An Organization Model
The AOM-3 Architecture Organization Structure and Role Models
What Do Architects Do?

November 11, 2005

Michael Bell

An Electronic Routing Idea!
Table of Contents

ABSTRACT ......................................................................................................................... 4

INTRODUCTION ............................................................................................................... 4

APPLICATIONS .............................................................................................................. 5

KEY QUESTIONS ........................................................................................................... 5

CONSTITUTION ............................................................................................................. 5
  Mission ......................................................................................................................... 6
  Structure ..................................................................................................................... 6
  Positioning .................................................................................................................. 6
  Survival ....................................................................................................................... 6

THE MODEL .................................................................................................................... 7
  Driving Forces and Components ............................................................................... 7
  Benefits ....................................................................................................................... 7

BUSINESS REQUIREMENTS – A DRIVING FORCE ................................................. 8
  Enterprise Driving Domains .................................................................................... 8
  Events ......................................................................................................................... 9
  Impact of Business Requirements on Architecture Organizational Model .............. 9
  Durability of Business Requirements ..................................................................... 10
  A Reactive Approach to Business Requirements ................................................... 10
  A Proactive Approach to Business Requirements .................................................. 11
  Mapping Business Requirements ........................................................................... 12

TECHNOLOGICAL REQUIREMENTS – A DRIVING FORCE .............................. 14
  Fundamental Technological Requirements ............................................................ 14
  Technological Requirements Derived from Business Requirements .................... 18
  Functional Necessities of Information Technology Organizations ....................... 20
  Innovation Initiatives Technological Requirements ............................................. 20

ARCHITECTURE DISCIPLINES – A MODEL COMPONENT .............................. 21
  Architecture Disciplines ......................................................................................... 21
  Solution Activities .................................................................................................. 22
  Derivation of Architecture Disciplines ................................................................... 22
  Contributions to Vision and Mission Statements ................................................... 24

ARCHITECTURE PRACTICES – A MODEL COMPONENT ................................. 25
  Architecture Practices ............................................................................................. 25
  Practices and Architecture Management Structure ............................................. 26
  Practices Structure .................................................................................................. 27

ARCHITECTURE ROLES ......................................................................................... 29
  Composition of Architecture Roles ......................................................................... 29
Grading Expertise ................................................................. 32
Methods of Deriving Architecture Roles ............................ 32

Architecture Role Models – A Model Component ............ 33
Creation of Role Models ....................................................... 33
Deriving Architecture Roles from Architecture Role Models 34
Architecture Role Patterns .................................................... 35
Working with Role Patterns ............................................... 36
Complementary Role Patterns ............................................. 37
Converting Complementary Role Patterns to Role Models .. 38
Overlapping Role Patterns .................................................. 39
Cloned Role Patterns ........................................................ 39
Applications of Role Patterns ............................................. 40

ARCHITECTURE ORGANIZATIONAL MODEL COMPONENTS
CORRELATION ........................................................................ 40
Correlations ............................................................................ 41

POSITIONING OF ARCHITECTURE ORGANIZATIONS .................. 46
The Act of Positioning ......................................................... 46
Positioning ................................................................. 46
Positioning Management & Strategies ............................... 46
Communications, Interaction Protocols & Processes ........... 49
Positioning Architecture Organizations .............................. 50

ARCHITECTURE ORGANIZATIONAL MANAGEMENT STRUCTURE ... 51
The Structure ................................................................. 51
Architecture Management Structure Model ......................... 51
Structure Dimensions & Views ........................................... 56
Management Controls & Protocols ..................................... 62

ENTERPRISE PERSPECTIVE ................................................. 67
Appendix A ................................................................. 68
ABSTRACT

This paper presents a reusable organizational model and methodology that provides guidelines and standards to the establishment, reorganization, and alignment activities of architecture organizations in an enterprise. It can be generalized, abstracted, and adopted for various enterprise organizational initiatives by utilizing the architecture organizational model as a use case to resolve their scenarios.

- Architecture organization model components are essential ingredients to the success of an architecture foundation within an enterprise because they are the building blocks of the organizational management structure and provide different views of its main operations, activities, and expertise level.

- Architecture operations and activities are derived from enterprise business needs and technological necessities. These driving forces play a major role in the motivation and the justification of architecture organization establishment, reorganization or alignment processes.

Software providers may find it suitable to providing conceptual guidelines for automating organizational models, building management structures, and constructing enterprise role models.

INTRODUCTION

Enterprise organizations have been embracing horizontal management foundations because they are able to provide organizational vision and mission, strategic planning and roadmap for execution, introduce standards, policies and best practices, encourage asset reusability, enhance consolidation mechanisms, improve asset integration, become a central information hub, and fill communications gaps. Hence, the need for an architecture organization in an enterprise is a sign of maturity and is a response to the inefficiencies of silo organizations, lack of information technology direction, and technical management disarray.

The AOM-3 methodology describes the constitution and the fundamental principals that an architecture organization can survive on. It also defines the main model components and the drivers behind the justification and the motivation for the establishment, restructuring or alignment of an organizational structure.

This model recognizes the significance of business and technological influences on the formation of an architecture organization. Business necessities are described as the motivation to its establishment; nonetheless technological needs impact the foundation of the organization charter, operations, and activities.

The response to the question ‘what architects do’ unfolds through the introduction to the model components. These knowledge driven management entities are depicted as the main pillars of an architecture organization, influence management structures, and help shape architecture roles, organizational occupations, individual activities, responsibilities, and accountabilities.
And finally, this model consists of institutional perspective practices, such as positioning strategies of architecture organizations in an enterprise, designing techniques of architecture management structures, validation methods of communications protocols, and testing management reporting system processes.

**APPLICATIONS**

**When should this model be used?**

- Establishment – when no such organization exists in the enterprise.
- Restructuring – when the effectiveness of an organization is questionable.
- Alignment – when business or technological requirements change.

**KEY QUESTIONS**

**What are the key questions that should be addressed?**

- How should architecture organizations be established, reorganized or aligned?
- How should business and technological requirements be mapped to architecture initiatives?
- What do architects do?
- How should architecture roles be defined?
- How should architecture role models be established?
- How should architecture organizations be positioned in the enterprise?
- How should labor be propagated in a structured organization?

**CONSTITUTION**

An architecture organization is a creation of the environment in which it operates. It should be established because of enterprise necessities to resolve core business and technological problems. The environment is responsible for shaping its four supporting constitutional dimensions: mission and operations, management structure, positioning in an enterprise, and the base conditions of its survival.
Mission
An architecture organization is a technical service provider entity that should proactively offer solutions to arising challenges, discover, generalize and abstract problems for the purpose of providing strategic and tactical resolutions via technology standardization to various enterprise organizations.

Structure
An architecture organization structure should be aligned with enterprise business strategies, business initiatives, business concepts, and core business requirements. The vibrant nature of business conditions and changes in business strategies should mandate dynamic architecture organizational structure adjustments to accommodate these transformations.

Enterprise technological requirements should be grounds to accelerate architecture organizational structure changes to provide timely solutions and support.

Positioning
An architecture organization should be horizontally positioned in an enterprise. It should assume a center stage for technological solutions, evolve into a socio-political center, be proactive and the predominant driving force for strategic technological initiatives.

Survival
The existence of an architecture organization in an enterprise depends on the quality of its personnel. It also depends on the leadership level they exercise toward its vision, mission, strategies and direction, governance policies, guidelines, standards, and best practices it enforces on its ability to comprehend the business, to model business processes, and digitize them. Survival then depends on quality operations, systems development life cycle effectiveness, and the methodologies it supports.
The Model

This model contains out-of-the-box components, which can be customized to your enterprise needs.

Driving Forces and Components
The architectural organization model is comprised of two driving forces and three main components:

- Main driving forces:
  - Business Requirements that justify the creation, reorganization or the realignment of an architecture organization.
  - Technological Requirements that justify the creation, reorganization or the realignment of an architecture organization.

- Main components:
  - Architecture Practices – main operations of an architecture organization.
  - Architecture Disciplines – subject areas or fields of expertise in an organizational practice.
  - Architecture Role Models – reusable templates that provide an organizational standard for architecture role creation.

Benefits
The AOM-3 model assures the following implementation benefits:

- Motivation and Justification: The model provides motivation and justification approaches to the establishment, reorganization or the realignment of an architecture organization.
- Methodology: The model provides a methodology for the construction, reorganization, and the alignment of an architecture organization management structure.
It also provides organizational positioning approaches and offers creation methods for architecture practices, disciplines, roles, and role models.

- Enablement of a Dynamic Organizational Structure: The model’s components and their correlations enable an architecture organizational structure to dynamically align with business and technological requirements.

### BUSINESS REQUIREMENTS – A DRIVING FORCE

#### Strategic and tactical business requirements are defined and analyzed in the problem domain by various business stakeholders, such as business strategists, product managers, business analysts, and business architects. You should study your organization business requirements, and understand how they can influence your architecture organization structure and occupation.

- Tracking market conditions
- Analyzing impact on the business caused by market and technological events
- Defining the problem domain
- Performing vertical and horizontal market research
- Providing market, customer, and product segmentation studies
- Performing SWOT analysis (strengths, weaknesses, opportunities, and threats)
- Defining business strategies
- Developing business models
- Launching business initiatives
- Developing product concepts
- Sponsoring new projects, and
- Defining business requirements.

Solution domain constituents, such as architecture organizations, should be well positioned and have the capacity to provide the following:

- Leadership, methodologies, technological expertise, and execution to resolve business problems.
- Abstracting and generalizing business requirements to provide solution models (solution templates), which can be proactively re-applied to similar future events (i.e., one of their main occupations).
- Management structures should be aligned with problem domain strategic necessities and requirements.
- Realignment of organizational structures should occur when business requirements change with market conditions, events, business strategies, and business models.
Communication between the problem domain and the solution domain organizations is frequently bi-directional. Business requirements are being derived from discovered problems, which are reviewed by solution domain providers that cycle back their views and ideas, negotiate on execution terms, and inform problem domain organizations progress status.

- Problem domain organizations deliver the ‘what to do’ business aspect of the operation.
- The solution domain is mainly responsible for transforming requirements into technological specifications - the ‘how to do’ portion. In many cases, the solution domain world needs to establish technical ‘what to do’ requirements because of language and dialect barriers between the two domains.

Events

Events are random or planned business or technological occurrences, which can influence business operations, interfere with strategic and tactical enterprise processes, impact business profitability and productivity, and impede technical operations.

Impact of Business Requirements on Architecture Organizational Model

Business requirements are deeply rooted and originate in the problem domain space. They describe the necessities to solutions that are caused by events, which may impact business conditions or profitability, impose business threats, and cause business weaknesses. Requirements identify opportunities in terms of business growth, higher productivity, process improvements, business strength, reliability, and stability.

For example: To improve trading system performance and user interface response time can be a strategic business requirement, caused by a recent surge in trading volumes due to improvements...
in market conditions and the economy. The inability to satisfy traders may result in loss of customers and can affect business and profitability growth.

An architecture organizational model greatly relies on business requirements because they provide:

- Motivation and justification to organization existence.
- Substance and the context in which it operates.
- A foundation to its management structure and influences the creation of architecture roles, responsibilities, and accountabilities.

Requirements contribute to the creation of architecture disciplines and practices, which are its main areas of operations and expertise.

For example: Improving trading system performance and response time can be further generalized to launching enhancement efforts for all enterprise systems.

- This carries long-term strategic implications on the solution domain and mostly on the architecture organization.
- The organization charter and justification for its existence should be based on revamping enterprise systems and making them more powerful to deal with higher market volume demands.
- Architecture management structure and architecture roles are impacted because of new performance tuning expertise needs.
- Architecture management responsibilities should focus on new types of concerns such as systems scalability, deployment, availability, and workload management.

**Durability of Business Requirements**

Strategic business requirements can be long lasting and relevant until the business problem ceases to exist. Ongoing event re-occurrences can also impact business activity. Performance of trading systems and applications response time can be considered a long-term business concern. This necessity should be accommodated and considered a higher priority by an architecture organization and its management.

Tactical business requirements are concerned with minor business impacts and can be placed in lower priority queue levels.

For example: Requirements such as improving the look-and-feel of the ‘order statements’ page or change the headings font from Times New Roman to Arial can be regarded as cosmetic changes to the trading system user interface and may not impose threats to business activity or business profitability. Not all look and feel requirements are viewed as marginal, change font size from 10 to 14 may be an essential need since it is affiliated with the growing number of senior citizens that use the system.

**A Reactive Approach to Business Requirements**

Event occurrences may introduce changes to business strategies, business model modifications, business concept alterations, business initiative revisions, and finally, creation of or modification to business requirements. This reactive approach pressures the business organization to constantly track and pursue unforeseeable problems that cannot be immediately controlled or resolved. The architecture organization
then is required to provide swift and timely high quality solutions that depend on its readiness level, management capacity, and skill set. Effective solutions can resolve or minimize the impact of problems.

**A Proactive Approach to Business Requirements**

Readiness initiatives are designed to prevent business harm by utilizing existing architecture solution models that have been developed and tested in past events and problems. Mapping and comparisons to past solutions and approaches can facilitate the establishment of or the modification of proactive business requirements, which can augment, enhance, or derive new sets of solution models. The development of an architecture solutions knowledgebase, in the form of models, can shorten the response time of architecture organizations to business necessities and provide high quality solutions that have been challenged and road-tested on similar circumstances. This method can position the business and technological organizations ahead of the curve, eliminating future elements of surprise and omitting unpredictability.

For example: A firm that has adopted a merger and acquisition strategy to incur business growth can face similar customer name and address data merger problems in each acquisition it pursues. The core of the problem can lie in the incompatibility of their business models; the merging firms have different beliefs and strategies concerning data centralization. This clash of strategies can result in long-term disputes and confusion, business disruptions, lost of productivity, and even impact business profitability until policies, standards, and best practices are established and take effect. Architecture solution models can alleviate the pain of customer name and address data mergers since the requirements can be further generalized and abstracted into a more generic form of necessity, in which it provides a broad solution to data centralization vs. data partitioning and segmentation in the firm. This proactive approach may not always provide ideal solutions to data mergers, but with each event, a new set of proactive business requirements can improve existing architecture solution models. A progressive approach to solving problems, learning from past experiences, and adding to the architecture solution knowledgebase can exponentially shorten the reaction time to problems that can impact the business.
Mapping Business Requirements

Tiers enable the ranking, evaluation, and positioning of business necessitates. They are arranged based on their potential impact on profitability, business opportunities, and revenue growth.

- Significant business requirements should be grouped into top tiers and classified as strategic targets and focal interest points to solution domain organizations.

- Tactical business needs may not require immediate attention and should be positioned in lower-level tiers, or kept in separate queues. This arrangement provides solution directions road map and prioritized activities. The following table depicts these concepts.

<table>
<thead>
<tr>
<th>Tier</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tier 1</td>
<td>Time to Market</td>
<td>Shorten the length of time it takes to develop and enhance trading systems and trading tools, from the product concept phase and the submission of business requirements to initial deployment in production</td>
</tr>
<tr>
<td>Tier 1</td>
<td>Systems Performance &amp; Response Time</td>
<td>Improve trending systems performance and user interface response time</td>
</tr>
<tr>
<td>Tier 1</td>
<td>Increase Storage Capacity</td>
<td>Increase storage capacity to accommodate five years history of trading data</td>
</tr>
<tr>
<td>Tier 1</td>
<td>Improve System Communications</td>
<td>Currently there are duplicated customer records in Account Name &amp; Address Lookup application, Customer Account Balances application and Account Holdings application. Enable the communications between these applications to eliminate the redundancy of these records</td>
</tr>
<tr>
<td>Tier 2</td>
<td>Improve User Interface</td>
<td>Provide customers with flexible trading system user interface, enable customization, personalization and preferences system capabilities</td>
</tr>
<tr>
<td>Tier 2</td>
<td>Provide Important Trading Tools &amp; Features</td>
<td>Provide customers with extended news on demand and financial market alert features and tools</td>
</tr>
<tr>
<td>Tier 3</td>
<td>Improve System Administration Views</td>
<td>Provide systems administrator to view customer information from a single user interface</td>
</tr>
<tr>
<td>Tier 3</td>
<td>Look &amp; Feel</td>
<td>Improve trading system look-and-feel</td>
</tr>
</tbody>
</table>

Map-able Business Requirements
Business requirements influence architecture organizational structure and operations by addressing occupational priorities, setting goals, providing context to vision and mission statements, and shaping roles and responsibilities. The following are the main benefits of map-able business requirements tiers:

- **Influence architecture’s main occupation:** Business requirements provide content and context to architecture disciplines and practices by addressing their priorities, influencing their activity schedules, and facilitating the establishment of goals and deliverables. Architecture strategy, vision, and mission should be part of the top-level business requirements tiers and provide solutions and roadmaps based on their precedence.

- **Facilitate conversion of business needs to technological necessities:** The mapping process between business and technological requirements includes translation and transformation of the business language into technological and architectural dialects. The business requirements format facilitates the coverage and response to business necessities by encouraging technical prioritization, which requires architecture to provide planning and strategies based on burning and immediate business needs and project sponsoring.

- **Shape roles, expertise, and skill sets:** Architecture roles, responsibilities, and accountabilities should be established based on needs that provide technical solutions to business problems and requirements. Expertise and skill sets should correspond to strategic business necessities.

- **Help determining the position of an architecture organization in the enterprise:** Positioning is the act of ranking organizations based on their occupations, responsibilities, and strategic values, which grant them management and supervision authorities over other organizations. An architecture organization that is committed to resolving top tier business requirements should be centered and horizontally positioned in an enterprise, enabling it to take the lead on technological strategic initiatives.

- **Assist with architecture organization management structure alignment:** Architecture organizations should constantly conform to new or revised business requirements and perpetually altering role responsibilities and accountabilities. They should also continuously modify their organizational management structures to accommodate business necessities.
TECHNOLOGICAL REQUIREMENTS – A DRIVING FORCE

Technological requirements are major contributors to the architecture organizational model since they provide different views, angles, and needs from a technological perspective. These necessities can surface from translation and transformation of business requirements to technological language and terms.

- Requirements can emerge from daily maintenance and technological support activities such as systems configuration and management, product middleware, and infrastructure support.
- Requirements can be derived from fundamental needs of technological organizations such as reusability and integration models.
- A shift in technological trends should be an important contributor to proactive technological requirements, which can be facilitated by research, proof of concepts, and analysis of new technologies.

The following diagram depicts four major technological requirements groups that can contribute to the establishment, reorganization, and the realignment of an architecture organization.

Fundamental Technological Requirements
Solutions to rudimentary technological requirements underline the conditions and foundations on which technology organizations operate. The demand for structure, disciplines, guidance, and technical expertise can result in development of core solutions that impact profitability and productivity. Requirements for governance rules can embrace technology standards, best practices, and policies. Processes and methodologies can be regarded as solutions to guidance and operational procedures that are required by
most organizations. Technological patterns and models can be the backbone and the supporting frameworks of many organizational challenges.
Solutions to fundamental technology necessities are concerned with information technology environmental and operational base practices and disciplines. The technological ‘how to do’ aspects should be expressed in a collection of architectural solution models (reusable templates) with a core knowledgebase comprised of methods, technological expertise, instructions, and routines that enable architecture organizations to provide quick solutions to arising problems.

For example: Asset reusability is a leading concern in many organizations since it can impact expenditure and savings. An establishment of an organizational reusability model can encourage utilization reoccurrences of organizational assets, result in consolidation of resources, and promote decoupling of intellectual properties for the purpose of reuse, when appropriate.

Software, hardware, methodologies, and processes are other examples of organizational assets that can be reused if a reusability model is in place to educate and guide professionals how to apply it. Repetitions of events can produce similar values or result in indistinguishable outcomes, such as meetings, design and architecture sessions, training and product evaluation studies, and research and demonstration of applications that can be avoided and consolidated by utilizing a knowledgebase repository.

Requirements for models are the driving force behind an architecture organization occupation since they provide motivation to the planning of architecture initiatives, creation of road maps, and the definition of goals. These deliverables are a profound substance to organization existence and survival.

Models are grouped into four major areas of interest, which are topics of concern that require technological solutions. These views of apprehension are super-models (model of models). These concerns are comprised of technological requirements that set the framework and the boundaries to solutions, but they are the requirements themselves that demand technical expertise to resolve information technology challenges. The four major organization concerns are:

1. Methodologies and Processes
2. Design and Construction
3. Management
4. Production.
The following table illustrates the correlation between concerns, requirements, and models. The four major organization concerns are comprised of various related requirements, which specifically require models to resolve problems.

- Concerns provide the ‘what’ context.
- Requirements demand the ‘how’ aspect.
- Models present the solution.

<table>
<thead>
<tr>
<th>Concern</th>
<th>Requirement</th>
<th>Solution Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Production Concern</td>
<td>How should organizational assets be managed?</td>
<td>Asset Management Model</td>
</tr>
<tr>
<td></td>
<td>How should organizational assets be secured?</td>
<td>Security Model</td>
</tr>
<tr>
<td></td>
<td>How should organizational assets be configured?</td>
<td>Configuration Model</td>
</tr>
<tr>
<td></td>
<td>How should organizational assets guarantee its services?</td>
<td>Provisioning Model</td>
</tr>
<tr>
<td>Management Concern</td>
<td>How should an organization be established, re-structured or aligned?</td>
<td>Organizational Model</td>
</tr>
<tr>
<td></td>
<td>How should roles be established?</td>
<td>Strategy Model</td>
</tr>
<tr>
<td></td>
<td>How should management provide strategies, direction, vision and mission?</td>
<td>Systems Development Life Cycle Model</td>
</tr>
<tr>
<td></td>
<td>How should projects be managed?</td>
<td></td>
</tr>
<tr>
<td>Methodologies &amp; Processes Concern</td>
<td>How should organizational assets be re-used?</td>
<td>Re-usability Model</td>
</tr>
<tr>
<td></td>
<td>How should organizational assets be integrated?</td>
<td>Integration Model</td>
</tr>
<tr>
<td></td>
<td>How should organizational models collaborate?</td>
<td>Collaboration Model</td>
</tr>
<tr>
<td>Discovery, Design &amp; Construction Concern</td>
<td>How should events be understood?</td>
<td>Business Architecture Model</td>
</tr>
<tr>
<td></td>
<td>How should the problem domain be understood?</td>
<td></td>
</tr>
<tr>
<td></td>
<td>How should business requirements be understood?</td>
<td></td>
</tr>
<tr>
<td></td>
<td>How should organizational assets and services be designed?</td>
<td>Design Model</td>
</tr>
<tr>
<td></td>
<td>How should organizational assets and services be constructed?</td>
<td>Construction Model</td>
</tr>
</tbody>
</table>

**Fundamental Concerns, Requirements and Models**
Technological Requirements Derived from Business Requirements

Technological requirements should be derived from business necessities and placed in parallel tiers. This map-able format enables accurate alignment of essential needs on both sides of the problem and the solution domain. It facilitates the translation and the transformation of requirements from one dialect to another. A single business requirement can derive multiple technical approaches, which should be expressed in technical terms. The following table illustrates these ideas in tier one; it includes the transformed technological requirement next to their corresponding business requirements.

- The *time-to-market* business requirement is positioned in the top tier and proposes to shorten the time of product creation from the inception phase to production.

- Shortening or speeding up a product life cycle requires an array of technological remedies, which can deal with the imposed challenge.

- The translation and the transformation of this necessity can result in multiple technical requirements, such as improvement in the current development life cycle, training to increase efficiency and encourage ‘buy vs. build’ policies for the purpose of development curve reduction.

- The transformation of *system performance and the user-interface response time* business requirement into technical necessities introduces strategic challenges to an architecture organization, e.g., middleware, infrastructure, and scalability technological improvements that can speed up trading systems performance and reduce the application response time.

This type of mapping and alignment mechanism should be repeated each time business requirements are being changed or redefined. New or modified strategic technological requirements should trigger an architecture organization alignment activity and impact architecture roles and management positions that influence the context of their occupation and final goals.
<table>
<thead>
<tr>
<th>Tier</th>
<th>Name</th>
<th>Description</th>
<th>Transformed Technological Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Time to Market</td>
<td>Shorten the length of time it takes to develop and enhance trading systems</td>
<td>System Development Life Cycle: Establish a systems development life-cycle process to expedite product</td>
</tr>
<tr>
<td></td>
<td></td>
<td>and trading tools, from the product concept phase and the submission of</td>
<td>construction and deployment to production</td>
</tr>
<tr>
<td></td>
<td></td>
<td>business requirements to initial production deployment</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Training Strategies: Develop training strategies to support advanced technologies and improve staff</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>efficiency</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Application Transitioning Process: Establish efficient application transitioning processes to support</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>quick product deployment to testing and production environments</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Continuous Integration: Develop a real-time product and application integration environment by</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>establishing continuous integration processes and employing automated deployment mechanisms</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Buy vs. Build: Establish organizational policies, standards, and best practices to support buy vs.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>build strategies</td>
</tr>
<tr>
<td></td>
<td>Systems Performance &amp;</td>
<td>Improve trading systems performance and user interface response time</td>
<td>Middleware Support: Provide enhanced middleware to support high performance data transportation between</td>
</tr>
<tr>
<td></td>
<td>Response Time</td>
<td></td>
<td>services and applications</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Infrastructure: Improve infrastructure and hardware to support high trading volumes</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Scalability: Enhance vertical and horizontal systems scalability, provide efficient workload</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>management, and enable distribution of services to maximize performance and response time</td>
</tr>
</tbody>
</table>

*Mapping Business to Technological Requirements Example*
Functional Necessities of Information Technology Organizations

Technological needs of various information technology organizations, such as assurance services, software development, and data services and infrastructure can contribute to the establishment, reorganization, and alignment of an architecture organization. This is due to the close proximity in which they operate and the mutual interest in resolving internal operational problems that ordinarily are not derived from business requirements.

- A horizontally positioned and centralized architecture organization can add tremendous bridging values to silo, semi-silo or independent organizations by filling communication gaps, coordinating activities, observing and discovering commonalities, encouraging reusability of assets, and promoting quality integration of applications.
- Architecture should become the central hub of information by training and maintaining knowledgebase repositories for depositing technological studies, solution models, and documentation.
- On the strategy level, an architecture organization should help define organization identity, assist with their mission, vision, and strategy establishment.

There are three major types of functional necessities that an architecture organization should engage:

1. Base Requirements: These are needs for the establishment of internal processes, methodologies, and environments that are affiliated with essential organizational operations.

   For example: The foundation or enhancement of virtual servers environment, migration of software or hardware, help with product evaluations and adaptation, establishment of development staging and deployment environments, establishment of source code transitioning and repositories, and assisting with infrastructure and network operation processes.

2. Consultation Requirements: These necessities are based on architecture consultation services provided to various organizations based on short-term needs or on a project basis, e.g., software design and modeling, systems architecture, network architecture, data farm and data mining design, and architecture.

3. Keep lights on (KLO) or keep the world going (KWG) requirements: Ongoing support initiatives such as data recovery, configuration management, installation management, production support, production monitoring, and security services are a part of baseline information technology organization activities, which are executed at all times regardless of any occurring business initiatives and events.

Innovation Initiatives Technological Requirements

Technology trends and industry innovations can play a major role in the creation, reorganization, and the alignment of an architecture organization because of their impact on readiness and adaptation activities, which should embrace modernization of information technology facilities, software platforms, systems, and infrastructure. Staying tuned with emerging and cutting-edge technologies can immensely contribute to the improvement of technical services. It can provide value to the business and impact productivity and profitability.

Requirements for information technology innovation initiatives can include the establishment of a research and development (R&D) organization along with the foundation of research and development
practices, standards, and policies. Other requirements can be affiliated with product evaluation and adaptation initiatives, which would require proof-of-concept and quality assurance efforts.

ARCHITECTURE DISCIPLINES – A MODEL COMPONENT

Architecture Disciplines

An architecture discipline is a solution domain framework and a specialized area of knowledge, an architecture asset, and an intellectual property. Disciplines are derived from business requirements, technological requirements or other necessities such as management and administration requirements. The disciplines are managed by subject matter experts that enable an architecture organization to resolve problems by providing “what to do” guidelines in the form of detailed technical requirements. This hands-on authority is responsible for assisting the architecture leadership to translate and transform business requirements into an elaborated technical work format, further dividing this labor and dispatching it to the lower chain of command in the architecture organization.

For example: Deployment management, integration management, business architecture management, and security management can be conceived as disciplines, which provide technical requirements, plans of engagement, directions, and detailed roadmaps.

Management and administration requirements can derive architecture management and administration management disciplines. Both are internal frameworks that are important to the creation of management and administration roles in an architecture organization.

Contributors to architecture disciplines
Solution Activities

A solution activity is a hands-on knowledge entity that contains prescriptions, routines, methodologies, and guidance to resolve problems in specific fields of expertise. It is managed by technical staff that is chartered to provide the technical ‘how to do’ detailed solution guidelines in the form of technical specifications. An architecture discipline is comprised of one or more solution activities, which can tackle problems from different angles and views. These activities are the architecture organization arms of execution positioned at the lowest chain of command in the management structure.

For example: A trading application has the need to communicate to a customer profile application by utilizing integration mechanisms. One of the selected solutions would be to utilize third party integration products. A product integration support solution activity can offer product selection and evaluation services, establish product adaptation processes, and provide integration best practices and policies. Furthermore, middleware strategies and architecture solution activity can be commissioned to strengthen the solution by offering hub and spoke architecture to help bridge the communication problems between the customer profile and trading applications. The collaboration between solution activities to resolve problems often can result in the elimination of concerns and in a collaborative successful implementation.

Derivation of Architecture Disciplines and Their Solution Activities

The ability to understand business and technological requirements and to translate them into architecture disciplines and solution activities can successfully facilitate the establishment of an architecture organization, shape its management structure and form its roles, accountabilities, and responsibilities. In organization restructuring or alignment initiatives, some existing disciplines may be de-commissioned leaving room for newly grown disciplines. The elimination or establishment of disciplines is mostly rooted in business and technological changes that result from arising problems.

Disciplines and their affiliated solution activities should be derived from strategic requirements, gathered from both the problem and the solution domain. It would make sense to base an architecture management structure on long-lasting demands and requirements that are rooted in fundamental business and technological concerns that keep resurfacing. Disciplines can originate from business requirements alone, from technological requirements only, or from both. Requirements that are not derived from either are usually affiliated with management and administrative needs.

The following illustration depicts how an architecture discipline can be derived from business and technological requirements. The discovery of duplicate customer records in various repositories and lack of a centralized administrative user interface application can provide different views of customer
balances, customer name and address, and customer trading information is a challenge to account management. Thus, the business organization demands a solution to better manage trading system complexities. It is interested in a technological proposal or plan that demonstrates a solution to resolve the centralization of customer data. Provide better systems integration mechanisms by employing a middleware layer to facilitate communications between several trading systems. Furthermore, the technological requirement to design a focal user-interface, which is capable of displaying several client information views can eliminate the replication of customer records and enforce data integrity standards.

In this example, business and technological requirements derive two major disciplines, which were selected to provide solutions.

1. The integration management discipline is responsible for providing the means to better systems integration by middleware enhancements and by incorporating integration products.
2. The software design discipline provides design blue prints and artifacts to demonstrate a centralized customer user interface approach.
Contributions to Vision and Mission Statements
There should be a need for the categorization and prioritization of architecture solutions to establish or revise architecture vision and mission foundation statements. Architecture disciplines and solution activities derived from the top tiers of business and technological requirements should be recognized as strategic elements of an architecture organization and contribute the most to its vision and mission proclamation. Tactical constituents may play a decreased role in shaping the direction and strategy of an organization, but can provide short and midterm solutions to carry out architecture tasks.
ARCHITECTURE PRACTICES – A MODEL COMPONENT

Architecture Practices

And now, establish architecture practices, which are main architecture organization occupations. This process is definitely a bottom up approach to forming architecture mechanisms that can provide solutions, and maximize alignment with major business and technological concerns.

Practices are main architecture operations, a coarse form of organizational solution management that provides core areas of expertise and problem solving capabilities. A skillful management force that understands the dynamics of labor division and dispatching, which can establish communication protocols and constitute organizational reporting systems, owns them. This layer of leadership provides the high level ‘what to do’ direction, nevertheless it does not furnish the ‘how to do’ aspects of the solution. A practice is not only comprised of architecture disciplines, it is also derived from their technical commonality. The categorization of disciplines into a subject area of architecture interests that influence the composition of practices and their management structure.

For example: A software architecture practice, which is required to provide software-related architecture services, may contain disciplines that handle and support software solutions such as software analysis and design disciplines, application architecture disciplines, and software modeling disciplines.

The following example illustrates the relationship between architecture practices and architecture disciplines. The depicted organization contains two major practices, which are comprised of disciplines and their related solution activities. Disciplines can be shared among architecture practices because of their needed expertise and capabilities.

An architecture management discipline can be included in the software architecture practice and in the hardware architecture practice because both have management capability needs.
Disciplines provide remedies to enterprise concerns by utilizing affiliated solution activities, which tackle problems from different points of view and furnish technical specifications to accomplish goals. Conversely, practices offer a crude form of the solution. They serve as a valuable foundation for the architecture organization and are conceived as its root management structure. The following example depicts a software architecture practice, which is comprised of different architecture disciplines, derived from business and technological requirements. The action of grouping disciplines must be based on a common denominator, e.g., software affiliated management tasks is the common attribute that makes this classification possible.

**Practices and Architecture Management Structure**

Your architecture management structure can be based on **practices** that are key operations in your IT organization.
Practices Structure
Business, technological, and other requirements can affect the ratio between practices and disciplines. It depends on organizational necessities and their base values, vision, and mission the architecture organization was established on. Since practices are a coarse form of management operations, they are conceived as high-level areas of expertise; therefore, an architecture organization should be comprised of few practices, regardless of its environment size. A large number of practices can increase the risk of redundancy and unnecessary operations overhead.

An investment-banking institute management commissioned their technology advisory board to establish an architecture organization based on business and technological requirements. These necessities focused on major areas of interest such as needs for repository strategies and storage integration facilities, new development of loan and credit applications, integration, and consolidation of various client name and address applications. They first established a considerable number of disciplines and affiliated solution activities; next, these disciplines were categorized into major areas of interest, which provided the motivation to establishing three major management practices: software development practice, data and repositories strategies practice, and global architecture logistics practice.

The suggested process to establishing practices in an organization is a bottom up approach, working from the details to a more generalized construct, to a coarse form of management. This method advocates first ascertaining the mapping of business and technological requirements to architecture disciplines, developing solution activities, and grouping these disciplines into architecture practices. The following diagram depicts the process of practice creation, as indicated in the following steps:

1. Business requirements are being delivered.
2. Business requirements are transformed to technological requirements.
3. Business and technological requirements derive disciplines.
4. Solution activities are created for each discipline.

5. Disciplines are grouped for the purpose of forming architecture practices.
ARCHITECTURE ROLES

The answer to the fundamental question *what do architects do?* can be found in the solution domain. Disciplines and solution activities are the best place to start from; business and technological requirements should be first in place. Once architecture disciplines are defined and solution activities are finalized, roles can be derived; subsequently they can be positioned in an organizational structure.

Composition of Architecture Roles
An architecture role should be comprised of solution activities that are conceived as its accountabilities, its course of actions, and its statement of work. Activities can be selected from various architecture disciplines spread over an assortment of architecture practices. The following illustration depicts the relationship between a role, architecture disciplines, solution activities, and architecture practices.
Realistically, a role that spans over multiple architecture practices may be hard to fill since it would require a unique skill set. Moreover, from a management perspective, it would make sense to create roles that operate within one architecture practice because it can reduce accountability conflicts and it can enable teams to focus and strengthen their knowledge in one area of expertise.

The following example illustrates the composition of an application architect role from various solution activities, which were selected from different architecture disciplines within one architecture practice. This role offers application architecture level support that provides analysis, design, architecture, and programming services. The strategy and the methodology disciplines were not included because they were irrelevant to this position.
Grading Expertise
Grading solution activities expertise level (range 0 - 100), can further refine architecture roles. This skill scoring method can provide better role tuning to accomplish tasks within the architecture disciplines. The following illustration depicts this idea.

- The *software design services* solution activity is in the *design* discipline and is defined in the *software architecture* practice. It is highly required for an application architect role, thus valued 100.
- Conversely, the required skill value of the *physical design services* solution activity contained in the *architecture* discipline is defined in the *software architecture* practice with a value 40 since such a role does not require a high skill value in hardware design.

**Tuning an Architecture Role**

Methods of Deriving Architecture Roles
There are two different methods of deriving roles. Architecture roles that can be derived from

- Solution activities (the above approach): This method can be successfully applied to small architecture organizations with fewer disciplines, solution activities, and roles to manage.
- Models: This approach is provided next, it supports a construction of a role model before deriving architecture roles.
Architecture Role Models – A Model Component

Role models are reusable skill templates that facilitate the creation and shaping of architecture roles by providing acceptable value ranges to solution activities expertise, by which roles can be evaluated against. Models provide supporting elements to the foundation of an organizational structure and set standards for various role-hiring initiatives in an architecture organization.

You will find it easy to hire architects if you follow architecture organizational role models requirements. These models should be aligned with enterprise strategic business and technological requirements, and derived from disciplines and solution activities.

Creation of Role Models
The process of creating role models is somehow similar to the establishment of architecture roles. Both are comprised of solution activities, selected from various architecture disciplines that are defined in one or more architecture practices. Models are different since they provide templates that define the required expertise value-range for each solution activity. Selection of expertise values should fall within this spectrum when creating roles from role models.

The following illustration depicts a security architect role model and the allowable range of expertise values for each solution activity (ranges are shown in dark gray on each solution activity). The security strategy services solution activity in the security discipline, defined in the architecture logistics practice, mandates a threshold expertise value of 85 (i.e., a low value) and a maximum value of 95. In the same fashion, the deployment models development solution activity defined in the deployment management discipline requires a threshold of 45 and a maximum value of 85. This is a much wider range thus a more lenient requirement.
Deriving Architecture Roles from Architecture Role Models

Solution activities expertise values in architecture roles should always be set above the corresponding architecture role model threshold values. Since models define an allowable expertise range per solution activity, multiple roles with different skill strengths can be derived to accommodate architecture organization resource planning and solution management requirements.

This approach refines the requirements for an architecture role because skill evaluations are confined to a range that is mandated by an architecture role model. Moreover, this range provides some flexibility in the definition of architecture roles because skills do not have to precisely match optimal values. The following illustrations depict two security architect roles derived from a security architecture role model.

1. Role 1 represents the minimum required expertise level for each solution activity that is defined in its corresponding role model (just above the threshold value).
2. Solution activities expertise values in role 2 match the highest values of its model.
Security Architect Role Model for the Architecture Logistics Practice

Operates in the Architecture Logistics Practice

Derived Security Architect Role with Minimum Requirements
Operates in the Architecture Logistics Practice

Derived Security Architect Role with Maximum Requirements
Operates in the Architecture Logistics Practice

Architecture Role Patterns

Role models are useful for individual architecture role placements, while role patterns reflect your organization broader needs, and assist with the mobilization of resources to most burning challenges.

An architecture organization is responsible for providing quality resource management by solving the complexity of expertise placement, providing effective talent coverage, and matching skills with solution activities. Architecture role patterns can facilitate these concerns because of their reusability capabilities, their ability to provide various views of the solution activity spectrum, and their capacity to provide good resource allocation mechanisms.

Role models and role patterns represent the same concept at different semantic levels or from different points of view. Role models are templates that facilitate the creations of individual architecture roles.
Role patterns are management tools that can provide horizontal expertise coverage to solution activities on a wider scale. They are more about capacity and spectrum coverage of architecture operations. Combining roles could provide the best possible expertise and solutions.

Graphically, an architecture role model is regarded as a role pattern when its solution activity expertise threshold, high values, and solution activity bar demarcation lines are stripped off. The following diagram illustrates an architecture role pattern of a security architect. It depicts a visual expression of its corresponding role model.

![Security Architecture Role Pattern](image)

**Security Architecture Role Pattern**

**Working with Role Patterns**

Three different mechanisms are provided to alleviate the difficulties of resource selection and assignment, expertise matching, and solution management in an organization. These three methods described in the next paragraphs are visually based on pattern relationship and composition approaches:

1. Complementary Role Patterns
2. Overlapping Role Patterns
3. Cloned Role Patterns.
Complementary Role Patterns

Role patterns are vital to the establishment of complementary skills in an architecture organization for the purpose of expertise diversification and supplementation. There are times when budgets do not allow for the acquisition of similar skill sets or the architecture organization strategy does not require roles that have common characteristics.

Complementary role patterns can be created by subtracting each solution activity expertise value contained in the base role pattern, the highest possible scale value (100) and assigning the result to the complementary role pattern. In the following illustration, complementary pattern expertise values supplement the values of the base architect role pattern (the dark gray pattern shown upside-down complements the base role pattern, which is expressed in a lighter gray). The solution activity development process support maximum expertise value defined in the process management discipline is valued as 15 in its security architect role pattern.

Security Architect Role Pattern and its Complementary Role Pattern

In the same fashion, the complementary pattern supplements the skills of a security architect role pattern, as illustrated in the following diagram (the model was turned upside down). The solution activity development process support maximum expertise value is 85 (supplements the security architect 15 value).
Complementary Role Pattern of Security Architect Role Pattern

Converting Complementary Role Patterns to Role Models
Complementary role patterns can be converted to role models by simply adding to each solution activity its acceptable expertise range (threshold and high value) and drawing the demarcation lines between solution activities. After this transformation, they should be considered as complementary models because they supplement the base role models that they are derived from. The following illustration clarifies this idea.

Role Model Complements a Base Architect Role Model

Architecture roles that are derived from complementary role models can be valuable to an architecture organization since they result from solution activities that are not counted for in their base models. This approach enhances the architecture organization’s horizontal coverage to a great extent, since these created roles complement each other and do not have overlapping skills.

Solution activities expertise that are expressed in complementary role models should only be regarded as supplemental skills to the base role model and should not be utilized for the creation of architecture roles that are the same type of its base role model.
For example: The complementary role model derived from the security architect role model cannot facilitate the creation of security architect roles because they do not possess the same security skills as the base role model demands.

**Overlapping Role Patterns**
The creation of overlapping role patterns should result in common expertise subsets, which can strengthen certain solution activities and provide extra support to problems that need immediate attention or require resource augmentation. This approach would probably be more realistic in larger corporations, since overlapping roles are necessary to provide better coverage on pressing issues. Thus, overlapping role patterns can provide the flexibility of strengthening solutions and provide less support to others in a spectrum of architecture disciplines.

An architecture organization that is required to offer application architecture consultation and software design support to a large number of projects would need to beef up its software design services solution activity capabilities by augmenting its resources and providing additional design need support. The following illustration depicts two architecture role patterns (in dark and light gray pasted on top of each other), which have overlapping expertise capabilities in the design discipline. The gray shaded role pattern expresses strength in the strategy discipline as well.

**Cloned Role Patterns**
Cloned role patterns are ‘carbon copies’ of the master model pattern. They are replicated for reuse in different environments and circumstances, such as in other architecture organizations or in other corporations if their business strategies, business initiatives, and business requirements are comparable and their technological requirements are similar.
For example: Commodity trading companies may face the same industry challenges in terms of competition, market volume, and demand of goods during their business life cycle, which may affect their business models, business strategies, and business initiatives. Low transaction volumes and high profit margins can yield to similar technological necessities such as utilization of document management to store large contracts, the need for storage devices, and middleware facilities that enable the transportation of large amounts of data between consumers and producers. Scalability and clustering of applications may not be a high technological demand in this case, since low transaction volumes do not require such facilitation.

The main benefits of reusing cloned role patterns are the expediency in which they can be adopted. Clones can eliminate the need for discovery, analysis, and requirements processes and can expedite the development process of establishing organizational standards. Reusing cloned models is not an optimal method to architecture organization development because they are borrowed from different operating environments and may be tailored to solving different problems.

Applications of Role Patterns
Weigh, coverage and replication are three important application mechanisms that an organization should employ when creating role patterns to support its solution activity management.

- The weigh method utilizes overlapping role patterns to strengthen the support of particular solution activities by evaluating the volume, urgency, and strategic value of their requirements.

- The coverage technique uses the complementary role pattern approach to widen the solution activity spectrum support in an enterprise.

- The replication approach should employ the cloned role pattern method for expedient reusability purposes across organizations and businesses.

Tactical and strategic solution activities can be managed in different ways depending on the overall strategy, vision, and mission of the architecture organization.

ARCHITECTURE ORGANIZATIONAL MODEL COMPONENTS CORRELATION

The AOM-3 model foundation is based on the three major components discussed so far; architecture disciplines, architecture practices, and architecture role models. Each possesses unique characteristics, ideas, and concepts that an enterprise should study when building, reorganizing or aligning an architecture organization. Comprehending the correlations among
its components, examining their interfaces, and observing element interactions can help develop the model.

**Correlations**
The components of an architecture organization model offer a set of correlation and construction rules to provide the enterprise with the establishment and realignment process of an architecture organization. Component relationships present aggregation and dependency type of affiliation that can assist with the conceptual composition and positioning of an architecture organization structure.

- An architecture organization consists of one or more architecture practices.

- An architecture practice contains one or more architecture disciplines.

- Architecture disciplines are derived and depend on business, technological, and other requirements. Technological requirements can be originated from technological necessities and from business requirements.
An architecture discipline is comprised of one or more solution activities.
• An architecture role has one or more solution activities.

• An architecture role model is comprised of one or more solution activities.

• An architecture role is created from an architecture role model and it depends on its solution activity expertise value boundaries.
• An architecture role pattern is generated from an architecture role model and it depends on its maximum solution activities expertise values.

• Complementary role pattern, overlapping role pattern, and cloned role pattern are types of an architecture role pattern.
The following diagram depicts the big picture of an architecture organization model based on its components and sub-components.
POSITIONING OF ARCHITECTURE ORGANIZATIONS

The Act of Positioning

Sizing the spectrum of organization influence is the act of positioning, which sets boundaries to its weight, control, and domination in an enterprise. Establishing organizational positioning dimensions can set these limits. A horizontally positioned organization is granted extended authorities and is approved to operate and execute across multiple domains and organizations. Vertical positioning limits organizational roles to a silo form of management, which enables an organization to focus on a narrower scope of execution. Positioning organizations is fundamental to the following:

- Establishment of management structures
- Constitution of authorities
- Construction of hierarchies
- Foundation of inter-organizational relationships
- Manifestation of enterprise cultures.

Positioning

Positioning is required during establishment, restructuring or alignment of an architecture organization. Dynamic and environmental aspects such as the human factor, socio-political influences, market conditions, new business strategies, business requirements, technological requirements or management changes may trigger this act.

For example: A surge in economy growth mandates the facilitation of high volume transactions by building reusable and scalable services, which may require extending the software development organization’s role and supremacy beyond its original charter. Thus, locating it in a partial dominating positioning dimension can enable it to lead initiatives in a few information technology groups, such as supervising data storage projects in the data services group or even mandating operational changes in the assurance services group.

Positioning Management & Strategies

Positioning management is about the formation of a dimension map, in which organizations recognize their site and operations boundaries in the enterprise. This type of influence can be expressed and graphically visualized by their allocated space and orientation on that map. There are a few mechanisms that can facilitate the creation of an enterprise positioning map:

- Fully horizontal positioning: This organization is empowered to oversee, lead, and direct all other vertically positioned organizations. In the following graphic, the enterprise is an “architecture centric” type, the enterprise became “architecture centric” because the architecture organization is fully horizontal.
- 47 -

Semi-horizontal positioning: This scenario grants leadership over few organizations, yet, others are allowed to operate under different management groups. The following diagram depicts an architecture organization that provides leadership to the development group, training services group, and assurance services group, but the engineering group is not included under this supervision.
Semi-horizontal positioning can come in different flavors and patterns. The following illustration provides a different level of complexity in which two architecture organizations operate separately in their own divisions.

For example:

- The medical application development division is dominated by architecture organization 1 and specializes in construction of products and deployment.
- The medical electronic storage division requires different types of specialties, which are provided by architecture organization 2.
- The data services group is semi-horizontally positioned under architecture organization 2 and it oversees operations of its three subordinate groups; business intelligent, data farm management and data analytics.

Alternatively, this positioning strategy can be replaced with a fully horizontal positioning pattern in which one architecture organization has two main practices; application development practice and electronic storage practice.

- Partial domination positioning: This positioning can guarantee partial involvement in other organizations’ affairs, domination over some groups, and partial or project co-management. Such strategy is recommended in environments that regard architecture organizations as facilitation and training groups, which are not engaged in leadership initiatives.
• Fully vertical positioning: Vertical positioning narrows the functionality scope of organizational occupation and form a silo style of management and operations. Such a tight spectrum view enables organizations to focus on implementation details and on the tactical aspects of the services they provide. This type of positioning impacts enterprise inter-communications and organizational level dialog, disregards reusability of core enterprise assets, and impairs product integration. Vertical positioning strategies should be limited to organizations that offer tactical solutions and are not major strategic players in the enterprise.

Communications, Interaction Protocols & Processes

Setting standards of communication between organizations, establishing interaction protocols, and defining inter-organizational operation processes should accompany the act of positioning. Additionally, each organization branded on the enterprise-positioning map should define its identity by providing its vision and mission, organizational structure, and governance rules conforming to the overall enterprise charter.
Positioning Architecture Organizations
An architecture organization should be fully-horizontally positioned, empowered to serve as an overseeing centric entity, set direction, and craft vision and mission of technological initiatives in an enterprise. It should be permitted to participate and lead strategic and tactical initiatives in various IT groups, exposed to their arising problems, and encouraged to provide solutions. The following are some of the compelling reasons to such accommodation:

- Increase communications between silo organizations in the enterprise.
- Facilitate reusability of assets among vertical groups.
- Reduce redundant IT and business architecture initiatives in corporations.
- Assist and lead integration initiatives across the enterprise.
- Aid various organizations in an IT institution to define their identity and to develop their strategies.
- Provide the enterprise with guidance, standards, best practices, and policies.

The following illustration depicts an architecture organization that is horizontally positioned relative to enterprise silo organizations.
ARCHITECTURE ORGANIZATIONAL MANAGEMENT STRUCTURE

Here is where all comes together! The architecture organization models provide solid structure to your organization.

The Structure
Architecture organizational structure is a core management entity. It is responsible for the following:

- Dividing labor into specific tasks.
- Achieving coordination among its initiatives.
- Assigning accountabilities and responsibilities to architecture roles.
- Defining authority boundaries to personnel.
- Designing organizational management hierarchies.
- Creating reporting systems.
- Establishing communication protocols among participating roles.

The structure of an architecture organization should correspond to architecture practices, architecture disciplines, and solution activities, which are discovered and derived from business, technological, and other requirements. It should be dynamic and provide flexibility and agility to business and technological trends. Alignment with strategic business and technological requirements should occur often. Restructuring efforts should take place when the effectiveness of the organization is questionable.

Architecture Management Structure Model
There should be four conceptual layers to an architecture management structure model, in which architecture management roles are granted ownership of these layers:

1. *Architecture leadership* layer
2. *Architecture practice management* layer
3. *Architecture discipline management* layer
4. *Architecture solution activities management* layer.

The following diagram depicts this model.
Architecture management layers should be responsible for translating, transforming, and interpreting requirements into the labor architecture type, which provides subordinating management layers with more granular, detailed, and specific tasks and goals to accomplish.

- Leadership Layer: Supervises the overall strategic direction of the organization by taking a proactive approach to resolving business and technological problems. It provides strategic and conceptual support, helps translating business and technological requirements into architecture tasks, and maps them to architecture practices. The translation, interpretation, and the transformation of business and technological requirements into architecture work is the art of analysis and discovery. This may involve not only the leadership layer but can include all four layers of the architecture organization, stakeholders in the business organization, and technology groups.

Propagation of architecture tasks to the practice management layer is the act of initiative coordination that is needed for the purpose of controlling the flow and the division of work within the organization.

For example: The business requirement provide customization, personalization, and preferences capabilities to enrich our customer experience on our Wall Street business news magazine site would require building infrastructure, designing and architecting an enterprise portal, and utilizing advanced technologies to achieve such business imperatives. The architecture leadership layer divides this labor into architecture tasks and assigns them to various architecture practices.
- 53 -

- The software architecture practice is required to design and architect a portal.
- The logistics architecture practice is engaged to provide planning for supporting the portal infrastructure and evaluating various portal products.
- The data architecture practice is commissioned to model the data and create repositories.

The following illustration depicts this idea.

Propagation of Labor – Architecture Leadership to Architecture Practice Layer

- Practice Management Layer: This layer is conceived as the foundation structure of an architecture organization, owned by architecture management roles and supervised by the architecture leadership layer. It is a coarse form of solution management that is responsible for further refining and propagating the labor to its subordinating discipline management layer and is accountable for coordinating tasks among architecture disciplines. Practice management is occupied with providing the high level ‘what do to’ guidance of the solution, furnishing business architecture insights, assisting with processes and methodologies, and pointing to high level technological remedies without providing the ‘how to do’ detailed portion of the solution.
For example: Management of the *software architecture* practice should provide the following:

- Delegate responsibilities.
- Involve and guide the *software & product integration management* discipline, the *design & architecture service* discipline, and the *business analysis management* discipline.
- Provide portal solutions development and support to customization, preferences, and personalization requirements.

The following diagram illustrates this approach.

**Propagation of Labor – Architecture Practice Layer to Architecture Discipline Layer**

- **Discipline Management Layer**: The discipline management layer is responsible for providing the detailed implementation of the ‘what do to’ portion of the solution. This layer should be concerned with the following:
  - Achieving successful solutions.
  - Propagating the labor to its subordinating solution activities management layer by matching types of work to expertise capabilities of its various owners.
  - Providing methods of implementation.
  - Coordinating solution activities.

The ‘what to do’ portion of its responsibilities should be translated into detailed technical requirements, tactical plans for each solution activity, road maps, and expected targets and goals.
As depicted in the following table, the design architecture discipline is responsible for interpreting and transforming the ‘provide portal design and architecture blue prints’ requirement into detailed technical requirements and distributing them to its subordinate solution activities.

<table>
<thead>
<tr>
<th>Solution Activity Management</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conceptualization</td>
<td>Develops software concepts from given portal requirements.</td>
</tr>
<tr>
<td>Visualization</td>
<td>Designs a presentation layer, which accommodates customers with preferences, customization, and personalization.</td>
</tr>
<tr>
<td>Software design</td>
<td>Provides design artifacts for handling the logic behind the presentation layer.</td>
</tr>
<tr>
<td>Software architecture</td>
<td>Provides scalable blue prints of components and services, designs middleware solutions, and provides an integration plan.</td>
</tr>
<tr>
<td>Software modeling</td>
<td>Designs and creates a working executable model to the suggested architecture and its capacity to support client customization, prioritization, and preferences services.</td>
</tr>
</tbody>
</table>

*Propagation of Labor – Architecture Discipline Layer to Architecture Solution Activity Layer*
Solution Activity Management Layer: This layer is a hands-on technical specification provider and tactical implementation entity. It is responsible for providing implementation detailed descriptions, elaborating on the ‘how to do’ portion of the solution. Its main focus should be on achieving the goals and the targets assigned by the discipline management layer.

As seen in the previous diagram, the solution activity layer should cover the spectrum of the solution by suggesting a range of combined activities that can resolve the problem. The conceptualization and the visualization groups should be concerned with providing detailed technical specifications, which should be used by the design, architecture, and the software modeling solution activity manager.

For example: Conceptualization activities should produce technology approaches for using the portal product, map product features to IT technology capabilities, introduce new software concepts to the development community, and suggest various implementation methods. Visualization activities should be concerned with solution presentation specifications, which should include the look and feel aspect, user interface behavior, and its capabilities.

Structure Dimensions & Views
Structure of architecture management is comprised of four major bonded dimensions:

- The structure depth of architecture practices that depicts the hierarchy of management in the organization.
- The spectrum of architecture disciplines that influences the solution coverage and capacity of an architecture organization.
- The human factor constituent.
- Architecture role compositions.

These dimensions can be observed from the different views as depicted in the following diagram.
Physical Dimension View: The physical view of an architecture organization is comprised of management structure depth and spectrum of solution activities dimensions. The following illustration depicts these dimensions from vertical and horizontal management structure perspectives.
1. The *Management structure depth* dimension is a vertical management structure perspective, which is expressed by its management hierarchy depth, and it can be measured from the architecture leadership layer down to the solution activity layer. The depth level of a management hierarchy can influence management control mechanisms of an organization, such as management delegation methods, reporting system, communication processes, propagation, and distribution of labor systems.

Deep hierarchies are structures that are comprised of multiple levels of management within single layers. Thick management layers are depicted in the following diagram.
Deep hierarchies are mainly suitable for architecture centric organizations – fully horizontally positioned. Their structure depth, wealth of management resources, and multi-level reporting system can be leveraged by various silo organizations in the enterprise. Conversely, such configuration can cause communication difficulties, challenges with the reporting system, and issues with delegation methods in an architecture organization.

Thin management layers can be generated by shallow structure configurations, which reduce management levels in layers. The following diagram depicts previous hierarchy information.
Shallow hierarchy configuration can be achieved by combining management layers (as shown in the previous illustration – practice and discipline layers were collapsed into one layer); alternatively, reduction of management levels in layers can achieve the same goal. This arrangement can reduce management overhead, improve communication protocols, and enable good reporting systems in an architecture organization.

Thin layers are mainly suited to hands-on type of environments in which the emphasis is on strengthening the solution management layer because of demand for rapid deliverables, swift architecture operations, and better labor division and distribution processes.

Deep and shallow hierarchies should be periodically aligned with the overall architecture strategy of an architecture organization.

2. The *spectrum of solution activities* dimension is a horizontal management structure perspective and it depends on its solution activity layer. This range provides the necessary hands-on coverage to execute solution activities and is one of the most important contributors to the success of an architecture organization.

Technical requirements are propagated down to the *solution activity management* layer and transformed into technical specifications, which may influence the spectrum of the *solution activity* layer by adding or eliminating solution activity roles and responsibilities. The *spectrum of solution activities* dimension of an organizational management structure should be dynamically aligned with propagated labor requirements by constantly adjusting the spectrum of its solution activity management layer. This chain reaction starts from business and technological requirements early in
the process and continues to affect the architecture organization down to its lower chain of command.

- Human Dimension View: The *human dimension* view of an organizational structure is the overall composition of individual qualities, the blend of intellect, and the combination of skill sets personnel bring to the table. Two dimensions are affiliated with this view, the *role composition* and the *human factor*. They are the most dynamic entities in an organization. They constantly change the backbone and the stability of the management structure, influence the total skill level, impact the execution capacity of its management, and contribute the most to the survival of an architecture organization.

1. *Role composition* dimension is the combination of architecture roles and their skill sets in an organization. The art of integrating these roles, their expertise, and knowledge contributes to the foundation of an architecture management structure. This mix of skills influences the capacity of an organization to understand requirements, translate them into specifications, provide solutions, and facilitate problem-solving initiatives. Role creation techniques such as *role models* and *role patterns*, discussed in ‘architecture roles’ is the key to the success of personnel placement and overall skill matching in an organization.

2. The *human factor* dimension is the mix of personal qualities in an organization, such as attitude, behavior, knowledge, intelligence, performance, self-discipline, efficiency, communication, leadership, self-motivation, focus, attention, and problem solving capabilities. This important blend has an immense impact on the overall strategy of an organization, day-to-day operations, management execution, and the competence of its operations.

- Management & Execution View: The *management structure depth* and the *human factor* are the two dimensions that represent this view. Execution is about performance and accomplishments and is fundamental to enterprise organization survival and existence. The management hierarchy formation is a configuration that can influence management capacity, its capabilities to understand problems, provide solutions, execute, manage staff, divide labor, and distribute it. The human factor that needs to operate from within such a construction should be capable to quickly adapt to changes by understanding new rules of operations, developing and improving communication protocols, and creating an efficient reporting system.

- Subject Matter Expertise View: This view provides insights of how proficient an organization is and how it is prepared to deal with problems and requirements from a skill set and knowledge perspective. *Role composition* and *spectrum of solution activities* dimensions can provide a quantitative expertise view of an organization as a whole. These combined and collective skills of personnel in an organization can influence the stability and the capability of an architecture organization. Thus, the readiness level should
constantly be challenged and measured for the purpose of providing continuity and contingency plans in case the organizational skill level deepens.

**Management Controls & Protocols**

Any establishment of or changes to a management structure in an organization should be tested and bullet proofed before management assumes its responsibilities on its layers. Testing a structure can be accomplished by walking through a number of use cases, which present real business and/or technological requirements. The following four different management control mechanisms should be challenged when testing:

1. **Communication Protocol:** This protocol controls messaging and information flow between team members in an organization. It is comprised of rules and standards of communication. These regulations should provide guidance and policies on how management, peers, and their subordinates should exchange information:
   - Who should be communicated with
   - What is the order or sequence of communications
   - How often
   - When should communications occur
   - What is the communications format
   - How communications return acknowledgment status.

2. **Management Reporting System:** This protocol enables managers and their subordinates to process and execute instructions, directions, commands, and orders. It is a roundtrip and a circular process, which always starts from the top and flows down to the bottom of the management structure. The flow should be streamed back to upper management in the format of confirmation, acknowledgment, work status, and reporting of achievements, accomplishments, challenges, and problems.

3. **Translation, Transformation, and Interpretation of Labor:** Requirements should be translated by architecture management and transformed into an architecture form of functional requirements and specifications for the purpose of execution. The initial crude requirements, which are being propagated down, should become more and more granular as they reach lower management layers.

4. **Division and distribution of Labor:** Each level in the management structure should divide the work and distribute it to its various subordinate members based on its context and suitability. Division and distribution is about expertise and capability matching between the types of labor and the receiving parties in the organization. Management that assumes control over work division and dispatching should be properly positioned in the organization hierarchy.
The following use case illustrates a business requirement for an enterprise reporting system.

**Use Case – Enterprise Reporting**

*Business management has identified needs for enterprise reporting software capabilities to provide ad-hoc financial status and market conditions information, which will facilitate executives to make changes to company’s business strategy.*

The following examples depict an architecture organization, which utilizes management control mechanisms to process and execute the enterprise reporting use case requirements.

Translation, transformation, and division of labor: Architecture leadership receives a use case from the business and translates the work from business requirements to technology requirements.

- The architecture leadership layer obtains the enterprise reporting use case from the business unit and learns more about the problem domain and its detailed requirements.
- The architecture leadership transforms the requirements into four distinguished sets of architecture requirements and units of work, which best fit this organization’s architecture practices.

Management reporting systems and distributions of labor Architecture leadership distributes the labor and requirements information to its four architecture practices:

<table>
<thead>
<tr>
<th>Practice</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Architecture Logistics</td>
<td>Asked to provide reporting infrastructure needs.</td>
</tr>
<tr>
<td>Software Architecture</td>
<td>Commissioned to select reporting products, design, and architect enterprise-reporting solutions.</td>
</tr>
<tr>
<td>Data Architecture</td>
<td>Instructed to provide reporting data strategies and database modeling and design services.</td>
</tr>
<tr>
<td>Architecture Initiative Coordination</td>
<td>Requested to facilitate vendor relationships, be in charge of product negotiations, allocate resources, and provide user and developer training.</td>
</tr>
</tbody>
</table>

The following diagram depicts these concepts.
Communication protocol: After the propagation of labor by the architecture leadership, the architecture practice management layer initiates a number of communications by following communication policies, standards, and best practices.

<table>
<thead>
<tr>
<th>Description</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>The software architecture practice management posts resource requests to</td>
<td>The request is accepted and resources are scheduled.</td>
</tr>
<tr>
<td>the architecture initiative coordination practice, asking for staff</td>
<td></td>
</tr>
<tr>
<td>augmentation and for extra hardware and software to help with the</td>
<td></td>
</tr>
<tr>
<td>enterprise reporting initiative.</td>
<td></td>
</tr>
<tr>
<td>The software architecture practice posts a request to the software</td>
<td></td>
</tr>
<tr>
<td>logistics practice to help with reporting product installation and</td>
<td></td>
</tr>
<tr>
<td>configuration.</td>
<td></td>
</tr>
<tr>
<td>The software architecture practice posts a request to the data architecture</td>
<td>Request was denied, the data architecture practice management is not</td>
</tr>
<tr>
<td>practice to facilitate with database modeling and reporting data strategies.</td>
<td></td>
</tr>
<tr>
<td>The data architecture practice posts a request to the architecture</td>
<td>The request is put on hold because of lack of training resources.</td>
</tr>
<tr>
<td>initiative coordination practice to help with reporting product training.</td>
<td></td>
</tr>
<tr>
<td>The software architecture practice management requests the architecture</td>
<td>Extra budget was approved.</td>
</tr>
<tr>
<td>leadership for extra budget.</td>
<td></td>
</tr>
<tr>
<td>The architecture initiative coordination practice requests the architecture</td>
<td>The request was denied.</td>
</tr>
<tr>
<td>leadership for training resources budget.</td>
<td></td>
</tr>
<tr>
<td>The software architecture management practice instructs the product</td>
<td></td>
</tr>
<tr>
<td>evaluation and selection discipline to start the product evaluation process.</td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>Action</td>
</tr>
<tr>
<td>-----------------------------------------------------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td>The <em>software architecture</em> management practice commissions the <em>design and architecture</em> discipline to provide an enterprise reporting conceptual design.</td>
<td></td>
</tr>
<tr>
<td>The <em>architecture logistics</em> practice sends a readiness alert message to its <em>configuration management</em> discipline.</td>
<td></td>
</tr>
</tbody>
</table>
The following diagram illustrates communications protocol in action, based on pre-written script policies and best practices.

**Communications Protocol Example**

Architecture management reporting system: Each one of the architecture practices further breaks down the work, provides guidance and methodologies, and distributes it to its subordinate architecture discipline layer. In this example the architecture logistics management practice instructs its discipline management to facilitate the enterprise reporting initiative, create requirements, and pass them on to their solutions activities management.

<table>
<thead>
<tr>
<th>Discipline</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Infrastructure Design</td>
<td>Assigned to develop requirements for designing the enterprise reporting deployment infrastructure.</td>
</tr>
<tr>
<td>Product Adaptation</td>
<td>Instructed to provide reporting product adaptation plans.</td>
</tr>
<tr>
<td>Product Management</td>
<td>Asked to manage and upgrade the reporting product and ensure product scalability and availability.</td>
</tr>
<tr>
<td>Infrastructure Security</td>
<td>Assigned to provide requirements for the development of reporting security infrastructure.</td>
</tr>
<tr>
<td>Configuration Management</td>
<td>Commissioned to provide reporting product installation and configuration.</td>
</tr>
</tbody>
</table>
ENTERPRISE PERSPECTIVE

A perspective of an enterprise can be understood, quantified, and visualized by the spectrum of its architecture organization disciplines and the depth of its architecture practices structure.
Appendix A
Architecture Organization Practices, Disciplines and Solution Activities Example