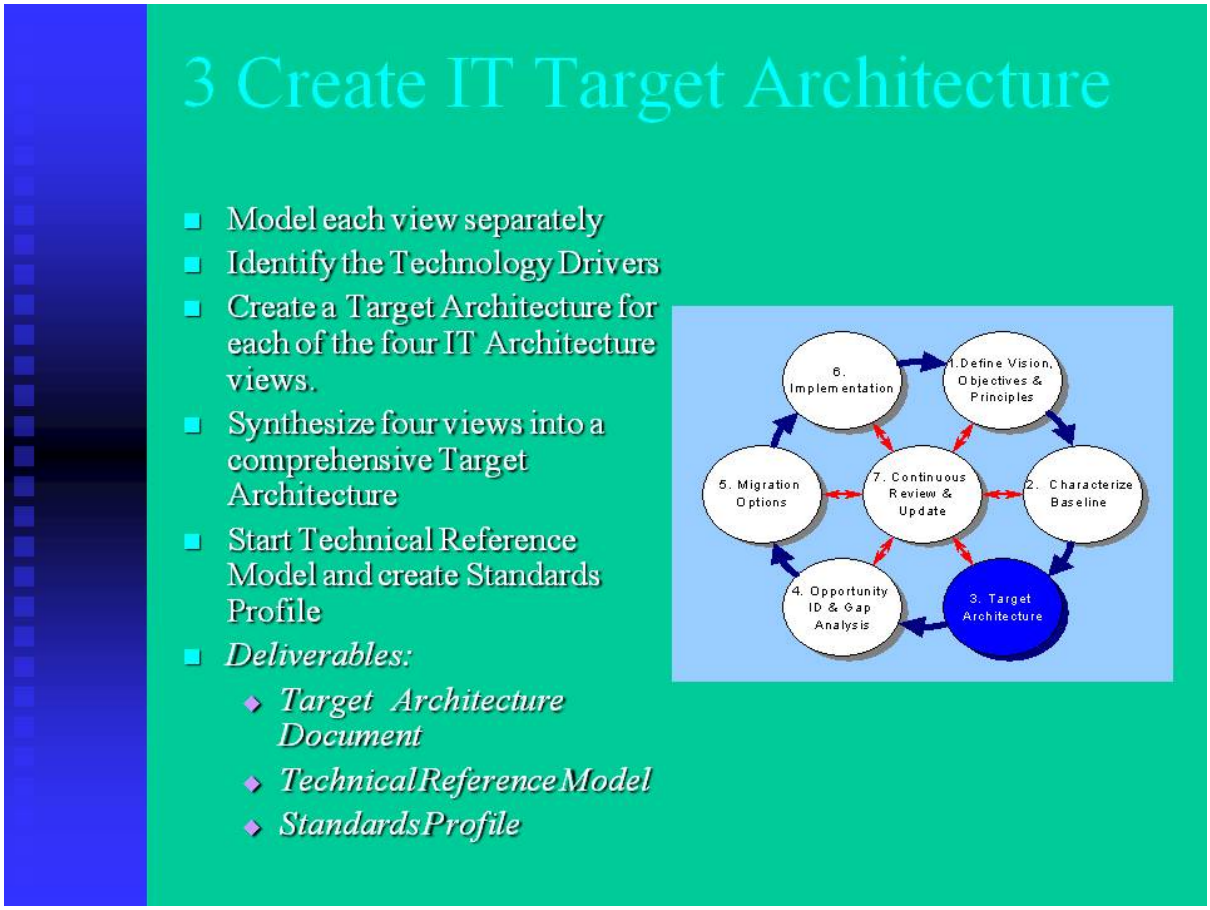
- The way Work activities are organized and the Locations where the work is carried out
- The Information Sets/Databases needed to perform the work
- The Applications/Software that capture and manipulate the information sets
- The Technology Infrastructure (hardware, network and communications) needed to run the applications



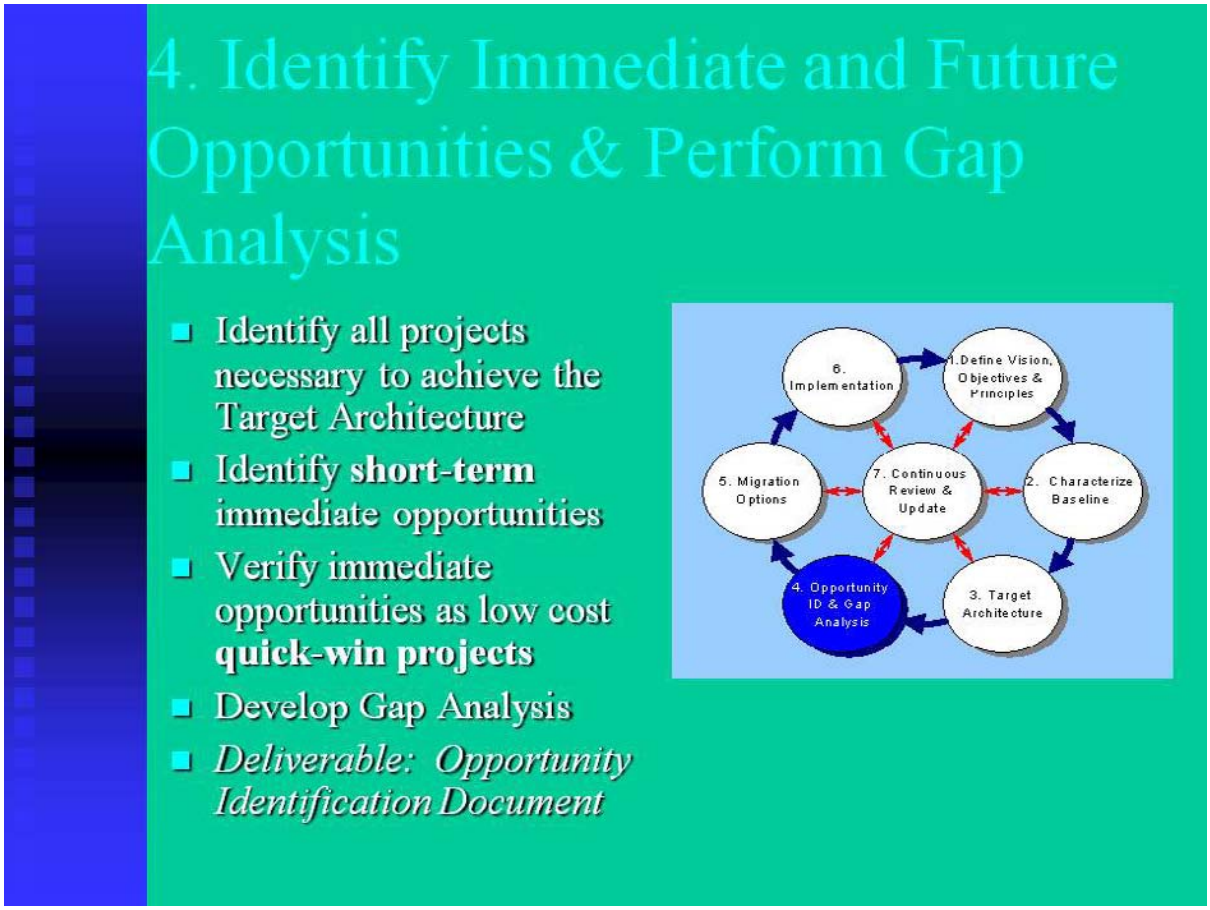
- Align the IT Architecture with strategic plan.
- Establish IT Architecture Vision.
- Clearly state and establish IT Architecture Principles for all IT Architecture views
- Define IT Requirements
- Link the IT Architecture to the IT Capital Planning process.
- Document Step 1 Activities



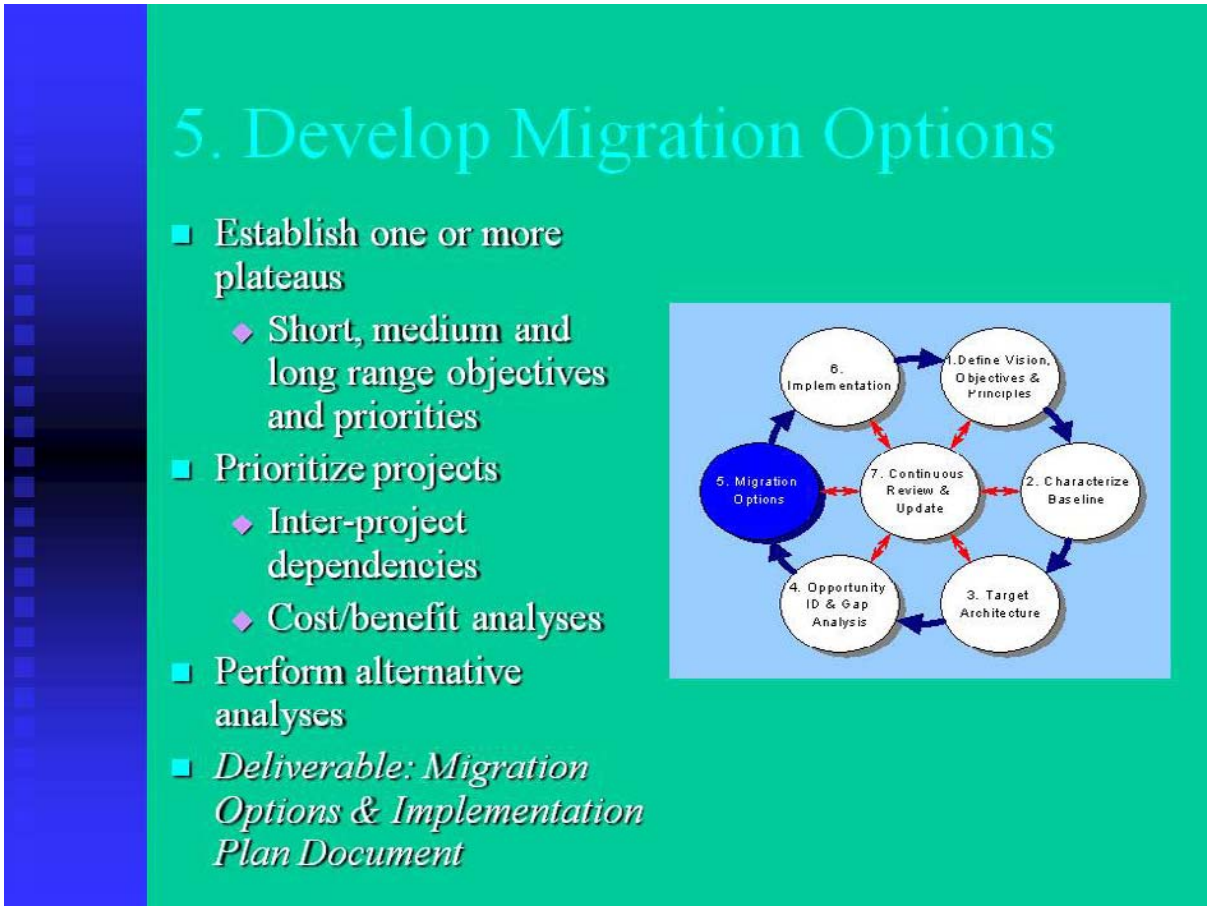
- Develop an inventory (IT Baseline) of existing Information Sets, Databases, Applications and Technology Infrastructure.
- Identify the information and data flows within the bureau and with constituents and collaborators.
- Document Step 2 Activities



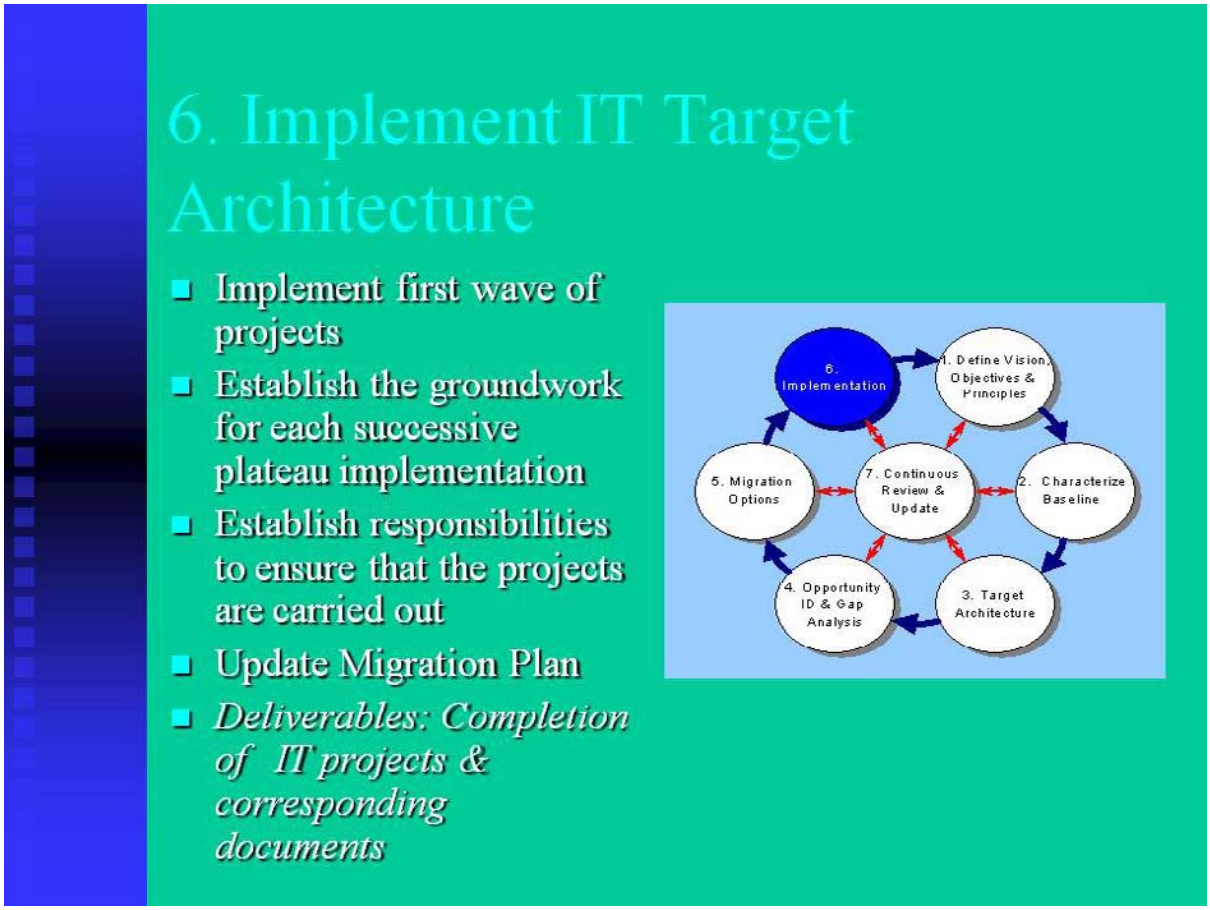
- Align the IT Architecture with strategic plan.
- Establish IT Architecture Vision.
- Clearly state and establish IT Architecture Principles for all IT Architecture views
- Define IT Requirements
- Link the IT Architecture to the IT Capital Planning process.
- Document Step 1 Activities



- Identify all projects necessary to achieve the IT Target Architecture.
- Perform a Gap Analysis.
- Identify short-term immediate opportunities.
- Identify short term immediate opportunities that can result in visible ‘quick-win’ projects.
- Document Step 4 Activities



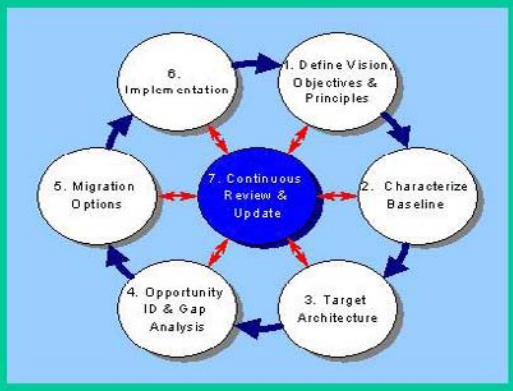
- Classify all projects as short (6 - 18 months), medium (18 - 36 months)and long (3 - 5 years) range.
- Prioritize within each classification (short, medium and long range), all projects
- Establish and document Data Dictionaries, Software Developmental Methodologies and Configuration Management Processes.
- Develop IT Architecture Migration Plan.
- Document Step 5 Activities.

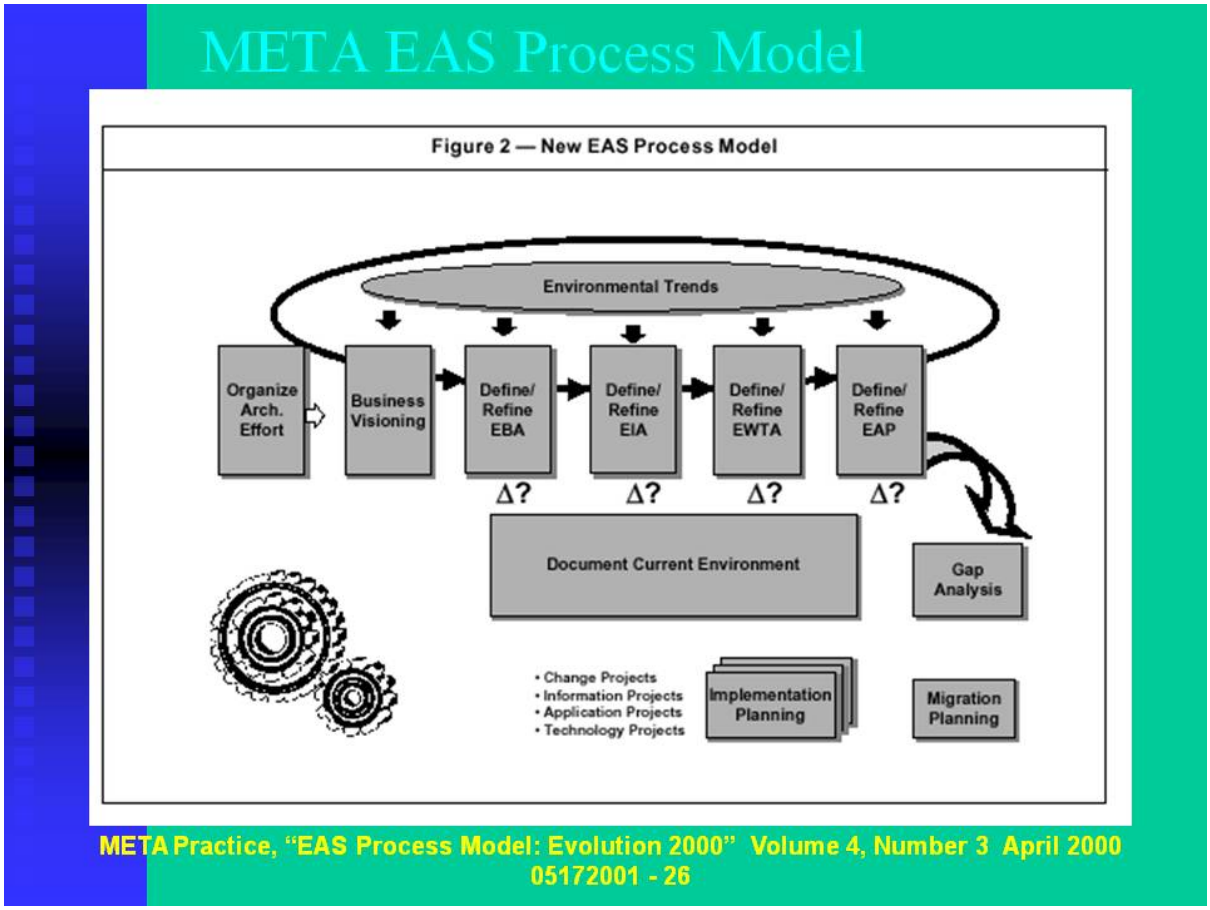


- Identify IT Architecture project leader for each IT Architecture project to be implemented.
- Establish roles and responsibilities for IT Architecture project implementation.
- Establish a project plan and milestone schedule for each project.
- Document Step 6 Activities.

7. Continuously Review and Update IT Enterprise Architecture

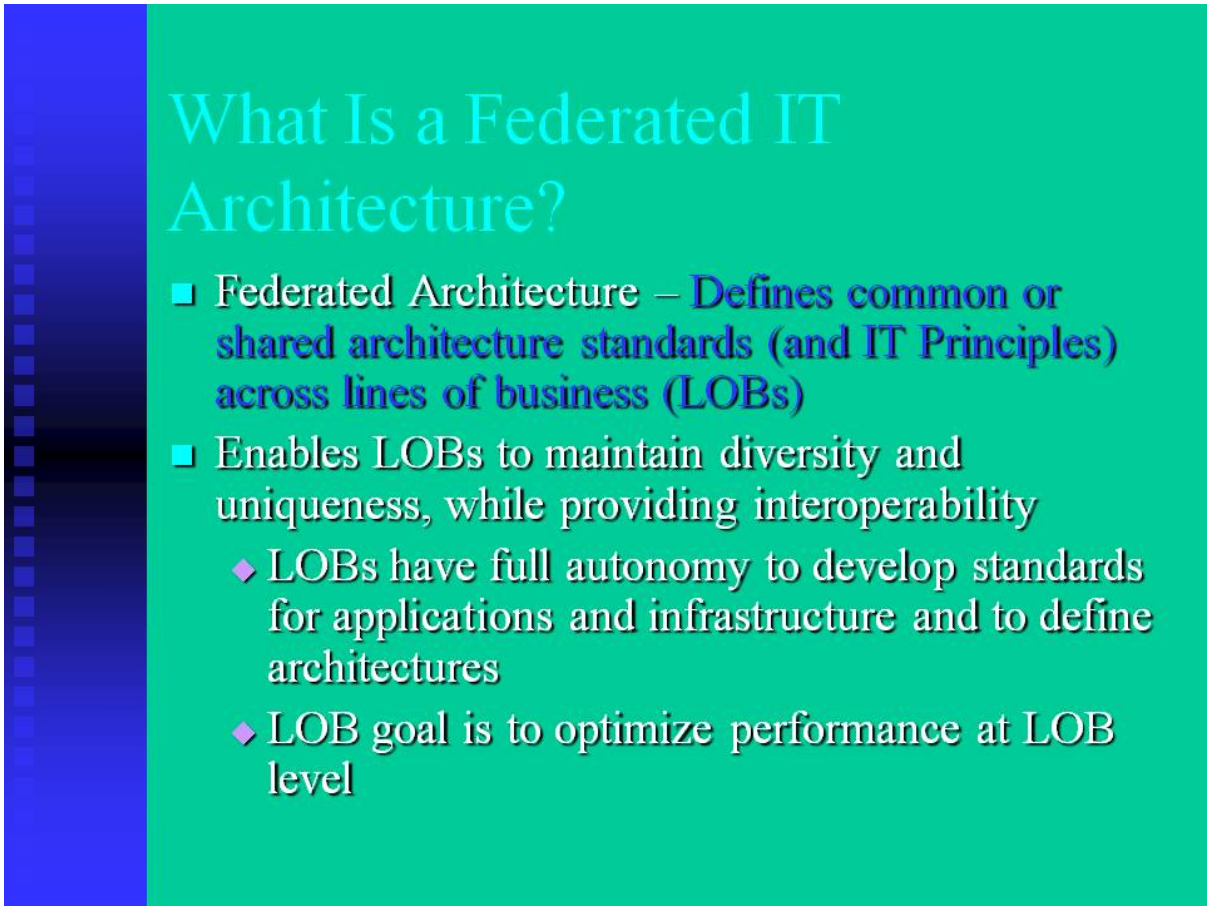
- Adjust the IT Enterprise Architecture decisions for unforeseen changes
- Make adjustments based on experience
- Ensure modifications reflect a realistic approach
- **Can cause a reentry into the process at any point**
- Update IT Architecture, annually



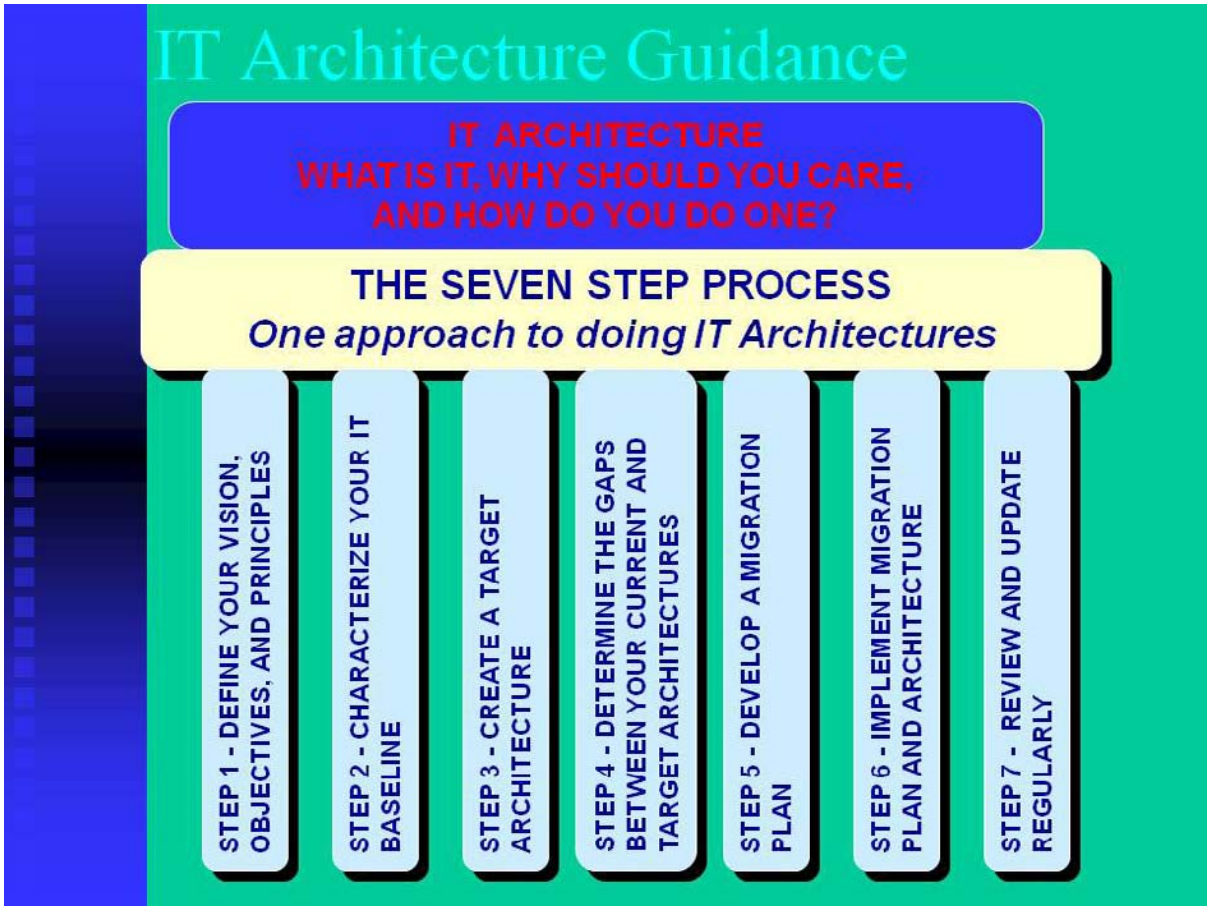


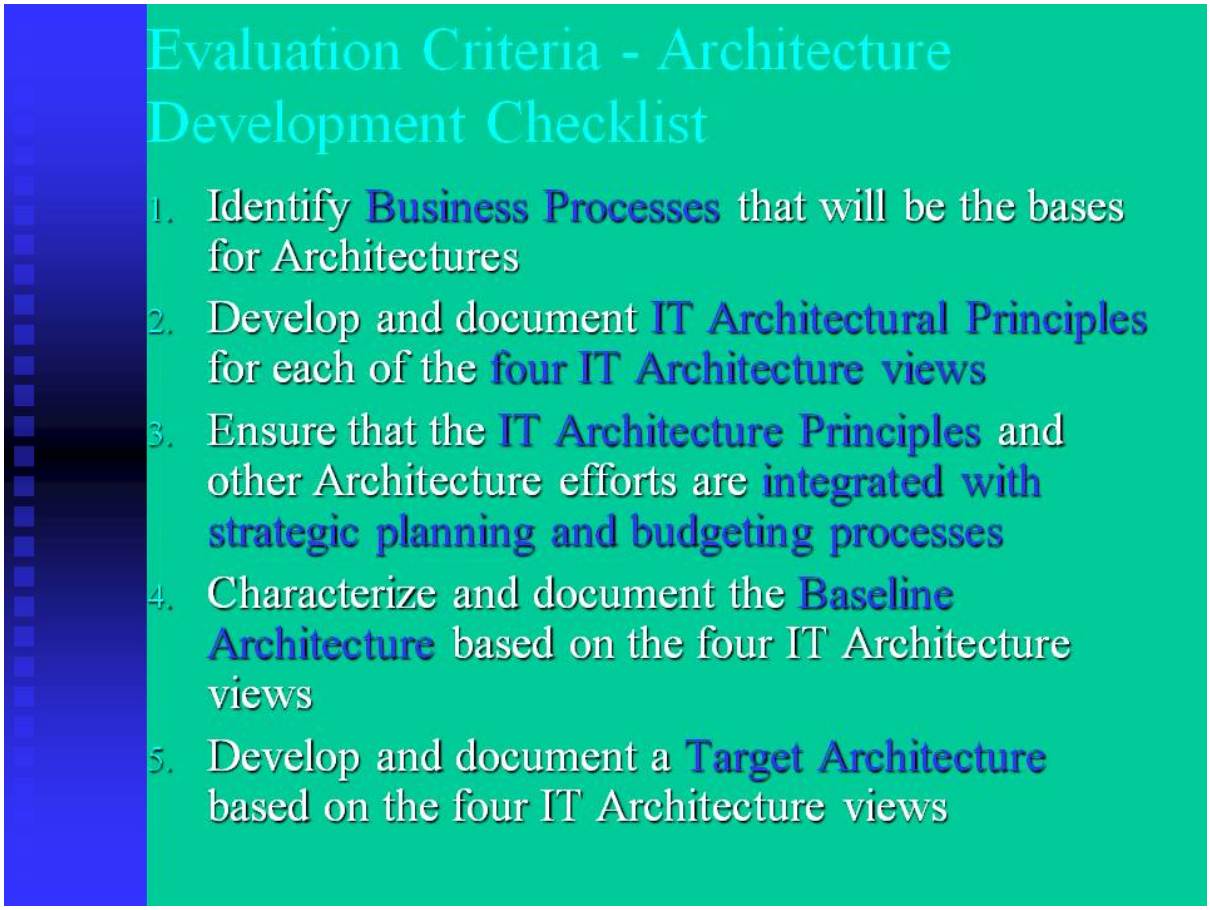
"EAS Process Model: Evolution 2000" Volume 4, Number 3 April 2000

Please see The Basics on Enterprise Architecture on page 77 within this workbook.



META Delta 46, "Federated Architectures: Integrating Autonomous LOBs", March 1, 1999.

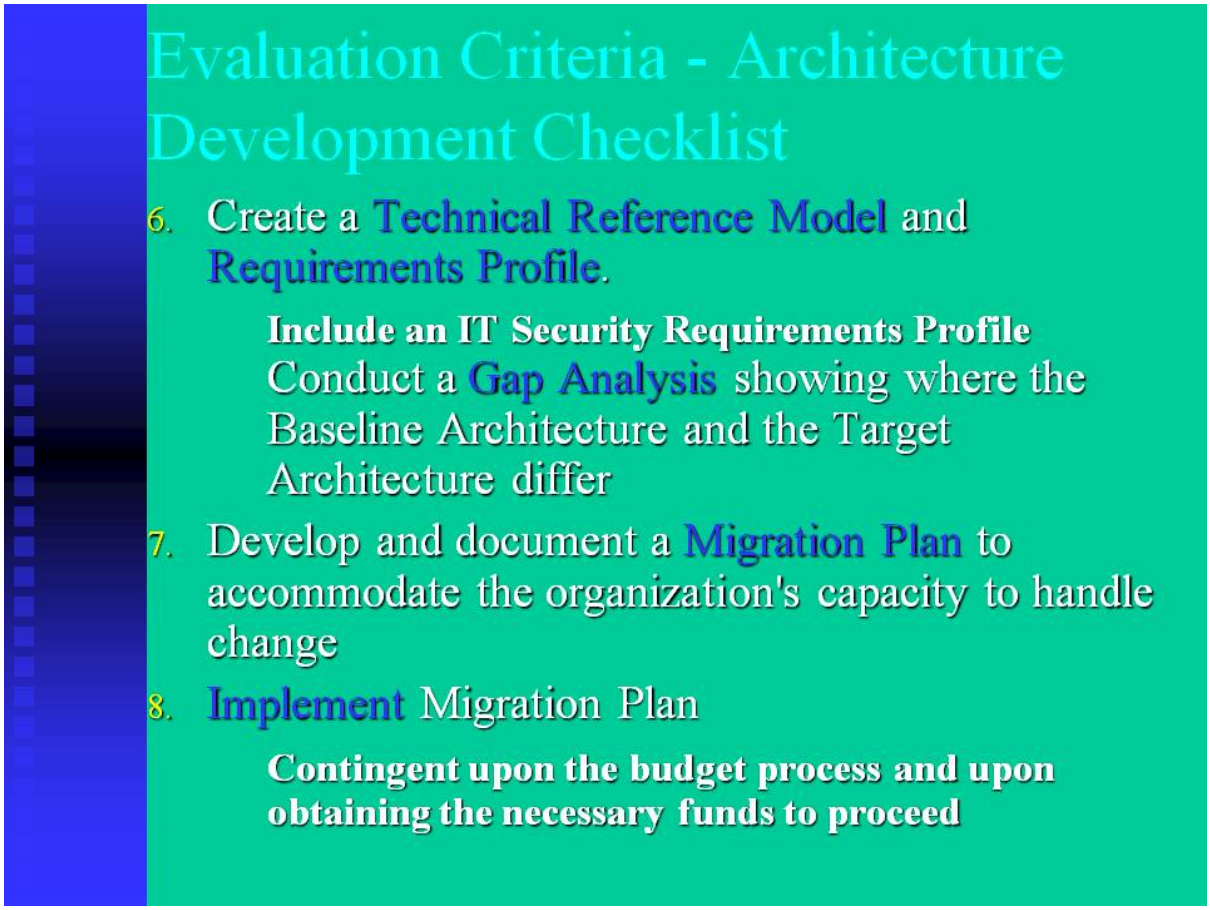

IT Architecture Guidance
IT ARCHITECTURE
WHAT IS IT, WHY SHOULD YOU CARE,
AND HOW DO YOU DO ONE?
THE SEVEN STEP PROCESS
One approach to doing IT Architectures
STEP 1 - DEFINE YOUR VISION, OBJECTIVES, AND PRINCIPLES
STEP 2 - CHARACTERIZE YOUR IT BASELINE
STEP 3 - CREATE A TARGET ARCHITECTURE
STEP 4 - DETERMINE THE GAPS BETWEEN YOUR CURRENT AND TARGET ARCHITECTURES
STEP 5 - DEVELOP A MIGRATION PLAN
STEP 6 - IMPLEMENT MIGRATION PLAN AND ARCHITECTURE
STEP 7 - REVIEW AND UPDATE REGULARLY

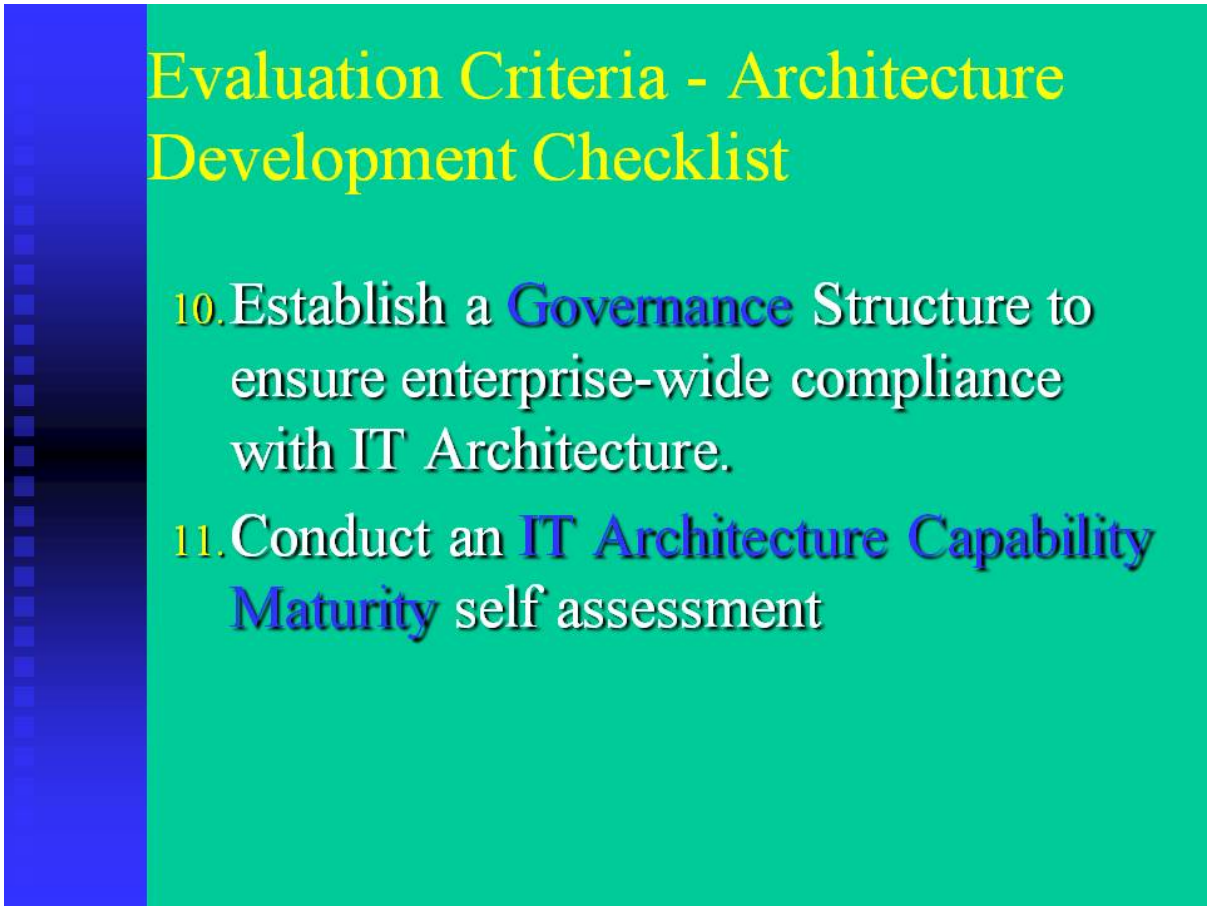


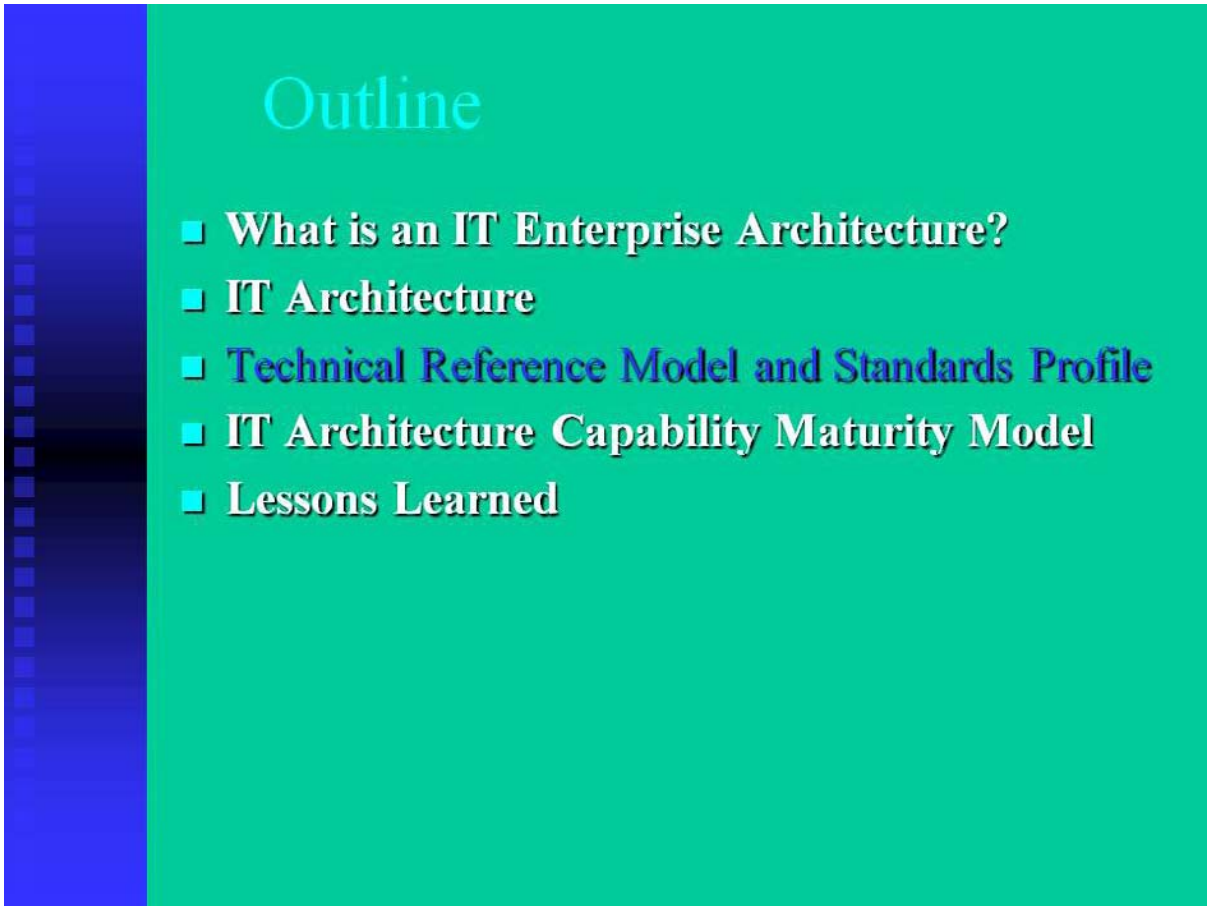
Three Columns: Standards and Related Comments Mandatory Guidance

Letter signed by Roger Baker, XXX CIO to CIO of Operating Unit

Evaluation Table accompanies Letter

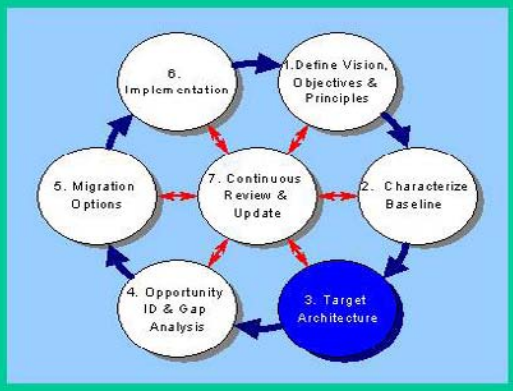

Evaluation Criteria - Architecture Development Checklist
6. Create a Technical Reference Model and Requirements Profile.
Include an IT Security Requirements Profile
Conduct a Gap Analysis showing where the Baseline Architecture and the Target Architecture differ
7. Develop and document a Migration Plan to accommodate the organization's capacity to handle change
8. Implement Migration Plan
Contingent upon the budget process and upon obtaining the necessary funds to proceed

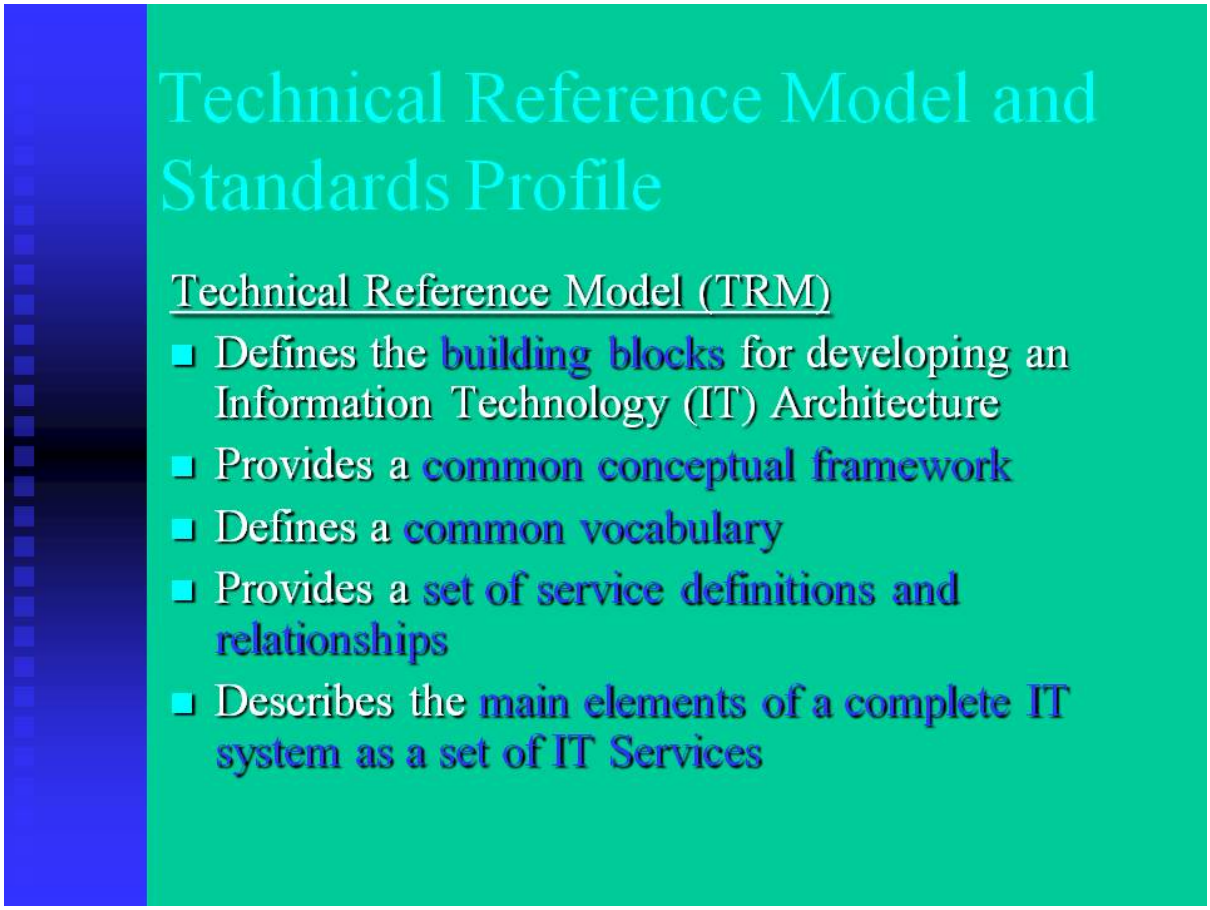

Evaluation Criteria - Architecture Development Checklist
10. Establish a Governance Structure to ensure enterprise-wide compliance with IT Architecture.
11. Conduct an IT Architecture Capability Maturity self assessment


Outline
What is an IT Enterprise Architecture?
IT Architecture
Technical Reference Model and Standards Profile
IT Architecture Capability Maturity Model
Lessons Learned

3. Create IT Target Architecture

- Model each view separately
- Identify the Technology Drivers
- Create a Target Architecture for each of the four IT Architecture views.
- Synthesize four views into a comprehensive Target Architecture
- **Develop Technical Reference Model and Standards Profile**
- ***Deliverables:***
  - *Target Architecture Document*
  - ***Technical Reference Model & Standards Profile, First Edition***

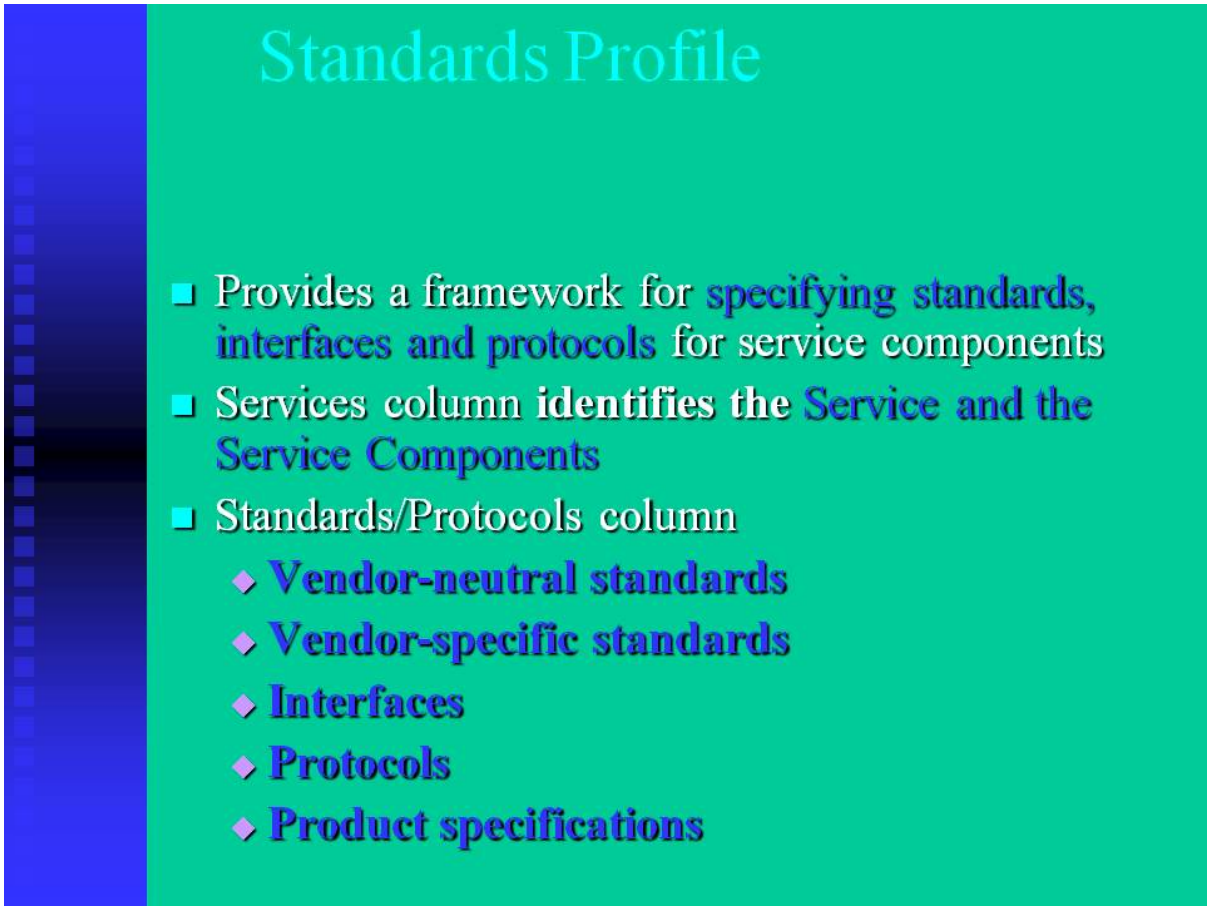




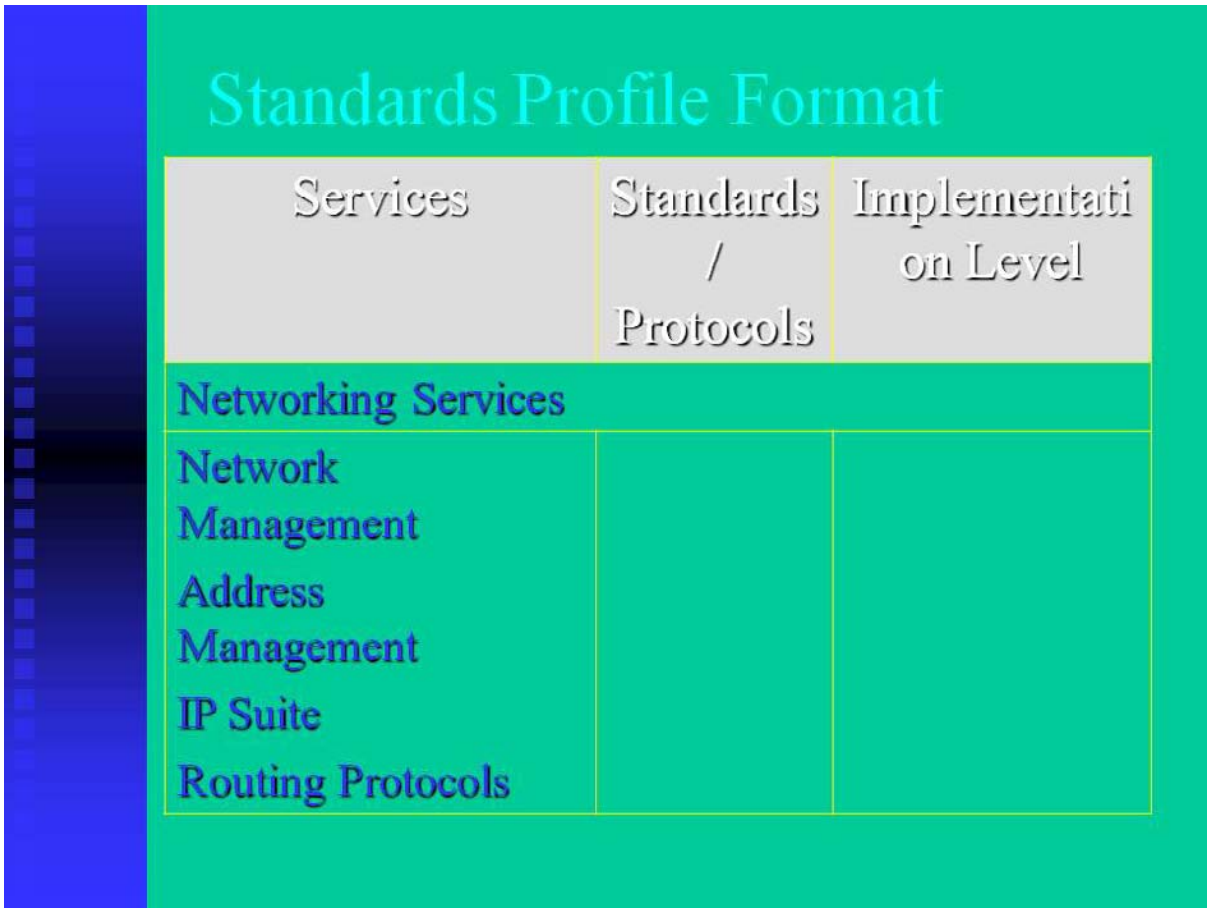
Objectives of TRM

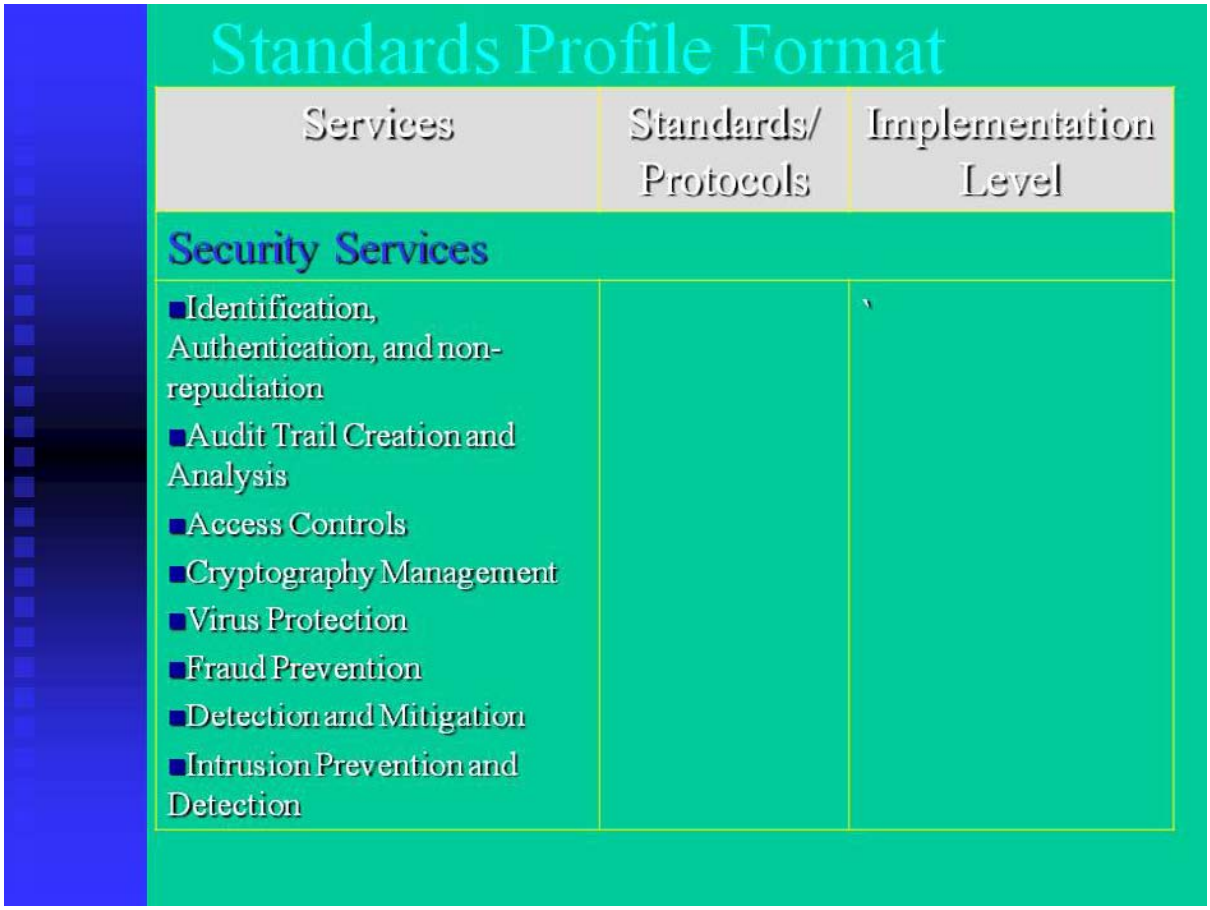
IT Service Categories

- Security
- Networking
- Operating System
- User (Person)/Computer Interface
- Data Management
- Data Interchange
- Multimedia/Graphics
- Communications
- Document Management
- Support
- Hardware



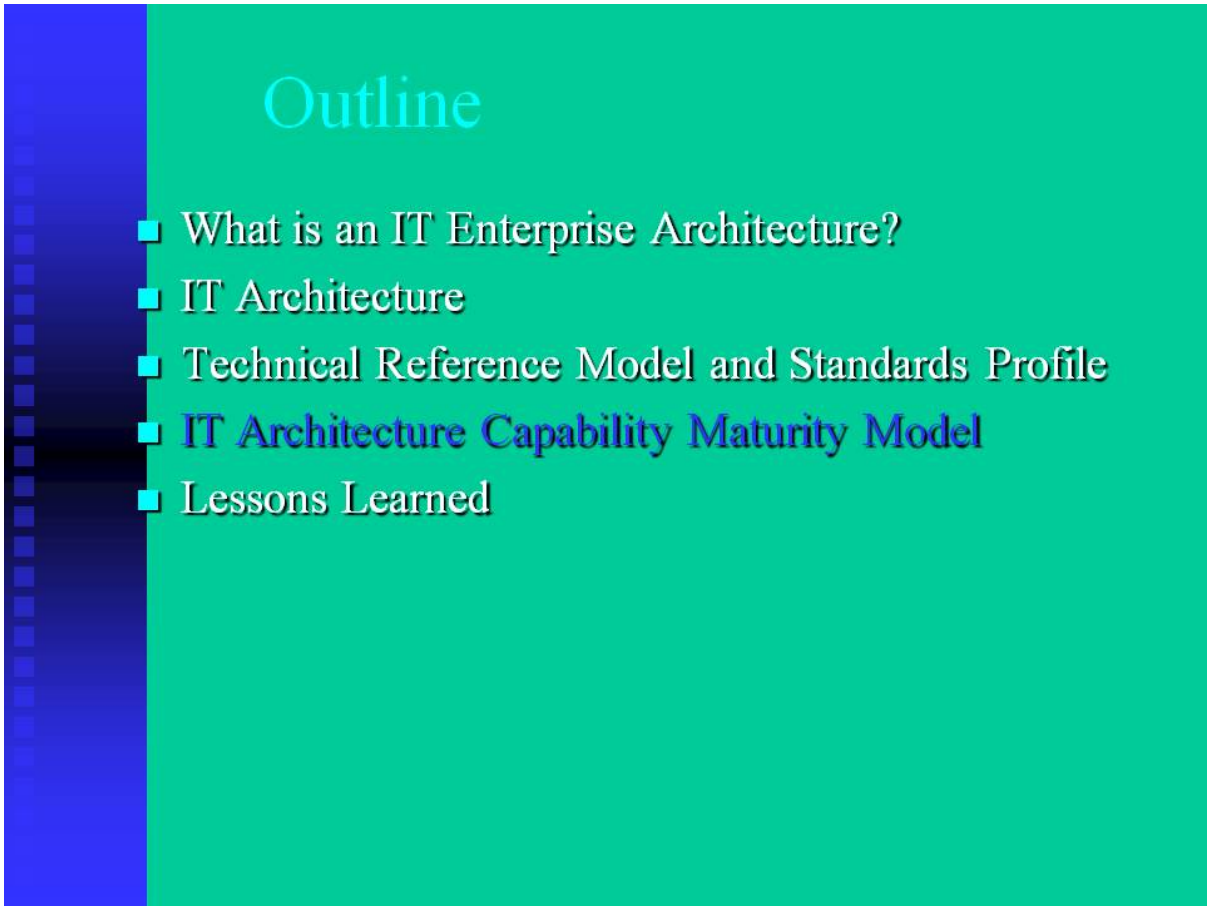
- Provides a framework for specifying standards, interfaces and protocols for service components
- Services column identifies the Service and the Service Components
- Standards/Protocols column
  - Vendor-neutral standards
  - Vendor-specific standards
  - Interfaces
  - Protocols
  - Product specifications
- Implementation Level column indicates the level of compliance or adoption for the invoked standard, protocol or interface
- Classifications Levels
  - Mandatory
  - Recommended
  - Emerging
  - Obsolete

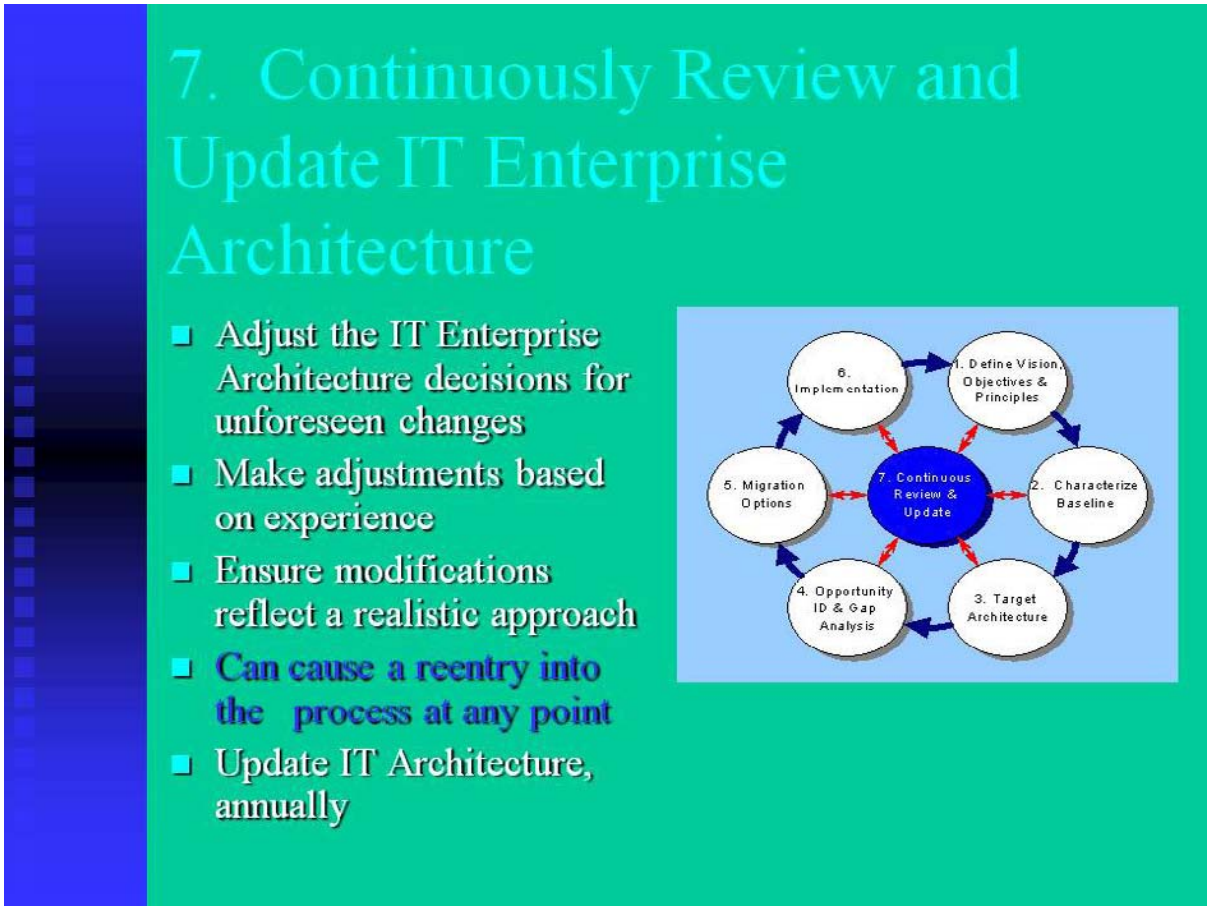

Standards Profile Format
Services
Standards / Protocols
Implementation Level
Networking Services
Network Management
Address Management
IP Suite
Routing Protocols



Standards Profile Format – Sample Write-up

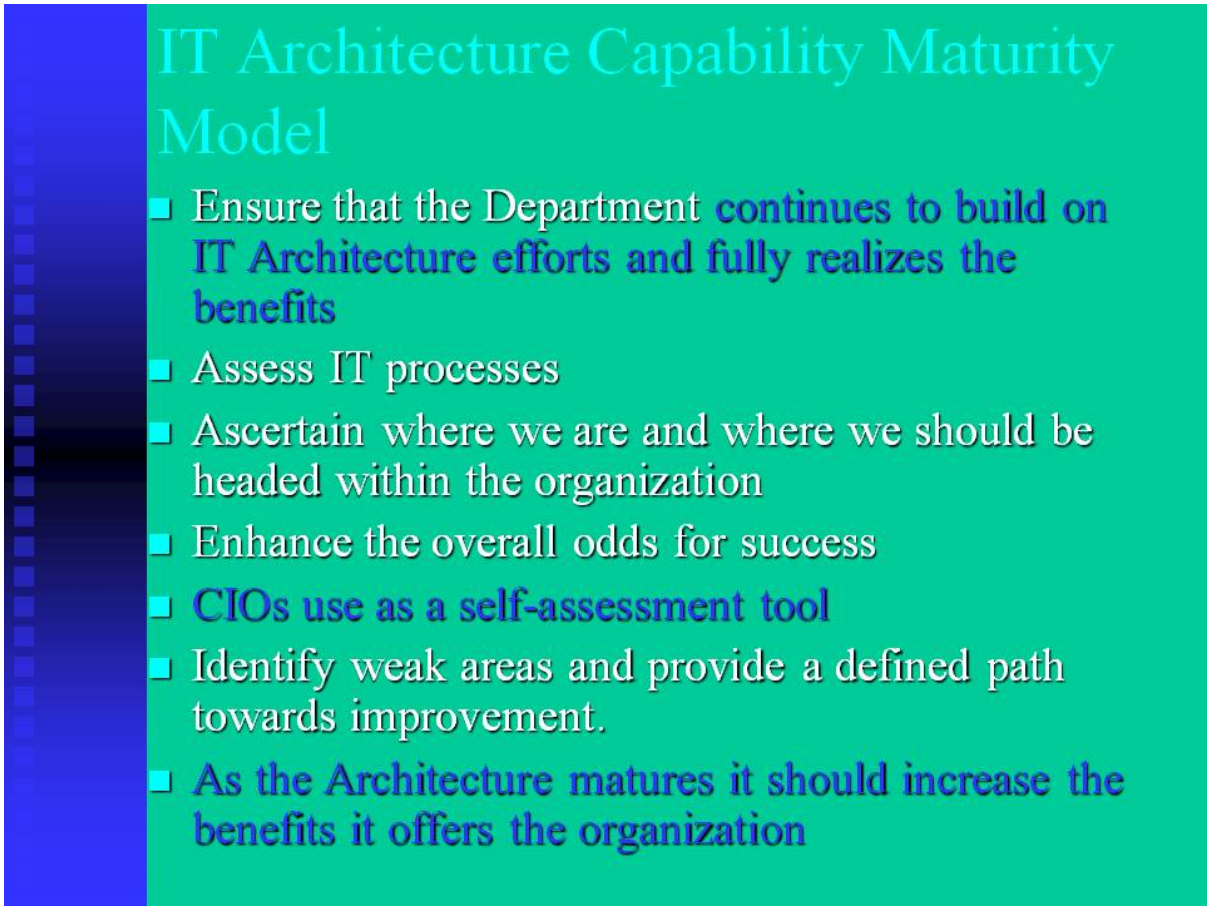
- TCP/IP Implementation: TCP/IP and its suite of protocols (IETF Implementation STD5, STD7)
- Implementation Level: Mandatory adoption
- **Description**: TCP/IP is a well-established, widely adopted industry standard communications protocol. It is the primary means of communications throughout the Internet. TCP/IP provides for a reliable, connection-oriented, end-to-end transport service on top of an unreliable network…. The new generation IP (IP Version 6 - also known as IPng) will alleviate the address depletion problems.
- Rationale: Adopting TCP/IP network protocol will simplify communications and data exchange … TCP/IP is supported by almost every supplier of communications products.
- Implications: TCP/IP can be used over almost any kind of network to provide quality end-to-end transmission service.


Outline
What is an IT Enterprise Architecture?
IT Architecture
Technical Reference Model and Standards Profile
IT Architecture Capability Maturity Model
Lessons Learned



IT Architecture Capability Maturity Model

- Ensure that the Department continues to build on IT Architecture efforts and fully realizes the benefits
- Assess IT processes
- Ascertain where we are and where we should be headed within the organization
- Enhance the overall odds for success
- CIOs use as a self-assessment tool
- Identify weak areas and provide a defined path towards improvement.
- As the Architecture matures it should increase the benefits it offers the organization


IT Architecture Capability Maturity Model
Ensure that the Department continues to build on IT Architecture efforts and fully realizes the benefits
Assess IT processes
Ascertain where we are and where we should be headed within the organization
Enhance the overall odds for success
CIOs use as a self-assessment tool
Identify weak areas and provide a defined path towards improvement.
As the Architecture matures it should increase the benefits it offers the organization

IT Architecture Capability Maturity Model[1]

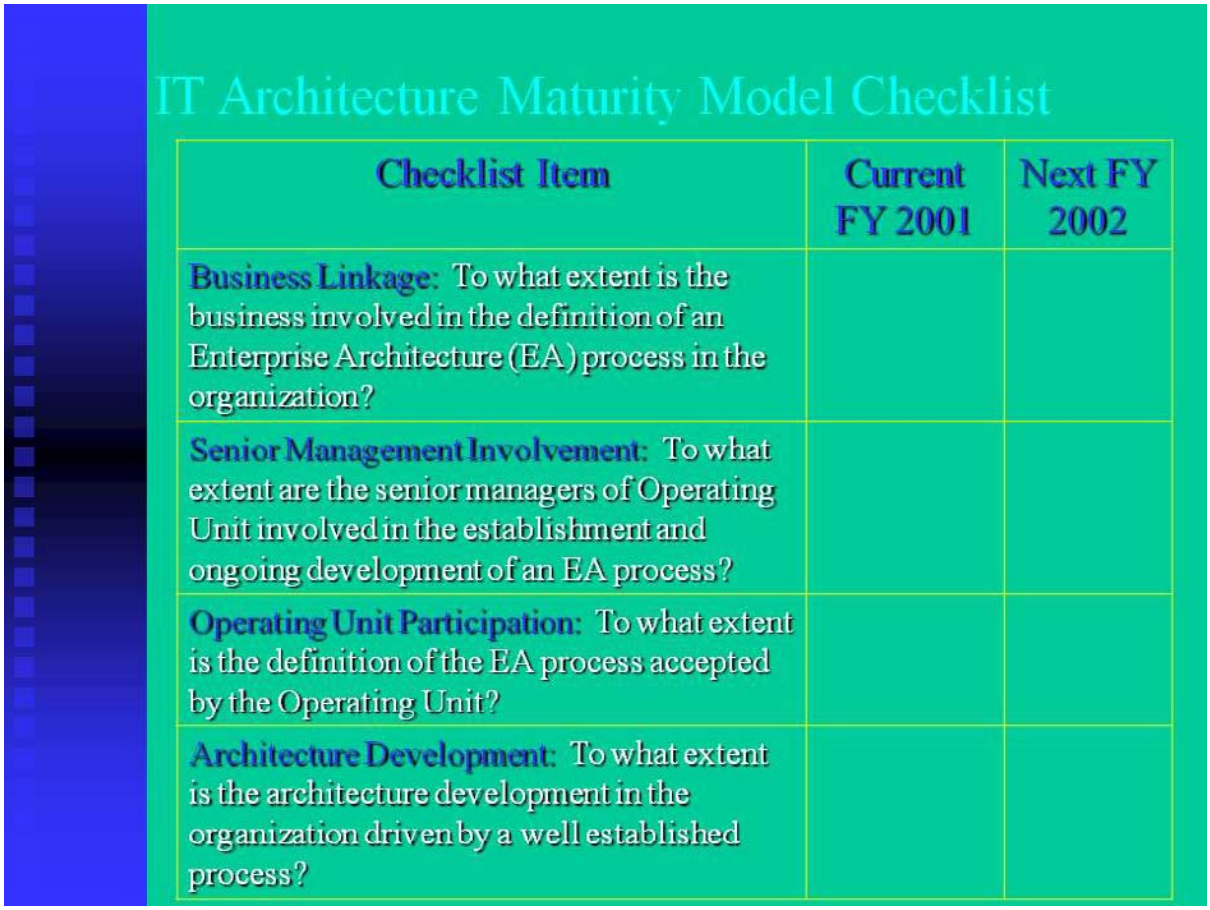
Level	Focus	Characteristics
3	Defined IT Architecture Including Detailed Written Procedures and Technical Reference Model	The architecture is well defined and communicated. The process is largely followed. Gap Analysis, Migration Plan, Technical Reference Model, Standards Profile, and Migration Plan are completed. Cost-benefits are considered in identifying projects. IT goals and methods are identified. Training and awareness programs provided at regular intervals. IT Architecture is integrated with strategic planning and budgeting processes.

Characteristics of Operating Units' IT Architecture Processes at Different Maturity Levels

- Business Linkage
- Senior-Management Involvement
- Operating Unit Participation
- Architecture Process Definition
- Architecture Development
- Architecture Communication
- Governance
- Program Management
- Holistic Enterprise Architecture
- IT Investment, and Procurement Strategy

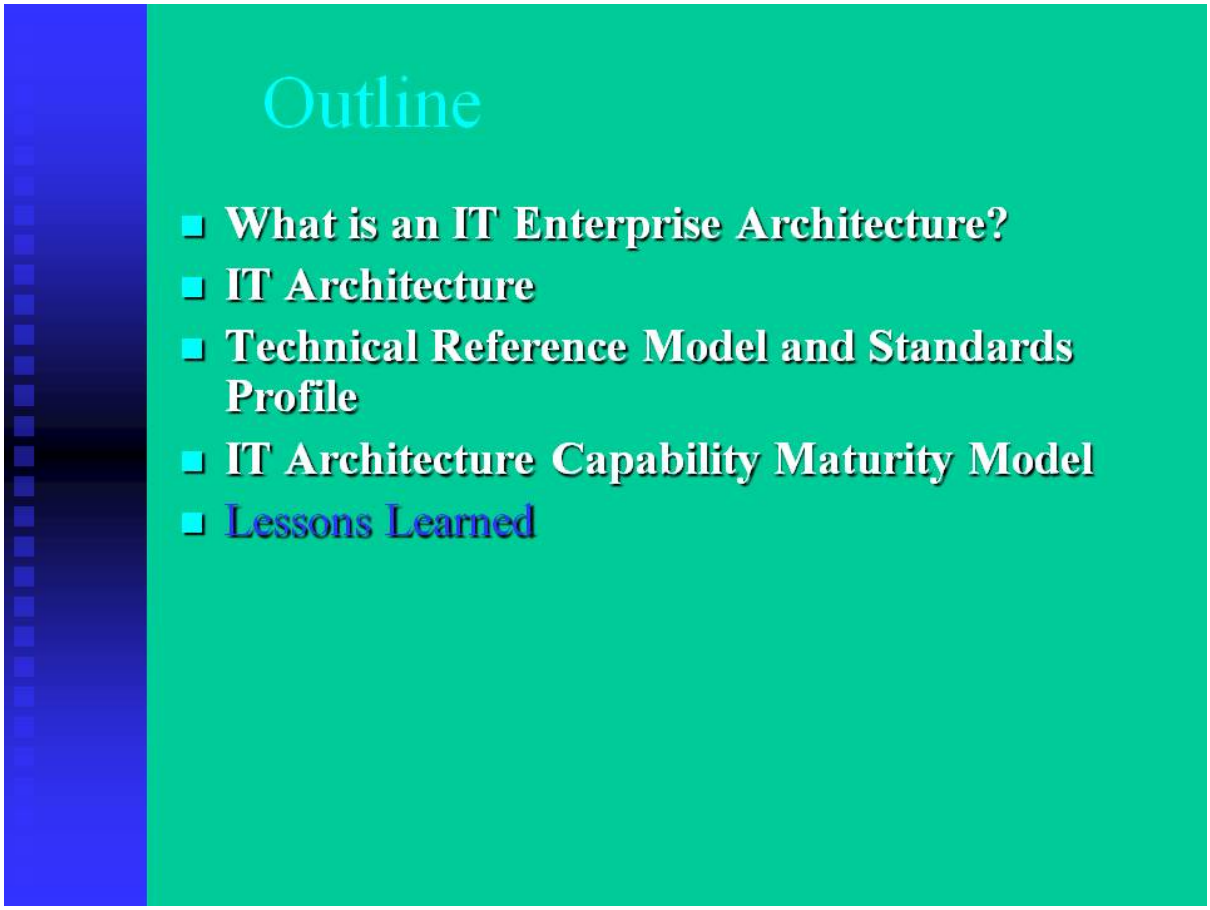
Characteristics at Different Maturity Levels

Characteris-tics	Level 0: No Architecture	Level 1: Initial	Level 2: Under Develop-ment	Level 3: Defined
Architecture Development	No architecture at all.	No architecture to speak of. Some standards, established by a variety of ad hoc means.	Architecture standards exist, but not necessarily linked to overarching conceptual architecture. Technical Reference Model (TRM) and Standards Profile framework established.	Architecture standards development linked to business drivers via conceptual architecture of principles and best practices. Partially completed TRM and Standards Profile.



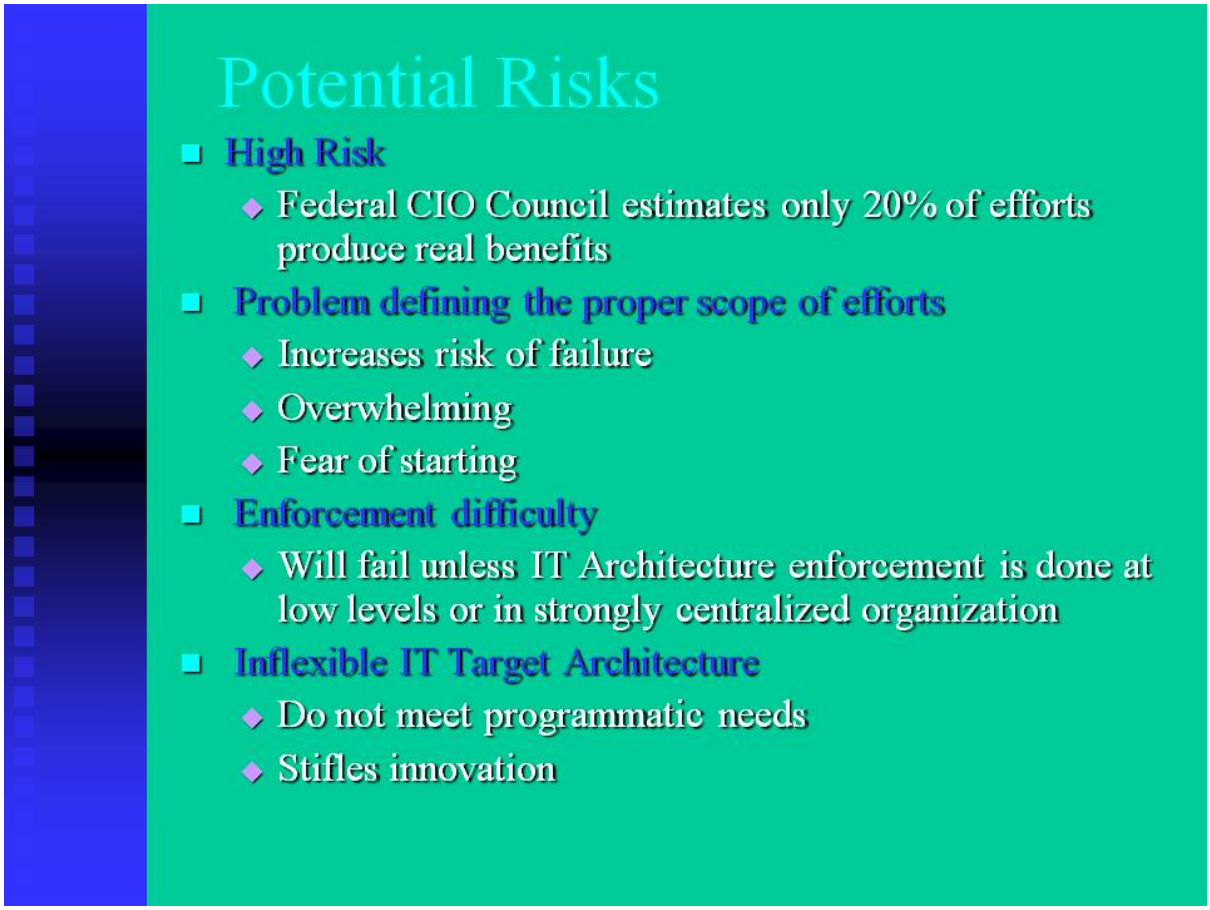
IT Architecture Maturity Model Checklist

Checklist Item	Current FY 2001	Next FY 2002
Business Linkage: To what extent is the business involved in the definition of an Enterprise Architecture (EA) process in the organization?		
Senior Management Involvement: To what extent are the senior managers of Operating Unit involved in the establishment and ongoing development of an EA process?		
Operating Unit Participation: To what extent is the definition of the EA process accepted by the Operating Unit?		
Architecture Development: To what extent is the architecture development in the organization driven by a well established process?		



Affinity Group Assumptions: Success Factors

- Business Linkage
- Senior-Management Involvement
- Operating Unit Participation
- Architecture Process Definition
- Architecture Development
- Architecture Communication
- Governance
- Program Management
- Holistic Enterprise Architecture
- IT Investment, and Procurement Strategy



Potential Benefits

- Integration - Make the flow of information and data easier
- Change - Ensure IT Infrastructure works and continues to work with rapidly increasing change
- Cost - Result in substantial cost reductions in IT procurements and support through standardization
- Convergence - Create more efficient business processes

Great flexibility and creativity are required to modify the process so that it works for your organization.

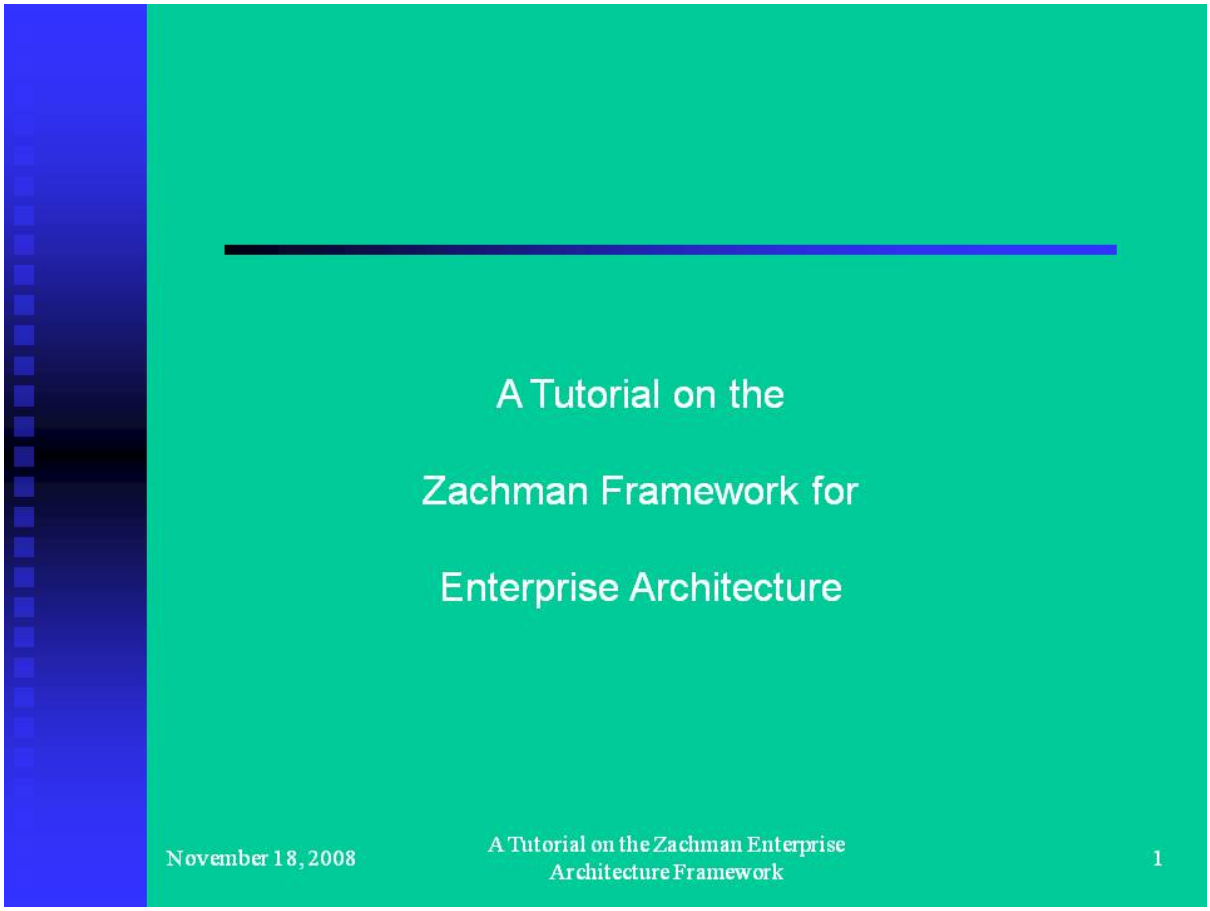
The process will be anything but smooth

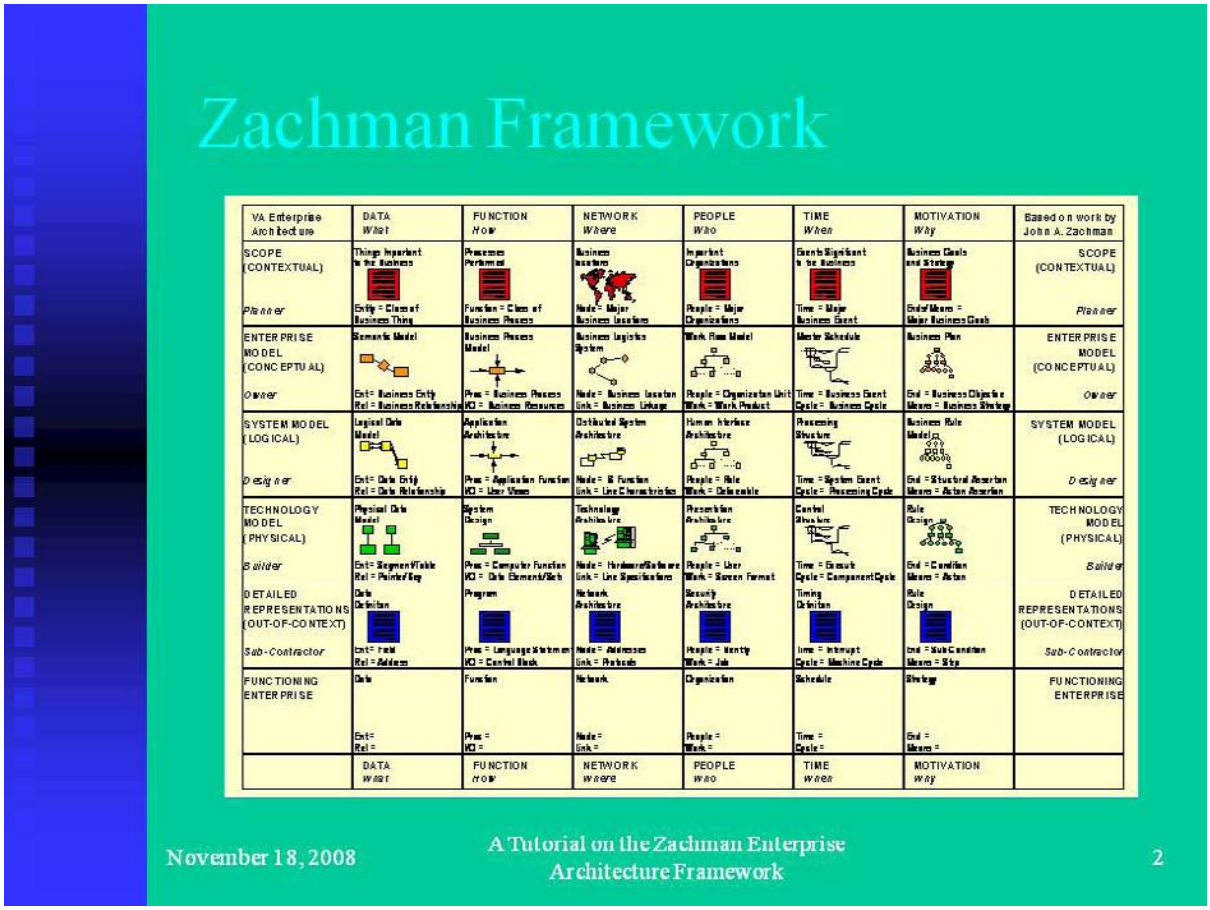
Requires continuous review and update

Important to involve as many IT and business staff in the process, as practicable. The more you involve the end user, the more they feel a part of and take an active role in the process


Lessons Learned (Continued)
Must begin with a clear definition and understanding of the Organization's vision, IT Principles and business context
Multi-organizational collaborative efforts can be very effective and successful
The IT Enterprise Architecture process is much more important than the IT Architecture Plan.
Technology and business drivers can rapidly change.
Nontechnical challenges are greater than the technical challenges.
JUST DO IT!

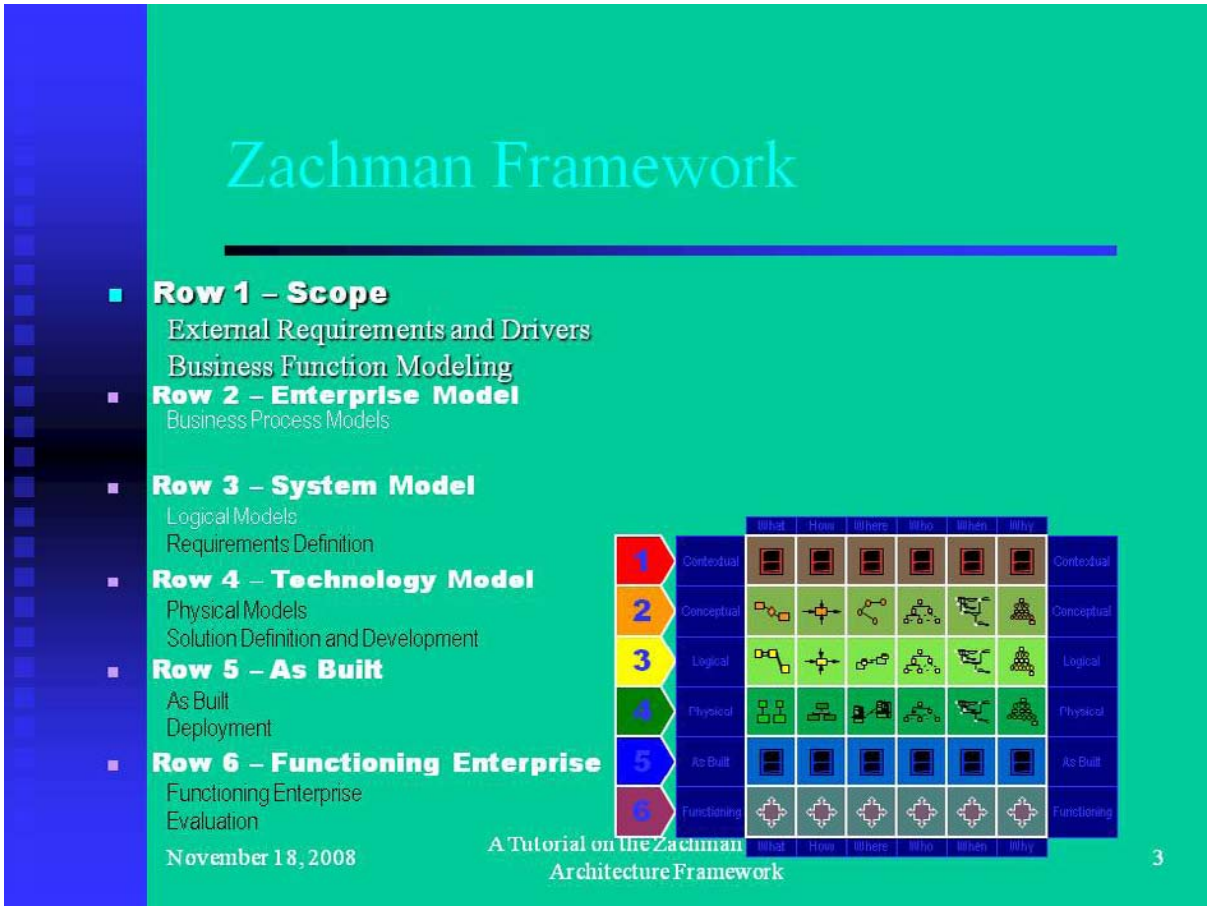
ZACHMAN FRAMEWORK



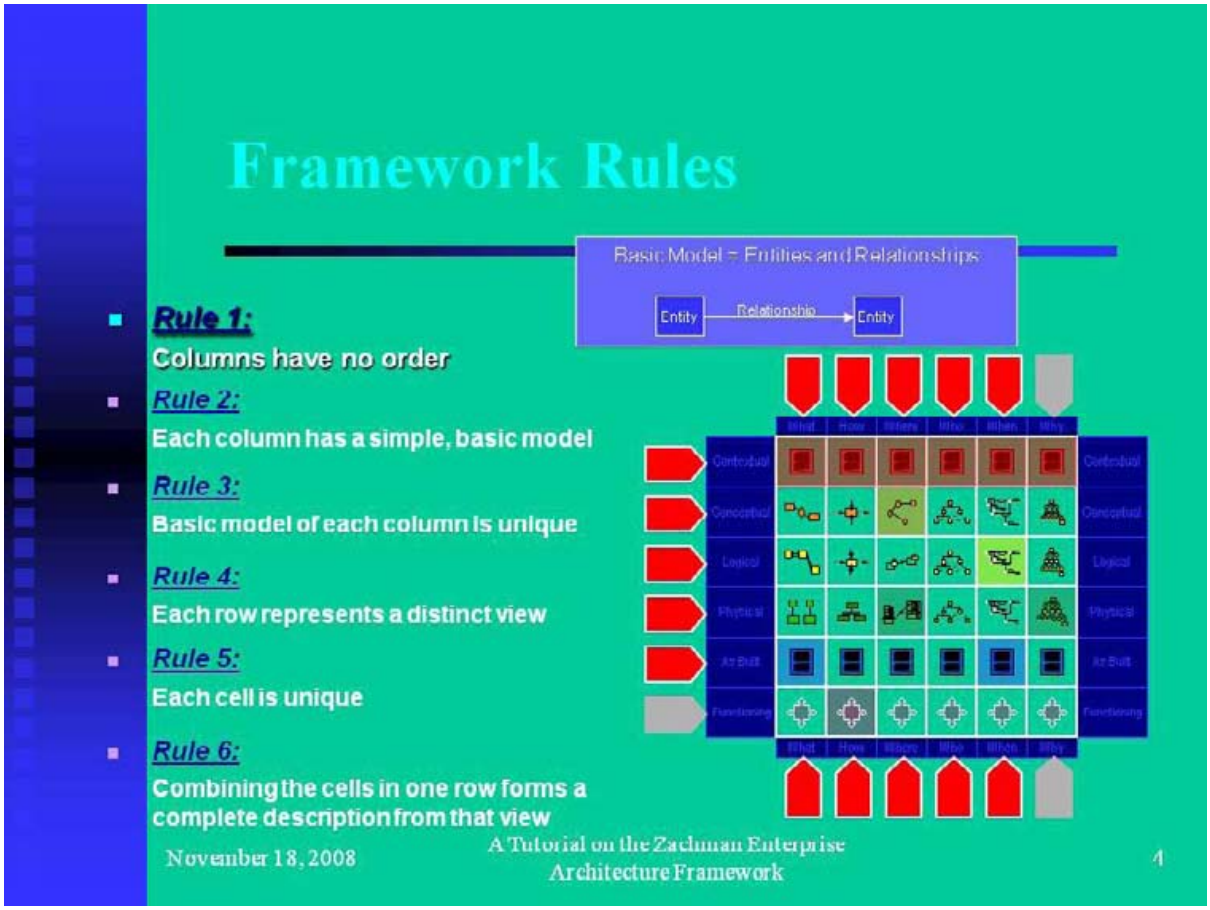


VA Enterprise Architecture	DATA *What*	FUNCTION *How*	NETWORK *Where*	PEOPLE *Who*	TIME *When*	MOTIVATION *Why*	Based on work by John A. Zachman
SCOPE (CONTEXTUAL) *Planner*	Things Important to the Business; Entity = Class of Business Thing	Processes Performed; Function = Class of Business Process	Business Locations; Node = Major Business Locations	Important Organizations; People = Major Organizations	Events Significant to the Business; Time = Major Business Event	Business Goals and Strategy; End/Means = Major Business Goals	SCOPE (CONTEXTUAL) *Planner*
ENTERPRISE MODEL (CONCEPTUAL) *Owner*	Semantic Model; Ent= Business Entity Rel = Business Relationship	Business Process Model; Proc = Business Process I/O = Business Resources	Business Logistics System; Node = Business Location Link = Business Linkage	Work Flow Model; People = Organization Unit Work = Work Product	Master Schedule; Time = Business Event Cycle = Business Cycle	Business Plan; End = Business Objective Means = Business Strategy	ENTERPRISE MODEL (CONCEPTUAL) *Owner*
SYSTEM MODEL (LOGICAL) *Designer*	Logical Data Model; Ent= Data Entity Rel = Data Relationship	Application Architecture; Proc = Application Function I/O = User Views	Distributed System Architecture; Node = IS Function Link = Line Characteristics	Human Interface Architecture; People = Role Work = Deliverable	Processing Structure; Time = System Event Cycle = Processing Cycle	Business Rule Model; End = Structural Assertion Means = Action Assertion	SYSTEM MODEL (LOGICAL) *Designer*
TECHNOLOGY MODEL (PHYSICAL) *Builder*	Physical Data Model; Ent= Segment/Table Rel = Pointer/Key	System Design; Proc = Computer Function I/O = Data Elements/Sets	Technology Architecture; Node = Hardware/Software Link = Line Specifications	Presentation Architecture; People = User Work = Screen Format	Control Structure; Time = Execute Cycle = Component Cycle	Rule Design; End = Condition Means = Action	TECHNOLOGY MODEL (PHYSICAL) *Builder*
DETAILED REPRESENTATIONS (OUT-OF-CONTEXT) *Sub-Contractor*	Data Definition; Ent= Field Rel = Address	Program; Proc = Language Statement I/O = Control Block	Network Architecture; Node = Addresses Link = Protocols	Security Architecture; People = Identity Work = Job	Timing Definition; Time = Interrupt Cycle = Machine Cycle	Rule Design; End = Sub-Condition Means = Step	DETAILED REPRESENTATIONS (OUT-OF-CONTEXT) *Sub-Contractor*
FUNCTIONING ENTERPRISE	Data; Ent= Rel =	Function; Proc = I/O =	Network; Node = Link =	Organization; People = Work =	Schedule; Time = Cycle =	Strategy; End = Means =	FUNCTIONING ENTERPRISE
	DATA *What*	FUNCTION *How*	NETWORK *Where*	PEOPLE *Who*	TIME *When*	MOTIVATION *Why*	

Please see Zachman Framework Document on page 87 within this workbook.



Please see Zachman for Enterprise Architecture on page 93 within this workbook.



Framework Rules

Zachman provided the following rules to assist the reader in understanding ISA and its application.

Rule 1: The columns have no order.

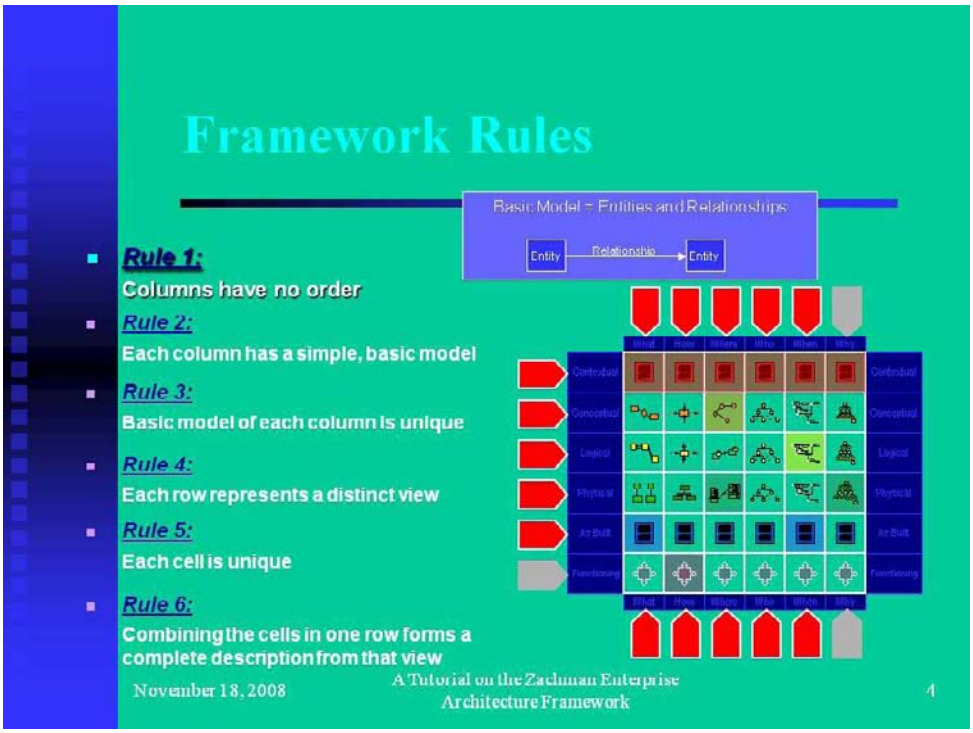
Order implies priorities and creates a bias toward one aspect at the expense of others. All columns are equally important because all are abstractions of the same enterprise.

Rule 2: Each column has a simple, basic model.

Each column represents an abstraction from the real world for convenience of description. These models include:

Data (what), Function (how), Network (where), People (who), Time (when), and Motivation (why)

Continued…



Rule 3: The basic model of each column must be unique.

The individual models may be related to one another because they are all abstractions of the same real-world enterprise, but each model represents a separate and unique concept.

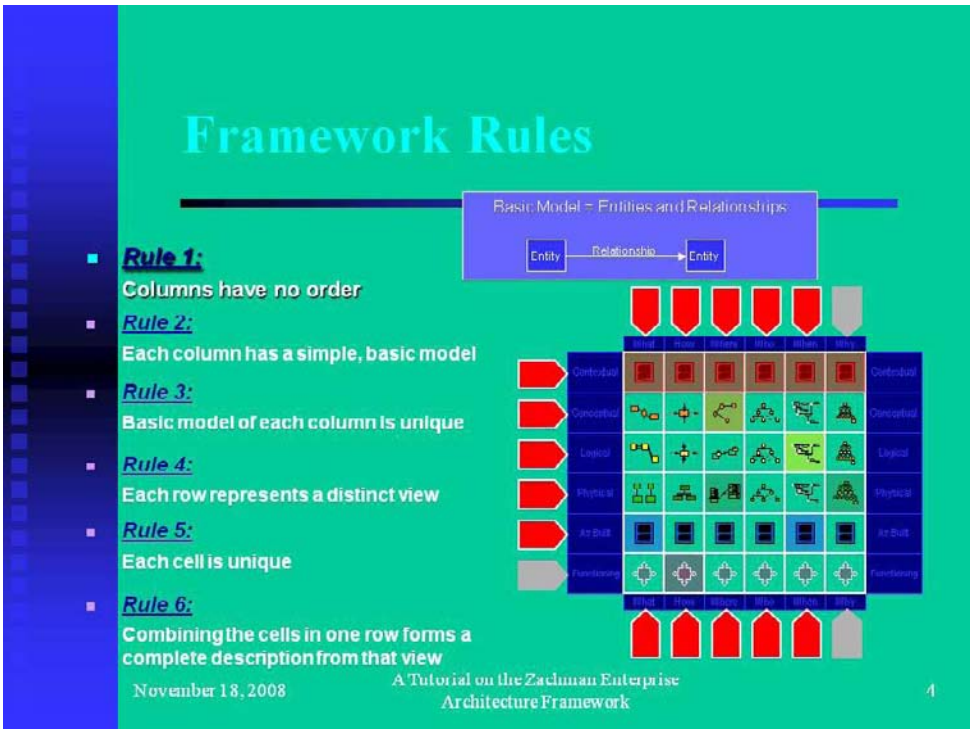
Rule 4: Each row represents a distinct perspective

This rule is most easily demonstrated by the Business Model, System Model, and Technology Model rows, which represent the owner's, architect's, and builder's perspectives. Each perspective is different because it deals with a different set of constraints. For example, the owner deals with usability constraints, both aesthetic and functional, and the architect deals with design constraints, the laws of physics or nature, and the builder deals with construction constraints, the state of the art in methods and technologies.

Rule 5: Each cell is unique.

Since each column has a unique basic model that makes each column unique and each row has a different perspective, each cell in the framework is unique. Zachman likens ISA to a "periodic table" for information entities, providing a classification

Continued…



scheme for information entities, allowing different entities to be combined to provide different views of an information system.

Because each cell is unique, different techniques and different graphic representations are appropriate for different cells. This also accounts for the large number of information systems design models and methodologies that have emerged over the years.

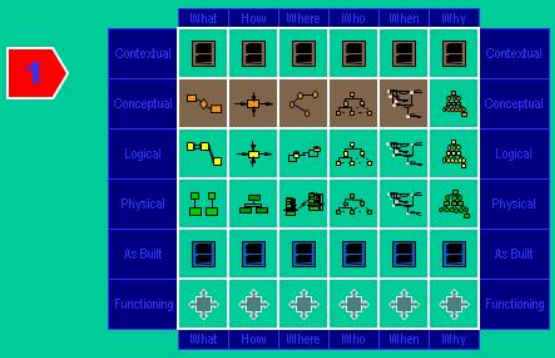
Rule 6: Combining the cells in one row forms a complete model.

The sum of all cells in a given row is the most complete depiction of reality from the perspective of that row. As new cells in a given row are defined each new cell description must be consistent with the perspective of that row. Each cell in a given row can be defined and is independent of any other cells in the row, yet each cell is but one abstraction of the same perspective of reality. Therefore, each cell is related to every other cell in the same row

Zachman Framework – Row 1 Scope/Planner's View

- **Motivation/Why**
  Business goals, objectives and performance measures related to each function
- **Function/How**
  High-level business functions
- **Data/What**
  High-level data classes related to each function
- **People/Who**
  Stakeholders related to each function
- **Network/Where**
  VA locations related to each function
- **Time/When**
  Cycles and events related to each function

- **External Requirements and Drivers**
- **Business Function Modeling**



November 18, 2008

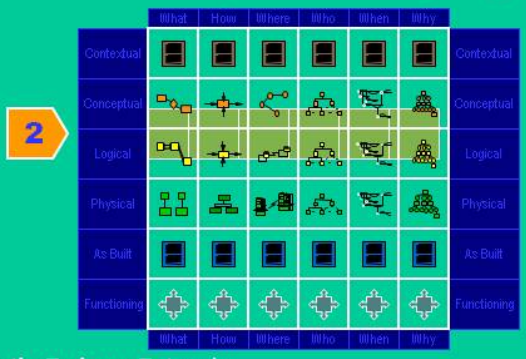
A Tutorial on the Zachman Enterprise Architecture Framework

5

Zachman Framework – Row 2
Enterprise Model/Designer's View

- **Motivation/Why**
  Policies, procedures and standards for each process
- **Function/How**
  Business processes
- **Data/What**
  Business data
- **People/Who**
  VA roles and responsibilities in each process
- **Network/Where**
  VA locations related to each process
- **Time/When**
  Events for each process and sequencing of integration and process improvements

- **Business Process Models**
- **Business Function Allocation**
- **Elimination of Function Overlap and Ambiguity**



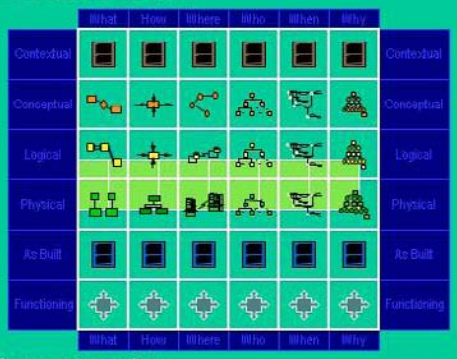
November 18, 2008 A Tutorial on the Zachman Enterprise Architecture Framework 6

Zachman Framework – Row 3
System Model/Designer's View

- **Motivation/Why**
  VA policies, standards and procedures associated with a business rule model
- **Function/How**
  Logical representation of information systems and their relationships
- **Data/What**
  Logical data models of data and data relationships underlying VA information
- **People/Who**
  Logical representation of access privileges constrained by roles and responsibilities
- **Network/Where**
  Logical representation of the distributed system architecture for VA locations
- **Time/When**
  Logical events and their triggered responses constrained by business events and their responses

- **Logical Models**
- **Project Management**
- **Requirements Definition**

3



November 18, 2008

A Tutorial on the Zachman Enterprise Architecture Framework

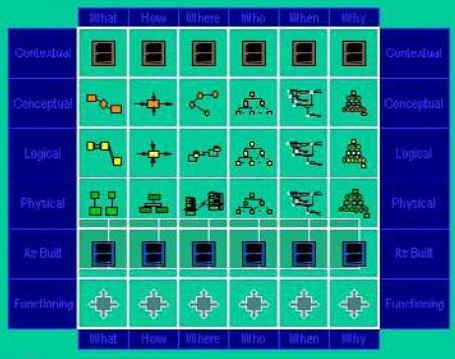
7

Zachman Framework – Row 4 Technology Model/Builder's View

- **Motivation/Why**
  VA business rules constrained by information systems standards
- **Function/How**
  Specifications of applications that operate on particular technology platforms
- **Data/What**
  Database management system (DBMS) type requirements constrained by logical data models
- **People/Who**
  Specification of access privileges to specific platforms and technologies
- **Network/Where**
  Specification of network devices and their relationships within physical boundaries
- **Time/When**
  Specification of triggers to respond to system events on specific platforms and technologies

- **Physical Models**
- **Technology Management**
- **Solution Definition and Development**

4



November 18, 2008 A Tutorial on the Zachman Enterprise Architecture Framework 8

Zachman Framework – Row 5
As Built/Integrator's View

- **Motivation/Why**
  VA business rules constrained by specific technology standards
- **Function/How**
  Programs coded to operate on specific technology platforms
- **Data/What**
  Data definitions constrained by physical data models
- **People/Who**
  Access privileges coded to control access to specific platforms and technologies
- **Network/Where**
  Network devices configured to conform to node specifications
- **Time/When**
  Timing definitions coded to sequence activities on specific platforms and technologies

- **As Built**
- **Configuration Management**
- **Deployment**

5

November 18, 2008

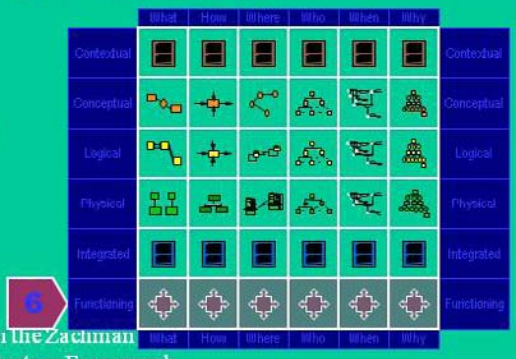
A Tutorial on the Zachman Enterprise Architecture Framework

9

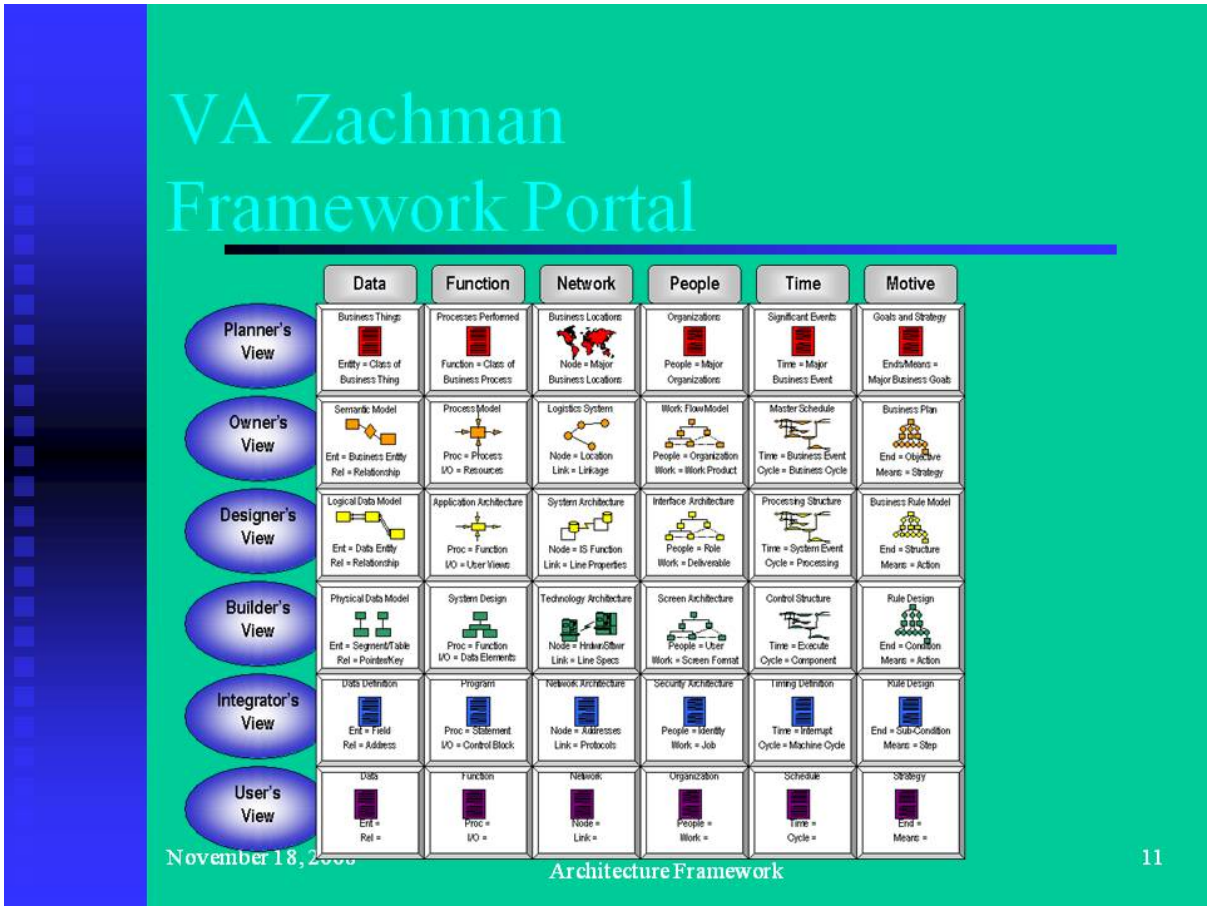
Zachman Framework – Row 6
Functioning Enterprise/User's View

- **Motivation/Why**
  Operating characteristics of specific technologies constrained by standards
- **Function/How**
  Functioning computer instructions
- **Data/What**
  Data values stored in actual databases
- **People/Who**
  VA personnel and key stakeholders working within their roles and responsibilities
- **Network/Where**
  Sending and receiving messages
- **Time/When**
  Timing definitions operating to sequence activities

- **Functioning Enterprise**
- **Operations Management**
- **Evaluation**



November 18, 2008
A Tutorial on the Zachman Architecture Framework
10

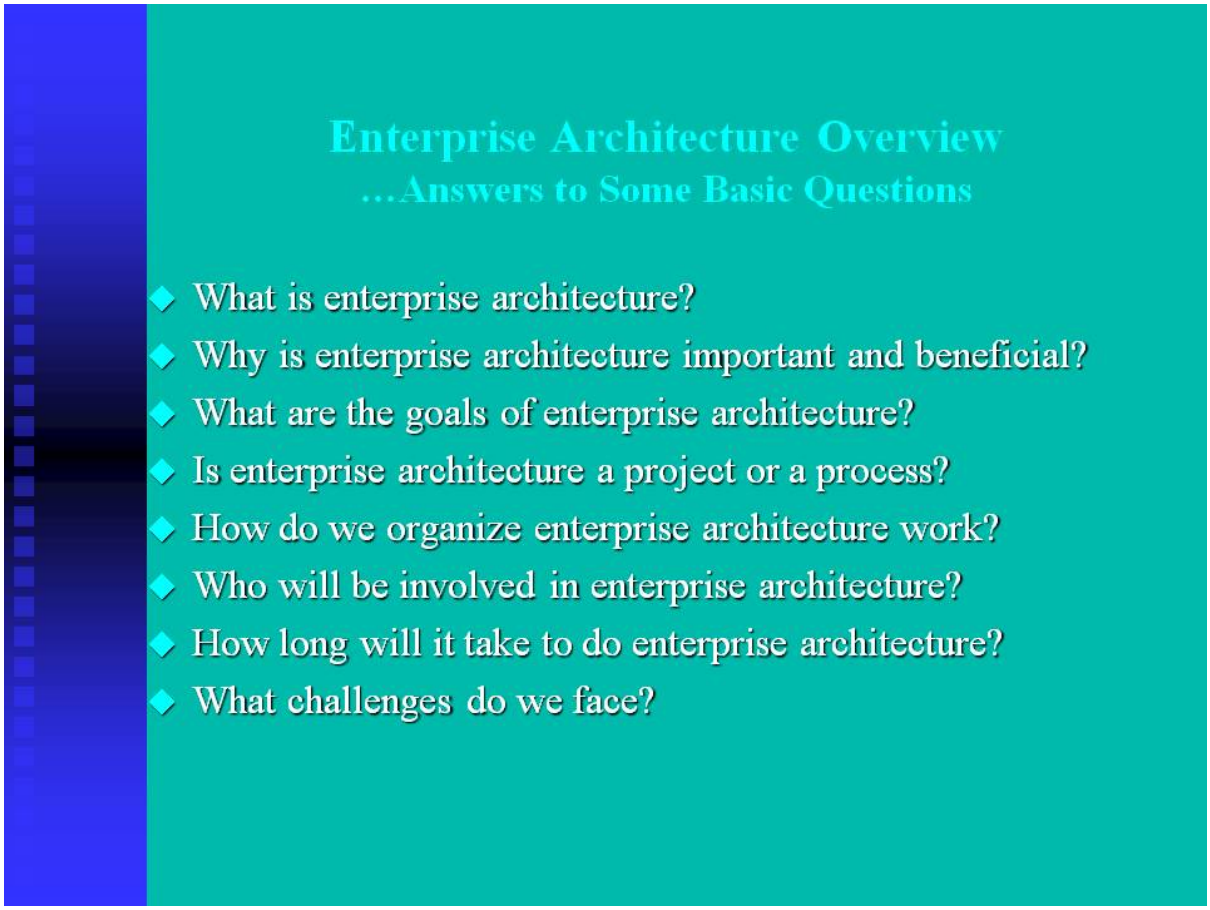

VA Zachman
Framework Portal
Data
Function
Network
People
Time
Motive
Planner's View
Owner's View
Designer's View
Builder's View
Integrator's View
User's View
Business Things
Entity = Class of Business Thing
Processes Performed
Function = Class of Business Process
Business Locations
Node = Major Business Locations
Organizations
People = Major Organizations
Significant Events
Time = Major Business Event
Goals and Strategy
Ends/Means = Major Business Goals
Semantic Model
Ent = Business Entity
Rel = Relationship
Process Model
Proc = Process
I/O = Resources
Logistics System
Node = Location
Link = Linkage
Work Flow Model
People = Organization
Work = Work Product
Master Schedule
Time = Business Event
Cycle = Business Cycle
Business Plan
End = Objective
Means = Strategy
Logical Data Model
Ent = Data Entity
Rel = Relationship
Application Architecture
Proc = Function
I/O = User Views
System Architecture
Node = IS Function
Link = Line Properties
Interface Architecture
People = Role
Work = Deliverable
Processing Structure
Time = System Event
Cycle = Processing
Business Rule Model
End = Structure
Means = Action
Physical Data Model
Ent = Segment/Table
Rel = Pointer/Key
System Design
Proc = Function
I/O = Data Elements
Technology Architecture
Node = Hdwe/Sftwr
Link = Line Specs
Screen Architecture
People = User
Work = Screen Format
Control Structure
Time = Execute
Cycle = Component
Rule Design
End = Condition
Means = Action
Data Definition
Ent = Field
Rel = Address
Program
Proc = Statement
I/O = Control Block
Network Architecture
Node = Addresses
Link = Protocols
Security Architecture
People = Identity
Work = Job
Timing Definition
Time = Interrupt
Cycle = Machine Cycle
Rule Design
End = Sub-Condition
Means = Step
Data
Ent =
Rel =
Function
Proc =
I/O =
Network
Node =
Link =
Organization
People =
Work =
Schedule
Time =
Cycle =
Strategy
End =
Means =
November 18, 2003
Architecture Framework
11

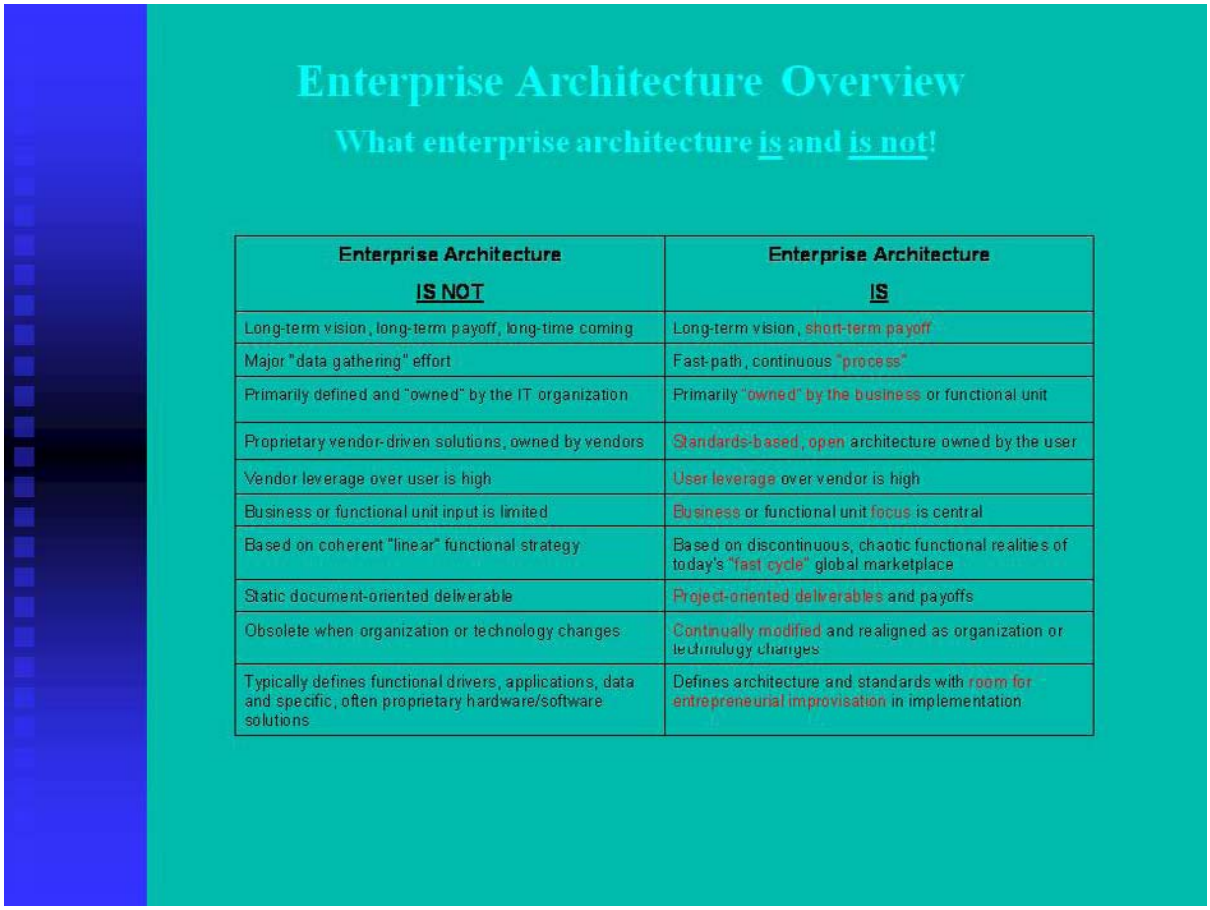
EXECUTIVE OVERVIEW

How can we improve the IT planning process?

Your priorities:

- Stakeholder-defined technology architecture guidelines/standards
- Integrated, flexible planning/project management process *(addressing funding issues and needs of different types and sizes of agencies)*
- Streamline/facilitate IT procurements
- IT intelligence sharing/information exchange


Enterprise Architecture Overview
…Answers to Some Basic Questions
What is enterprise architecture?
Why is enterprise architecture important and beneficial?
What are the goals of enterprise architecture?
Is enterprise architecture a project or a process?
How do we organize enterprise architecture work?
Who will be involved in enterprise architecture?
How long will it take to do enterprise architecture?
What challenges do we face?



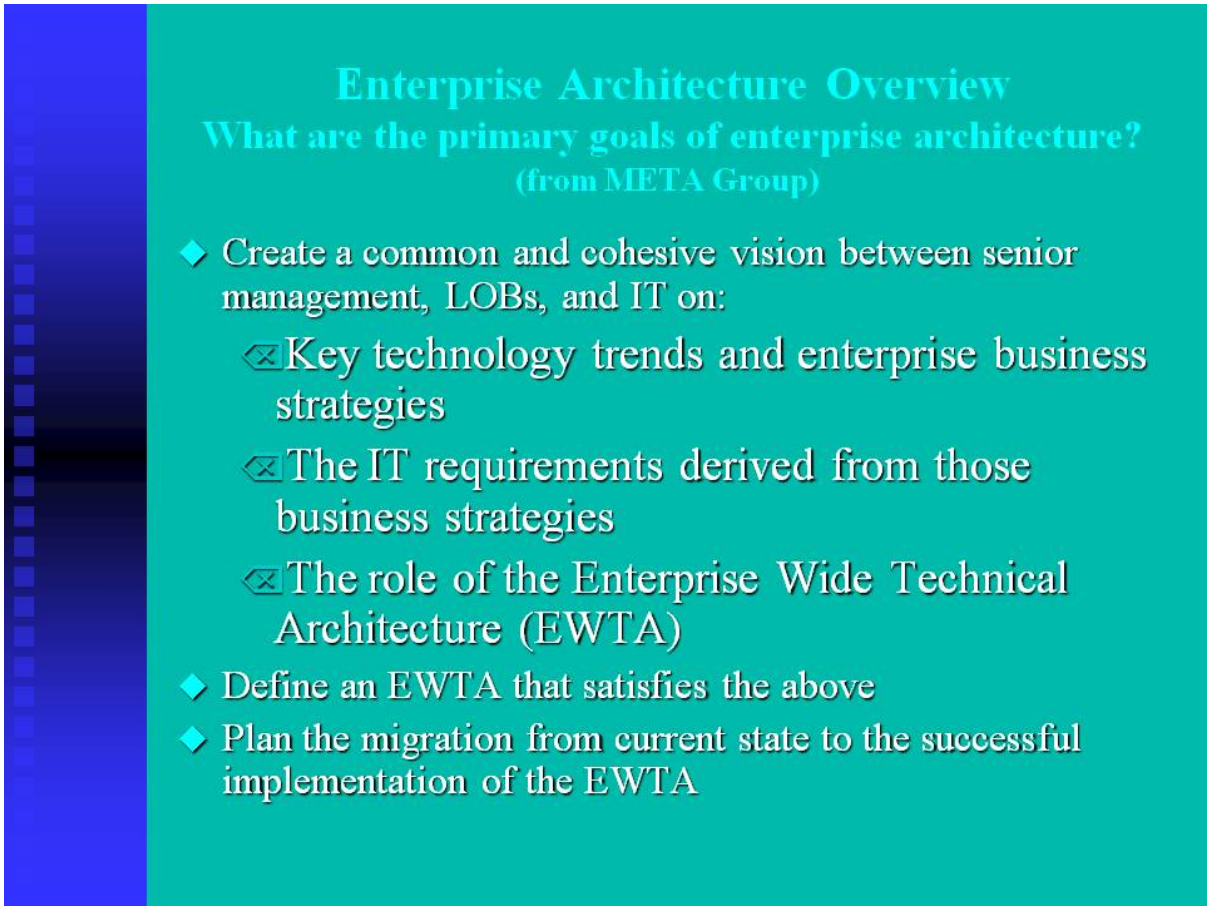
Enterprise Architecture Overview - What is enterprise architecture ?

Three views:

- Enterprise architecture is *the bridge between business and technology…* it is about expressing technology in business terms and business in technology terms.
- Enterprise architecture is about *business and technology alignment…* the better you understand your business, the more effective an enterprise architecture can be!
- Enterprise architecture is the fine art of telling the business what you are going to use to achieve what they say they want to do (even before they say it).

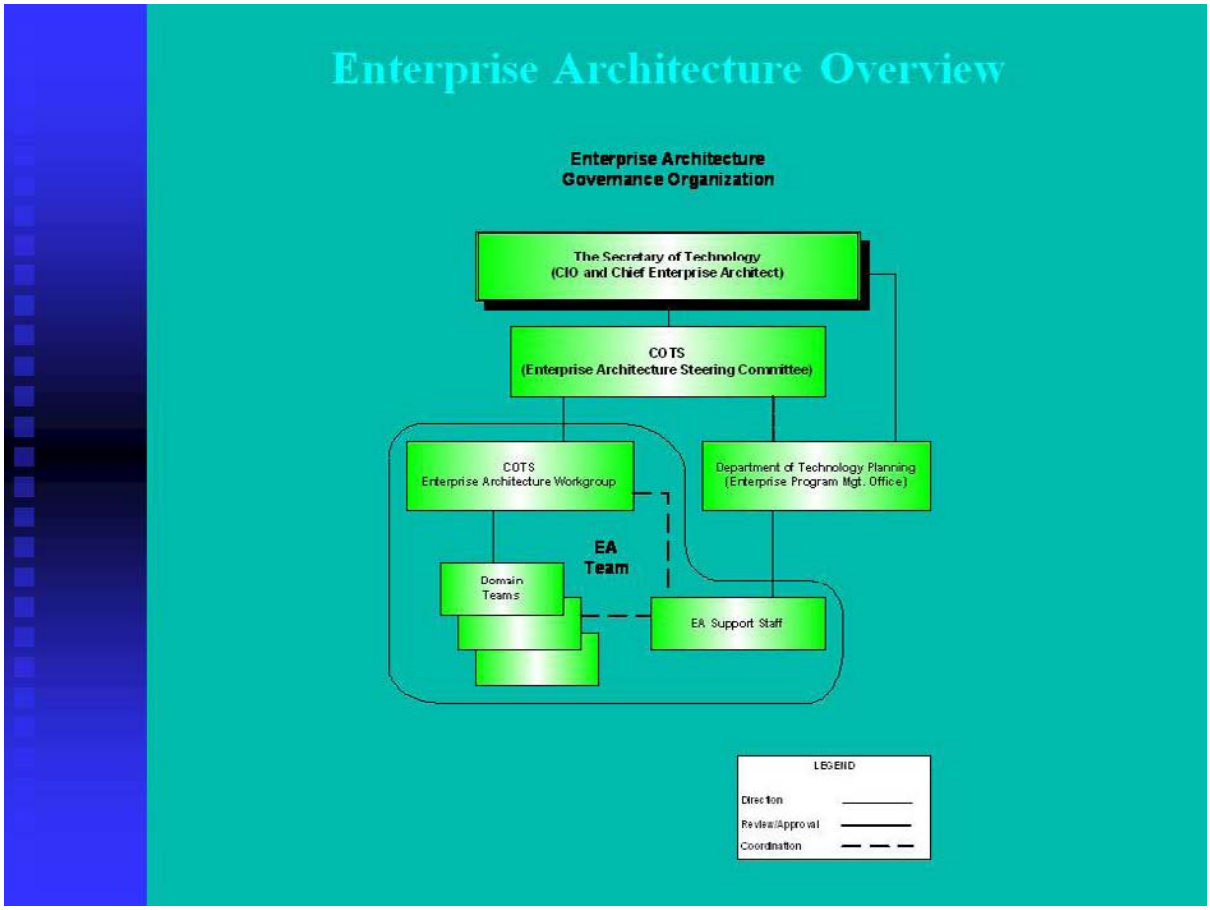
Business Drives Architecture!


Enterprise Architecture Overview
Why is enterprise architecture important and beneficial?
It aligns IT support with business objectives.
It provides a unified enterprise-wide IT vision.
It creates business-focused processes for IT development.
It implements best practices and principles that empower IT to rapidly implement high value solutions in response to emerging business needs.
It employs a consistent framework that better supports future technology decisions.
It provides a mechanism to make more effective IT investments at lower TCO.
It applies enterprise-wide IT standards and processes.
The Bottom Line... IT Better, Faster, Cheaper!



Is enterprise architecture a project or a process?

Enterprise architecture is both...



Who will be involved in enterprise architecture?

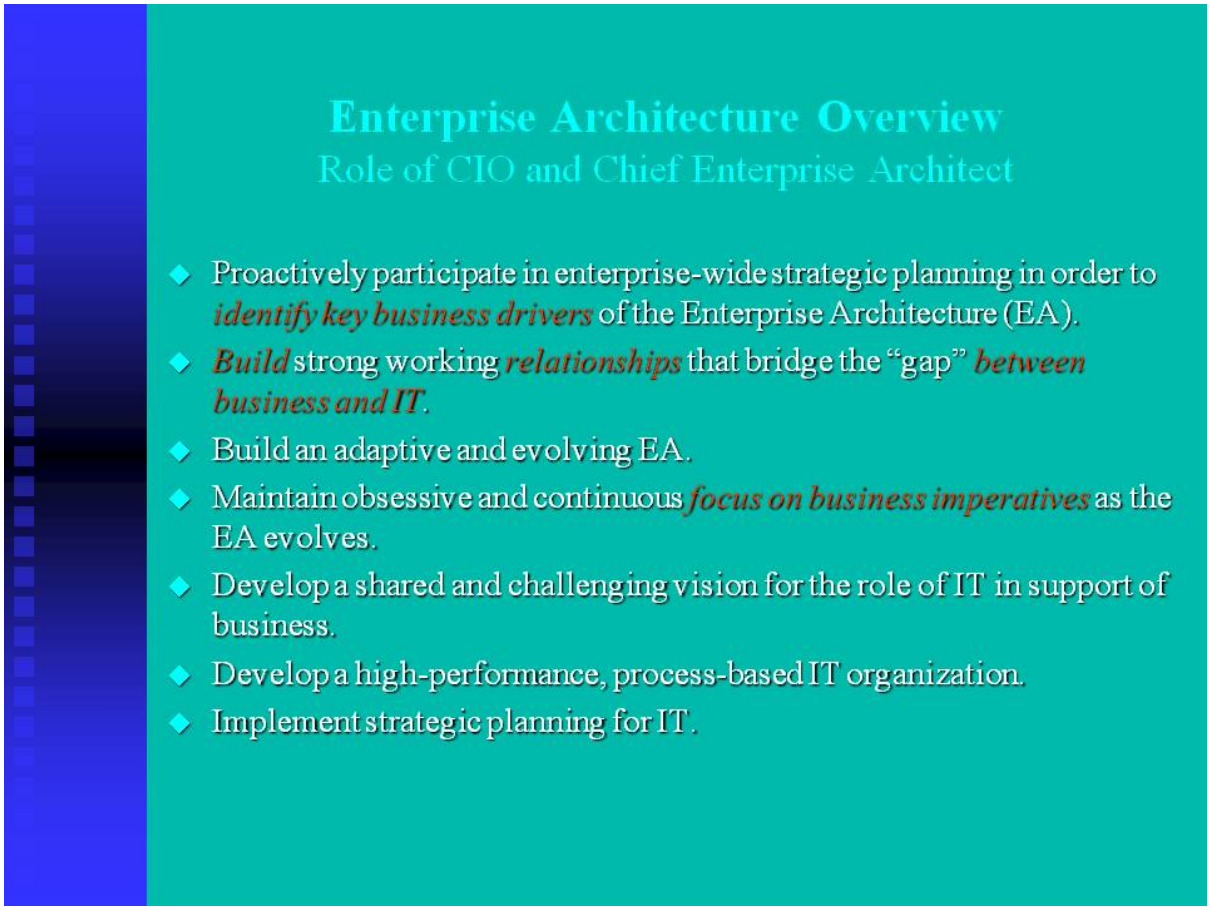
Ultimately?

...Everyone!

Initially?

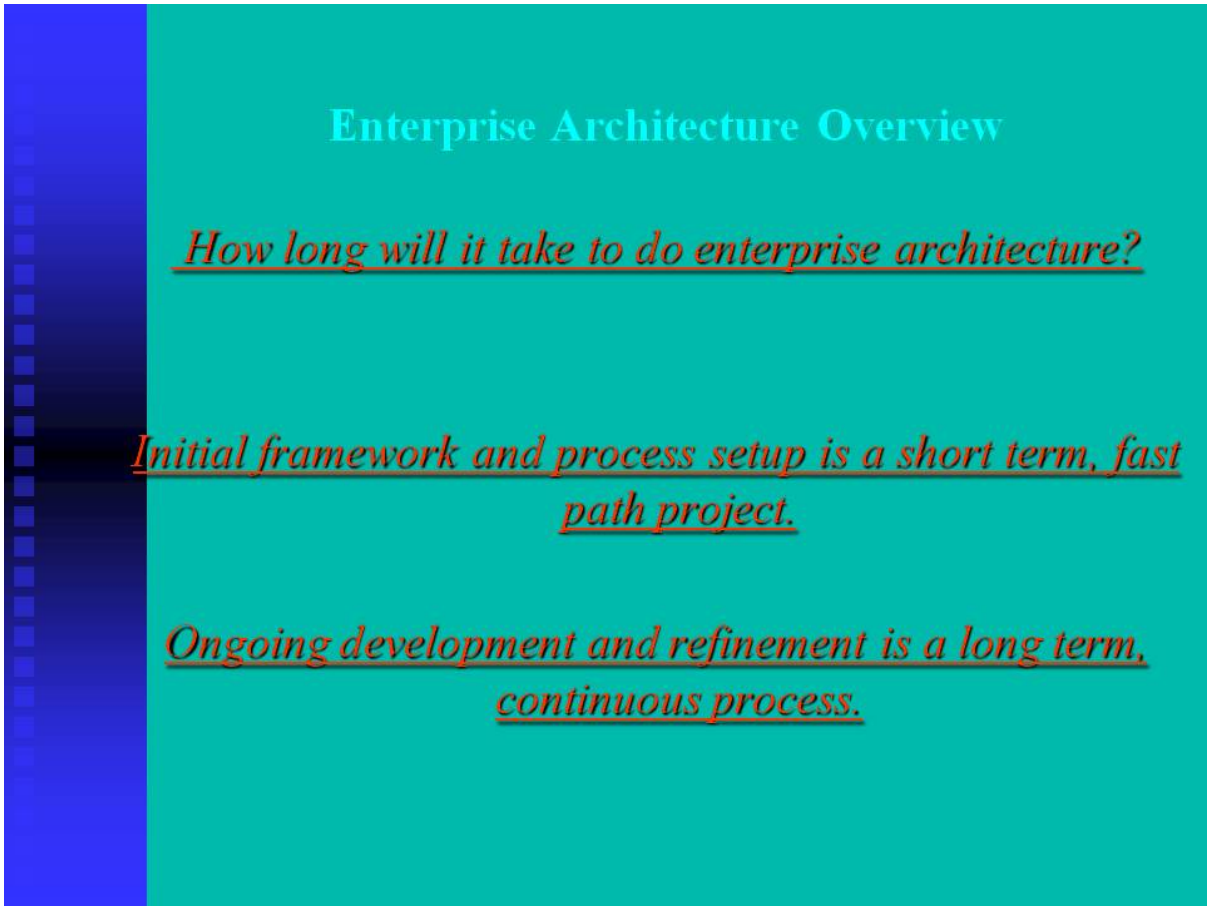
…business and technology experts

…assigned specific project (then process) roles:



General Qualifications of EA Team Members

- Collaborative perspective.
- Interpersonal communication and relationship building skills.
- Action oriented and results driven.
- Big picture sensibility, understands the business of the enterprise.
- Knowledge across multiple domains.
- Extensive experience in at least one domain.
- Sense of balance.
- Good communicator.
- Creative mindset.



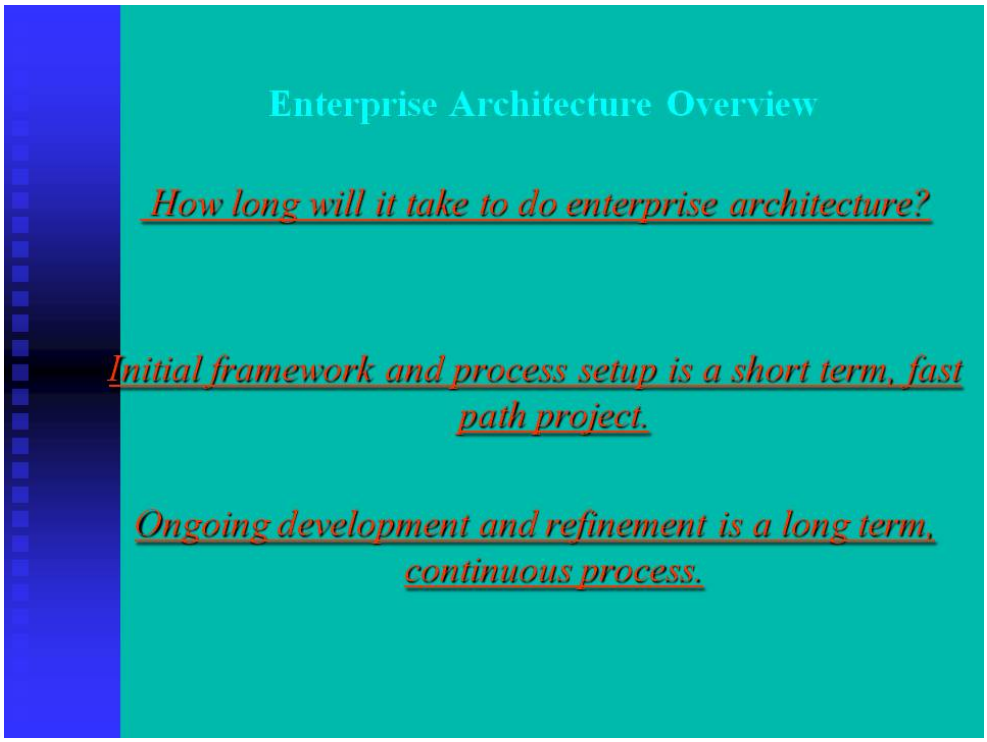
First Year Work Plan

Complete Project Initiation Activities :

- Research and define EA strategy. ✔
- Establish EA process model. ✔
- Outline a proposed first year work plan. ✔
- Build EA Web site. ✔
- Establish the full governance organization (Ongoing).

Complete Phase 1 (Common Requirements Vision):

- Define technology trends. ✔
- Identify enterprise business strategies (EBS). ✔
- Translate EBS into business information requirements (BIR) and requirements for technical architecture (RTA).
- Describe preliminary application portfolio implications (API).

Continued… 

Begin Phase 2:

- Document current environment and technology baseline, the “as is” architecture.
- (Note: This effort can begin concurrent with Phase 1 activities.)
- Define a conceptual architecture, based on principles and best practices.
- Begin to define domain architectures.
- Begin gap analysis.

Bottom Line

- EA is not only a collection of technology projects, but rather a process that seeks to achieve cohesive, enterprise-wide results.
- EA “brings it all together.”
- EA positions IT to deliver

SUPPORTING DOCUMENTS

Through the documents, look for text surrounded by << and >> these are indicators for you to create some specific text.

Watch also for highlighted text which provides further guidance and instructions.

THE OVERVIEW: ENTERPRISE ARCHITECTURE

Recently there has been a surprising increase of interest in developing and using Enterprise Architectures as the "blueprint" for all enterprise engineering activities including data warehousing, software engineering, information engineering, commercial software evaluation, enterprise application integration, business to business and business to customer e-commerce, and the list is endless. The remarkable truth is that all of these activities are facilitated, and in some instances even made possible, by an enterprise architecture.

This is not a new concept. In 1987, John Zachman formally proposed the use of such architectures for software engineering (J. Zachman, "A Framework for Information Systems Architecture", IBM Systems Journal, 1987, 26(3), pp 276-292). Since then, Zachman has extended the concept to include more than software engineering and countless organizations have successfully established and used the enterprise architecture that Zachman envisioned.

What is new is the realization by many organizations that an enterprise architecture, properly engineered, could possibly be the single most effective tool they can use to ensure that the right information gets to the right people, in the right format, and at the right time.

An enterprise architecture links an enterprise's strategic plan and performance plans ("business architecture") with its enterprise data architecture, enterprise application architecture and enterprise technical architecture.
A well-documented architecture is a logical organization of information pertaining to the following multi-level, multi-dimensional, enterprise-wide elements:

- ***Strategic goals, objectives, and strategies***
- ***Business rules and measures***
- ***Information requirements***
- ***Application systems***
- ***Relationships between applications and data elements***
- ***Technology infrastructure***

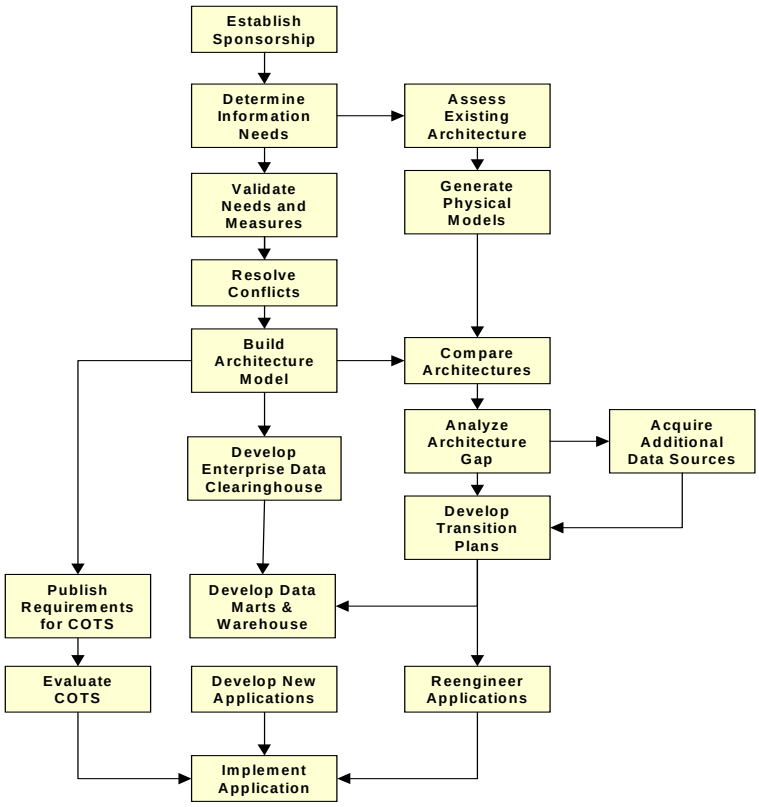
An enterprise architecture also contains guidelines, standards, policies and business rules that define the enterprise's software engineering environment.
Every enterprise has an architecture. However, it is usually undocumented and the elements are inconsistent. Most likely some of the architecture elements are embodied in strategic and performance plans, published and unpublished policies and procedures, and system documentation. Unfortunately, there are also a lot of enterprise architecture elements that are embedded in software application code, and even more that only exist as employee "tribal knowledge."

An enterprise that has fully documented its architecture can use it to accomplish the following:

- ***Facilitate change management by linking strategic requirements to systems that support them and by linking the business model to application designs***
- ***Enable strategic information to be consistently and accurately derived from operational data***
- ***Promote data sharing, thus reducing data redundancy and reducing maintenance costs***
- ***Improve productivity through component development, management and reuse***
- ***Reduce software development cycle time***
- ***Evaluate commercial products and services***
- ***Share information with customers and business partners***

In order to establish an enterprise architecture that can be used to gain these benefits, it must be "engineered." This requires rigorous and repeatable approach that is strategically-driven, information-centric, model-based, and technology independent. It also requires tools that fully support the approach.
The remainder of this paper describes an effective, productive, common sense approach to engineering an enterprise architecture. This approach can also be extended to include all the activities necessary for effective enterprise strategic information management. Enterprise Architecture Engineering -- At a Glance The following diagram illustrates the Enterprise Architecture Engineering approach described in this paper.

Enterprise (Architecture) Engineering



This approach to enterprise architecture engineering results in a useable enterprise management tool that exactly meets the needs of an enterprise, business or government, large or small. Designing and developing an architecture using this approach involves seven very different activities:

1. Model the new architecture
2. Reverse engineer the existing architecture(s)
3. Compare architectures
4. Analyze the gap
5. Transition to new architecture
6. Acquire, develop, and reengineer systems, including data warehouses and data marts based upon new architecture
7. Define new architecture

Identify Enterprise Needs

Identifying enterprise needs is the first, crucial task when engineering an enterprise architecture. When developing operational information systems, there is often one single sponsor or one group of users with a clear view of what they need, what the system should look like, and how it should function. Unlike single-dimensional application development, when developing an enterprise architecture, there are normally multiple potential sponsors, each with a different idea of what the enterprise architecture is and what it should provide, and all requesting or demanding action. Because of this lack of a single focused direction, identifying precise enterprise needs is critical to the success of an enterprise architecture project. Different views of the architecture should be established to represent the varying perspectives of the architecture stakeholders and enterprise information owners.

Enterprise needs (the business architecture) should be expressed in terms of business functions, business rules, measures of effectiveness, and critical success factors. An enterprise's business plans typically provide the basis for defining preliminary enterprise needs. In addition, it may be necessary to interview key enterprise managers and analyze other pertinent documentation in order to ascertain the enterprise's strategic information needs.

The best technique for defining and refining preliminary enterprise information needs is to conduct a series of facilitated focus group sessions. (If no business or performance plans exist, similar sessions first should be used to create the plans.) The participants in any of these sessions should be decision-makers for whom the information is important.

Validate Needs and Measures

After identifying and defining enterprise needs, it is advantageous to communicate them throughout the enterprise. One of the best justifications for undertaking an architecture project is the synergy achieved through the process of defining and then communicating its critical success factors and measures. Everyone becomes aware of precisely what defines success and how it is measured. In addition, the measures undergo a "reality check" by people who were not involved in their development, but who may be measured by them and who will be involved in creating the raw data

from which the measures will be derived. Their feedback is used for refining the measures.

Completely validating enterprise measures includes describing the cycles or time periods used. Are quarters, months, or hours appropriate for capturing useful measurement data? How much historical data will be needed? These vary greatly by enterprise. The United States Federal Reserve Bank views enterprise measures in monthly, quarterly and annual increments and uses years of historical data to determine trends in the economy. An insurance company requires decades of actuarial data for meaningful measures. A telephone sales operation, on the other hand, uses hourly enterprise measures and may keep only a few weeks of historical information.

Resolve Data Conflicts

A well-defined enterprise architecture model cannot contain homonyms, synonyms, or other data definition conflicts. It must be semantically clear, correct and consistent. Data definition conflicts often exist because enterprises have one or more major terms that are used by everyone in the enterprise, but mean different things in different organizational units. One of the most common examples is "customer."

> To the Accounting Department, "customer" could mean the organization (or individual) that receives a bill. "Customer" could also mean an individual receiving service or buying a product. To the Sales Department, "customer" could mean the organizations on which the salesperson calls. Additionally, each department may use different names to describe the same data element (Customer vs. Client vs. Prospect vs…).

Providing any single interpretation as the only enterprise definition of “customer” would not meet the needs of the enterprise and would predispose its information resource management efforts to failure. The conflicts must be resolved so that the enterprise architecture not only provides the corporate definition of customer, it also reflects the various definitions understood and used by every enterprise element. It must also include links between every enterprise architecture data element and every application data element that implements it (more about this later).

Great pains should be taken to resolve all data conflicts in the enterprise architecture before attempting to implement applications that support the new architecture.

Build the Enterprise Architecture Model

Creating an enterprise architecture is a lot like creating an architecture for a building. Both involve a disciplined development cycle, use rigorous techniques, and require the right tools.

A building is constructed using architectural diagrams (conceptual models and blueprints) that clearly depict the building's infrastructure (structural elements, walls, electrical wiring, plumbing, etc.). Enterprise architectures include architectural models of enterprise infrastructure (policies, goals, measures, critical success factors, data elements, etc.).

Blueprints are also used to enlarge a building or make any significant modifications. Without a diagram of the infrastructure, such changes are quite difficult and very costly -- and can even be dangerous. It is the same with enterprise architectures. Managing change means first updating the enterprise's architecture model so that it reflects changes (new product lines or services, new policies, or new laws for example) and then modifying the information systems to support the changed enterprise.

Enterprise software engineering is easier and less costly when based upon an accurate architectural model of the enterprise. Further, strategic information based upon the architecture is easier to use and consistently produces desired outcomes when decision-makers have access to an enterprise architecture that accurately reflects enterprise infrastructure.

It is imperative that an enterprise architecture reflects both strategic information requirements and day-to-day operational data requirements. The architecture must be closely linked to the enterprise strategic plan and corporate performance measures. The framework for the architecture is described in the illustration below (for more information about the Zachman Framework, visit the Zachman Institute for Framework Advancement web site www.zifa.com).

	DATA What	FUNCTION How	NETWORK Where	PEOPLE Who	TIME When	MOTIVATION Why
SCOPE Contextual	Enterprise Data Dictionary	Functional Hierarchy Breakdown	Locations List	Organization & Stakeholder View list	Business Event	Strategic Plan Goals, CSF etc
ENTERPRISE MODEL Conceptual	Summary Information Model	Activities within Function & Costing	Linkage of Location to Motivation	Organization & Stakeholder Hierarchy	Business Event Hierarchy	Business & Operational Plan
SYSTEM MODEL Logical	Logical Data Model	Process Model or Detailed Activities	Location data/process links	Activity for Org view & Org Data Authority	Logical Process Model or Sequence diagram	Rules & Sys Requirements w/ Logical links
TECHNICAL MODEL Physical	Physical Database Design	Object model/ Components or structured*[2]	Network Design*[3]	Security design for App/DB/Net*[4]	Job Scheduler*[5] & App Modules for Events	Links to DB/App that implements
DETAILS Implemented	SQL DDL/ DBMS Tables*[1]	Source Code/ Executable Applications*[2]	Network Cabling & protocols*[3]	Net/App/ DBMS Security	Scheduled Batch & Online apps (Function Details)	App modules & DB tables, Data and Function Details

*Other tool(s) recommended. See notes below.

1) Your selected DBMS is the appropriate tool for this cell.
2) Your application development environment (including language, Java, C++ VB, etc) fits here.
A roundtrip Object Modeling tool may also be appropriate.
3) A tool that helps to map detailed network traffic and network node design would be appropriate here.
4) Main security mechanism (Application, Database or Network) is selected and the tool used to manage
security will be highly dependent on the method selected for your particular environment.
5) Job Scheduling management software recommended to manage batch/offline process scheduling

THE BASICS ON ENTERPRISE ARCHITECTURE

Introduction

Having a complete and well documented Information Technology Enterprise Architecture allows for an organization to make effective decisions about which IT projects to pursue and the technology or products to use in the implementation.

What Is Enterprise Architecture?

The first phase of the Systems Development Life Cycle (SDLC) is called **Project**

Identification and Selection. It is in this initial phase that potential projects are identified and ranked. Then particular projects are chosen for further investigation, such as a more in-depth cost benefit analysis, or for implementation. So, what criteria are used to determine which IT projects are going to be pursued or discarded? If a project is to be pursued further, what technology will be used to implement and what base infrastructure is needed? These are the answers that an

Enterprise Architecture can answer. Known by a variety of other names such as Information Architecture, Application Architecture, Business System Architecture, Enterprise Wide Technical Architecture, the basic process is the same – to develop a high level plan of how IT will meet future business problems. To briefly define each of the above mentioned “subsets” of IT Planning:

- ***Information Architecture*** *– very similar to Enterprise Architecture, but focuses more on the application and data aspects of an IT system. Often includes the Application Architecture (mentioned below) and the corporate data model.*
- ***Application Architecture*** *– highlights the data flows between applications in a integrated information system.*
- ***Business System Architecture*** *– a mix of the strategic plans for both IT and business resources. It is normally in pictorial form and used for high-level planning.*
- ***Enterprise Wide Technical Architecture*** *– Another name for Enterprise Architecture, that better stresses that technical components as well as the infrastructure components of EA (http://www.ewita.com).*

A common word in each of the individual components above is **architecture**. In this context, an **architecture** is merely "*a set of shared definitions and constraints that are expected to effect a time, cost or risk reduction in future application development or operations"* (Gartner). The concept of an architecture is then applied to individual components of a company's IT infrastructure, or taken all together, covering the entire **enterprise**. Each of the components mentioned above complement each other to form a common goal – effective and efficient IS planning.

A corporation of any size is going to spend a considerable amount of energy preparing a strategic plan for their business. Corporate strategic planning is where a corporation takes their current business environment and decides where they want to be in the future. They then construct a strategic plan on how to get from the current to the future state. A very analogous process can be applied to Information Technology – where the current infrastructure is examined, then the desired future architecture is laid out based on both the business plans as well as what is known about future technology. A set of projects is then constructed to achieve this goal. It is important that both business needs and technical needs are considered. Upgrades, replacements and improvements can't be performed for technology's sake unless there is a business need for it. Conversely, it wouldn't make a lot of sense to built complex business processes on top of obsolete information technology infrastructure.

Traditionally, problems are identified and a project is initiated to solve that particular problem. A planning-based approach looks further into the future and seeks to find a solution to the problem both as it exists today and into the future. This is particularly important for projects that span multiple departments or business units in an organization or many organizations. Also, as IT costs are becoming a larger and larger piece of the budget, the cost of an IT "mistake" is more closely scrutinized. Internet and E-commerce applications and their associated technology are rapidly growing. An organization may outgrow a specific solution but a strategy is more long term. Granted, there are times when a tactical, short-term solution is the best approach to address a very specific, immediate need. However, it must be pointed out to the business leaders that this is the approach being taken, and the short term

gains may be offset by future support costs, conversion costs, or other implications of not working within a well-defined strategy.

Building an IT strategy based on a specific architecture can be considered a good investment – spend the time today for a payoff in the future. Or it could be considered a risky “bet.” Whether or not it is considered a “bet” or an “investment” relies on the level of predictability of the environment. A limitation of an architecture is that it relies on being able to predict the business needs into the future, and the basics of the technical needs. For example, it is probably a pretty safe bet that a company is going to want to share files and printers well into the future. They are also going to be able to want to communicate effectively with their suppliers and retailers. If the exact technical requirements (of these business requirements) were known, then all that is needed is a design, not an architecture. However, the architecture is going to describe these requirements at a higher level - such as the type of programming languages (e.g. object oriented) or Intel hardware (vs. Sun). However, the less certain about the future environment, the shorter the life span of the architecture and it will have a more narrow scope. In highly volatile environments, maybe the architecture is only good for a year or two, but constantly revised each year for the next two.

Using the term "strategy", especially when discussing the development of an Enterprise Architecture, can be confusing. A *strategy* and an *architecture* are relatively analogous terms. However, an *architecture* is often thought more of as a static picture that you draw on the wall. A *strategy* is more like putting the architecture into motion, defining not only *what* is to be accomplished, but *how* it is going to be accomplished.

IT architecture is analogous to a homeowner discussing high level plans for a house with an architect. After the plans are made, the architect or the builder (designer) can make tactical changes later on to meet other requirements, but the overall framework will stay the same. The same thing applies to IT. An Enterprise Architecture is a high level planning guide for building the infrastructure out. It has to be flexible enough, however, to accept minor changes on down the road.

Why Should An Organization Consider Enterprise Architecture?

The role of IT in organizations is changing. In the past, IT was a cost center – it didn't add to the bottom line, nor did it help gain a competitive advantage. The best that IT could do was reduce costs. However, in the past 10 years, CEOs are realizing that that IT can indeed directly increase revenue. From 1992 to 2002, the percentage of IT investments that grow revenue within a corporation will rise from 30% to an estimated 80% (Meta). This is significant. IT is now more than just overhead for a corporation, it is a business asset that much be controlled, monitored, and managed like any other asset such as buildings, factories, or machinery. Enterprise Architecture also allows a company to treat all of its IT assets as a portfolio rather than individual items. Just as you manage a stock portfolio for certain attributes, such as risk, etc., IT infrastructure should be managed the same way.

Enterprise architecture is the way to strategically manage a company's investment in IT. In times of industry consolidation, mergers, acquisitions, spin-offs, drops in the stock market, and a rougher economy, IT enterprise architecture might be the first thing to take a back seat to more pressing business problems. However, this is the worst thing a company can do. At times like these, a company needs to be able to quickly realign their IT strategy with their changing business strategy. If a corporation doesn't have a well documented IT strategy, as an enterprise architecture, then these changes are going to be very hard to make.

Some of the other benefits in having a mature enterprise architecture lie in several areas. First, by having a well established idea of what infrastructure is going to be needed in a company, efforts can go into building it, and building it with growth in mind. Then, as specific business applications wish to utilize this infrastructure for their needs, it is already built and ready to go. Standards are another area that, while may seem as a hindrance to some, can be enforced by having a documented enterprise architecture. When a company supports standards, especially in the IT arena, support costs are drastically lowered, allowing the IT staff to do one thing, and do it well vs. having to support a wide-variety of systems, but doing none of them particularly well.

Having an enterprise architecture can also help a company retain talented staff by being able to better focus training and other employee development funds, allowing the IT staff to focus on services for their internal “clients”, reduces the number of technologies the support staff has to support (see *standards* discussion above), and helps better focus the employees efforts, making their job descriptions much more clear.

As was previously mentioned, creating and maintaining an Enterprise Architecture is an investment for a company to make, with the payoff occurring sometime in the future. The goal is to be able to deliver the right information to the right decision makers at the right time.

Seventy percent of large companies are currently too busy fighting fires, solving short-term tactical solutions and not developing long-term strategic plans. Just as planning your business strategy takes a different mindset, so does developing an Enterprise Architecture. Often a company’s culture and current employees are more focused on the project at hand rather than looking at their environment from a enterprise standpoint. Because of this, simply trying to “make” your IT department think strategically will most likely fail. When Enterprise Architecture is run as a program, with dedicated staff and an on-going review of the plans, a corporation is much more likely to have the efforts supported and accepted by the business and IT staff.

Once the architecture is defined, then individual business units merely have to see where they “plug-in” to it to get their work done. For example, if the architecture plans says we are going towards a unified ERP solution, such as SAP, and a business unit wants a payroll system, maybe looking at SAP is a good start. Or if Unix systems are the “standard”, then that part of the design is already done. Finally, if the architecture defines that employee portals are important to the company to boost productivity, then a project to put a portal component together will be approved much faster than a standalone system would.

Having an Enterprise IT Architecture is also a valuable tool in reaching the 2nd and 3rd level of the SEI's Capability Maturity Model where the model calls for repeatable

tasks and a defined organizational mission. An Enterprise Architecture forces an organization to document their IT plans and align them with the business needs. Once the plans are in place, and standards are set, it is much easier to use the same methodologies, if not the same technologies to deliver business solutions. Business leadership is critical in making this move, and with enough commitment, a well defined architecture can easily push a business to the next level on the SEI model (level 2 or 3) (Meta - Enterprise Architecture Process...).

Architectures are not set in stone. However, if a business case is made for something that deviates from these standards, these plans are not show stoppers, merely a reason to stop and think about it. It is this "stopping and thinking" that shows where an Enterprise Architecture adds value. It gives the IT decision makers a baseline of the current and future environment so they can decide how (if at all) this new project fits into their architecture plans and how well it will work as the plans mature. It is up to management to decide whether or not to pursue this deviation from the standard. Or perhaps it is a signal that the standard needs to be revisited. Again, as was previously mentioned, the architecture plans need to be flexible enough to allow these kinds of changes as the technology and the business also changes.

Who Is An Architect And What Do They Do?

Below are six major "parts" of constructing an Enterprise Architecture that an Enterprise Architect would be responsible for: (*This information was taken from an Information World article by Paul Cottey*)
Identify the current state of how IT is currently being used in the organization, what technologies are being used and how they are deployed along with the business value that IT is providing.

- Examine and document any current initiatives currently being planned or implemented that involves IT resources.
- Document the future state if the current plans execute and business continues as usual.
- Document the desired future state, where the business would like to be from an IT perspective.

- Document the "gap" between the two future states. This is the difference between where the IT organization is heading and where it would like to be.
- Construct a list of projects and initiatives to put the IT infrastructure back on track towards the desired future state. This transition plan will attempt to close the gap previously identified.

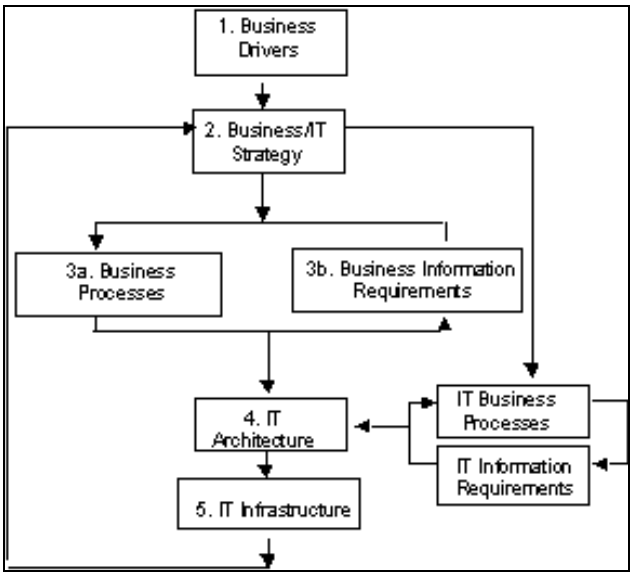
This is not necessarily a linear process, however, each phase of this process may trigger changes in previous phases. Also, the constant changes in the business environment will alter the content and priority of many of the IT projects. Equally important to keep in mind that this is a *continuous* process whereas these steps are repeated, or the architecture be at least reviewed on a regular basis. One obvious reason for this is that technology is changing rapidly and what might make sense one day might not make sense the next. Other reasons are more business focused, such as the changing strategic plan of the business, competition, laws, regulations, and economic pressures. During hard financial times, taking risks, especially in the IT area, may not make as much sense as it would during times of economic booms. All of these factors must be taken into consideration as the Enterprise Architecture and the resulting architecture plans are reviewed .

An Enterprise Architecture team is often made up of three areas:

- *Business Leaders*
- *Business Unit or Specialty Architects*
- *Enterprise Architects*

First, having a business leader heavily involved in enterprise architecture is necessary because only they understand the real information needs of the company and how they fit in with the business plans. This may be an executive, CIO, VP, or similar high-ranking individual. They may also act as a "sponsor" type of role in developing and presenting the plans to upper management. This is a very important role. Irrespective of the quality of the Enterprise Architecture, the success of the effort is primarily dependent on the support and enforcement of management.

The following picture demonstrates the connection between business drivers and IT infrastructure.



Executives and managers can bring the business drivers to the table while the IT experts identify the appropriate IT components.
Next, there are the architects or planners of individual business units. This group may also be specialty groups for specific IT projects or technologies that have their own planning teams. These are the people who know their area the best and can provide the necessary insight. Often, these may be the same people who do some of the implementation. Finally, the enterprise level architects consolidate the business information and goals with the individual business units and specialty groups. Enterprise level architects may also deal with other high level architects in the areas of data management, application development, or infrastructure (Gartner – What Do IT Architects Do?).

Once data is accumulated from all areas of the business and technical areas, then the architect's job is to define all of the major elements, interfaces, and partitions between the different areas. Even though much of the information will have to be found at the departmental level, the architect should take a *enterprise* view of the organization during the analysis. Priorities and responsibilities of individual departments may change over time, but the overall nature of the organization should be relatively stable. Depending on the organization, further responsibilities of the architects may include managing the lifecycle of major groups of technologies, procurement of infrastructure level technologies, methodologies for design and documentation, standards for lifecycle development and operational processes,

training needs, data architecture (naming standards, storage, etc.) (Gartner – What Do IT Architects Do?). Creating standards between different areas of the company will ensure that applications can be effectively integrated with each other.

A paper on Enterprise Architecture wouldn't be complete without mentioning the Zachman Framework. This framework provides a basic structure for organizing a business's architecture through *dimensions* such as data, function, network, people, time, and motivation. The last three dimensions were more recent additions to the framework. Each dimension is then described through the perspectives of different players in business's IT organization such as the planner, owner, designer/architect, builder, subcontractor, and user. This very technical and rather theoretical framework is rarely used "as-is" rather it is the basis for other, more practical applications of Enterprise Architecture. The Zachman Framework should be something that any IT Architect should at least be familiar with, and to understand some of the questions of who, what, when, why, and how - and from who's perspective are the answers. Answering these questions will be beneficial in producing a complete Enterprise Architecture. Another advantage of the Zachman's Framework is the idea of *classification*. Classification allows you to "eat the elephant one bite at a time" and identify different pieces of the IT infrastructure through different perspectives in logical pieces. By doing this, you have a much more manageable list of components to analyze.

A final part of the architect’s job involves actually using the architecture that they have created. Communication is probably the biggest part – an architecture isn’t any good if the other managers and technical designers in the company are not aware of it. By educating these people, and showing them roadmaps for technology, they can make better decisions. The architects should be responsible for dealing with exceptions and helping to minimize their impact on the stability of the infrastructure. Despite much industry wide concern about outsourcing, these enterprise architect jobs are not likely to move outside of the organization. The type of business insight, longevity, and systems thinking are hard to find in a typical IT outsourcing arrangement. As important as Enterprise Architecture is to a company, having stability in this role is crucial.

Conclusion

In the fast paced changing world of both business and information technology, companies today need to have a well-thought out, documented, and communicated IT Enterprise Architecture plan to ensure that IT provides a competitive advantage.

ZACHMAN FRAMEWORK DOCUMENT

As a company tries to become better in the creation of information systems a regular discussion occurs about the need or value of an information architecture. The success of an individual application is not dependent on having an information architecture, but the corporate goal of a seamless, integrated corporate information system requires corporate planning. A major part of this planning may involve architecture. This article reflects some of my initial thoughts on a long-term project that I have just started that deals with the Zachman Framework. For information on the general issue of information architectures refer to references at the end of this article.

The Zachman Framework basically deals with the common sense rules that we were given in middle school for writing a story. We were told to provide information about Who, What, When, Where, Why, and How in every story. This same common sense partitioning of knowledge occurs in the columns of the Zachman Framework. I will not try to defend the Zachman Framework over other frameworks, but any framework that is adopted or developed should be easy to explain to management and the involved professionals.

A standard presentation for the Zachman Framework is given in Figure 1. The knowledge that defines an application can be partitioned within the Zachman Framework. Each column has a need for a precise model that defines the knowledge associated with that aspect of the application. A current effort exists to provide the meta-model for each column. This effort would benefit from using the NLM/ORM/NIAM approach of examples and sentences to define these requirements. The type of knowledge captured in individual cells will not be discussed in this article, but an overview of the columns and a discussion of the rows will be presented.

	DATA	FUNCTION	NETWORK	PEOPLE	TIME	MOTIVATION
SCOPE	Class of business thing	Class of business process	Major business location	Major organizati on unit	Major business event	Major business goal
ENTERPRISE MODEL	Business entity	Business process	Business location	Organizati on unit	Business event	Business objective
SYSTEM MODEL	Data entity	Application function	Node function	Role	System event	Criterion
TECHNOLOGY MODEL	Segment or row	Computer function	System software	User	Execute	Condition
COMPONENTS	Field	Language statement	Address	Identity	Interrupt	Sub condition

Figure 1: *Zachman Framework*

The columns of the Zachman Framework originally included only DATA, FUNCTION, and MOTIVATION. Information that could be partitioned into these areas regularly occurs in information system designs. The NETWORK, PEOPLE, and TIME columns were added later. This information is required before an application can be successfully implemented, but it may not be a standard part of an information system design. Each column can be associated with one of the standard questions for story writing.

The first column***, DATA***, is the "What" things that are involved in the subject area. The data consists of a list of things that are important to the business. The Entity is a major data item that stands for a class of a business thing. Entities are connected via relationships. In NLM these relationships are expressed as sentence patterns and include some constraints.

The second column, ***FUNCTION***, is the "How" things are done in the subject area. This consists of a list of processes that are performed. In the NLM procedure, processes are triggered by an event that requires action be taken. Processes can also generate derived knowledge that may be stored or used to trigger other processes.

The third column, ***NETWORK***, is the "Where" things are done in the subject area. This could be a list of locations or network nodes that need to be supported to allow

the knowledge capture and use within the subject area. This availability of access is required for a successful implementation of an application although there is no procedure step to capture this knowledge in NLM. One interpretation of this column could be the physical location for the capture or use of local sentence patterns that participate in the global definition of knowledge. The design should understand if the available network supports that current application. The additional knowledge that is captured to support this conclusion should be documented. The design of this network is outside of the knowledge definition specified using the NLM procedures unless the network is "subject area" of the analysis.

The fourth column, ***WHO,*** is the "Who" that are allowed to interact with knowledge. It contains knowledge about the people and organizations within the business. Some of these rules deal with who is allowed to participate in the events that are triggered by an event rule. Other rules deal with who is responsible for a piece of knowledge that is created. This could deal with either a process in a work-breakdown structure or a sentence pattern within a conceptual schema. Again, the successful implementation of an application requires this information. The inclusion of this knowledge creates a better design. This knowledge certainly can be placed in an NLM diagram, but it may not be done without standard design practices established within an architecture.

The fifth column, ***TIME***, is the "When" something is done. This could be a fixed time that triggers a process, the time of an event that triggers another process, or a time sequencing that says this process must be performed before a follow-on process. Again, the successful implementation of an application requires this information. The formal requirement that this be addressed within a design will improve the resulting quality of the design.

The sixth column, ***MOTIVATION***, is the "Why" things are done. Business goals and strategies are listed here. These goals and strategies can be expressed as natural language sentences. In many cases, the sentences that measure these goals and strategies will have derivation rules for determining the performance of the company or organization. The derived fact instances are compared with stated goals that are called out in a business objective.

My attempt at adding value to the successful adoption of this framework was to propose standard deliverables for each of the rows. These deliverables should have the same type of common sense approach that appears in the columns. Each deliverable will have a standard form for its content, a person responsible for placing the knowledge in the standard form, a person that is accountable for the knowledge content, and a person who can validate the knowledge content. The following discussion may more directly apply to the original Zachman Framework columns, but it also applies to the other columns. The standard Zachman Framework row definitions suggest that precision will improve as a project moves down through the rows. My approach here is that each row must have sufficient precision for validation. The validation of the output from a row must be done before the work is initiated in the subsequent row.

The first row, ***SCOPE***, presents high level guidance for the project. The contents of this document, which may be referred to as a Statement of Work, are usually guided by non-technical management and written by subject matter experts who do not have information modeling skills. The document will have wonderful ideas as well as incomplete, inconsistent, irrelevant, and redundant information. Random examples may also be presented. The person performing the technical work is most likely the subject matter expert. The person who is accountable for this work is the manager who has procured funding to support the project. The person who validates that this project is required will also be from the management ranks.

The second row, ***ENTERPRISE MODEL***, presents the subject matter experts understanding of the subject area. This row involves the first formal modeling effort. The person creating the model is a skilled information analyst. The Statement of Work and the initial examples provide the starting point for model creation. The analyst must ensure that the subject matter expert provides the needed input to the modeling process. The person with the most subject area knowledge is the subject matter expert. This expert must be able to fully understand and thus become accountable for contents of the model. For this reason the requirements should be developed using the NLM procedure so that the subject matter expert only has to answer "Yes" or "No" to straight forward questions about the subject area. I have, also, seen a Use Case Model developed at this time that can serve the dual purpose

of providing consistent and relevant examples and then be used for training future users of the processes. I will be investigating this in future articles. The person performing the analysis is the information analyst. The person accountable for the model content is the subject matter expert. The person who can validate the model is another subject matter expert. The integrity of the rules developed by the subject matter expert must be maintained throughout the remaining rows.

The third row, ***SYSTEM MODEL***, presents the required or approved graphical model that is targeted at implementation, the corporate repository, or both. Standard models may be required for implementation or to support the corporate data administration function. The analyst can perform the conversion of the NLM sentence based model into alternate graphical forms and be held accountable for maintaining the validated knowledge. Another information analyst can validate that the knowledge was properly converted. The presentation of this knowledge can be in either a relational based or object oriented graphical model.

The fourth row, ***TECHNOLOGY MODEL***, presents an extended model that accounts for a specific targeted platform and deals with performance issues. This row prepares the model for implementation. The person performing this activity is a database administrator. The model presented to the row directly supports the future implementation. Additions to the model can include indexes and meaningless keys to support performance. These additions do not change any of the fact types validated by the subject matter expert. The person who is accountable for this model is also the database administrator. The person validating the resulting model is another database administrator.

The fifth row, ***COMPONENTS***, presents the code that implements the application. The person performing this function is the implementer. The implementer is also accountable for implementing the provided design. The implemented system should be validated by a tester based on the sentence based NLM model and supported by the use case descriptions. This final validation ensures that the desires of the subject matter expert were implemented. The subject matter expert thus becomes accountable for the implemented system in the same manner that the senior engineer is accountable for a production component.

There should be no creative input to the design after validation by the subject matter expert. Any suggested changes or additions to the subject matter expert's model must be properly modeled so that the subject matter expert can validate the changes. This must be done whenever issues come up in rows 3 to 5.

The contents of the row 1 documents can be precisely rewritten from the knowledge stated in row 2. I consider this an academic effort (Although, this is exactly what I do for developing my modeling course examples.) for corporate applications. The knowledge content created for the subject area should be available for the benefit of future projects. This knowledge is first stated in row 2 and then graphically presented in row 3. A corporate repository would at least hold the row 3 and hopefully hold row 2 knowledge. The row 1 documentation is not usually part of the corporate record, so the effort to redo it better has little long-term value. In the implementation of a framework, I do not want to suggest effort be put into what analysts may consider busy work in the name of architecture compliance. I have enough trouble trying to convince analysts and management that the necessary modeling should be precisely done before coding starts.

An information architecture will force a more formal information system process to exist. The effective reuse of the defined knowledge and the future integration of applications should be sufficient justification for establishing an architecture.
Happy Modeling!

ZACHMAN FOR ENTERPRISE ARCHITECTURE

The entire enterprise architecture -- enterprise information needs, corporate performance measures, and critical success factors, along with corresponding logical and physical models -- should be documented in a central repository with sufficient detail to meet all enterprise management needs. For example, the logical data model should include operational data entities that are necessary for day-to-day enterprise activities as well as strategic information data entities that, when implemented, will tell executives and managers how well their enterprise is performing.

Activities critical to capturing a clear understanding of an enterprise's information architecture include providing a clear and unambiguous definition of every data entity, describing the way each is used, defining derivation formulas and aggregation categories, and documenting data collection and retention time periods. The resulting enterprise architecture model, which links enterprise needs with data entities and enterprise rules, becomes both requirement documentation and a source for communicating the contents of the architecture (and its metadata) to its users.

Reverse Engineer Existing Environment

While the desired "To Be" architecture is being defined and modeled, the existing operational systems should be reverse engineered to provide an "As Is" architecture containing physical data models and data dictionaries.

Assess Current Data Architecture

An enterprise has an architecture, whether documented or not. Before an enterprise can implement a new architecture, it must document its existing architecture. With the focused and committed involvement of enterprise managers, there must be a detailed assessment of the current architecture. This helps to develop the scope and schedule for all follow-on activity. In order to complete eventual transition to the new architecture, it is necessary to determine the relative condition of the current environment. The factors to consider are the "popularity" of the file management systems, the physical implementation (pages of code, reports, screens, data

structures/schemas, and business rules), and the existence and relevance of investigative evidence to be included in the detailed assessment.

Generate Data Dictionaries

This effort can be facilitated through the use of automated tools. There are various sources for these tools, many of which were used to solve Year 2000 problems. Even with automated tools, however, this activity will be resource-intensive. An enterprise architecture does not initially require a detailed dictionary for every existing data source. Detailed reverse engineering should be performed only when a particular data collection has been identified for conversion.

Develop Physical Data Models

The enterprise architecture should contain a physical model for each application/database/data store that includes the tables, columns and table relationships derived from its data dictionary. Every model should include a CRUD matrix identifying under what conditions the elements are created, read, updated and deleted.

Map "As Is" Physical Model to "To Be" Logical Model

When both architectures are available, the physical data models (and data dictionaries) of the old architecture should be mapped to the logical data model of the new architecture. This mapping should include the relationship between operational elements and strategic information, including requirements for data cleansing, summarizing and transformation. Mapping existing data elements to those desired will help establish the corporate value of current operational data elements and will allow gap analysis.

Analyze the Gap

Once mapping is complete, the gap between the "As Is" and "To Be" architectures should be analyzed. Elements in the old architecture that match elements in the new architecture will provide indicators of current operational data value (elements that should be converted). Any shortfall in current data will indicate a need for new applications or data sources. Identifying data elements in the old architecture that do

not map to the new will help determine applications and files that will not need to be converted during transition and which can likely be eliminated.
Analysis of the two architectures also will help identify data problems that can be fixed immediately with limited resources and that will result in early, significant return on investment.

Develop Transition Plan

Once analysis of the relationship between the old and new architectures is complete, a plan for transition to the new architecture should be defined. The plan should be based upon the needs and priorities identified during the development of the new architecture, and should include tasks, timetables, deliverables, costs, and other resource requirements.

Engineer an Enterprise Data Clearinghouse

The transition from current applications to new applications that support the new architecture can be facilitated through the implementation of an enterprise data clearinghouse.
This special-purpose database, based upon the new enterprise architecture, provides a central store for "cleansed" data from operational systems that can be used as input to other applications. This immediately reduces interfaces while improving data quality. The clearinghouse can also provide the source for executive and decision-support information (see below). At the same time the clearinghouse provides a bridge for migrating to new applications that implement the new architecture.
The enterprise data clearinghouse is built over time as data necessary for functions/projects/business areas are analyzed and the transformation of the data elements from old to new architecture is implemented. Once the "trusted data" is stored in the clearinghouse it is available to any application that needs it. The data can also have more strategic uses.
One use for a clearinghouse is as a staging point or source for a strategic data warehouse or data mart. A data warehouse is a subject-oriented repository designed for enterprise-wide information access. It

provides tools to satisfy the information needs of enterprise managers at all organizational levels — not just for complex data queries, but as a general facility for getting quick, accurate, and often insightful information. A data warehouse is designed so that users can recognize the information they want and access that information using simple tools. Engineering a data warehouse at the same time the clearinghouse is being implemented will provide executives with strategic information immediately.

Transition to New Architecture

Programs need to be developed that transform critical key data from the existing operational systems and use the data to populate the enterprise data clearinghouse. Because the operational data has already been transformed to meet new information requirements, it will be easier to transition the current operational systems from the "As Is" to the "To Be" architecture and allow them to be migrated or replaced one at a time in priority order.

As each new application is implemented, data and information collection from current applications will be discontinued and replaced with information from the new application. For some new applications, there will be data from existing systems that must be converted to provide historical data.

The conversions may be manual, partially automated, or fully automated, depending on the accessibility and integrity of the data. Regardless of the conversion method, a strategy for improving data integrity should be developed, when required to cleanse operational data.

Enterprise Architecture Engineering Tools

Effective Enterprise Architecture Engineering requires effective tools. Enterprise needs and measures must be documented in a manner that allows them to be accessible and flexible. Further, enterprise needs should be linked directly to a strategic information model, enterprise data dictionary, legacy data base models, data integration and transformation process models, and data warehouse and data mart database design models -- all in a single relational repository. This linkage enables consistently successful development and implementation of high-quality

data warehouses and data marts. It also allows quick reaction to changes in environment, policy, or customer requirements.

Enterprise Architecture Engineering tools should have the capability to document an enterprise's multi-level, strategic plans, and link them to appropriate performance measures. More importantly, the tools should also allow information requirements that support an enterprise's strategic plans and performance measures to be modeled and translated into database, data mart, data warehouse, decision support system (DSS), and executive information system (EIS) models and designs. These designs can then be developed and implemented, ensuring that the enterprise’s information systems meet its information needs.

An Enterprise Architecture Engineering tool set should also allow multiple physical (de-normalized) database designs to be created from and/or linked to subsets of a single, logical (fully-normalized) enterprise data model in the same repository while allowing every logical model element to be linked to strategic, business, and system requirements. This facilitates both data warehouse architecture development and strategically-driven, customer-focused, model-based, software component engineering.

SAMPLE IMPLEMENTATION PLAN

Project Name

Enterprise Architecture Planning Implementation, Project Number: ORGANIZATION###

Description

The Department of Information Technology will implement a formal process for standards development and technology selection called Enterprise Architecture Planning (EAP). EAP is a formalized, industry recognized process that establishes the steps necessary for corporations and governments to analyze and select IT across the enterprise. The EAP process includes the following:

- Recognizing and integrating statewide business strategies and objectives into IT technology decisions
- Establishing a governance process and organizational structure to ensure all State agencies follow the IT standards established
- Steps for ensuring that IT decisions are made to optimize the effectiveness of IT for the whole of state government, not for one agency over another
- Defining an architecture and creating technology standards so that agencies can make quicker decisions with the assurance that the results will integrate well within the overall enterprise
- Creating a mechanism to review and act upon requests for IT that deviate from the established architecture and technology standards

Project Manager

Project oversight will be provided by xxx, ORGANIZATION'S Director of Enterprise Architecture Planning.

Project Need, Benefit and Impact

Problems and Opportunities

The following sections deal with identified problems followed by opportunities for improvement.

Problems

- Both ORGANIZATION and the agencies use independent internal processes to select IT solutions. These independent processes usually focus on individual IT project strategies and objectives, not on statewide business strategies and objectives.
- ORGANIZATION selects IT to meet statewide infrastructure requirements but does not optimally involve the agencies to ensure the results will fulfill their current and future business needs.
- Agencies select IT to meet their respective business needs but do not optimally involve ORGANIZATION nor other agencies to ensure the results will integrate well with the overall state enterprise.
- The current processes for IT selection increase the likelihood of non-standardization, duplication of systems, statewide infrastructure integration problems, and increased cost for supporting a heterogeneous environment. Opportunities
- The EAS service has a business-oriented process for analyzing and selecting information technologies (IT) across the whole of state government.
- ORGANIZATION will be afforded the opportunity to develop technology standards utilizing agency input.
- ORGANIZATION will be afforded the opportunity to better its image and have the agencies look to ORGANIZATION as a valued resource to guide and assist them in fulfilling their information management needs and meeting their business objectives.

Project Impact

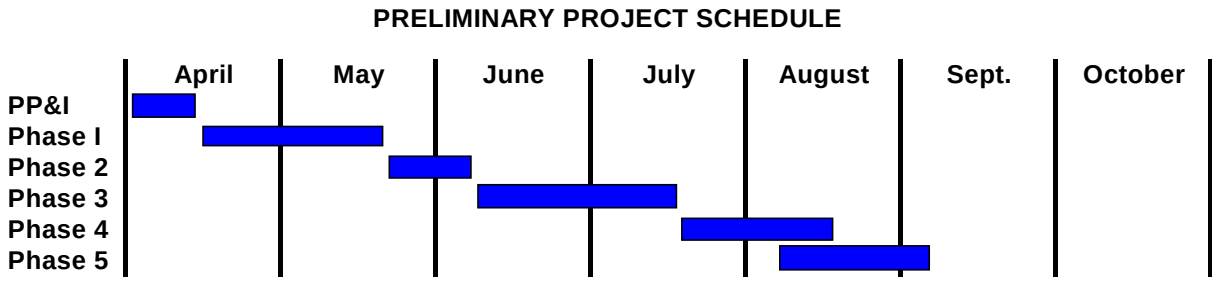
All ORGANIZATION service areas and all state agencies will be impacted by this project. The principles, standards, and best practices defined by this architecture will be utilized by all agencies in the executive branch. The statewide products and services contracts that will be put in place to implement the architecture standards will be available to all branches of state government and all the state's political subdivisions. The other branches of government will be encouraged to participate in the EAP process, to implement the resulting standards and best practices, and to utilize the statewide contracts.

IT Architecture

This project will define the Statewide IT architecture that will be used by all agencies, including ORGANIZATION.

Schedule

We believe that the scope of the effort described herein will take approximately twenty-four to twenty-eight weeks, with timely participation from the ORGANIZATION and agency staff involved. The intention is to be far enough along in the process, to



be able to document the IT projects that will require FY2002-FY2003 budget options by the budget submission deadline in September of 200x.

Project Objectives

Successful implementation of this project will result in accomplishing the following objectives.

- Increase state IT effectiveness by improving the process for IT selection to ensure that IT decisions optimize services for the overall enterprise.
- Control the growing complexity (and related support requirements) of the statewide infrastructure in order to minimize potential costs by setting statewide standards for IT.
- Enable State IT service groups to react faster to changing requirements by establishing an architecture to guide the decision making process for selecting IT.
- Integrate the EAP process into ORGANIZATION and Agency IT processes.
- Document the EAP process, policies, and procedures.
- Acquire the necessary personnel to perform and support the EAP process as an ongoing program.

- Educate participating ORGANIZATION and agency personnel on the EAP process.
- Identify statewide business strategies and objectives as the basis for the architecture.
- Create the architecture principles upon which IT standards will be derived.
- Create the IT standards for the state enterprise.
- Perform a gap analysis comparing current IT systems with the new architecture and IT standards.
- Create an Architecture Migration plan to implement the architecture.

Project Scope

Architecture Project Planning and Initiation

The objective of this initial effort is to organize the project to ensure success and refine the scope of the engagement. The resulting project plan and business case will establish a detailed ORGANIZATION and EXTERNAL CONSULANT staffing plan, identify critical success factors, formalize the technology domains to be addressed, and identify project dependencies. The results of this initial phase will ensure that all stakeholders and participants understand what is required for success.

Activities will include:

- Developing the detailed project plan.
- Reviewing existing ORGANIZATION and agencies' business strategies, strategic automation plans, and existing IT architecture documents to determine the quality of available information and how best to provide any missing information.
- Reviewing the organization(s) charts and deriving the major functional areas.
- Documenting the key players (stakeholders).
- Documenting the critical success factors for the project.
- Determining the scope and boundaries for each of the selected architecture domains.
- Analyzing project risks in terms of size, scope and project structure.

- Finalizing the ORGANIZATION project team members for each phase and major task area.
- Developing training curriculums and conducting one or more training sessions in the architecture development process for ORGANIZATION senior management, members of the Business & Technology Strategy Board, the Core Architecture Team, the Domain Teams, and the Infrastructure Teams.
- Preparing the initial communication plan for business and IT staffs (including Web publication).
- Preparing and conducting kick-off sessions for business and IT staffs.
- Establishing a Governance process and organizational structure.
- Determining the points of convergence between Agency business and IT planning, and the EWTA process.
- Determining the points of convergence between the State's budget process and the EWTA process.
- Developing an approach for addressing the state's E-Government and ERP initiatives in phases one through five of the EWTA process and
- Establishing the format for each deliverable and the process templates.

The must-do strategies of ORGANIZATION and the agencies will form the basis of the architecture and help IT focus on the most important requirements of the state's business and service programs. The Business Drivers and Business Information Requirements will be reviewed in a group setting (Business and IT Strategy Board) to help ensure business buy-in and participation in the architecture process.

Phase One: Clarify Business Strategy and Establish Business Participation: Create Business Drivers, Business Information Requirements and Architecture Vision

EXTERNAL CONSULANT will review existing business strategy documentation to be provided by ORGANIZATION. The documentation and any associated business strategy documents will establish a base for the Common Requirements Vision. EXTERNAL CONSULANT and ORGANIZATION will synthesize the current and in-progress strategy documents to create the initial Business Drivers and Business

Information Requirements. A series of interviews with senior business executives in the agencies is likely to be needed.

Phase One activities include:

- Frequent Architecture Team meetings for planning of activities, synthesis and review of materials, and creation of deliverables.
- Review of existing and in-progress business strategy documents.
- On-site working sessions with IT management to identify business requirements, critical issues and currently deployed technologies.
- Creation and documentation of ORGANIZATION and agencies' Business Drivers and Business Information Requirements.
- Management presentation and critique of Business Drivers and Business Information Requirements.
- Presentation to Business and IT Strategy Board to discuss the findings and gain support for the next phase.
- A two-day, onsite Architecture boot camp conducted with both an EAS analyst and a member of the project team. Senior management will likely only attend the first day of this training session. (This might be a Project Planning and Initiation deliverable depending upon when ORGANIZATION wants to conduct the training)
- Critique of Phase One execution including recommendations for improvements next time.

Phase Two: Create Architecture Requirements and Architecture Principles

Phase Two begins the transition from a business context to an IT context, documenting the technology qualities required by the architecture to enable the business strategy. The Core Architecture Team will utilize the Business Drivers and Business Information Requirements documented and agreed upon in Phase One to derive the Architecture Requirements.

The other deliverables of Phase Two are the Architecture Principles adapted to the state's environment from industry and IT best practices. These principles guide the design choices, standards and products recommended by the Domain Teams and enable the long-term adaptability of the IT environment. Each principle documents

Limit the number of product permutations to facilitate integration, simplify support efforts, and reduce long-term costs.

Rationale
In the absence of major business advantage, deploying multiple products delivering similar functionality needlessly increases complexity and cost. By limiting the number of technologies, and creating a more consistent environment, the State optimizes the cost for implementing, supporting, and maintaining the IT environment.
Furthermore, reducing the number of technology permutations to consider during the planning process simplifies the integration of new functionality and allows the IT environment to adapt more quickly to business change: a time reduction for technology transition and implementation.

Implications
- Consider the requirements of potential users in the greater community when investigating new technologies.
- Retire technologies providing insufficient functionality or those with high management and support costs, and transition them to new, common services.
- Include the cost of introducing new technology in cost/benefit analysis (i.e., implementation, retirement, user/IT staff training, support, and maintenance cost)
- Allow for increasing the breadth of product implementations without repeated contract negotiations by identifying the cost savings for enterprise-wide versus niche implementation in product licensing agreements.

the rationale and implications specific to the state's environment. Below is an example of an Architecture Principle:

Phase Two activities include:

- Creation and documentation of the state's Architecture Requirements.
- Creation and documentation of the state's Architecture Principles.
- Reconciliation of state's Architecture Principles with Agency IT Principles where they have already been defined.
- Identification and final selection of relevant Domain Architectures.
- Identification of specific recommendations for the detailed development of the individual domain architectures.
- Identification of domain team participants.
- Development of a current environment status report documenting our insights to date.

- Presentation to Business & Technology Strategy Board and discussion of the findings, to ensure continued business participation and gain support for the next phase.
- Critique of Phase Two execution including recommendations for improvements next time.

Phase Three: Develop Domain Architecture Detail

Phase Three creates the detailed domain architectures for the state. If needed, EXTERNAL CONSULANT will provide domain experts to supplement the EXTERNAL CONSULANT architecture team, provide knowledge transfer and produce deliverables. If EXTERNAL CONSULANT domain experts are needed, their fees are included in the option portion of the fixed fee estimate in the EXTERNAL CONSULANT Consulting proposal.
Representative Phase Three activities include, but are not limited to:

- Train domain teams.
- Creation of the principles guiding the design and engineering within each individual Domain Architecture selected in Phase Two.
- Work with the core architecture team on the definition of a taxonomy of technology domains and their relationship to infrastructure patterns.
- Work with the domain teams to define domain architectures. ORGANIZATION will determine which domain teams will use EXTERNAL CONSULANT domain experts to accomplish their work.
- Work with the core architecture team on the definition of infrastructure patterns and their relationship with technology domains.
- Work with the core architecture team on the diagramming of the relationships between technology domains and infrastructure patterns.
- Work with the infrastructure teams on the standards, configurations and best practices for each of the infrastructure patterns.
- Critique of Phase Three execution including recommendations for improvements next time.

Phase Four: Gap Analysis and Migration Planning

The Gap Analysis phase begins to bring together application, information (data), organization and technology planning. Gap Analysis begins with a review of the existing architecture baseline information. In some cases, EXTERNAL CONSULANT works with clients to create or compress documentation about the existing environment. In other cases, just documentation review and interpretation are required. The approach for ORGANIZATION will be developed during the Process Setup phase. It is the hope of the project team that the Agency IT Planning process for 200x will deliver an updated view of the state's installed base of IT in time to be used in this phase of the EAP process. We will identify the gaps between the ORGANIZATION and agencies' current state and the future/target architecture. For each gap identified, we will develop alternative solutions to "fill" the gap, e.g., new applications, new databases, new technology, new skills or management approaches. The result is a series of recommendations, with priorities and interdependencies documented.

Migration planning defines the effort necessary to implement the recommendations from gap analysis. We will identify projects, initiatives and policies to be delivered over a three-year timeframe.

Phase Four activities include:

- Review existing state and agencies' architecture documentation.
- Create documentation regarding the existing architecture, where needed.
- Identify gaps between the as-is and future target.
- Develop recommendations to close gaps.
- Prioritize and identify interdependencies of recommendations.
- Identify projects, initiatives and policies for each recommendation.
- Define a high-level migration plan for a three-year timeframe.
- Define scope and timeline for each project.
- Create a program plan to coordinate across all projects.
- Critique of Phase Four execution including recommendations for improvements next time.

Phase Five: Implementation Planning

During the final phase, we will identify the needed resources and develop the project profiles to implement the target enterprise architecture. The resources required to plan implementation projects include IT infrastructure management and staff, application management and staff, database management and staff and agency program management and staff.

Phase Five activities include:

- Identify resources for implementation planning.
- Develop schedule for completion of implementation planning.
- Develop project profiles , to include: primary design goals, scope, reuse opportunities, business or IT management sponsors, estimated timelines and resource requirements and implementation dependencies.
- Reconcile results of Agency IT Planning (e.g. Agency IT projects) with the results of implementation planning.
- Develop enterprise program management charter and plan that all project plans roll up to.
- Critique of Phase Five execution including recommendations for improvements next time.

Project Approach

Approach

Enterprise Architecture Planning will be implemented at the state using the EAP process model developed by EXTERNAL CONSULANT'S Enterprise Architecture Strategies service. This model identifies the process steps necessary to perform EAP and create an Architecture Migration plan. This model also provides the capability to minimize the time necessary to perform EAP by providing a 'fast path' approach that focuses only on the information necessary to create the Enterprise-Wide Technical Architecture. This model will be modified as necessary to meet the state's unique requirements.

Project Structure

This project will be organized using two logical stages consisting of major milestones:

Stage I –EAP Process Setup

- Define the EAP process for State
- Document the EAP process, policies, and procedures
- Acquire the necessary personnel to perform and support the EAP process
- Install the necessary tools required to perform and report on the EAP process
- Educate participating personnel (and the State at large) on the EAP process
- Integrate the EAP process into State IT processes

Stage II – Perform EAP Process

- Identify the statewide business strategies and objectives as the base for the architecture
- Create the architecture principles upon which IT standards will be derived
- Create the IT standards for the enterprise
- Perform a gap analysis comparing current IT systems with the new standards
- Create an Architecture Migration plan to implement the architecture
- Project Requirements
- Completely documented EAP system (process, organization, communication)
- Neutral EAP process and process owner
- Senior level management supports the EAP process
- State knowledgeable on EAP process through training, presentations, and documentation
- EAP utilizes both business program and State IT staffs on EAP teams
- Target time duration for EAP setup and first run execution is four to six months
- EAP process is based on EXTERNAL CONSULANT'S EAS model and modified only when necessary to meet state's unique requirements
- The implemented EAP process and documentation will be reviewed and evaluated by EXTERNAL CONSULANT EAS service.
- Staffing requirements identified by the EAP process are met

- Methods for handling exceptions to the architecture are agreed to by the entire enterprise
- The enterprise agrees to abide by the EAP process, architecture, and technology standards
- Existing process and procedures within the organization affected by the EAP process are identified and updated
- The organization's planning processes are modified and documented as needed to incorporate the EAP process
- Location(s) for EAP meetings, teamwork, and project are identified and available
- EAP process will use sound project management principles and follow state project management standards and best practices

Project Organization

Stakeholders

Primary Stakeholders:

- Chief Information Officer (CIO) [EAP sponsor]
- Agency Heads
- Agency Deputies and Chief Technology Officer (CTO) [State Business & Technology Strategy Board lead]
- ORGANIZATION [EAP process owner]

Secondary Stakeholders:

- State IT Business & Technology Strategy Board (EAP architecture and Architecture Migration plan owner)
- ORGANIZATION Directors and Managers
- Agency IT Executives, Directors, and Managers whose divisions are connected to the Statewide Infrastructure

FURTHER INFORMATION

For more information on other products available from The Art of Service, you can visit our website: http://www.theartofservice.com

If you found this guide helpful, you can find more publications from The Art of Service at: http://www.amazon.com

INDEX*

C

T