

The Open Group Standard

TOGAF® Standard — ADM Techniques

The Open Group

Copyright © 2005-2022, The Open Group

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission of the copyright owners.

Any use of this publication for commercial purposes is subject to the terms of the Annual Commercial License relating to it. For further information, see www.opengroup.org/legal/licensing.

The Open Group Standard
TOGAF® Standard — ADM Techniques
ISBN: 1-947754-90-4
Document Number: C220

Published by The Open Group, 2005-2022.

Any comments relating to the material contained in this document may be submitted by email to:
OGspecs@opengroup.org

Contents

Chapter	1	Introduction	1
Chapter	2	Architecture Principles	3
	2.1	Introduction	3
	2.2	Characteristics of Architecture Principles	4
	2.3	Components of Architecture Principles	4
	2.4	Developing Architecture Principles	5
	2.4.1	Qualities of Principles	5
	2.5	Applying Architecture Principles	6
	2.6	Example Set of Architecture Principles	7
	2.6.1	Business Principles	7
	2.6.2	Data Principles	11
	2.6.3	Application Principles	16
	2.6.4	Technology Principles	17
Chapter	3	Stakeholder Management	21
	3.1	Introduction	21
	3.2	Approach to Stakeholder Management	21
	3.3	Steps in the Stakeholder Management Process	22
	3.3.1	Identify Stakeholders	22
	3.3.2	Classify Stakeholder Positions	24
	3.3.3	Determine Stakeholder Management Approach	24
	3.3.4	Tailor Engagement Deliverables	25
	3.4	Template Stakeholder Map	25
Chapter	4	Architecture Patterns	39
	4.1	Introduction	39
	4.1.1	Background	39
	4.1.2	Content of a Pattern	40
	4.1.3	Terminology	41
	4.2	Some Pattern Resources	42
Chapter	5	Gap Analysis	45
	5.1	Introduction	45
	5.2	Suggested Steps	46
	5.3	Example	46
Chapter	6	Migration Planning Techniques	49
	6.1	Implementation Factor Catalog	49
	6.2	Consolidated Gaps, Solutions, & Dependencies Matrix	50
	6.3	Architecture Definition Increments Table	50
	6.4	Transition Architecture State Evolution Table	51

	6.5	Business Value Assessment Technique	52
Chapter	7	Interoperability Requirements	53
	7.1	Overview.....	53
	7.2	Defining Interoperability	53
	7.3	Enterprise Operating Model	55
	7.4	Refining Interoperability	55
	7.5	Determining Interoperability Requirements	56
	7.6	Reconciling Interoperability Requirements with Potential Solutions	58
Chapter	8	Business Transformation Readiness Assessment.....	59
	8.1	Introduction.....	59
	8.1.1	Business Transformation Enablement Program (BTEP).....	60
	8.2	Determine Readiness Factors	60
	8.3	Present Readiness Factors	62
	8.4	Assess Readiness Factors	63
	8.4.1	Readiness Factor Vision.....	64
	8.4.2	Readiness Factor Rating	64
	8.4.3	Readiness Factor Risks & Actions.....	65
	8.5	Readiness and Migration Planning	66
	8.6	Marketing the Implementation Plan	66
	8.7	Conclusion.....	66
Chapter	9	Risk Management.....	67
	9.1	Introduction.....	67
	9.2	Risk Classification	68
	9.3	Risk Identification	68
	9.4	Initial Risk Assessment.....	68
	9.5	Risk Mitigation and Residual Risk Assessment.....	70
	9.6	Conduct Residual Risk Assessment.....	70
	9.7	Risk Monitoring and Governance (Phase G).....	70
	9.8	Summary	71
Chapter	10	Architecture Alternatives and Trade-Offs.....	73
	10.1	Concept	73
	10.2	Method.....	73
	10.2.1	Criteria	74
	10.2.2	Identify Alternatives	75
	10.2.3	Choose from Alternatives and Define in Detail.....	76
		Index	77
List of Figures			
	3-1	Sample Stakeholders and Categories	23
	3-2	Stakeholder Power Grid.....	24
	5-1	Gap Analysis Example	46
	6-1	Implementation Factor Catalog.....	49
	6-2	Consolidated Gaps, Solutions, and Dependencies Matrix	50

Contents

6-3	Architecture Definition Increments Table	50
6-4	Transition Architecture State Evolution Table	51
6-5	Sample Project Assessment with Respect to Business Value and Risk.....	52
7-1	Business Information Interoperability Matrix.....	57
7-2	Information Systems Interoperability Matrix	57
8-1	Business Transformation Readiness Assessment — Maturity Model	63
8-2	Summary Table of Business Transformation Readiness Assessment	64
9-1	Risk Classification Scheme	69
9-2	Sample Risk Identification and Mitigation Assessment Worksheet	70
10-1	Architecture Trade-Off Method	73

List of Tables

2-1	Recommended Format for Defining Principles	4
3-1	Example Stakeholder Analysis	24

Preface

The Open Group

The Open Group is a global consortium that enables the achievement of business objectives through technology standards. With more than 870 member organizations, we have a diverse membership that spans all sectors of the technology community — customers, systems and solutions suppliers, tool vendors, integrators and consultants, as well as academics and researchers.

The mission of The Open Group is to drive the creation of Boundaryless Information Flow™ achieved by:

- Working with customers to capture, understand, and address current and emerging requirements, establish policies, and share best practices
- Working with suppliers, consortia, and standards bodies to develop consensus and facilitate interoperability, to evolve and integrate specifications and open source technologies
- Offering a comprehensive set of services to enhance the operational efficiency of consortia
- Developing and operating the industry's premier certification service and encouraging procurement of certified products

Further information on The Open Group is available at www.opengroup.org.

The Open Group publishes a wide range of technical documentation, most of which is focused on development of Standards and Guides, but which also includes white papers, technical studies, certification and testing documentation, and business titles. Full details and a catalog are available at www.opengroup.org/library.

The TOGAF® Standard

The TOGAF Standard is an open, industry consensus framework for Enterprise Architecture.

It is a foundational framework, which means that it is applicable to the development of any kind of architecture in any context. This foundational framework is supplemented by The Open Group TOGAF Library,¹ an extensive and growing portfolio of guidance material, providing practical guidance in the application of the TOGAF framework in specific contexts.

The TOGAF Documentation

The TOGAF documentation consists of a set of documents:

- The TOGAF Standard, which describes the generally applicable approach to Enterprise and IT Architecture
- The TOGAF Library, a portfolio of additional guidance material, which supports the practical application of the TOGAF approach

1. The TOGAF Library (see www.opengroup.org/togaf-library) is a structured library of resources that support the TOGAF Standard.

Preface

This Document

This is the TOGAF Standard — ADM Techniques.

Intended Audience

The TOGAF Standard is intended for Enterprise Architects, Business Architects, IT Architects, Data Architects, Systems Architects, Solution Architects, and anyone responsible for the architecture function within an organization.

Acknowledgements

The Open Group is grateful for the contribution of many individuals and organizations in the development of the TOGAF Standard. See the TOGAF Standard — Introduction and Core Concepts for details.

Trademarks

ArchiMate, DirecNet, Making Standards Work, Open O logo, Open O and Check Certification logo, The Open Group, TOGAF, UNIX, UNIXWARE, and the Open Brand X logo are registered trademarks and Boundaryless Information Flow, Build with Integrity Buy with Confidence, Commercial Aviation Reference Architecture, Dependability Through Assuredness, Digital Practitioner Body of Knowledge, DPBoK, EMMM, FACE, the FACE logo, FHIM Profile Builder, the FHIM logo, FPB, Future Airborne Capability Environment, IT4IT, the IT4IT logo, O-AA, O-DEF, O-HERA, O-PAS, Open Agile Architecture, Open FAIR, Open Footprint, Open Process Automation, Open Subsurface Data Universe, Open Trusted Technology Provider, OSDU, Sensor Integration Simplified, SOSA, and the SOSA logo are trademarks of The Open Group.

FICO is a registered trademark of Fair Isaac Corporation in the United States and other countries.

Java is a registered trademark of Oracle and/or its affiliates.

PMBOK is a registered trademark of the Project Management Institute, Inc. which is registered in the United States and other nations.

PRINCE2 is a registered trademark of AXELOS Limited.

The Open Group acknowledges that there may be other company names and products that might be covered by trademark protection and advises the reader to verify them independently.

Referenced Documents

Please refer to the TOGAF Standard — Introduction and Core Concepts: Appendix A for documents referenced in the TOGAF Standard.

Chapter 1: Introduction

This chapter provides an introduction to the guidance provided in the TOGAF Standard — ADM Techniques (this document).

Guidelines included within this document are as follows:

- Architecture Principles (see [Chapter 2](#)) describes principles for the use and deployment of IT resources across the enterprise, and how to develop the set of general rules and guidelines for the architecture being developed
- Stakeholder Management (see [Chapter 3](#)) describes stakeholder management, an important discipline that successful architecture practitioners can use to win support for their projects
- Architecture Patterns (see [Chapter 4](#)) provides guidance on using architectural patterns
- Gap Analysis (see [Chapter 5](#)) describes the technique known as gap analysis; it is widely used in the TOGAF ADM to validate an architecture that is being developed
- Migration Planning Techniques (see [Chapter 6](#)) describes a number of techniques to support migration planning in Phases E and F
- Interoperability Requirements (see [Chapter 7](#)) describes a technique for determining interoperability requirements
- Business Transformation Readiness Assessment (see [Chapter 8](#)) describes a technique for identifying business transformation issues
- Risk Management (see [Chapter 9](#)) describes a technique for managing risk during an architecture/business transformation project
- Architecture Alternatives and Trade-Offs (see [Chapter 10](#)) describes a technique to identify alternative Target Architectures and perform trade-offs between the alternatives

Chapter 2: Architecture Principles

This chapter describes principles for use in the development of an Enterprise Architecture.

2.1 Introduction

Principles are general rules and guidelines, intended to be enduring and seldom amended, that inform and support the way in which an organization sets about fulfilling its mission.

In their turn, principles may be just one element in a structured set of ideas that collectively define and guide the organization, from values through to actions and results.

Depending on the organization, principles may be established within different domains and at different levels. Two key domains inform the development and utilization of architecture:

- **Enterprise Principles** provide a basis for decision-making throughout an enterprise, and inform how the organization sets about fulfilling its mission

Such principles are commonly found as a means of harmonizing decision-making across an organization. In particular, they are a key element in a successful Architecture Governance strategy (see the TOGAF Standard — Enterprise Architecture Capability and Governance).

Within the broad domain of enterprise principles, it is common to have subsidiary principles within a business or organizational unit. Examples include IT, HR, domestic operations, or overseas operations. These principles provide a basis for decision-making within the subsidiary domain and will inform architecture development within the domain. Care must be taken to ensure that the principles used to inform architecture development align to the organizational context of the Architecture Capability.

- **Architecture Principles** are a set of principles that relate to architecture work

They reflect a level of consensus across the enterprise, and embody the spirit and thinking of existing enterprise principles. Architecture Principles govern the architecture process, affecting the development, maintenance, and use of the Enterprise Architecture.

It is common to have sets of principles form a hierarchy, in that segment principles will be informed by, and elaborate on, the principles at the enterprise level. Architecture Principles will be informed and constrained by enterprise principles.

Architecture Principles may restate other enterprise guidance in terms and form that effectively guide architecture development.

The remainder of this section deals exclusively with Architecture Principles.

2.2 Characteristics of Architecture Principles

Architecture Principles define the underlying general rules and guidelines for the use and deployment of all IT resources and assets across the enterprise. They reflect a level of consensus among the various elements of the enterprise, and form the basis for making future IT decisions.

Each Architecture Principle should be clearly related back to the business objectives and key architecture drivers.

2.3 Components of Architecture Principles

It is useful to have a standard way of defining principles. In addition to a definition statement, each principle should have associated rationale and implications statements, both to promote understanding and acceptance of the principles themselves, and to support the use of the principles in explaining and justifying why specific decisions are made.

A recommended template is given in [Table 2-1](#).

Name	Should both represent the essence of the rule as well as be easy to remember. Specific technology platforms should not be mentioned in the name or statement of a principle. Avoid ambiguous words in the Name and in the Statement such as: "support", "open", "consider", and for lack of good measure the word "avoid", itself, be careful with "manage(ment)", and look for unnecessary adjectives and adverbs (fluff).
Statement	Should succinctly and unambiguously communicate the fundamental rule. For the most part, the principles statements for managing information are similar from one organization to the next. It is vital that the principles statement is unambiguous.
Rationale	Should highlight the business benefits of adhering to the principle, using business terminology. Point to the similarity of information and technology principles to the principles governing business operations. Also describe the relationship to other principles, and the intentions regarding a balanced interpretation. Describe situations where one principle would be given precedence or carry more weight than another for making a decision.
Implications	Should highlight the requirements, both for the business and IT, for carrying out the principle — in terms of resources, costs, and activities/tasks. Although it may often be apparent that current systems, standards, or practices would be incongruent with the principle upon adoption, context will drive the degree of scope. The impact to the business and consequences of adopting a principle should be clearly stated. The reader should readily discern the answer to: "How does this affect me?". It is important not to oversimplify, trivialize, or judge the merit of the impact. Some of the implications will be identified as potential impacts only, and may be speculative rather than fully analyzed.

Table 2-1 Recommended Format for Defining Principles

An example set of Architecture Principles following this template is given in [Section 2.6](#).

2.4 Developing Architecture Principles

Architecture Principles are typically developed by the Enterprise Architects, in conjunction with the key stakeholders, and are approved by the Architecture Board.

Architecture Principles will be informed by principles at the enterprise level, if they exist.

Architecture Principles must be clearly traceable and clearly articulated to guide decision-making. They are chosen so as to ensure alignment of the architecture and implementation of the Target Architecture with business strategies and visions.

Specifically, the development of Architecture Principles is typically influenced by the following:

- **Enterprise mission and plans:** the mission, plans, and organizational infrastructure of the enterprise
- **Enterprise strategic initiatives:** the characteristics of the enterprise — its strengths, weaknesses, opportunities, and threats — and its current enterprise-wide initiatives (such as process improvement and quality management)
- **External constraints:** market factors (time-to-market imperatives, customer expectations, etc.); existing and potential legislation
- **Current systems and technology:** the set of information resources deployed within the enterprise, including systems documentation, equipment inventories, network configuration diagrams, policies, and procedures
- **Emerging industry trends:** predictions about economic, political, technical, and market factors that influence the enterprise environment

2.4.1 Qualities of Principles

Merely having a written statement that is called a principle does not mean that the principle is good, even if everyone agrees with it.

A good set of principles will be founded in the beliefs and values of the organization and expressed in language that the business understands and uses. Principles should be few in number, future-oriented, and endorsed and championed by senior management. They provide a firm foundation for making architecture and planning decisions, framing policies, procedures, and standards, and supporting resolution of contradictory situations. A poor set of principles will quickly become disused, and the resultant architectures, policies, and standards will appear arbitrary or self-serving, and thus lack credibility. Essentially, principles drive behavior.

There are five criteria that distinguish a good set of principles:

- **Understandable:** the underlying tenets can be quickly grasped and understood by individuals throughout the organization

The intention of the principle is clear and unambiguous, so that violations, whether intentional or not, are minimized.

- **Robust:** enable good quality decisions about architectures and plans to be made, and enforceable policies and standards to be created

Each principle should be sufficiently definitive and precise to support consistent decision-making in complex, potentially controversial situations.

- **Complete:** every potentially important principle governing the management of information and technology for the organization is defined — the principles cover every situation perceived

- **Consistent:** strict adherence to one principle may require a loose interpretation of another principle

The set of principles must be expressed in a way that allows a balance of interpretations. Principles should not be contradictory to the point where adhering to one principle would violate the spirit of another. Every word in a principle statement should be carefully chosen to allow consistent yet flexible interpretation.

- **Stable:** principles should be enduring, yet able to accommodate changes

An amendment process should be established for adding, removing, or altering principles after they are ratified initially.

2.5 Applying Architecture Principles

Architecture Principles are used to capture the fundamental truths about how the enterprise will use and deploy IT resources and assets. The principles are used in a number of different ways:

1. To provide a framework within which the enterprise can start to make conscious decisions about Enterprise Architecture and projects that implement the target Enterprise Architecture
2. As a guide to establishing relevant evaluation criteria, thus exerting strong influence on the selection of products, solutions, or solution architectures in the later stages of managing compliance to the Enterprise Architecture
3. As drivers for defining the functional requirements of the architecture
4. As an input to assessing both existing implementations and the strategic portfolio, for compliance with the defined architectures; these assessments will provide valuable insights into the transition activities needed to implement an architecture, in support of business goals and priorities
5. The Rationale statements within an Architecture Principle highlight the business value of implementations consistent with the principle and provide guidance for difficult decisions with conflicting drivers or objectives
6. The Implications statements within an Architecture Principle provide an outline of the key tasks, resources, and potential costs to the enterprise of following the principle; they also provide valuable inputs to future transition initiative and planning activities
7. Support the Architecture Governance activities in terms of:
 - Providing a "back-stop" for the standard Architecture Compliance assessments where some interpretation is allowed or required
 - Supporting the decision to initiate a dispensation request where the implications of a particular architecture amendment cannot be resolved within local operating procedure

Principles may be inter-related, and need to be applied as a set.

Principles will sometimes compete; for example, the principles of "accessibility" and "security"

tend towards conflicting decisions. Each principle must be considered in the context of "all other things being equal".

At times a decision will be required as to which principle will take precedence on a particular issue. The rationale for such decisions should always be documented.

A common reaction on first reading of a principle is "this is obvious and does not need to be documented". The fact that a principle seems self-evident does not mean that the guidance in a principle is followed. Having principles that appear obvious helps ensure that decisions actually follow the desired outcome.

Although specific penalties are not prescribed in a declaration of principles, violations of principles generally cause operational problems and inhibit the ability of the organization to fulfil its mission.

2.6 Example Set of Architecture Principles

Too many principles can reduce the flexibility of the architecture. Many organizations prefer to define only high-level principles, and to limit the number to between 10 and 20.

The following example illustrates both the typical content of a set of Architecture Principles, and the recommended format for defining them, as explained above.

2.6.1 Business Principles

Principle 1: Primacy of Principles

Statement: These principles of information management apply to all organizations within the enterprise.

Rationale: The only way we can provide a consistent and measurable level of quality information to decision-makers is if all organizations abide by the principles.

Implications:

- Without this principle, exclusions, favoritism, and inconsistency would rapidly undermine the management of information
- Information management initiatives will not begin until they are examined for compliance with the principles
- A conflict with a principle will be resolved by changing the framework of the initiative

Principle 2: Maximize Benefit to the Enterprise

Statement: Information management decisions are made to provide maximum benefit to the enterprise as a whole.

Rationale: This principle embodies "service above self". Decisions made from an enterprise-wide perspective have greater long-term value than decisions made from any particular organizational perspective. Maximum return on investment requires information management decisions to adhere to enterprise-wide drivers and priorities. No minority group will detract from the benefit of the whole. However, this principle will not preclude any minority group from getting its job done.

- Implications:
- Achieving maximum enterprise-wide benefit will require changes in the way we plan and manage information — technology alone will not bring about this change
 - Some organizations may have to concede their own preferences for the greater benefit of the entire enterprise
 - Where feasible, application development priorities must be established by the entire enterprise for the entire enterprise
 - Applications components should be shared across organizational boundaries
 - Information management initiatives should be conducted in accordance with the enterprise plan
- Individual organizations should pursue information management initiatives which conform to the blueprints and priorities established by the enterprise. The plan will be changed as needed.
- As needs arise, priorities must be adjusted; a forum with comprehensive enterprise representation should make these decisions

Principle 3: Information Management is Everybody's Business

- Statement: All organizations in the enterprise participate in information management decisions needed to accomplish business objectives.
- Rationale: Information users are the key stakeholders, or customers, in the application of technology to address a business need. In order to ensure information management is aligned with the business, all organizations in the enterprise must be involved in all aspects of the information environment. The business experts from across the enterprise and the technical staff responsible for developing and sustaining the information environment need to come together as a team to jointly define the goals and objectives of IT.
- Implications:
- To operate as a team, every stakeholder, or customer, will need to accept responsibility for developing the information environment
 - Commitment of resources will be required to implement this principle

Principle 4: Business Continuity

- Statement: Enterprise operations are maintained in spite of system interruptions.
- Rationale: As system operations become more pervasive, we become more dependent on them; therefore, we must consider the reliability of such systems throughout their design and use. Business premises throughout the enterprise must be provided with the capability to continue operations regardless of external events. Hardware failure, natural disasters, and data corruption should not be allowed to disrupt or stop enterprise activities. The enterprise must be capable of operating on alternative information delivery mechanisms.
- Implications:
- Dependency on shared system applications mandates that the risks of business interruption must be established in advance and managed
- Management includes but is not limited to periodic reviews, testing for

vulnerability and exposure, or designing mission-critical services to ensure business continuity through redundant or alternative capabilities.

- Recoverability, redundancy, and maintainability should be addressed at the time of design
- Applications must be assessed for criticality and impact on the enterprise mission, in order to determine what level of continuity is required and what corresponding recovery plan is necessary

Principle 5: Common Use Applications

Statement: Development of applications used across the enterprise is preferred over the development of similar or duplicative applications which are only provided to a particular organization.

Rationale: Duplicative capability is expensive and proliferates conflicting data.

Implications:

- Organizations which depend on a capability which does not serve the entire enterprise must change over to the replacement enterprise-wide capability; this will require establishment of and adherence to a policy requiring this
- Organizations will not be allowed to develop capabilities for their own use which are similar/duplicative of enterprise-wide capabilities; in this way, expenditures of scarce resources to develop essentially the same capability in marginally different ways will be reduced
- Data and information used to support enterprise decision-making will be standardized to a much greater extent than previously

This is because the smaller, organizational capabilities which produced different data (which was not shared among other organizations) will be replaced by enterprise-wide capabilities. The impetus for adding to the set of enterprise-wide capabilities may well come from an organization making a convincing case for the value of the data/information previously produced by its organizational capability, but the resulting capability will become part of the enterprise-wide system, and the data it produces will be shared across the enterprise.

Principle 6: Service Orientation

Statement: The architecture is based on a design of services which mirror real-world business activities comprising the enterprise (or inter-enterprise) business processes.

Rationale: Service orientation delivers enterprise agility and Boundaryless Information Flow.

Implications:

- Service representation utilizes business descriptions to provide context (i.e., business process, goal, rule, policy, service interface, and service component) and implements services using service orchestration
- Service orientation places unique requirements on the infrastructure, and implementations should use open standards to realize interoperability and location transparency

- Implementations are environment-specific; they are constrained or enabled by context and must be described within that context
- Strong governance of service representation and implementation is required
- A "Litmus Test", which determines a "good service", is required

Principle 7: Compliance with Law

Statement:	Enterprise information management processes comply with all relevant laws, policies, and regulations.
Rationale:	Enterprise policy is to abide by laws, policies, and regulations. This will not preclude business process improvements that lead to changes in policies and regulations.
Implications:	<ul style="list-style-type: none"> ■ The enterprise must be mindful to comply with laws, regulations, and external policies regarding the collection, retention, and management of data ■ Education and access to the rules <p>Efficiency, need, and common sense are not the only drivers. Changes in the law and changes in regulations may drive changes in our processes or applications.</p>

Principle 8: IT Responsibility

Statement:	The IT organization is responsible for owning and implementing IT processes and infrastructure that enable solutions to meet user-defined requirements for functionality, service levels, cost, and delivery timing.
Rationale:	Effectively align expectations with capabilities and costs so that all projects are cost-effective. Efficient and effective solutions have reasonable costs and clear benefits.
Implications:	<ul style="list-style-type: none"> ■ A process must be created to prioritize projects ■ The IT function must define processes to manage business unit expectations ■ Data, application, and technology models must be created to enable integrated quality solutions and to maximize results

Principle 9: Protection of Intellectual Property

Statement:	The enterprise's Intellectual Property (IP) must be protected. This protection must be reflected in the IT architecture, implementation, and governance processes.
Rationale:	A major part of an enterprise's IP is hosted in the IT domain.
Implications:	<ul style="list-style-type: none"> ■ While protection of IP assets is everybody's business, much of the actual protection is implemented in the IT domain — even trust in non-IT processes can be managed by IT processes (email, mandatory notes, etc.)

- A security policy, governing human and IT actors, will be required that can substantially improve protection of IP; this must be capable of both avoiding compromises and reducing liabilities
- Resources on such policies can be found at the SANS Institute (refer to www.sans.org/security-resources/policies)

2.6.2 Data Principles

Principle 10: Data is an Asset

Statement:	Data is an asset that has value to the enterprise and is managed accordingly.
Rationale:	Data is a valuable corporate resource; it has real, measurable value. In simple terms, the purpose of data is to aid decision-making. Accurate, timely data is critical to accurate, timely decisions. Most corporate assets are carefully managed, and data is no exception. Data is the foundation of our decision-making, so we must also carefully manage data to ensure that we know where it is, can rely upon its accuracy, and can obtain it when and where we need it.
Implications:	<ul style="list-style-type: none"> ■ This is one of three closely-related principles regarding data: data is an asset; data is shared; and data is easily accessible <p>The implication is that there is an education task to ensure that all organizations within the enterprise understand the relationship between value of data, sharing of data, and accessibility to data.</p> <ul style="list-style-type: none"> ■ Stewards must have the authority and means to manage the data for which they are accountable ■ We must make the cultural transition from "data ownership" thinking to "data stewardship" thinking ■ The role of data steward is critical because obsolete, incorrect, or inconsistent data could be passed to enterprise personnel and adversely affect decisions across the enterprise ■ Part of the role of data steward, who manages the data, is to ensure data quality <p>Procedures must be developed and used to prevent and correct errors in the information and to improve those processes that produce flawed information. Data quality will need to be measured and steps taken to improve data quality — it is probable that policy and procedures will need to be developed for this as well.</p> <ul style="list-style-type: none"> ■ A forum with comprehensive enterprise-wide representation should decide on process changes suggested by the steward ■ Since data is an asset of value to the entire enterprise, data stewards accountable for properly managing the data must be assigned at the enterprise level

Principle 11: Data is Shared

Statement: Users have access to the data necessary to perform their duties; therefore, data is shared across enterprise functions and organizations.

Rationale: Timely access to accurate data is essential to improving the quality and efficiency of enterprise decision-making. It is less costly to maintain timely, accurate data in a single application, and then share it, than it is to maintain duplicative data in multiple applications. The enterprise holds a wealth of data, but it is stored in hundreds of incompatible stovepipe databases. The speed of data collection, creation, transfer, and assimilation is driven by the ability of the organization to efficiently share these islands of data across the organization.

Shared data will result in improved decisions since we will rely on fewer (ultimately one virtual) sources of more accurate and timely managed data for all of our decision-making. Electronically shared data will result in increased efficiency when existing data entities can be used, without re-keying, to create new entities.

Implications:

- This is one of three closely-related principles regarding data: data is an asset; data is shared; and data is easily accessible

The implication is that there is an education task to ensure that all organizations within the enterprise understand the relationship between value of data, sharing of data, and accessibility to data.

- To enable data sharing we must develop and abide by a common set of policies, procedures, and standards governing data management and access for both the short and the long term
- For the short term, to preserve our significant investment in legacy systems, we must invest in software capable of migrating legacy system data into a shared data environment
- We will also need to develop standard data models, data elements, and other metadata that defines this shared environment and develop a repository system for storing this metadata to make it accessible
- For the long term, as legacy systems are replaced, we must adopt and enforce common data access policies and guidelines for new application developers to ensure that data in new applications remains available to the shared environment and that data in the shared environment can continue to be used by the new applications
- For both the short term and the long term we must adopt common methods and tools for creating, maintaining, and accessing the data shared across the enterprise
- Data sharing will require a significant cultural change
- This principle of data sharing will continually "bump up against" the principle of data security — under no circumstances will the data sharing principle cause confidential data to be compromised

- Data made available for sharing will have to be relied upon by all users to execute their respective tasks

This will ensure that only the most accurate and timely data is relied upon for decision-making. Shared data will become the enterprise-wide "virtual single source" of data.

Principle 12: Data is Accessible

Statement: Data is accessible for users to perform their functions.

Rationale: Wide access to data leads to efficiency and effectiveness in decision-making, and affords a timely response to information requests and service delivery. Using information must be considered from an enterprise perspective to allow access by a wide variety of users. Staff time is saved and consistency of data is improved.

Implications:

- This is one of three closely-related principles regarding data: data is an asset; data is shared; and data is easily accessible

The implication is that there is an education task to ensure that all organizations within the enterprise understand the relationship between value of data, sharing of data, and accessibility to data.

- Accessibility involves the ease with which users obtain information
- The way information is accessed and displayed must be sufficiently adaptable to meet a wide range of enterprise users and their corresponding methods of access
- Access to data does not constitute understanding of the data — personnel should take caution not to misinterpret information
- Access to data does not necessarily grant the user access rights to modify or disclose the data

This will require an education process and a change in the organizational culture, which currently supports a belief in "ownership" of data by functional units.

Principle 13: Data Trustee

Statement: Each data element has a trustee accountable for data quality.

Rationale: One of the benefits of an architected environment is the ability to share data (e.g., text, video, sound, etc.) across the enterprise. As the degree of data sharing grows and business units rely upon common information, it becomes essential that only the data trustee makes decisions about the content of data. Since data can lose its integrity when it is entered multiple times, the data trustee will have sole responsibility for data entry which eliminates redundant human effort and data storage resources.

Note: A trustee is different than a steward — a trustee is responsible for accuracy and currency of the data, while responsibilities of a steward may be broader and include data standardization and definition tasks.

- Implications:
- Real trusteeship dissolves the data "ownership" issues and allows the data to be available to meet all users' needs
This implies that a cultural change from data "ownership" to data "trusteeship" may be required.
 - The data trustee will be responsible for meeting quality requirements levied upon the data for which the trustee is accountable
 - It is essential that the trustee has the ability to provide user confidence in the data based upon attributes such as "data source"
 - It is essential to identify the true source of the data in order that the data authority can be assigned this trustee responsibility
This does not mean that classified sources will be revealed nor does it mean the source will be the trustee.
 - Information should be captured electronically once and immediately validated as close to the source as possible
Quality control measures must be implemented to ensure the integrity of the data.
 - As a result of sharing data across the enterprise, the trustee is accountable and responsible for the accuracy and currency of their designated data element(s) and, subsequently, must then recognize the importance of this trusteeship responsibility

Principle 14: Common Vocabulary and Data Definitions

- Statement: Data is defined consistently throughout the enterprise, and the definitions are understandable and available to all users.
- Rationale: The data that will be used in the development of applications must have a common definition throughout the Headquarters to enable sharing of data. A common vocabulary will facilitate communications and enable dialog to be effective. In addition, it is required to interface systems and exchange data.
- Implications:
- We are lulled into thinking that this issue is adequately addressed because there are people with "data administration" job titles and forums with charters implying responsibility
Significant additional energy and resources must be committed to this task. It is key to the success of efforts to improve the information environment. This is separate from but related to the issue of data element definition, which is addressed by a broad community — this is more like a common vocabulary and definition.
 - The enterprise must establish the initial common vocabulary for the business; the definitions will be used uniformly throughout the enterprise
 - Whenever a new data definition is required, the definition effort will be co-ordinated and reconciled with the corporate "glossary" of data descriptions
The enterprise data administrator will provide this co-ordination.

- Ambiguities resulting from multiple parochial definitions of data must give way to accepted enterprise-wide definitions and understanding
- Multiple data standardization initiatives need to be co-ordinated
- Functional data administration responsibilities must be assigned

Principle 15: Data Security

Statement: Data is protected from unauthorized use and disclosure. In addition to the traditional aspects of national security classification, this includes, but is not limited to, protection of pre-decisional, sensitive, source selection-sensitive, and proprietary information.

Rationale: Open sharing of information and the release of information via relevant legislation must be balanced against the need to restrict the availability of classified, proprietary, and sensitive information.

Existing laws and regulations require the safeguarding of national security and the privacy of data, while permitting free and open access. Pre-decisional (work-in-progress, not yet authorized for release) information must be protected to avoid unwarranted speculation, misinterpretation, and inappropriate use.

Implications:

- Aggregation of data, both classified and not, will create a large target requiring review and de-classification procedures to maintain appropriate control

Data owners and/or functional users must determine whether the aggregation results in an increased classification level. Appropriate policy and procedures will be needed to handle this review and de-classification. Access to information based on a need-to-know policy will force regular reviews of the body of information.

- The current practice of having separate systems to contain different classifications needs to be rethought

Is there a software solution to separating classified and unclassified data? The current hardware solution is unwieldy, inefficient, and costly. It is more expensive to manage unclassified data on a classified system. Currently, the only way to combine the two is to place the unclassified data on the classified system, where it must remain.

- In order to adequately provide access to open information while maintaining secure information, security needs must be identified and developed at the data level, not the application level

- Data security safeguards can be put in place to restrict access to "view only" or "never see"

Sensitivity labeling for access to pre-decisional, decisional, classified, sensitive, or proprietary information must be determined.

- Security must be designed into data elements from the beginning; it cannot be added later

Systems, data, and technologies must be protected from unauthorized access and manipulation. Information at Headquarters must be

safeguarded against inadvertent or unauthorized alteration, sabotage, disaster, or disclosure.

- New policies are needed on managing duration of protection for pre-decisional information and other works-in-progress, in consideration of content freshness

2.6.3 Application Principles

Principle 16: Technology Independence

Statement: Applications are independent of specific technology choices and therefore can operate on a variety of technology platforms.

Rationale: Independence of applications from the underlying technology allows applications to be developed, upgraded, and operated in the most cost-effective and timely way. Otherwise technology, which is subject to continual obsolescence and vendor dependence, becomes the driver rather than the user requirements themselves.

Realizing that every decision made with respect to IT makes us dependent on that technology, the intent of this principle is to ensure that Application Software is not dependent on specific hardware and operating systems software.

- Implications:**
- This principle will require standards which support portability
 - For Commercial Off-The-Shelf (COTS) and Government Off-The-Shelf (GOTS) applications, there may be limited current choices, as many of these applications are technology and platform-dependent
 - Subsystem interfaces will need to be developed to enable legacy applications to interoperate with applications and operating environments developed under the Enterprise Architecture
 - Middleware should be used to decouple applications from specific software solutions
 - As an example, this principle could lead to use of Java[®], and future Java-like protocols, which give a high degree of priority to platform-independence

Principle 17: Ease-of-Use

Statement: Applications are easy to use. The underlying technology is transparent to users, so they can concentrate on tasks at hand.

Rationale: The more a user has to understand the underlying technology, the less productive that user is. Ease-of-use is a positive incentive for use of applications. It encourages users to work within the integrated information environment instead of developing isolated systems to accomplish the task outside of the enterprise's integrated information environment. Most of the knowledge required to operate one system will be similar to others. Training is kept to a minimum, and the risk of using a system improperly is low.

Using an application should be as intuitive as driving a different car.

- Implications:
- Applications will be required to have a common "look-and-feel" and support ergonomic requirements; hence, the common look-and-feel standard must be designed and usability test criteria must be developed
 - Guidelines for user interfaces should not be constrained by narrow assumptions about user location, language, systems training, or physical capability
- Factors such as linguistics, customer physical infirmities (visual acuity, ability to use keyboard/mouse), and proficiency in the use of technology have broad ramifications in determining the ease-of-use of an application.

2.6.4 Technology Principles

Principle 18: Requirements-Based Change

Statement: Only in response to business needs are changes to applications and technology made.

Rationale: This principle will foster an atmosphere where the information environment changes in response to the needs of the business, rather than having the business change in response to IT changes. This is to ensure that the purpose of the information support — the transaction of business — is the basis for any proposed change.

Unintended effects on business due to IT changes will be minimized.

A change in technology may provide an opportunity to improve the business process and, hence, change business needs.

- Implications:
- Changes in implementation will follow full examination of the proposed changes using the Enterprise Architecture
 - There is no funding for a technical improvement or system development unless a documented business need exists
 - Change management processes conforming to this principle will be developed and implemented
 - This principle may bump up against the responsive change principle
- We must ensure the requirements documentation process does not hinder responsive change to meet legitimate business needs. The purpose of this principle is to keep the focus on business, not technology needs — responsive change is also a business need.

Principle 19: Responsive Change Management

- Statement:** Changes to the enterprise information environment are implemented in a timely manner.
- Rationale:** If people are to be expected to work within the enterprise information environment, that information environment must be responsive to their needs.
- Implications:**
- Processes for managing and implementing change must be developed that do not create delays
 - A user who feels a need for change will need to connect with a "business expert" to facilitate explanation and implementation of that need
 - If changes are going to be made, the architecture must be kept updated
 - Adopting this principle might require additional resources
 - This will conflict with other principles (e.g., maximum enterprise-wide benefit, enterprise-wide applications, etc.)

Principle 20: Control Technical Diversity

- Statement:** Technological diversity is controlled to minimize the non-trivial cost of maintaining expertise in and connectivity between multiple processing environments.
- Rationale:** There is a real, non-trivial cost of infrastructure required to support alternative technologies for processing environments. There are further infrastructure costs incurred to keep multiple processor constructs interconnected and maintained.
- Limiting the number of supported components will simplify maintainability and reduce costs.
- The business advantages of minimum technical diversity include: standard packaging of components; predictable implementation impact; predictable valuations and returns; redefined testing; utility status; and increased flexibility to accommodate technological advancements. Common technology across the enterprise brings the benefits of economies of scale to the enterprise. Technical administration and support costs are better controlled when limited resources can focus on this shared set of technology.
- Implications:**
- Policies, standards, and procedures that govern the acquisition of technology must be tied directly to this principle
 - Technology choices will be constrained by the choices available within the technology blueprint
- Procedures for augmenting the acceptable technology set to meet evolving requirements will have to be developed and put in place.
- The technology baseline is not being frozen
- Technology advances are welcomed and will change the technology blueprint when compatibility with the current infrastructure, improvement in operational efficiency, or a required capability has been demonstrated.

Principle 21: Interoperability

Statement: Software and hardware should conform to defined standards that promote interoperability for data, applications, and technology.

Rationale: Standards help ensure consistency, thus improving the ability to manage systems and improve user satisfaction, and protect existing IT investments, thus maximizing return on investment and reducing costs. Standards for interoperability additionally help ensure support from multiple vendors for their products, and facilitate supply chain integration.

Implications:

- Interoperability standards and industry standards will be followed unless there is a compelling business reason to implement a non-standard solution
- A process for setting standards, reviewing and revising them periodically, and granting exceptions must be established
- The existing IT platforms must be identified and documented

Chapter 3: Stakeholder Management

3.1 Introduction

Stakeholder management is an important discipline that successful architecture practitioners can use to win support from others. It helps them ensure that their projects succeed where others fail.

The benefits of successful stakeholder management are that:

- The most powerful stakeholders can be identified early and their input can then be used to shape the architecture; this ensures their support and improves the quality of the models produced
- Support from the more powerful stakeholders will help the engagement win more resources, thus making the architecture engagement more likely to succeed
- By communicating with stakeholders early and frequently, the architecture team can ensure that they fully understand the architecture process, and the benefits of Enterprise Architecture; this means they can support the architecture team more actively when necessary
- The architecture team can more effectively anticipate likely reactions to the architecture models and reports, and can build into the plan the actions that will be needed to capitalize on positive reactions while avoiding or addressing any negative reactions
- The architecture team can identify conflicting or competing objectives among stakeholders early and develop a strategy to resolve the issues arising from them

It is essential in any initiative to identify the individuals and groups within the organization who will contribute to the development of the architecture, identify those that will gain and those that will lose from its introduction, and then develop a strategy for dealing with them.

3.2 Approach to Stakeholder Management

Stakeholder analysis should be used during Phase A (Architecture Vision) to identify the key players in the engagement, and also be updated throughout each phase; different stakeholders may be uncovered as the engagement progresses through into Opportunities & Solutions, Migration Planning, and Architecture Change Management.

Complex architectures are extremely hard to manage, not only in terms of the architecture development process itself, but also in terms of obtaining agreement from the large numbers of stakeholders touched by it.

For example, just as a building architect will create wiring diagrams, floor plans, and elevations to describe different facets of a building to its different stakeholders (electricians, owners, planning

officials), so an Enterprise Architect must create different architecture views of the Business, Information Systems, and Technology Architecture for the stakeholders who have concerns related to these aspects.

The TOGAF Standard specifically identifies this issue throughout the ADM through the following concepts (see the TOGAF Standard — Architecture Content):

- Architecture View
- Architecture Viewpoint
- Concern
- Stakeholder

3.3 Steps in the Stakeholder Management Process

The following sections detail recommended stakeholder management activity.

3.3.1 Identify Stakeholders

Identify the key stakeholders of the Enterprise Architecture.

The first task is to work out who the main Enterprise Architecture stakeholders are. As part of this, think of all the people who are affected by it, who have influence or power over it, or have an interest in its successful or unsuccessful conclusion.

It might include senior executives, project organization roles, client organization roles, system developers, alliance partners, suppliers, IT operations, customers, etc.

When identifying stakeholders there is a danger of concentrating too heavily on the formal structure of an organization as the basis for identification. Informal stakeholder groups may be just as powerful and influential as the formal ones.

Most individuals will belong to more than one stakeholder group, and these groups tend to arise as a result of specific events.

Look at who is impacted by the Enterprise Architecture project:

- Who gains and who loses from this change?
- Who controls change management of processes?
- Who designs new systems?
- Who will make the decisions?
- Who procures IT systems and who decides what to buy?
- Who controls resources?
- Who has specialist skills the project needs?
- Who has influence?

In particular, influencers need to be identified. These will be well respected and moving up, participate in important meetings and committees (look at meeting minutes), know what's going on in the company, be valued by their peers and superiors, and not necessarily be in any formal

position of power.

Although stakeholders may be both organizations and people, ultimately the Enterprise Architecture team will need to communicate with people. It is the correct individual stakeholders within a stakeholder organization that need to be formally identified.

3.3.1.1 Sample Stakeholder Analysis

A sample stakeholder analysis that distinguishes 22 types of stakeholder, in five broad categories, is shown in [Figure 3-1](#). Any particular architecture project may have more, fewer, or different stakeholders; and they may be grouped into more, fewer, or different categories.

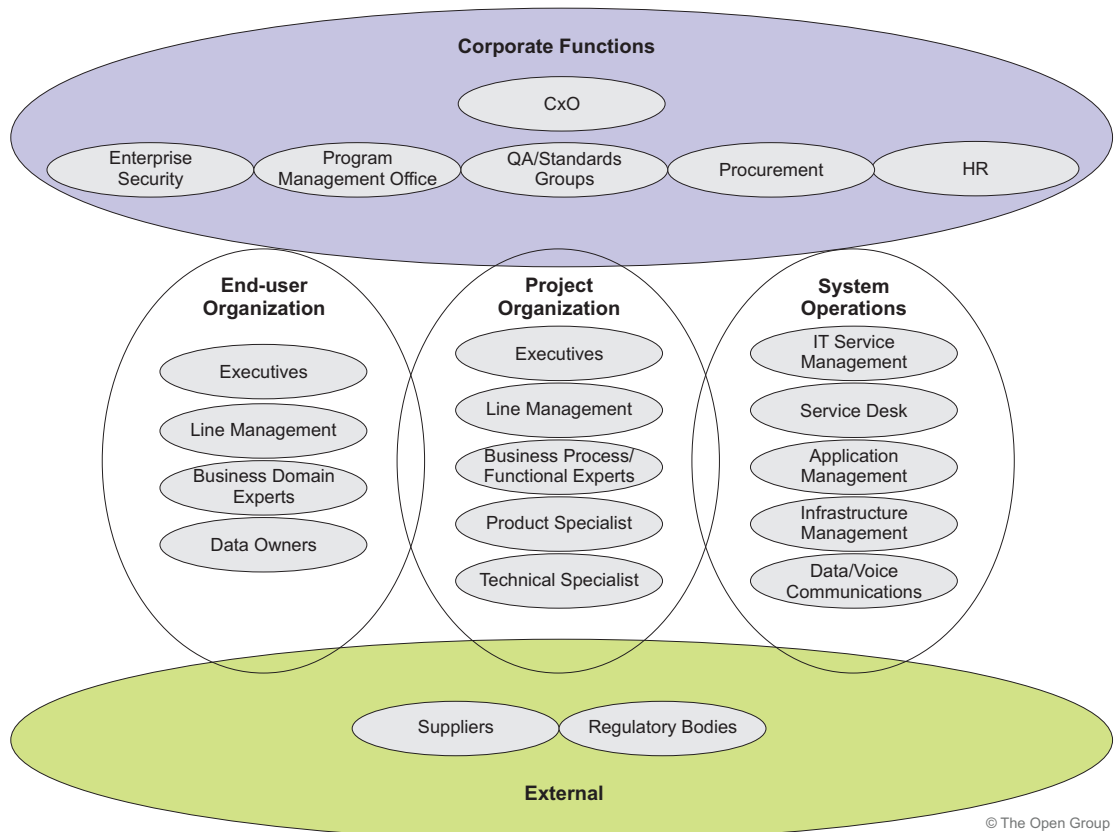


Figure 3-1 Sample Stakeholders and Categories

Consider both the Visible team — those obviously associated with the project/change — and the Invisible team — those who must make a real contribution to the project/change for it to be successful but who are not obviously associated with it (e.g., providers of support services).

3.3.2 Classify Stakeholder Positions

Develop a good understanding of the most important stakeholders and record this analysis for reference and refresh during the project. An example stakeholder analysis is shown in [Table 3-1](#).

Stakeholder Group	Stakeholder	Ability to Disrupt Change	Current Understanding	Required Understanding	Current Commitment	Required Commitment	Required Support
CIO	John Smith	H	M	H	L	M	H
CFO	Jeff Brown	M	M	M	L	M	M

Table 3-1 Example Stakeholder Analysis

It is also important to assess the readiness of each stakeholder to behave in a supportive manner (i.e., demonstrate commitment to the Enterprise Architecture initiative).

This can be done by asking a series of questions:

- Is that person ready to change direction and begin moving towards the Target Architecture? If so, how ready?
- Is that person capable of being a credible advocate or agent of the proposed Enterprise Architecture initiative? If so, how capable?
- How involved is the individual in the Enterprise Architecture initiative? Are they simply an interested observer, or do they need to be involved in the details?
- Has that person made a contractual commitment to the development of the Enterprise Architecture, and its role in the governance of the development of the organization?

Then, for each person whose commitment is critical to ensure success, make a judgment as to their current level of commitment and the desired future level of commitment.

3.3.3 Determine Stakeholder Management Approach

The previous steps identified a long list of people and organizations that are affected by the Enterprise Architecture project.

Some of these may have the power either to block or advance. Some may be interested in what the Enterprise Architecture initiative is doing; others may not care. This step enables the team to easily see which stakeholders are expected to be blockers or critics, and which stakeholders are likely to be advocates and supporters of the initiative.

Work out stakeholder power, influence, and interest, so as to focus the Enterprise Architecture engagement on the key individuals. These can be mapped onto a power/interest matrix, which also indicates the strategy to adopt for engaging with them. [Figure 3-2](#) shows an example power grid matrix.

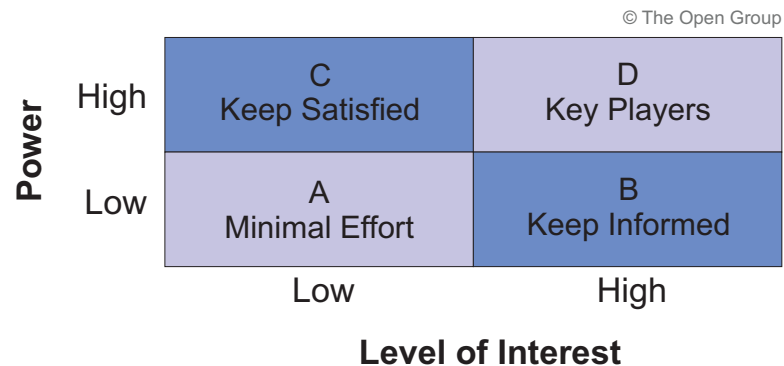


Figure 3-2 Stakeholder Power Grid

3.3.4 Tailor Engagement Deliverables

Identify catalogs, matrices, and diagrams that the architecture engagement needs to produce and validate with each stakeholder group to deliver an effective architecture model.

It is important to pay particular attention to stakeholder interests by defining specific catalogs, matrices, and diagrams that are relevant for a particular Enterprise Architecture model. This enables the architecture to be communicated to, and understood by, all the stakeholders, and enables them to verify that the Enterprise Architecture initiative will address their concerns.

3.4 Template Stakeholder Map

The following table provides an example stakeholder map for a TOGAF architecture project which has stakeholders as identified in [Figure 3-1](#).

Stakeholder	Key Concerns	Class	Catalogs, Matrices, and Diagrams
CxO (Corporate Functions); e.g., CEO, CFO, CIO, COO	The high-level drivers, goals, and objectives of the organization, and how these are translated into an effective process and IT architecture to advance the business.	KEEP SATISFIED	Business Footprint diagram Goal/Objective/Business Service diagram Organization Decomposition diagram Business Capabilities catalog Capability/Organization matrix Business Capability Map Strategy/Capability matrix Capability/Organization matrix Business Model diagram Value Stream catalog Value Stream Stages catalog Value Stream/Capability matrix Value Stream Map

Stakeholder	Key Concerns	Class	Catalogs, Matrices, and Diagrams
Program Management Office (Corporate Functions); e.g., Project Portfolio Managers	Prioritizing, funding, and aligning change activity. An understanding of project content and technical dependencies between projects supports portfolio management decision-making.	KEEP SATISFIED	Requirements catalog Project Context diagram Benefits diagram Business Footprint diagram Application Communication diagram Organization map Business Capabilities catalog Capability/Organization matrix Business Capability Map Strategy/Capability matrix Capability/Organization matrix Business Model diagram Value Stream catalog Value Stream Stages catalog Value Stream/Capability matrix Value Stream Map
Procurement (Corporate Functions); e.g., Acquirers	Understanding what building blocks of the architecture can be bought, and what constraints (or rules) are relevant to the purchase. Acquirers will shop with multiple vendors looking for the best cost solution while adhering to the constraints (or rules) derived from the architecture, such as standards. The key concern is to make purchasing decisions that fit the architecture.	KEY PLAYERS	Technology Portfolio catalog Technology Standards catalog

Stakeholder	Key Concerns	Class	Catalogs, Matrices, and Diagrams
Human Resources (HR) (Corporate Functions); e.g., HR Managers, Training & Development Managers	The roles and actors are required to support the architecture and changes to it. The key concern is managing people transitions.	KEEP INFORMED	Organization Decomposition diagram Organization/Actor catalog Location catalog Application and User Location diagram Business Capabilities catalog Capability/Organization matrix Business Capability Map Strategy/Capability matrix Capability/Organization matrix Business Model diagram
Enterprise Security (Corporate Functions); e.g., Corporate Risk Management, Security Officers, IT Security Managers	Ensuring that the information, data, and systems of the organization are available to only those that have permission, and protecting the information, data, and systems from unauthorized tampering.	KEY PLAYERS	Product Lifecycle diagram Data Dissemination diagram Data Security diagram Actor/Role matrix Networked Computing Hardware diagram Network and Communications diagram

Stakeholder	Key Concerns	Class	Catalogs, Matrices, and Diagrams
QA/Standards Group (Corporate Functions); e.g., Data Owners, Process Owners, Technical Standards Bodies	Ensuring the consistent governance of the organization's business, data, application, and technology assets.	KEY PLAYERS	Process/Event/Control/Product catalog Contract/Measure catalog Application Portfolio catalog Interface catalog Technology Standards catalog Technology Portfolio catalog Value Stream catalog Value Stream Stages catalog Value Stream/Capability matrix Value Stream Map

Stakeholder	Key Concerns	Class	Catalogs, Matrices, and Diagrams
Executive (End-user Organization); e.g., Business Unit Directors, Business Unit CxOs, Business Unit Head of IT/Architecture	The high-level drivers, goals, and objectives of the organization, and how these are translated into an effective process and architecture to advance the business.	KEEP SATISFIED	Business Footprint diagram Goal/Objective/Business Service diagram Organization Decomposition diagram Process Flow diagram Application Communication diagram Business Capabilities catalog Capability/Organization matrix Business Capability Map Strategy/Capability matrix Capability/Organization matrix Business Model diagram Value Stream catalog Value Stream Stages catalog Value Stream/Capability matrix Value Stream Map

Stakeholder	Key Concerns	Class	Catalogs, Matrices, and Diagrams
Line Management (End-user Organization); e.g., Senior Business Managers, Operations Regional Managers, IT Managers	Top-level functions and processes of the organization, and how the key applications support these processes.	KEY PLAYERS	Business Footprint diagram Organization Decomposition diagram Organization map Process Flow diagram Application Communication diagram Application and User Location diagram Business Capabilities catalog Capability/Organization matrix Business Capability Map Strategy/Capability matrix Capability/Organization matrix Business Model diagram Value Stream catalog Value Stream Stages catalog Value Stream/Capability matrix Value Stream Map

Stakeholder	Key Concerns	Class	Catalogs, Matrices, and Diagrams
Business Domain Experts (End-user Organization); e.g., Business Process Experts, Business/Process Analyst, Process Architect, Process Designer, Functional Managers, Business Analyst	Functional aspects of processes and supporting systems. This can cover the human actors involved in the system, the user processes involved in the system, the functions required to support the processes, and the information required to flow in support of the processes.	KEY PLAYERS	Business Interaction matrix Actor/Role matrix Business Service/Information diagram Organization map Product Lifecycle diagram Business Use-Case diagram Application Use-Case diagram Application Communication diagram Data Entity/Business Function matrix Value Stream catalog Value Stream Stages catalog Value Stream/Capability matrix Value Stream Map
IT Service Management (Systems Operations); e.g., Service Delivery Manager	Ensuring that IT services provided to the organization meet the service levels required by that organization to succeed in business.	KEEP INFORMED	Technology Standards catalog Technology Portfolio catalog Contract/Measure catalog Process/Application Realization diagram Enterprise Manageability diagram

Stakeholder	Key Concerns	Class	Catalogs, Matrices, and Diagrams
IT Operations — Applications (System Operations); e.g., Application Architecture, System & Software Engineers	Development approach, software modularity and re-use, portability migration, and interoperability.	KEY PLAYERS	Process/Application Realization diagram Application/Data matrix Application Migration diagram Software Engineering diagram Platform decomposition Diagram Networked Computing/ Hardware diagram Software distribution Diagram
IT Operations — Infrastructure (System Operations); e.g., Infrastructure Architect, Wintel support, Mid-range support, Operational DBA, Service Desk	Location, modifiability, re-usability, and availability of all components of the system. Ensuring that the appropriate components are developed and deployed within the system in an optimal manner.	KEY PLAYERS	Platform Decomposition diagram Technology Standards catalog Technology Portfolio catalog Enterprise Manageability diagram Networked Computing/ Hardware diagram Processing diagram Environments and Locations diagram
IT Operations — Data/Voice Communications (System Operations); e.g., Network Management	Location, modifiability, re-usability, and availability of communications and networking services. Ensuring that the appropriate communications and networking services are developed and deployed within the system in an optimal manner.	KEY PLAYERS	Network and Communications diagram

Stakeholder	Key Concerns	Class	Catalogs, Matrices, and Diagrams
Executive (Project Organization); e.g., Sponsor, Program Manager	On-time, on-budget delivery of a change initiative that will realize expected benefits for the organization.	KEEP INFORMED	Requirements catalog Principles catalog Value Chain diagram Solution Concept diagram Organization map Application and User Location diagram Business Capabilities catalog Capability/Organization matrix Business Capability Map Strategy/Capability matrix Capability/Organization matrix Business Model diagram Value Stream catalog Value Stream Stages catalog Value Stream/Capability matrix Value Stream Map

Stakeholder	Key Concerns	Class	Catalogs, Matrices, and Diagrams
Line Management (Project Organization); e.g., Project Manager	Operationally achieving on-time, on-budget delivery of a change initiative with an agreed scope.	KEEP INFORMED	Application Communication diagram Organization map Environments and Locations diagram Business Capabilities catalog Capability/Organization matrix Business Capability Map Strategy/Capability matrix Capability/Organization matrix Business Model diagram Value Stream catalog Value Stream Stages catalog Value Stream/Capability matrix Value Stream Map

Stakeholder	Key Concerns	Class	Catalogs, Matrices, and Diagrams
Business Process/Functional Expert (Project Organization); e.g., Financials FICO® Functional Consultant, HR Functional Consultant	Adding more detail to the functional requirements of a change initiative based on experience and interaction with business domain experts in the end-user organization.	KEY PLAYERS	Process Flow diagram Business Use-Case diagram Business Service/Information diagram Organization map Application Communication diagram Business Capabilities catalog Capability/Organization matrix Business Capability Map Strategy/Capability matrix Capability/Organization matrix Business Model diagram Value Stream catalog Value Stream Stages catalog Value Stream/Capability matrix Value Stream Map
Product Specialist (Project Organization); e.g., Portal Product Specialist	Specifying technology product designs in order to meet project requirements and comply with the Architecture Vision of the solution. In a packages and packaged services environment, product expertise can be used to identify product capabilities that can be readily leveraged and can provide guidance on strategies for product customization.	KEY PLAYERS	Software Engineering diagram Application/Data matrix

Stakeholder	Key Concerns	Class	Catalogs, Matrices, and Diagrams
Technical Specialist (Project Organization); e.g., Application Architect	Specifying technology product designs in order to meet project requirements and comply with the Architecture Vision of the solution.	KEY PLAYERS	Software Engineering diagram Platform Decomposition diagram Process/Application Realization diagram Application/Data matrix Application Migration diagram
Regulatory Bodies (Outside Services); e.g., Financial Regulator, Industry Regulator	Receipt of the information they need in order to regulate the client organization, and ensuring that their information requirements are properly satisfied. Interested in reporting processes, and the data and applications used to provide regulatory return information.	KEEP SATISFIED	Business Footprint diagram Application Communication diagram

Stakeholder	Key Concerns	Class	Catalogs, Matrices, and Diagrams
Suppliers (Outside Services); e.g., Alliance Partners, Key Suppliers	Ensuring that their information exchange requirements are met in order that agreed service contracts with the client organizations can be fulfilled.	KEEP SATISFIED	Business Footprint diagram Business Service/Information diagram Application Communication diagram Business Capabilities catalog Capability/Organization matrix Business Capability Map Strategy/Capability matrix Capability/Organization matrix Business Model diagram Value Stream catalog Value Stream Stages catalog Value Stream/Capability matrix Value Stream Map

Chapter 4: Architecture Patterns

This chapter provides guidelines for using architecture patterns.

4.1 Introduction

Patterns for describing Enterprise Architectures are becoming increasingly important to practitioners. The diverse and multi-disciplinary nature of Enterprise Architecture requires that patterns be developed in different disciplines, domains, and levels of detail.

Previous versions of this standard did not fully embrace architecture patterns due to their perceived lack of maturity. Today, many organizations are using patterns to describe their architectures at various levels ranging from software design patterns to business patterns. It remains true that there is no single standard for describing Enterprise Architecture patterns. However, it can be said that there is a pattern for describing patterns.

4.1.1 Background

A "pattern" has been defined as: "an idea that has been useful in one practical context and will probably be useful in others" (Source: Analysis Patterns — Re-usable Object Models, by M. Fowler).

In the TOGAF Standard, patterns are considered to be a way of putting building blocks into context; for example, to describe a re-usable solution to a problem. Building blocks are what you use: patterns can tell you how you use them, when, why, and what trade-offs you have to make in doing so. Patterns offer the promise of helping the architect to identify combinations of Architecture and/or Solution Building Blocks (ABBs/SBBs) that have been proven to deliver effective solutions in the past, and may provide the basis for effective solutions in the future.

Pattern techniques are generally acknowledged to have been established as a valuable architectural design technique by Christopher Alexander, a buildings architect, who described this approach in his book *The Timeless Way of Building*, published in 1979. This book provides an introduction to the ideas behind the use of patterns, and Alexander followed it with two further books (*A Pattern Language* and *The Oregon Experiment*) in which he expanded on his description of the features and benefits of a patterns approach to architecture.

Software and buildings architects have many similar issues to address, and so it was natural for software architects to take an interest in patterns as an architectural tool. Many papers and books have been published on them since Alexander's 1979 book, perhaps the most renowned being *Design Patterns: Elements of Re-usable Object-Oriented Software* (Gamma et al., 1994). This book describes simple and elegant solutions to specific problems in object-oriented software design.

4.1.2 Content of a Pattern

Several different formats are used in the literature for describing patterns, and no single format has achieved widespread acceptance. However, there is broad agreement on the types of things that a pattern should contain. The headings which follow are taken from *Pattern-Oriented Software Architecture: A System of Patterns* (Buschmann et al., 1996). The elements described below will be found in most patterns, even if different headings are used to describe them.

Name	A meaningful and memorable way to refer to the pattern, typically a single word or short phrase.
Problem	A description of the problem indicating the intent in applying the pattern — the intended goals and objectives to be reached within the context and forces described below (perhaps with some indication of their priorities).
Context	The preconditions under which the pattern is applicable — a description of the initial state before the pattern is applied.
Forces	<p>A description of the relevant forces and constraints, and how they interact/conflict with each other and with the intended goals and objectives. The description should clarify the intricacies of the problem and make explicit the kinds of trade-offs that must be considered. (The need for such trade-offs is typically what makes the problem difficult, and generates the need for the pattern in the first place.) The notion of "forces" equates in many ways to the "qualities" that architects seek to optimize, and the concerns they seek to address, in designing architectures. For example:</p> <ul style="list-style-type: none"> — Security, robustness, reliability, fault-tolerance — Manageability — Efficiency, performance, throughput, bandwidth requirements, space utilization — Scalability (incremental growth on-demand) — Extensibility, evolvability, maintainability — Modularity, independence, re-usability, openness, composability (plug-and-play), portability — Completeness and correctness — Ease-of-construction — Ease-of-use — etc., . . .
Solution	A description, using text and/or graphics, of how to achieve the intended goals and objectives. The description should identify both the solution's static structure and its dynamic behavior — the people and computing actors, and their collaborations. The description may include guidelines for implementing the solution. Variants or specializations of the solution may also be described.
Resulting Context	The post-conditions after the pattern has been applied. Implementing the solution normally requires trade-offs among competing forces.

This element describes which forces have been resolved and how, and which remain unresolved. It may also indicate other patterns that may be applicable in the new context. (A pattern may be one step in accomplishing some larger goal.) Any such other patterns will be described in detail under Related Patterns.

Examples One or more sample applications of the pattern which illustrate each of the other elements: a specific problem, context, and set of forces; how the pattern is applied; and the resulting context.

Rationale An explanation/justification of the pattern as a whole, or of individual components within it, indicating how the pattern actually works, and why — how it resolves the forces to achieve the desired goals and objectives, and why this is "good". The Solution element of a pattern describes the external structure and behavior of the solution: the Rationale provides insight into its internal workings.

Related Patterns

The relationships between this pattern and others. These may be predecessor patterns, whose resulting contexts correspond to the initial context of this one; or successor patterns, whose initial contexts correspond to the resulting context of this one; or alternative patterns, which describe a different solution to the same problem, but under different forces; or co-dependent patterns, which may/must be applied along with this pattern.

Known Uses Known applications of the pattern within existing systems, verifying that the pattern does indeed describe a proven solution to a recurring problem. Known Uses can also serve as Examples.

Patterns may also begin with an Abstract providing an overview of the pattern and indicating the types of problems it addresses. The Abstract may also identify the target audience and what assumptions are made of the reader.

4.1.3 Terminology

Although design patterns have been the focus of widespread interest in the software industry for several years, particularly in the object-oriented and component-based software fields, it is only recently that there has been increasing interest in architecture patterns — extending the principles and concepts of design patterns to the architecture domain.

The technical literature relating to this field is complicated by the fact that many people in the software field use the term "architecture" to refer to software, and many patterns described as "architecture patterns" are high-level software design patterns. This simply makes it all the more important to be precise in the use of terminology.

4.1.3.1 Architecture Patterns and Design Patterns

The term "design pattern" is often used to refer to any pattern which addresses issues of software architecture, design, or programming implementation. In *Pattern-Oriented Software Architecture: A System of Patterns*, the authors define these three types of patterns as follows:

- An **Architecture Pattern** expresses a fundamental structural organization or schema for software systems

It provides a set of predefined subsystems, specifies their responsibilities, and includes rules and guidelines for organizing the relationships between them.

- A **Design Pattern** provides a scheme for refining the subsystems or components of a software system, or the relationships between them

It describes a commonly recurring structure of communicating components that solves a general design problem within a particular context.

- An **Idiom** is a low-level pattern specific to a programming language

An idiom describes how to implement particular aspects of components or the relationships between them using the features of the given language.

These distinctions are useful, but it is important to note that architecture patterns in this context still refers solely to software architecture. Software architecture is certainly an important part of the focus of the TOGAF Standard, but it is not its only focus.

In this section we are concerned with patterns for enterprise system architecting. These are analogous to software architecture and design patterns, and borrow many of their concepts and terminology, but focus on providing re-usable models and methods specifically for the architecting of enterprise information systems — comprising software, hardware, networks, and people — as opposed to purely software systems.

4.1.3.2 *Patterns and the Architecture Continuum*

Although architecture patterns have not (as yet) been integrated into the TOGAF Standard, each of the first four main phases of the ADM (Phases A through D) gives an indication of the stage at which relevant re-usable architecture assets from the Enterprise Architecture Continuum should be considered for use. Architecture patterns are one such asset.

An enterprise that adopts a formal approach to the use and re-use of architecture patterns will normally integrate their use into the Enterprise Architecture Continuum.

4.1.3.3 *Patterns and Views*

Architecture views are selected parts of one or more models representing a complete system architecture, focusing on those aspects that address the concerns of one or more stakeholders. Patterns can provide help in designing such models, and in composing views based on them.

4.1.3.4 *Patterns and Business Scenarios*

Relevant architecture patterns may well be identified in the work on business scenarios.

4.2 Some Pattern Resources

- The Patterns Home Page (refer to hillside.net/patterns) hosted by the Hillside Group provides information about patterns, links to online patterns, papers, and books dealing with patterns, and patterns-related mailing lists
- The Patterns-Discussion FAQ (refer to <http://purl.org/theopengroup/pd-FAQ>) maintained by Doug Lea provides a very thorough and highly readable FAQ about patterns
- *Patterns and Software: Essential Concepts and Terminology* by Brad Appleton (refer to www.bradapp.com/docs/patterns-intro.html) provides another thorough and readable account of the patterns field

- The Service-Oriented Architecture (SOA) Patterns community website (refer to www.soapatterns.org/), dedicated to the ongoing development and expansion of the SOA design pattern catalog
- The Cloud Computing Design Patterns community website (refer to www.cloudpatterns.org)

Chapter 5: Gap Analysis

The technique known as gap analysis is widely used in the TOGAF Architecture Development Method (ADM) to validate an architecture that is being developed. The basic premise is to highlight a shortfall between the Baseline Architecture and the Target Architecture; that is, items that have been deliberately omitted, accidentally left out, or not yet defined.

5.1 Introduction

A key step in validating an architecture is to consider what may have been forgotten. The architecture must support all of the essential information processing needs of the organization. The most critical source of gaps that should be considered is stakeholder concerns that have not been addressed in prior architectural work.

Potential sources of gaps include:

- Business domain gaps:
 - People gaps (e.g., cross-training requirements)
 - Process gaps (e.g., process inefficiencies)
 - Tools gaps (e.g., duplicate or missing tool functionality)
 - Information gaps
 - Measurement gaps
 - Financial gaps
 - Facilities gaps (buildings, office space, etc.)
- Data domain gaps:
 - Data not of sufficient currency
 - Data not located where it is needed
 - Not the data that is needed
 - Data not available when needed
 - Data not created
 - Data not consumed
 - Data relationship gaps

- Applications impacted, eliminated, or created
- Technologies impacted, eliminated, or created

5.2 Suggested Steps

The suggested steps are as follows:

- Draw up a matrix with all the ABBs of the Baseline Architecture on the vertical axis, and all the ABBs of the Target Architecture on the horizontal axis
- Add to the Baseline Architecture axis a final row labeled "New", and to the Target Architecture axis a final column labeled "Eliminated"
- Where an ABB is available in both the Baseline and Target Architectures, record this with "Included" at the intersecting cell
- Where an ABB from the Baseline Architecture is missing in the Target Architecture, each must be reviewed

If it was correctly eliminated, mark it as such in the appropriate "Eliminated" cell. If it was not, an accidental omission in the Target Architecture has been uncovered that must be addressed by reinstating the ABB in the next iteration of the architecture design — mark it as such in the appropriate "Eliminated" cell.

- Where an ABB from the Target Architecture cannot be found in the Baseline Architecture, mark it at the intersection with the "New" row as a gap that needs to be filled, either by developing or procuring the building block

When the exercise is complete, anything under "Eliminated" or "New" is a gap, which should either be explained as correctly eliminated, or marked as to be addressed by reinstating or developing/procuring the building block.

5.3 Example

Figure 5-1 shows an example analysis for ABBs that are services from the Network Services category of the TOGAF Technical Reference Model (TRM), and shows a number of services from the Baseline Architecture missing from the Target Architecture.

Target → Architecture Baseline Architecture ↓	Video Conferencing Services	Enhanced Telephony Services	Mailing List Services	Eliminated Services ↓
Broadcast Services				Intentionally eliminated
Video Conferencing Services	Included			
Enhanced Telephony Services		Potential match		
Shared Screen Services				Unintentionally excluded - a gap in Target Architecture
New →		Gap: Enhanced services to be developed or produced	Gap: To be developed or produced	© The Open Group

Figure 5-1 Gap Analysis Example

Chapter 6: Migration Planning Techniques

This chapter contains a number of techniques used to support migration planning in Phases E and F.

6.1 Implementation Factor Catalog

The technique of creating an Implementation Factor catalog can be used to document factors impacting the architecture Implementation and Migration Plan.

The catalog should include a list of the factors to be considered, their descriptions, and the deductions that indicate the actions or constraints that have to be taken into consideration when formulating the plans.

Factors typically include:

- Risks
- Issues
- Assumptions
- Dependencies
- Actions
- Impacts

An example catalog is shown in [Figure 6-1](#).

Implementation Factor Catalog		
Factor	Description	Deduction
<Name of Factor>	<Description of Factor>	<Impact on Migration Plan>
Change in Technology	Shut down the message centers, saving 700 personnel, and have them replaced by email.	<ul style="list-style-type: none">• Need for personnel training, re-assignment• Email has major personnel savings and should be given priority
Consolidation of Services		
Introduction of New Customer Service		

© The Open Group

Figure 6-1 Implementation Factor Catalog

6.2 Consolidated Gaps, Solutions, & Dependencies Matrix

The technique of creating a Consolidated Gaps, Solutions, and Dependencies matrix allows the architect to group the gaps identified in the domain architecture gap analysis results and assess potential solutions and dependencies to one or more gaps.

This matrix can be used as a planning tool when creating work packages. The identified dependencies will drive the creation of projects and migration planning in Phases E and F.

An example matrix is shown in [Figure 6-2](#).

Consolidated Gaps, Solutions, and Dependencies Matrix				
No.	Architecture	Gap	Potential Solutions	Dependencies
1	Business	New Order Processing Process	Use COTS software tool process Implement custom solution	Drives applications (2)
2	Application	New Order Processing Application	COTS software tool X Develop in-house	
3	Information	Consolidated Customer Information Base	Use COTS customer base Develop customer data mart	

© The Open Group

Figure 6-2 Consolidated Gaps, Solutions, and Dependencies Matrix

6.3 Architecture Definition Increments Table

The technique of creating an Architecture Definition Increments table allows the architect to plan a series of Transition Architectures outlining the status of the Enterprise Architecture at specified times.

A table should be drawn up, as shown in [Figure 6-3](#), listing the projects and then assigning their incremental deliverables across the Transition Architectures.

Architecture Definition - Project Objectives by Increment (Example Only)				
Project	April 2018/2019	April 2019/2020	April 2020/2021	Comments
	Transition Architecture 1: Preparation	Transition Architecture 2: Initial Operational Capability	Transition Architecture 3: Benefits	
Enterprise e-Services Capability	Training and Business Process	e-Licensing Capability	e-Employment Benefits	
IT e-Forms	Design and Build			
IT e-Information Environment	Design and Build Information Environment	Client Common Data Web Content Design and Build	Enterprise Common Data Component Management Design and Build	
...

© The Open Group

Figure 6-3 Architecture Definition Increments Table

6.4 Transition Architecture State Evolution Table

The technique of creating the Transition Architecture State Evolution table allows the architect to show the proposed state of the architectures at various levels using the defined taxonomy (e.g., the TOGAF TRM).

A table should be drawn, listing the services from the taxonomy used in the enterprise, the Transition Architectures, and proposed transformations, as shown in Figure 6-4.

All SBBs should be described with respect to their delivery and impact on these services. They should also be marked to show the progression of the Enterprise Architecture. In the example, where target capability has been reached, this is shown as "new" or "retain"; where capability is transitioned to a new solution, this is marked as "transition"; and where a capability is to be replaced, this is marked as "replace".

Architectural State using the Technical Reference Model				
Sub-Domain	Service	Transition Architecture 1	Transition Architecture 2	Transition Architecture 3
Infrastructure Applications	Information Exchange Services	Solution System A (replace)	Solution System B-1 (transition)	Solution System B-2 (new)
	Data Management Services	Solution System D (retain)	Solution System D (retain)	Solution System D (retain)
...

© The Open Group

Figure 6-4 Transition Architecture State Evolution Table

Another technique (not shown here) is to use color coding in the matrix; for example:

- Green: service SBB in place (either new or retained)
- Yellow: service being transitioned into a new solution
- Red: service to be replaced

6.5 Business Value Assessment Technique

A technique to assess business value is to draw up a matrix based on a value index dimension and a risk index dimension. An example is shown in [Figure 6-5](#). The value index should include criteria such as compliance to principles, financial contribution, strategic alignment, and competitive position. The risk index should include criteria such as size and complexity, technology, organizational capacity, and impact of a failure. Each criterion should be assigned an individual weight.

The index and its criteria and weighting should be developed and approved by senior management. It is important to establish the decision-making criteria before the options are known.

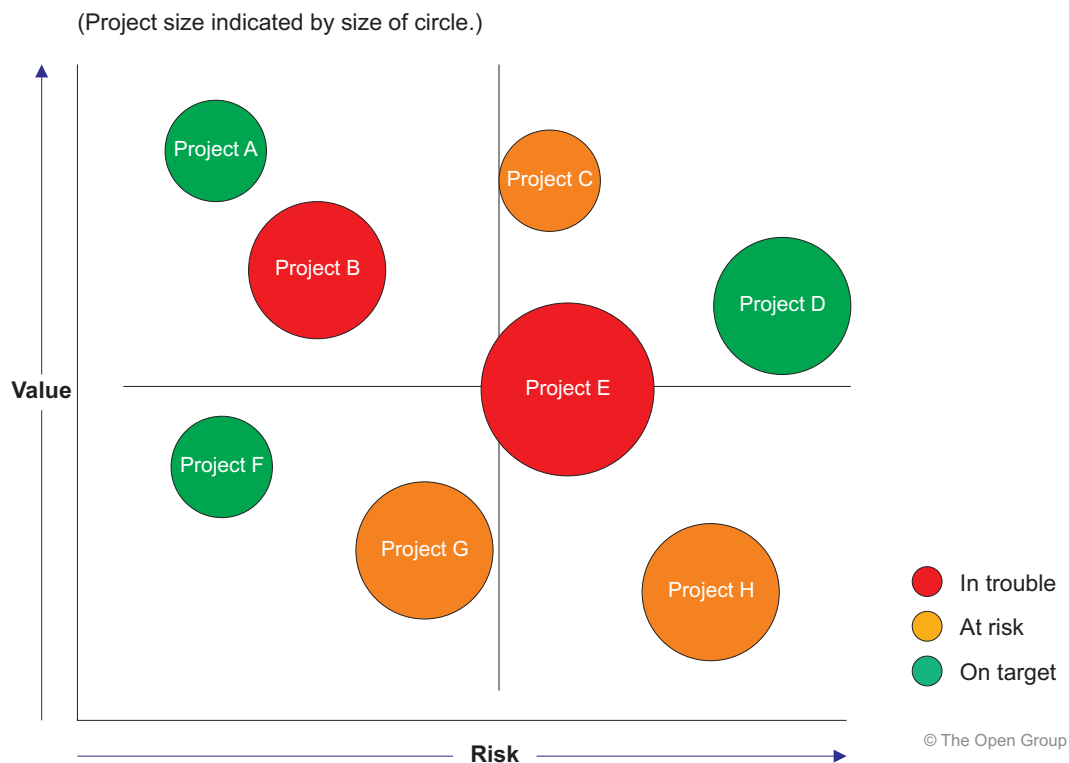


Figure 6-5 Sample Project Assessment with Respect to Business Value and Risk

Chapter 7: Interoperability Requirements

This chapter provides guidelines for defining and establishing interoperability requirements.

7.1 Overview

A definition of interoperability is "the ability to share information and services". Defining the degree to which the information and services are to be shared is a very useful architectural requirement, especially in a complex organization and/or extended enterprise.

The determination of interoperability is present throughout the ADM as follows:

- In the Architecture Vision (Phase A), the nature and security considerations of the information and service exchanges are first revealed within the business scenarios
- In the Business Architecture (Phase B), the information and service exchanges are further defined in business terms
- In the Data Architecture (Phase C), the content of the information exchanges is detailed using the corporate data and/or information exchange model
- In the Application Architecture (Phase C), the way that the various applications are to share the information and services is specified
- In the Technology Architecture (Phase D), the appropriate technical mechanisms to permit the information and service exchanges are specified
- In Opportunities & Solutions (Phase E), the actual solutions (e.g., COTS packages) are selected
- In Migration Planning (Phase F), the interoperability is logically implemented

7.2 Defining Interoperability

There are many ways to define interoperability and the aim is to define one that is consistently applied within the enterprise and extended enterprise. It is best that both the enterprise and the extended enterprise use the same definitions.

Many organizations find it useful to categorize interoperability as follows:

- **Operational or Business Interoperability** defines how business processes are to be shared
- **Information Interoperability** defines how information is to be shared
- **Technical Interoperability** defines how technical services are to be shared or at least connect to one another

From an IT perspective, it is also useful to consider interoperability in a similar vein to Enterprise Application Integration (EAI); specifically:

- **Presentation Integration/Interoperability** is where a common look-and-feel approach through a common portal-like solution guides the user to the underlying functionality of the set of systems
- **Information Integration/Interoperability** is where the corporate information is seamlessly shared between the various corporate applications to achieve, for example, a common set of client information

Normally this is based upon a commonly accepted corporate ontology and shared services for the structure, quality, access, and security/privacy for the information.

- **Application Integration/Interoperability** is where the corporate functionality is integrated and shareable so that the applications are not duplicated (e.g., one change of address service/component; not one for every application) and are seamlessly linked together through functionality such as workflow

This impacts the business and infrastructure applications and is very closely linked to corporate business process unification/interoperability.

- **Technical Integration/Interoperability** includes common methods and shared services for the communication, storage, processing, and access to data primarily in the application platform and communications infrastructure domains

This interoperability is premised upon the degree of rationalization of the corporate IT infrastructure, based upon standards and/or common IT platforms. For example, multiple applications sharing one infrastructure or 10,000 corporate websites using one centralized content management/web server (rather than thousands of servers and webmasters spread throughout the country/globe).

Many organizations create their own interoperability models, such as illustrated in the example below from the Canadian Government. They have a high-level definition of the three classes of interoperability and identify the nature of the information and services that they wish to share. Interoperability is coined in terms of e-enablers for e-Government. Their interoperability breakdown is as follows:

- Information Interoperability:
 - Knowledge management
 - Business intelligence
 - Information management
 - Trusted identity
- Business Interoperability:
 - Delivery networks
 - e-Democracy
 - e-Business
 - Enterprise resource management

- Relationship and case management
- Technical Interoperability:
 - IT infrastructure

In certain architectural approaches, such as system of systems or a federated model, interoperability is a strongly recommended best practice that will determine how the systems interact with each other. A key consideration will be the enterprise's business operating model.

7.3 Enterprise Operating Model

Key to establishing interoperability is the determination of the corporate operating model, where the operating model is "the necessary level of business process integration and standardization for delivering goods and services to customers. An operating model describes how a company wants to thrive and grow. By providing a more stable and actionable view of the company than strategy, the operating model drives the design of the foundation for execution."¹

For example, if lines of business or business units only need to share documents, then the ABBs and SBBs may be simpler than if there is a need to share structured transaction data. Similarly, if the Architecture Vision includes a shared services environment, then it is useful to define the level the services are to be shared.

The corporate operating model will normally indicate what type of interoperability approach will be appropriate. This model should be determined in Phase A (Architecture Vision) if not in Phase B (Business Architecture), and definitely by Phase E (Opportunities & Solutions).

Complex enterprises and/or extended enterprises (e.g., supply chain) may have more than one type of operating model. For example, it is common for the internal operating model (and supporting interoperability model) to differ from the one used for the extended enterprise.

7.4 Refining Interoperability

Implementing interoperability requires the creation, management, acceptance, and enforcement of realistic standards that are SMART (Specific, Measurable, Actionable, Realistic, and Time-bound). Clear measures of interoperability are key to success.

Architecture is the key for identifying standards and facilitated sessions will examine potential pragmatic ways (that fit within the current or emerging business culture) to achieve the requisite degree of interoperability.

Interoperability should be refined so that it meets the needs of the enterprise and/or extended enterprise in an unambiguous way. The refined interoperability measures (degrees, types, and high-level targets) should be part of or referred to the Enterprise Architecture strategic direction.

These measures are instantiated within a transformation strategy that should be embedded within the Target Architecture definition and pragmatically implemented in the Transition Architectures. Upon completion, also update the consolidated gap analysis results and dependencies to ensure that all output from facilitated sessions is captured.

An example of specifying interoperability is the Degrees of Interoperability (used within the

1. *Enterprise Architecture as Strategy* (Ross et al., 2006) provides potential models.

Canadian Department of National Defense and NATO). These organizations were focused on the sharing of information and came up with four degrees of interoperability as follows:

- **Degree 1: Unstructured Data Exchange** involves the exchange of human-interpretable unstructured data, such as the free text found in operational estimates, analysis, and papers
- **Degree 2: Structured Data Exchange** involves the exchange of human-interpretable structured data intended for manual and/or automated handling, but requires manual compilation, receipt, and/or message dispatch
- **Degree 3: Seamless Sharing of Data** involves the automated sharing of data amongst systems based on a common exchange model
- **Degree 4: Seamless Sharing of Information** is an extension of Degree 3 to the universal interpretation of information through data processing based on co-operating applications

These degrees should be further refined and made technically meaningful for each of the degrees. An example refinement of Degree 3 with four subclassifications follows:

- 3A: Formal Message Exchange
- 3B: Common Data Exchange
- 3C: Complete Data Exchange
- 3D: Real-time Data Exchange

The intent is to specify the detailed degrees of interoperability to the requisite level of detail so that they are technically meaningful.

These degrees are very useful for specifying the way that information has to be exchanged between the various systems and provide critical direction to the projects implementing the systems.

Similar measures should be established to determine service/business and technical interoperability.

7.5 Determining Interoperability Requirements

Co-existence between emerging and existing systems, especially during transformation, will be a major challenge and facilitated sessions should attempt to figure out what has to be done to reduce the pain. It is imperative to involve the operations management staff and architects in this step as they will be responsible for operating the portfolio deliverables.

For example, there might be a need for a "wrapper" application (an application that acts as the interface [a.k.a. interpreter] between the legacy application and the emerging infrastructure). Indeed, pragmatically, in the "if it works do not fix it" world, the "wrapper" might become a permanent solution.

Regardless, using the gap analysis results and business scenarios as a foundation, discuss the IT issues and work them through to ensure that all of the gaps are clearly identified and addressed and verify that the organization-specific requirements will be met.

It is important to note that the ensuing development process must include recognition of dependencies and boundaries for functions and should take account of what products are available in the marketplace. An example of how this might be expressed can be seen in the

building blocks example (see the TOGAF Standard — Architecture Content).

If a mechanism such as the Degrees of Interoperability is used, then a matrix showing the interoperability requirements is a useful tool, as illustrated in [Figure 7-1](#) and [Figure 7-2](#), noting that the degree of information sharing is not necessarily symmetrical or bidirectional between systems and/or stakeholders.

The matrix below can be used within the enterprise and/or within the extended enterprise as a way of detailing that information and/or services can be shared. The matrix should start in the Business Architecture (Phase B) to capture the nature of the sharing of information between stakeholders, and evolve to determine what systems share what information in Phase C.

Phase B: Inter-stakeholder Information Interoperability Requirements (Using degrees of information interoperability)							
Stakeholders	A	B	C	D	E	F	G
A		2	3	2	3	3	3
B	2		3	2	3	2	2
C	3	3		2	2	2	3
D	2	2	2		3	3	3
E	4	4	2	3		3	3
F	4	4	2	3	3		2
G	2	2	3	3	3	3	

© The Open Group

Figure 7-1 Business Information Interoperability Matrix

[Figure 7-1](#) shows that Stakeholder A requires structured data exchange (Degree 2) with Stakeholders/Systems B and D, and seamless sharing of data (Degree 3) with Stakeholders/Systems C, E, F, and G.

The business information interoperability matrix should be refined within the Information Systems Architecture using refined measures and specifying the actual systems used by the stakeholders. A sample is shown in [Figure 7-2](#).

Phase C: Inter-system Interoperability Requirements							
	System A	System B	System C	System D	System E	System F	System G
System A		2A	3D	2B	3A	3A	3B
System B	2E		3F	2C	3A	2B	2C
System C	3E	3F		2B	2A	2A	3B
System D	2B	2B	2B		3A	3A	3B
System E	4A	4B	2B	3A		3B	3B
System F	4A	4A	2B	3B	3A		2D
System G	2B	2B	3A	3A	3B	3B	

© The Open Group

Figure 7-2 Information Systems Interoperability Matrix

In [Figure 7-2](#), both the nature of the exchange is more detailed (e.g., Degree 3A *versus* only Degree 3) and the sharing is between specific systems rather than stakeholders. For example, System A shares information with the other systems in accordance with enterprise technical standards.

In many organizations the Business Architectures describe the nature of the information shared between stakeholders and/or organizations (e.g., in defense the term is "operational node"), and the Data Architecture specifies the information shared between systems.

Update the defined target data and Application Architecture (Approved) with the interoperability issues that were raised.

7.6 Reconciling Interoperability Requirements with Potential Solutions

The Enterprise Architect will have to ensure that there are no interoperability conflicts, especially if there is an intention to re-use existing SBBs and/or COTS.

The most significant issue to be addressed is in fact business interoperability. Most SBBs or COTS will have their own business processes embedded. Changing the embedded business processes will often require so much work that the advantages of re-using solutions will be lost. There are numerous examples of this in the past.

Furthermore, there is the workflow aspect between the various systems that has to be taken into account. The Enterprise Architect will have to ensure that any change to the business interoperability requirements is signed off by the Business Architects and architecture sponsors in a revised Statement of Architecture Work.

Chapter 8: Business Transformation Readiness Assessment

This chapter describes a technique known as Business Transformation Readiness Assessment, used for evaluating and quantifying an organization's readiness to undergo change.

This chapter builds on work by the Canadian Government and its Business Transformation Enablement Program (BTEP).

8.1 Introduction

Enterprise Architecture is a major endeavor within an organization and most often an innovative Architecture Vision (Phase A) and supporting Architecture Definition (Phases B to D) will entail considerable change. There are many dimensions to change, but by far the most important is the human element. For example, if the enterprise envisages a consolidation of information holdings and a move to a new paradigm such as service orientation for integrated service delivery, then the human resource implications are major. Potentially coupled with a change-averse culture and a narrowly skilled workforce, the most sound and innovative architecture could go nowhere.

Understanding the readiness of the organization to accept change, identifying the issues, and then dealing with them in the Implementation and Migration Plans is key to successful architecture transformation in Phases E and F. This will be a joint effort between corporate (especially human resources) staff, lines of business, and IT planners.

The recommended activities in an assessment of an organization's readiness to address business transformation are:

- Determine the readiness factors that will impact the organization
- Present the readiness factors using maturity models
- Assess the readiness factors, including determination of readiness factor ratings
- Assess the risks for each readiness factor and identify improvement actions to mitigate the risk
- Work these actions into Phase E and F Implementation and Migration Plan

8.1.1 Business Transformation Enablement Program (BTEP)

The Canadian Government Business Transformation Enablement Program (BTEP) provides guidance on how to identify the business transformation-related issues.

The BTEP recommends that all projects conduct a transformation readiness assessment to at least uncover the business transformation issues. This assessment is based upon the determination and analysis/rating of a series of readiness factors. The outcome is a deeper understanding of the challenges and opportunities that could be presented in the course of the endeavor. Many of the challenges translate directly into risks that have to be addressed, monitored, and, if possible, mitigated.

The following sections describe Business Transformation Readiness Assessment using the BTEP method, including some lessons learned. Readers should keep in mind that most organizations will have their own unique set of factors and criteria, but most are similar.

8.2 Determine Readiness Factors

The first step is to determine what factors will impact on the business transformation associated with the migration from the Baseline to Target Architectures.

This can be best achieved through the conduct of a facilitated workshop with individuals from different parts of the organization. It is important that all perspectives are sought as the issues will be varied. In this workshop it is very useful to start off with a tentative list of factors that participants can re-use, reject, augment, or replace.

An example set of factors drawn from the BTEP follows:

- **Vision** is the ability to clearly define and communicate what is to be achieved

This is where management is able to clearly define the objectives, in both strategic and specific terms. Leadership in defining vision and needs comes from the business side with IT input. Predictable and proven processes exist for moving from vision to statement of requirements. The primary drivers for the initiative are clear. The scope and approach of the transformation initiative have been clearly defined throughout the organization.

- **Desire, Willingness, and Resolve** is the presence of a desire to achieve the results, willingness to accept the impact of doing the work, and the resolve to follow through and complete the endeavor

There is active discussion regarding the impact that executing the project may have on the organization, with a clear indication of the intent to accept the impacts. Key resources (e.g., financial, human, etc.) are allocated for the endeavor and top executives project the clear message that the organization will follow through; a message that identifies the effort as well as the benefits. Organizationally there is a history of finishing what is started and of coming to closure on issues in the timeframes needed and there is agreement throughout the organization that the transformation initiative is the "right" thing to do.

- **Need**, in that there is a compelling need to execute the endeavor

There are clear statements regarding what the organization will not be able to do if the project does not proceed, and equally clear statements of what the project will enable the organization to do. There are visible and broadly understood consequences of endeavor failure and success criteria have been clearly identified and communicated.

- **Business Case** exists that creates a strong focus for the project, identifying benefits that must be achieved and thereby creating an imperative to succeed

The business case document identifies concrete benefits (revenues or savings) that the organization is committed to deliver and clearly and unquestionably points to goals that the organization is committed to achieving.

- **Funding**, in the form of a clear source of fiscal resources, exists that meets the endeavor's potential expenditures
- **Sponsorship and Leadership** exists and is broadly shared, but not so broad as to diffuse accountability

Leadership keeps everyone "on board" and keeps all focused on the strategic goals. The endeavor is sponsored by an executive who is appropriately aligned to provide the leadership the endeavor needs and able to articulate and defend the needs of the endeavor at the senior management level. These executive sponsors are and will remain engaged throughout.

- **Governance** is the ability to engage the involvement and support of all parties with an interest in or responsibility to the endeavor with the objective of ensuring that the corporate interests are served and the objectives achieved

There are clearly identified stakeholders and a clear sense of their interest in and responsibility to the project; a culture that encourages participation towards corporate rather than local objectives; a history of being able to successfully manage activities that cross interest areas; a culture that fosters meaningful, as opposed to symbolic, participation in management processes; and a commitment to ongoing project review and challenge and openness to outside advice.

- **Accountability** is the assignment of specific and appropriate responsibility, recognition of measurable expectations by all concerned parties, and alignment of decision-making with areas of responsibility and with where the impact of the decisions will be felt

Accountability is aligned with the area where the benefits of success or consequences of failure of the endeavor will be felt as well as with the responsibility areas.

- **Workable Approach and Execution Model** is an approach that makes sense relative to the task, with a supporting environment, modeled after a proven approach

There are clear notions of the client and the client's role relative to the builder or prime contractor and the organization is experienced with endeavors of this type so that the processes, disciplines, expertise, and governance are already in place, proven, and available to apply to the transformation endeavor. All the players know their roles because they have played them before with success. In particular, the roles of "client" and "systems builder" are mature and stable. There is a communication plan covering all levels of the organization and meeting the needs ranging from awareness to availability of technical detail. There is a reward and recognition plan in place to recognize teams and individuals who use good change management practices, planning and prevention of crisis behaviors, and who reinforce behaviors appropriate to the new way of doing business. It is clear to everyone how implementation will occur, how it will be monitored, and how realignment actions will be made and there are adequate resources dedicated for the life of the transformation.

- **IT Capacity to Execute** is the ability to perform all the IT tasks required by the project, including the skills, tools, processes, and management capability

There has been a recent successful execution of a similar endeavor of similar size and complexity and there exist appropriate processes, discipline, skills, and a rationale model for deciding what skills and activities to source externally.

- **Enterprise Capacity to Execute** is the ability of the enterprise to perform all the tasks required by the endeavor, in areas outside of IT, including the ability to make decisions within the tight time constraints typical to project environments based upon the recent successful execution of a similar endeavor of at least half the size and complexity

There exist non-IT-specific processes, discipline, and skills to deal with this type of endeavor. The enterprise has a demonstrated ability to deal with the type of ongoing project/portfolio management issues and requirements. There is a recognition of the need for knowledge and skill-building for the new way of working as well as the value of a formal gap analysis for skills and behavior.

- **Enterprise Ability to Implement and Operate** the transformation elements and their related business processes, absorb the changes arising from implementation, and ongoing ability to operate in the new environment

The enterprise has a recent proven ability to deal with the change management issues arising from new processes and systems and has in place a solid disciplined and process-driven service management program that provides operations, maintenance, and support for existing systems.

Once the factors have been identified and defined, it is useful to call a follow-on workshop where the factors shall be assessed in some detail in terms of their impact/risk. The next section will deal with preparing for an effective assessment of these factors.

8.3 Present Readiness Factors

Once the factors are determined, it is necessary to present them in such a way that the assessment is clear and the maximum value is derived from the participants.

One such presentation is through the use of maturity models. If each factor is converted into a maturity model (a re-usable governance asset as well) accompanied by a standard worksheet template containing all of the information and deductions that have to be gathered, it can be a very useful tool.

The maturity model should enable participants to:

- Assess their current (Baseline Architecture) maturity level
- Determine the target maturity level that would have to be achieved to realize the Target Architecture
- Determine an intermediate target that would be achievable in a lesser timeframe

The care spent preparing the models (which is not insignificant) will be recouped by a focused workshop that will rapidly go through a significant number of factors.

It is important that each factor be well-defined and that the scope of the Enterprise Architecture endeavor (preliminary planning) be reflected in the models to keep the workshop participants focused and productive.

Circulating the models before the workshop for comments would be useful, if only to ensure that they are complete as well as allowing the participants to prepare for the workshop. Note that the model shown below also has a recommended target state put in by the Enterprise Architect; this again acts as governance.

An example of a maturity model is shown in [Figure 8-1](#) for one of the BTEP factors.

Business Transformation Readiness Assessment - Maturity Model					
Factor 2: Need for Enterprise Information Architecture			Class	Organizational Context	
			BTEP Readiness Factor	YES	
Definition	There is recognition by the organization that information is a strategic corporate asset requiring stewardship. There is also recognition that the data is not universally understandable, of requisite quality, and accessible.				
Maturity Model Levels					
0 Not defined	1 <i>Ad Hoc</i>	2 Repeatable	3 Defined	4 Managed	5 Optimized
Information is not recognized as an asset. There is no clear stewardship of data.	Data Management (DM) concepts are intuitively understood and practiced on an <i>ad hoc</i> basis. Stewardship of the data is informal. Data is recognized by certain internal experts and senior management as being of strategic importance to the organization. Focus is primarily on technically managing redundant data at the applications level.	Many parts of the organization value information/data as a strategic asset. Internal DM experts maintain clear lines of responsibility and stewardship of the data, organized along lines of business and at all senior levels. Staff put into practice DM principles and standards in their daily activities.	Data is recognized as a strategic asset in most parts of the organization, and throughout most levels from operations to senior management. Resources are committed to ensuring strong stewardship of data at the lower management and information expert levels.	Data is recognized as a strategic asset in all parts of the organization, and throughout most levels from operations to senior management. Resources are committed to ensuring strong stewardship of data at the senior management and information expert levels.	Data is treated in all levels throughout the organization as a strategic asset to be exploited and re-used. Data products and services are strongly integrated with the management practice of the organization. All staff are empowered and equipped to take stewardship of information, and are seen as "knowledge workers".
				Recommended Target State	

© The Open Group

Figure 8-1 Business Transformation Readiness Assessment — Maturity Model

8.4 Assess Readiness Factors

Ideally, the factors should be assessed in a multi-disciplinary workshop. Using a mechanism such as maturity models, Enterprise Architects will normally have to cover a great deal of ground in little time.

The use of a series of templates for each factor would expedite the assessment, and ensure consistency across the wide range of factors.

The assessment should address three things, namely:

- Readiness Factor Vision
- Readiness Factor Rating
- Readiness Factor Risks & Actions

8.4.1 Readiness Factor Vision

The vision for a readiness factor is the determination of where the enterprise has to evolve to address the factor. First, the factor should be assessed with respect to its base state and then its target state.

For example, if the "IT capacity to execute" factor is rated as low, the factor should ideally be at "high" to realize the Target Architecture Vision. An intermediate target might be useful to direct the implementation. Maturity models are excellent vehicles to guide this determination.

8.4.2 Readiness Factor Rating

Once the factor visions are established, then it is useful to determine how important each factor is to the achievement of the Target Architecture as well as how challenging it will be to migrate the factor into an acceptable visionary state.

The BTEP uses a Readiness Rating Scheme that can be used as a start point for any organization in any vertical. Each one of the readiness factors are rated with respect to:

- **Urgency**, whereby if a readiness factor is urgent, it means that action is needed before a transformation initiative can begin
- **Readiness Status**, which is rated as either Low (needs substantial work before proceeding), Fair (needs some work before proceeding), Acceptable (some readiness issues exist; no showstoppers), Good (relatively minor issues exist), or High (no readiness issues)
- **Degree of Difficulty to Fix** rates the effort required to overcome any issues identified as either No Action Needed, Easy, Moderate, or Difficult

Although a more extensive template can be used in the workshop, it is useful to create a summary table of the findings to consolidate the factors and provide a management overview. A summary is shown in [Figure 8-2](#).

Business Factor Assessment Summary				
Ser	Readiness Factor	Urgency	Readiness Status	Degree of Difficulty to Fix
1	Vision			
2	Desire/willingness/resolve			
3	Need			
4	Business case			
5	Funding			
6	Sponsorship and leadership			
7	Governance			
8	Accountability			
9	Workable approach and execution model			
10	IT capacity to execute			
11	Departmental capacity to execute			
12	Ability to implement and operate			

© The Open Group

Figure 8-2 Summary Table of Business Transformation Readiness Assessment

8.4.3 Readiness Factor Risks & Actions

Once the factors have been rated and assessed, derive a series of actions that will enable the factors to change to a favorable state.

Each factor should be assessed with respect to risk using the process highlighted in [Chapter 9](#), including an estimate of impact and frequency.

Each factor should be discretely assessed and a series of improvement actions outlined. Before starting anew, existing actions outlined in the architectures should be checked first before creating new ones.

These newly identified actions should then be formally incorporated into the emerging Implementation and Migration Plan.

From a risk perspective, these actions are designed to mitigate the risks and produce an acceptable residual risk. As risks, they should be part of the risk management process and closely monitored as the Enterprise Architecture is being implemented.

8.5 Readiness and Migration Planning

The assessment exercise will provide a realistic assessment of the organization and will be a key input into the strategic migration planning that will be initiated in Phase E and completed in Phase F. It is important to note whether the business transformation actions will be on the vision's critical path and, if so, determine how they will impact implementation. There is no point deploying new IT capability without employees trained to use it and support staff ready to sustain it.

The readiness factors, as part of an overall Implementation and Migration Plan, will have to be continuously monitored (Phase G) and rapid corrective actions taken through the IT governance framework to ensure that the defined architectures can be implemented.

The readiness factors assessment will be a living document and during the migration planning and execution of the Transition Architectures, the business transformation activities will play a key role.

8.6 Marketing the Implementation Plan

The Architecture Definition should not be widely circulated until the business transformation issues are identified and mitigated, and the associated actions part of an overall "marketing" plan for the vision and the Implementation and Migration Plan.

For example, the consolidation of information holdings could result in hundreds of lost jobs and this vision should not be announced before a supporting business transformation/human resources plan is formulated to retrain or support the workers' quest for new employment.

The business transformation workshops are a critical part of the Communications Plan whereby key individuals from within the organization gather to assess the implications of transforming the enterprise. To do this they will become aware of the Architecture Vision and architecture definition (if they were not already involved through the business scenarios and Business Architecture). This group will feel ownership of the Enterprise Architecture, recognizing the Enterprise Architect as a valuable steward.

Their determination of the factors will again create a culture of understanding across the enterprise and provide useful insights for the Implementation and Migration Plan.

The latter plan should include a Communications Plan, especially to keep the affected personnel informed. In many cases collaborating with the unions and shop stewards will further assist a humane (and peaceful) transition to the target state.

8.7 Conclusion

In short, Enterprise Architecture implementation will require a deep knowledge and awareness of all of the business transformation factors that impact transitioning to the visionary state. With the evolution of IT, the actual technology is not the real issue any more in Enterprise Architecture, but the critical factors are most often the cultural ones. Any Implementation and Migration Plan has to take both into consideration. Neglecting these and focusing on the technical aspects will invariably result in an implementation that falls short of realizing the real promise of a visionary Enterprise Architecture.

Chapter 9: Risk Management

This chapter describes risk management, which is a technique used to mitigate risk when implementing an architecture project.

9.1 Introduction

There will always be risk with any architecture/business transformation effort. It is important to identify, classify, and mitigate these risks before starting so that they can be tracked throughout the transformation effort.

Mitigation is an ongoing effort and often the risk triggers may be outside the scope of the transformation planners (e.g., merger, acquisition) so planners must monitor the transformation context constantly.

It is also important to note that the Enterprise Architect may identify the risks and mitigate certain ones, but it is within the governance framework that risks have to be first accepted and then managed.

There are two levels of risk that should be considered, namely:

1. **Initial Level of Risk:** risk categorization prior to determining and implementing mitigating actions
2. **Residual Level of Risk:** risk categorization after implementation of mitigating actions (if any)

The process for risk management is described in the following sections and consists of the following activities:

- Risk classification
- Risk identification
- Initial risk assessment
- Risk mitigation and residual risk assessment
- Risk monitoring

9.2 Risk Classification

Risk is pervasive in any Enterprise Architecture activity and is present in all phases within the ADM. From a management perspective, it is useful to classify the risks so that the mitigation of the risks can be executed as expeditiously as possible.

One common way for risks to be classified is with respect to impact on the organization (as discussed in [Section 9.4](#)), whereby risks with certain impacts have to be addressed by certain levels of governance.

Risks are normally classified as time (schedule), cost (budget), and scope but they could also include client transformation relationship risks, contractual risks, technological risks, scope and complexity risks, environmental (corporate) risks, personnel risks, and client acceptance risks.

Another way of delegating risk management is to further classify risks by architecture domains. Classifying risks as business, information, applications, and technology is useful but there may be organizationally-specific ways of expressing risk that the corporate Enterprise Architecture directorate should adopt or extend rather than modify.

Ultimately, Enterprise Architecture risks are corporate risks and should be classified and as appropriate managed in the same or extended way.

9.3 Risk Identification

The maturity and transformation readiness assessments will generate a great many risks. Identify the risks and then determine the strategy to address them throughout the transformation.

The use of Capability Maturity Models (CMMs) is suitable for specific factors associated with architecture delivery to first identify baseline and target states and then identify the actions required to move to the target state. The implications of *not* achieving the target state can result in the discovery of risks. Refer to [Chapter 8](#) for specific details.

Risk documentation is completed in the context of a Risk Management Plan, for which templates exist in standard project management methodologies — e.g., Project Management Body of Knowledge (PMBOK®) and PRINCE2® — as well as with the various government methodologies.

Normally these methodologies involve procedures for contingency planning, tracking and evaluating levels of risk, reacting to changing risk level factors, as well as processes for documenting, reporting, and communicating risks to stakeholders.

9.4 Initial Risk Assessment

The next step is to classify risks with respect to effect and frequency in accordance with scales used within the organization. Combine effect and frequency to come up with a preliminary risk assessment.

There are no hard and fast rules with respect to measuring effect and frequency. The following guidelines are based upon existing risk management best practices.

Effect could be assessed using the following example criteria:

- **Catastrophic** infers critical financial loss that could result in bankruptcy of the organization
- **Critical** infers serious financial loss in more than one line of business leading to a loss in productivity and no return on investment on the IT investment
- **Marginal** infers a minor financial loss in a line of business and a reduced return on investment on the IT investment
- **Negligible** infers a minimal impact on a line of business' ability to deliver services and/or products

Frequency could be indicated as follows:

- **Frequent**: likely to occur very often and/or continuously
- **Likely**: occurs several times over the course of a transformation cycle
- **Occasional**: occurs sporadically
- **Seldom**: remotely possible and would probably occur not more than once in the course of a transformation cycle
- **Unlikely**: will probably not occur during the course of a transformation cycle

Combining the two factors to infer impact would be conducted using a heuristically-based but consistent classification scheme for the risks. A potential scheme to assess corporate impact could be as follows:

- **Extremely High Risk (E)**: the transformation effort will most likely fail with severe consequences
- **High Risk (H)**: significant failure of parts of the transformation effort resulting in certain goals not being achieved
- **Moderate Risk (M)**: noticeable failure of parts of the transformation effort threatening the success of certain goals
- **Low Risk (L)**: certain goals will not be wholly successful

These impacts can be derived using a classification scheme, as shown in [Figure 9-1](#).

Corporate Risk Impact Assessment					
Effect	Frequency				
	Frequent	Likely	Occasional	Seldom	Unlikely
Catastrophic	E	E	H	H	M
Critical	E	H	H	M	L
Marginal	H	M	M	L	L
Negligible	M	L	L	L	L

© The Open Group

Figure 9-1 Risk Classification Scheme

9.5 Risk Mitigation and Residual Risk Assessment

Risk mitigation refers to the identification, planning, and conduct of actions that will reduce the risk to an acceptable level.

The mitigation effort could be a simple monitoring and/or acceptance of the risk to a full-blown contingency plan calling for complete redundancy in a Business Continuity Plan (with all of the associated scope, cost, and time implications).

Due to the implications of this risk assessment, it has to be conducted in a pragmatic but systematic manner. With priority going to frequent high impact risks, each risk has to be mitigated in turn.

9.6 Conduct Residual Risk Assessment

Once the mitigation effort has been identified for each one of the risks, re-assess the effect and frequency and then recalculate the impacts and see whether the mitigation effort has really made an acceptable difference. The mitigation efforts will often be resource-intensive and a major outlay for little or no residual risk should be challenged.

Once the initial risk is mitigated, then the risk that remains is called the "residual risk". The key consideration is that the mitigating effort actually reduces the corporate impact and does not just move the risk to another similarly high quadrant. For example, changing the risk from frequent/catastrophic to frequent/critical still delivers an extremely high risk. If this occurs, then the mitigation effort has to be re-considered.

The final deliverable should be a transformation risk assessment that could be structured as a worksheet, as shown in [Figure 9-2](#).

Risk ID	Risk	Preliminary Risk			Mitigation	Residual Risk		
		Effect	Frequency	Impact		Effect	Frequency	Impact

© The Open Group

Figure 9-2 Sample Risk Identification and Mitigation Assessment Worksheet

9.7 Risk Monitoring and Governance (Phase G)

The residual risks have to be approved by the IT governance framework and potentially in corporate governance where business acceptance of the residual risks is required.

Once the residual risks have been accepted, then the execution of the mitigating actions has to be carefully monitored to ensure that the enterprise is dealing with residual rather than initial risk. The risk identification and mitigation assessment worksheets are maintained as governance artifacts and are kept up-to-date in Phase G (Implementation Governance) where risk monitoring is conducted.

Implementation governance can identify critical risks that are not being mitigated and might require another full or partial ADM cycle.

9.8 Summary

Risk management is an integral part of Enterprise Architecture. Practitioners are encouraged to use their corporate risk management methodology or extend it using the guidance in this chapter. In the absence of a formal corporate methodology, architects can use the guidance in this chapter as a best practice.

Chapter 10: Architecture Alternatives and Trade-Offs

This chapter describes a technique to identify alternative Target Architectures and perform trade-offs between the alternatives.

10.1 Concept

There is often more than one possible Target Architecture that would conform to the Architecture Vision, Architecture Principles, and Requirements. It is important to identify alternative Target Architectures and build understanding of different possibilities and identify trade-offs between the alternatives. Creating an architecture normally requires trade-offs among competing forces. Presenting different alternatives and trade-offs to stakeholders helps architects to extract hidden agendas, principles, and requirements that could impact the final Target Architecture.

10.2 Method

It is most common that a single alternative does not exist that will meet all stakeholders' concerns. The TOGAF Standard supports a technique to investigate different alternatives and to discuss these with the stakeholders. Commonly, alternatives are defined per domain. This is done to simplify the analysis of the different alternatives. Of course, the alternatives per domain can be merged into an overall analysis of the alternatives for the whole architecture.

Figure 10-1 illustrates the architecture trade-off method.

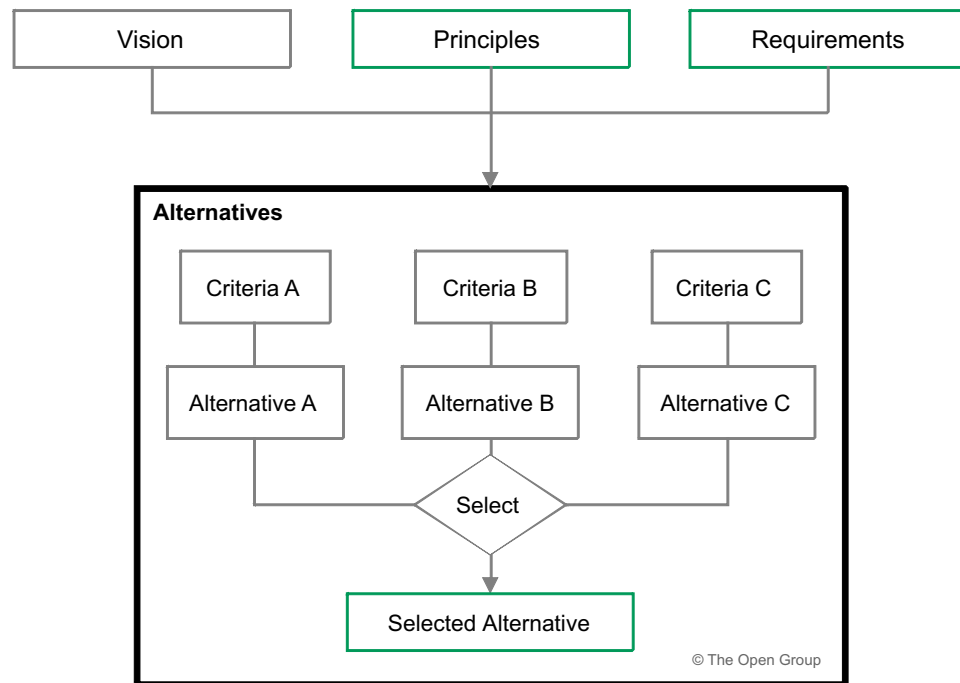


Figure 10-1 Architecture Trade-Off Method

The first part of the method uses the vision, principles, requirements, and other information to select sets of criteria fitting for different alternatives.

This second part of the method defines alternatives based on the criteria and builds understanding of each.

The third part of this method will either select one of the alternatives, or else combine features from more than one, to create the proposed alternative. Perform the following activities in just enough detail to support that decision. The method can be used for any phase at any level of an architecture.

10.2.1 Criteria

The criteria are used for the different alternatives and are derived from many different inputs to the architecture. Consider the influence of architecture principles, requirements, vision, and stakeholder concerns.

Each alternative will have distinct advantages or disadvantages that will need to be discussed and agreed with stakeholders. Additional viewpoints and views may be needed to allow stakeholders to explore the alternatives and understand any dependencies, risks, and uncertainties.

Typical examples of alternative types (based on criteria) include:

- Flexible alternative
- Time and cost of realizing the alternative, including any transitions and plateaus ("islands of stability")
- Time period over which estimated benefits of the alternative will be achieved
- Adherence to architecture styles or guidelines
- Solution delivery method (e.g., re-use, develop, buy)
- Minimal impact on business capabilities during implementation of the alternative
- Minimized risk associated with the alternative and any mitigating actions needed
- etc., . . .

10.2.2 Identify Alternatives

Identify a set of possible alternatives using the Architecture Vision, Principles, and Requirements.

For each alternative:

1. Define the overview criteria for the alternative
Use the Architecture Vision, Principles, and Requirements to define the criteria for the alternative. The criteria can be applied at different abstraction levels and ADM phases to identify different architecture alternatives.
2. Describe the architecture for the alternative
Create a set of necessary architecture views to reach a proper understanding of the impact of the alternative. Add any other needed information. Do not go into too much detail. It is important, however, to carry out a good impact assessment and identify interdependencies between alternatives and the existent landscape and have a complete picture of the implications of the alternative implementation.
3. Estimate gaps between the baseline and this alternative
Based on current understanding of the baseline state, outline the gaps that exist between the baseline and this alternative. If the baseline has not yet been defined, this gap analysis will be informal. More detail on how to do gap analysis appears in [Chapter 5](#).
4. Understand the impacts and trade-offs of the alternative across the Architecture Landscape:
 - Identify the impact that the alternative will have on any existing architectures, and on any Transition Architecture within the Architecture Landscape
 - Identify the impact the alternative will have on any running or planned implementation projects
 - Identify constraints imposed on this alternative by any running or planned implementation projects

- Identify impacts on architecture in other ADM phases in this architecture project
- Identify Architecture Requirements/Change Requests from this architecture that will constrain other architectures
- Identify the final value delivered by the alternative, to what extent it covers the gap to reach the future state, and the purpose of the iteration

10.2.3 Choose from Alternatives and Define in Detail

This step draws on the alternatives to select or define an alternate alternative. Use trade-off analysis to resolve conflicts between alternatives:

1. Understand the strengths and weaknesses of each alternative
2. Compare the alternatives based on how well they align with the defined criteria
3. Select the most suitable alternative or combine features from more than one of the alternatives, to define an alternate alternative in collaboration with stakeholders
4. Assemble the alternative:
 - Finalize the description of the alternative
 - Ensure that all the architecture viewpoints identified have been worked through for the alternative
 - Ensure that the alternative is defined in enough detail to support decision-making
5. Resolve impacts across the Architecture Landscape
6. Conduct formal stakeholder review to determine alternative decision and funding

The impact analysis of the alternatives was done in just enough detail to choose between them.

Index

ABB.....	39
application principles	
examples	16
architecture alternatives.....	73
Architecture Continuum	42
architecture pattern.....	39, 41
content	40
Architecture Principles	3
characteristics	4
components.....	4
criteria	5
architecture view	22, 42
architecture viewpoint	22
BTEP	59-60
Readiness Rating Scheme.....	64
Business Continuity Plan	70
business principles	
examples	7
Business Transformation Readiness Assessment.....	59
concern	22
COTS	16
data principles	
examples	11
Degrees of Interoperability.....	56
design pattern	41
enterprise operating model.....	55
enterprise principles	3
gap analysis	45
GOTS.....	16
governance	70
Hillside Group	42
idiom	42
initial risk assessment.....	68
interoperability	
categories.....	53
refining	55
requirements	53
Java	16
maturity model	62
Migration Planning	66
pattern resources.....	42
patterns home page.....	42
patterns-discussion FAQ.....	42
readiness factors.....	60

readiness planning.....	66
residual risk assessment	70
risk assessment	68
risk classification	68
risk identification	68
risk management	67
best practice	69
Risk Management Plan.....	68
risk monitoring	70
SANS Institute	11
SBB	39
SMART	55
stakeholder	22
stakeholder management	21
process steps	22
steward	13
technology principles	
examples	17
transformation readiness assessment	60
TRM	46
trustee.....	13