

The Open Group Standard

TOGAF® Standard — Enterprise Architecture Capability and Governance

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 — Enterprise Architecture Capability and Governance

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	Establishing an Architecture Capability	3
	2.1	Overview.....	3
	2.2	Phase A: Architecture Vision	4
	2.3	Phase B: Business Architecture	5
	2.4	Phase C: Data Architecture	6
	2.5	Phase C: Application Architecture	6
	2.6	Phase D: Technology Architecture	6
	2.7	Phase E: Opportunities & Solutions	6
	2.8	Phase F: Migration Planning	6
	2.9	Phase G: Implementation Governance	6
	2.10	Phase H: Architecture Change Management	7
	2.11	Requirements Management	7
Chapter	3	Architecture Governance	9
	3.1	Introduction.....	9
	3.1.1	Levels of Governance within the Enterprise	9
	3.1.2	Nature of Governance	10
	3.1.3	Technology Governance	11
	3.1.4	IT Governance	11
	3.1.5	Architecture Governance: Overview	12
	3.2	Architecture Governance Framework	13
	3.2.1	Architecture Governance Framework — Conceptual Structure	13
	3.2.2	Architecture Governance Framework — Organizational Structure	16
	3.3	Architecture Governance in Practice	18
	3.3.1	Architecture Governance — Key Success Factors	18
	3.3.2	Elements of an Effective Architecture Governance Strategy	19
Chapter	4	Architecture Board.....	21
	4.1	Role	21
	4.2	Responsibilities.....	21
	4.3	Setting Up the Architecture Board	22
	4.3.1	Triggers.....	22
	4.3.2	Size of the Board	23
	4.3.3	Board Structure	24
	4.4	Operation of the Architecture Board	24
	4.4.1	General	24
	4.4.2	Preparation	25

	4.4.3	Agenda	25
Chapter	5	Architecture Contracts.....	27
	5.1	Role	27
	5.2	Contents	29
	5.2.1	Statement of Architecture Work.....	29
	5.2.2	Contract between Architecture Design and Development Partners	29
	5.2.3	Contract between Architecting Function and Business Stakeholders	30
	5.3	Relationship to Architecture Governance	30
Chapter	6	Architecture Compliance.....	31
	6.1	Introduction.....	31
	6.2	Terminology: The Meaning of Architecture Compliance	31
	6.3	Architecture Compliance Reviews	33
	6.3.1	Purpose	33
	6.3.2	Timing	34
	6.3.3	Governance and Personnel Scenarios	35
	6.4	Architecture Compliance Review Process	36
	6.4.1	Overview.....	36
	6.4.2	Roles	37
	6.4.3	Steps	38
	6.5	Architecture Compliance Review Checklists	39
	6.5.1	Hardware and Operating System Checklist.....	39
	6.5.2	Software Services and Middleware Checklist.....	40
	6.5.3	Applications Checklists	42
	6.5.4	Information Management Checklists	44
	6.5.5	Security Checklist	45
	6.5.6	System Management Checklist	46
	6.5.7	System Engineering/Overall Architecture Checklists	47
	6.5.8	System Engineering/Methods & Tools Checklist	50
	6.6	Architecture Compliance Review Guidelines	51
	6.6.1	Tailoring the Checklists	51
	6.6.2	Conducting Architecture Compliance Reviews.....	52
		Index	53

List of Figures

3-1	Architecture Governance Framework — Conceptual Structure	13
3-2	Architecture Governance Framework — Organizational Structure	16
6-1	Levels of Architecture Conformance.....	31
6-2	Architecture Compliance Review Process.....	36

Contents

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 — Enterprise Architecture Capability and Governance.

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.

COBIT is a registered trademark of the Information Systems Audit and Control Association (ISACA) and the IT Governance Institute.

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 — Enterprise Architecture Capability and Governance (this document).

In order to successfully operate an architecture function within an enterprise, it is necessary to put in place appropriate organization structures, processes, roles, responsibilities, and governance.

Guidelines included within this document are as follows:

- Establishing an Architecture Capability (see [Chapter 2](#)) provides guidelines on how to use the ADM to establish an Architecture Capability
- Architecture Governance (see [Chapter 3](#)) provides a framework and guidelines for Architecture Governance
- Architecture Board (see [Chapter 4](#)) provides guidelines for establishing and operating an Enterprise Architecture Board
- Architecture Contracts (see [Chapter 5](#)) provides guidelines for defining and using Architecture Contracts
- Architecture Compliance (see [Chapter 6](#)) provides guidelines for ensuring project compliance to the architecture

Chapter 2: Establishing an Architecture Capability

This chapter provides guidelines on how to use the ADM to establish an Architecture Capability.

2.1 Overview

As with any business capability, the establishment of an Enterprise Architecture Capability can be supported by the TOGAF Architecture Development Method (ADM). Successful use of the ADM will provide a customer-focused, value-adding, and sustainable architecture practice that enables the business, helps maximize the value of investments, and pro-actively identifies opportunities to gain business benefits and manage risk.

Establishing a sustainable architecture practice within an organization can be achieved by adhering to the same approach that is used to establish any other capability — such as a Business Process Management (BPM) capability — within an organization. The ADM is an ideal method to be used to architect and govern the implementation of such a capability. Applying the ADM with the specific Architecture Vision to establish an architecture practice within the organization would achieve this objective.

The establishment of the architecture practice should not be seen as a phase of an architecture project, or a one-off project, but rather as an ongoing discipline that provides the context, environment, and resources to govern and enable architecture delivery to the organization. As an architecture project is executed within this environment it might request a change to the architecture practice that would trigger another cycle of the ADM to extend the architecture practice.

Implementing any capability within an organization would require the design of the four domain architectures: Business, Data, Application, and Technology. Establishing the architecture practice within an organization would therefore require the design of:

- The **Business Architecture** of the architecture practice that will highlight the Architecture Governance, architecture processes, architecture organizational structure, architecture information requirements, architecture products, etc.
- The **Data Architecture** that would define the structure of the organization's Enterprise Continuum and Architecture Repository
- The **Application Architecture** specifying the functionality and/or applications services required to enable the architecture practice
- The **Technology Architecture** that depicts the architecture practice's infrastructure requirements and deployment in support of the architecture applications and Enterprise Continuum

The steps in establishing an architecture practice are explained below, against the context of the ADM phases. The reader should therefore refer to the relevant ADM phase in the TOGAF Standard — Architecture Development Method to understand the complete scope of each step.

In this section, key aspects will be highlighted for each ADM phase that should be considered and are specific to establishing an architecture practice. The intent is therefore not to repeat each ADM phase description, but to guide the reader to apply each ADM phase within the context of establishing an architecture practice.

2.2 Phase A: Architecture Vision

The purpose of this phase within the context of establishing an architecture practice is to define or review the vision, stakeholders, and principles of the architecture practice. The focus in this phase would be on the architecture practice as a whole and not on a particular architecture project.

The following should be considered in terms of understanding the steps in the context of establishing an architecture practice:

- **Establish the Project:** this step should focus on defining the stakeholders in the architecture practice

The stakeholders would include the roles and organization units participating in the architecture practice, as well as people who will benefit from the deliverables generated by the architecture practice and who can therefore be defined as customers of the architecture practice.

- **Identify Stakeholders and Concerns, Business Requirements, and Architecture Vision:** this step generates the first, very high-level definitions of the baseline and target environments, from the perspectives of business information systems and technology, for the architecture practice
- **Identify Business Goals and Business Drivers:** an understanding of the business goals and drivers is essential to align the architecture practice to the business
- **Define Scope:** defining the scope of the architecture practice is a high-level project plan of what is scheduled to be addressed in terms of architecture for the next period
- **Define Constraints:** the focus in this step is the enterprise-wide constraints that impact on all architecture projects
- **Review Architecture Principles, including Business Principles:** the intent in this step is to define the principles that govern and guide the running of the architecture practice

Where Architecture Principles usually govern the architecture deliverables, the architecture practice principles address the architecture practice organization, content, tools, and process.
- **Develop Statement of Architecture Work and Secure Approval:** this step generates the architecture practice vision and scope

Another step that can be considered during this phase is to conduct an architecture maturity assessment. Refer to the TOGAF® Series Guide: Architecture Maturity Models for guidance on this topic.

2.3 Phase B: Business Architecture

Key areas of focus during this phase of establishing or refining the Business Architecture of the architecture practice are:

- An **Architecture Ontology** defining the architectural terms and definitions that will be used in the organization in order to establish a common understanding of these terms
- The **Architecture Process** where the ADM would form the base of the process and need to be customized to meet the organization's requirements and architecture practice vision

Refer to the TOGAF Standard — Architecture Development Method for guidance on developing this process. The required Architecture Governance processes should be included in the overall architecture process.

- The **Architecture Viewpoints and Views** that list all the viewpoints and views that should be addressed by the architecture practice

The identified architecture practice stakeholders would guide the development of this definition. One of the viewpoints to be included is the Architecture Governance viewpoint; refer to the TOGAF Standard — Architecture Content for guidance on this output.

- The **Architecture Framework** describing the various architecture deliverables that will be generated by the architecture practice, the inter-relationships and dependencies between the architecture deliverables, as well as the rules and guidelines governing the design of these deliverables

The defined architecture viewpoints and views should be used to guide the definition of the architecture framework. The TOGAF Standard — Architecture Development Method and the TOGAF Standard — Architecture Content are useful references that will assist in describing the architecture framework.

- The **Architecture Accountability Matrix** defining the roles in the architecture practice and allocating accountability of the roles to architecture deliverables and processes

This matrix would include the required Architecture Governance structures and roles. The TOGAF Standard — Architecture Development Method as well as [Chapter 4](#), [Chapter 3](#), and the TOGAF® Series Guide: Architecture Skills Framework would provide guidance on this output.

- The **Architecture Performance Metrics** identifying and describing the metrics that will be used to monitor the performance of the architecture practice against its stated architecture practice vision and objectives
- The **Architecture Governance Framework** which is a specific view of the defined architecture process and Architecture Accountability Matrix

2.4 Phase C: Data Architecture

The Data Architecture of the architecture practice would specify and govern the structure of the organization's Enterprise Continuum and Architecture Repository. The Data Architecture should be defined based on the architecture framework. The Data Architecture is sometimes referred to as the metamodel of the architecture practice.

2.5 Phase C: Application Architecture

The Application Architecture of the architecture practice defines the functionality required to generate, maintain, publish, distribute, and govern the architecture deliverables as defined in the architecture framework. A key focus should be on the modeling toolsets required for modeling, but it should not be the only focus. Refer to the TOGAF Standard — Architecture Development Method for guidance on selecting a toolset. Publishing the architecture deliverables to address specific views in the architecture framework would sometimes require specialized or customized functionality and should not be neglected.

2.6 Phase D: Technology Architecture

The Technology Architecture of the architecture practice should define technology infrastructure supporting the architecture practice.

2.7 Phase E: Opportunities & Solutions

A critical factor to consider during this phase of planning the establishment of the architecture practice is the organizational change that is required and how this will be achieved.

2.8 Phase F: Migration Planning

The focus should not only be on the Information Systems Architecture components in this phase, but include the Business Architecture. The adoption of the architecture process and framework will have a major impact on the overall establishment of the architecture practice in the organization.

2.9 Phase G: Implementation Governance

The implementation of the Business Architecture of the architecture practice should be the focus of this phase. Changing practices within the organization to adopt a more structured and disciplined approach will be a challenge and should be addressed by the appropriate organizational change techniques.

2.10 Phase H: Architecture Change Management

Changes to the architecture of the architecture practice should be managed by this phase. These changes are usually triggered during the execution of architecture projects. A typical change would be the requirement for a new architecture deliverable. This would impact on all the architecture domains of the architecture practice.

2.11 Requirements Management

Understanding and managing the requirements for the architecture practice is crucial. Requirements should be clearly articulated and align to the architecture practice vision.

Additional guidance on how to construct an Enterprise Architecture Capability can be found in the TOGAF® Series Guide: The TOGAF Leader's Guide to Establishing and Evolving an EA Capability.

Chapter 3: Architecture Governance

This chapter provides a framework and guidelines for Architecture Governance.

3.1 Introduction

This section describes the nature of governance, and the levels of governance.

3.1.1 Levels of Governance within the Enterprise

Architecture Governance is the practice and orientation by which Enterprise Architectures and other architectures are managed and controlled at an enterprise-wide level.

Architecture Governance typically does not operate in isolation, but within a hierarchy of governance structures, which, particularly in the larger enterprise, can include all of the following as distinct domains with their own disciplines and processes:

- Corporate Governance
- Technology Governance
- IT Governance
- Architecture Governance

Each of these domains of governance may exist at multiple geographic levels — global, regional, and local — within the overall enterprise.

Corporate governance is thus a broad topic, beyond the scope of an Enterprise Architecture framework such as the TOGAF framework.

This and related subsections are focused on Architecture Governance; but they describe it in the context of enterprise-wide governance, because of the hierarchy of governance structures within which it typically operates, as explained above.

In particular, this and following sections aim to:

- Provide an overview of the nature of governance as a discipline in its own right
- Describe the governance context in which Architecture Governance typically functions within the enterprise
- Describe a practical Architecture Governance Framework that can be adapted and applied, both for Enterprise Architecture and for other forms of IT architecture

3.1.2 Nature of Governance

3.1.2.1 Governance: A Generic Perspective

Governance is essentially about ensuring that business is conducted properly. It is less about overt control and strict adherence to rules, and more about guidance and effective and equitable usage of resources to ensure sustainability of an organization's strategic objectives.

The following outlines the basic principles of corporate governance, as identified by the Organization for Economic Co-operation and Development (OECD):

- Focuses on the rights, roles, and equitable treatment of shareholders
- Disclosure and transparency and the responsibilities of the board
- Ensures:
 - Sound strategic guidance of the organization
 - Effective monitoring of management by the board
 - Board accountability for the company and to the shareholders
- Responsibilities of the Board:
 - Reviewing and guiding corporate strategy
 - Setting and monitoring achievement of management's performance objectives

Supporting this, the OECD considers a traditional view of governance as: "... the system by which business corporations are directed and controlled. The corporate governance structure specifies the distribution of rights and responsibilities among different participants in the corporation — such as the board, managers, shareholders, and other stakeholders — and spells out the rules and procedures for making decisions on corporate affairs. By doing this, it also provides the structure through which the company objectives are set, and the means of attaining those objectives and monitoring performance" (Source: OECD, 2001).

3.1.2.2 Characteristics of Governance

The following characteristics have been adapted from *Corporate Governance* (Naidoo, 2002) and are positioned here to highlight both the value and necessity for governance as an approach to be adopted within organizations and their dealings with all involved parties:

Discipline	All involved parties will have a commitment to adhere to procedures, processes, and authority structures established by the organization.
Transparency	All actions implemented and their decision support will be available for inspection by authorized organization and provider parties.
Independence	All processes, decision-making, and mechanisms used will be established so as to minimize or avoid potential conflicts of interest.
Accountability	Identifiable groups within the organization — e.g., governance boards who take actions or make decisions — are authorized and accountable for their actions.
Responsibility	Each contracted party is required to act responsibly to the organization and its stakeholders.

Fairness All decisions taken, processes used, and their implementation will not be allowed to create unfair advantage to any one particular party.

3.1.3 Technology Governance

Technology governance controls how an organization utilizes technology in the research, development, and production of its goods and services. Although it may include IT governance activities, it often has a broader scope.

Technology governance is a key capability, requirement, and resource for most organizations because of the pervasiveness of technology across the organizational spectrum.

Recent studies have shown that many organizations have a balance in favor of intangibles rather than tangibles that require management. Given that most of these intangibles are informational and digital assets, it is evident that businesses are becoming more reliant on IT: and the governance of IT — IT governance — is therefore becoming an even more important part of technology governance.

These trends also highlight the dependencies of businesses on not only the information itself but also the processes, systems, and structures that create, deliver, and consume it. As the shift to increasing value through intangibles increases in many industry sectors, so risk management must be considered as key to understanding and moderating new challenges, threats, and opportunities.

Not only are organizations increasingly dependent on IT for their operations and profitability, but also their reputation, brand, and ultimately their values are also dependent on that same information and supporting technology.

3.1.4 IT Governance

IT governance provides the framework and structure that links IT resources and information to enterprise goals and strategies. Furthermore, IT governance institutionalizes best practices for planning, acquiring, implementing, and monitoring IT performance, to ensure that the enterprise's IT assets support its business objectives.

In recent years, IT governance has become integral to the effective governance of the modern enterprise. Businesses are increasingly dependent on IT to support critical business functions and processes; and to successfully gain competitive advantage, businesses need to manage effectively the complex technology that is pervasive throughout the organization in order to respond quickly and safely to business needs.

In addition, regulatory environments around the world are increasingly mandating stricter enterprise control over information, driven by increasing reports of information system disasters and electronic fraud. The management of IT-related risk is now widely accepted as a key part of enterprise governance.

It follows that an IT governance strategy, and an appropriate organization for implementing the strategy, must be established with the backing of executive management, clarifying who owns the enterprise's IT resources, and, in particular, who has ultimate responsibility for their enterprise-wide integration.

3.1.4.1 *An IT Controls Framework — COBIT*

As with corporate governance, IT governance is a broad topic, beyond the scope of an Enterprise Architecture framework such as the TOGAF framework. A good source of detailed information on IT governance is the COBIT® framework (Control Objectives for Information and related Technology). This is an open standard for control over IT, developed and promoted by the IT Governance Institute (ITGI), and published by the Information Systems Audit and Control Foundation (ISACF). COBIT controls may provide useful aides to running a compliance strategy.

3.1.5 **Architecture Governance: Overview**

3.1.5.1 *Architecture Governance Characteristics*

Architecture Governance is the practice and orientation by which Enterprise Architectures and other architectures are managed and controlled at an enterprise-wide level. It includes the following:

- Implementing a system of controls over the creation and monitoring of all architectural components and activities, to ensure the effective introduction, implementation, and evolution of architectures within the organization
- Implementing a system to ensure compliance with internal and external standards and regulatory obligations
- Establishing processes that support effective management of the above processes within agreed parameters
- Developing practices that ensure accountability to a clearly identified stakeholder community, both inside and outside the organization

3.1.5.2 *Architecture Governance as a Board-Level Responsibility*

This section aims to provide the impetus for opening up IT and Architecture Governance so that the business responsibilities associated with architecture activities and artifacts can be elucidated and managed.

3.1.5.3 *The TOGAF Standard and Architecture Governance*

Phase G of the TOGAF ADM (see the TOGAF Standard — Architecture Development Method) is dedicated to implementation governance, which concerns itself with the realization of the architecture through change projects. Implementation governance is just one aspect of Architecture Governance, which covers the management and control of all aspects of the development and evolution of Enterprise Architectures and other architectures within the enterprise.

Architecture Governance needs to be supported by an Architecture Governance Framework (described in [Section 3.2](#)) which assists in identifying effective processes so that the business responsibilities associated with Architecture Governance can be elucidated, communicated, and managed effectively.

3.2 Architecture Governance Framework

This section describes a conceptual and organizational framework for Architecture Governance

As previously explained, Phase G of the TOGAF ADM (see the TOGAF Standard — Architecture Development Method) is dedicated to implementation governance, which concerns itself with the realization of the architecture through change projects.

Implementation governance is just one aspect of Architecture Governance, which covers the management and control of all aspects of the development and evolution of Enterprise Architectures and other architectures within the enterprise.

Architecture Governance needs to be supported by an Architecture Governance Framework, described below. The governance framework described is a generic framework that can be adapted to the existing governance environment of an enterprise. It is intended to assist in identifying effective processes and organizational structures, so that the business responsibilities associated with Architecture Governance can be elucidated, communicated, and managed effectively.

3.2.1 Architecture Governance Framework — Conceptual Structure

3.2.1.1 Key Concepts

Conceptually, Architecture Governance is an approach, a series of processes, a cultural orientation, and set of owned responsibilities that ensure the integrity and effectiveness of the organization's architectures.

The key concepts are illustrated in [Figure 3-1](#).

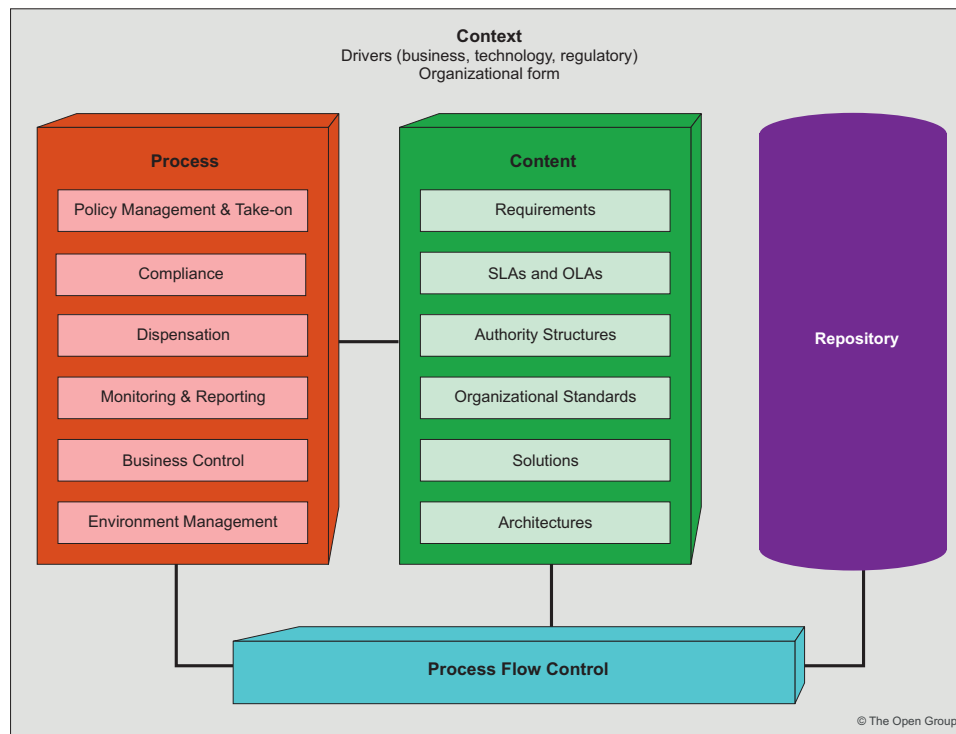


Figure 3-1 Architecture Governance Framework — Conceptual Structure

The split of process, content, and context are key to the support of the Architecture Governance initiative, by allowing the introduction of new governance material (legal, regulatory, standards-based, or legislative) without unduly impacting the processes. This content-agnostic approach ensures that the framework is flexible. The processes are typically independent of the content and implement a proven best practice approach to active governance.

The Architecture Governance Framework is integral to the Enterprise Continuum, and manages all content relevant both to the architecture itself and to Architecture Governance processes.

3.2.1.2 Key Architecture Governance Processes

Governance processes are required to identify, manage, audit, and disseminate all information related to architecture management, contracts, and implementation. These governance processes will be used to ensure that all architecture artifacts and contracts, principles, and Operational-Level Agreements (OLAs) are monitored on an ongoing basis with clear auditability of all decisions made.

Policy Management and Take-On

All architecture amendments, contracts, and supporting information must come under governance through a formal process in order to register, validate, ratify, manage, and publish new or updated content. These processes will ensure the orderly integration with existing governance content such that all relevant parties, documents, contracts, and supporting information are managed and audited.

Compliance

Compliance assessments against Service-Level Agreements (SLAs), OLAs, standards, and regulatory requirements will be implemented on an ongoing basis to ensure stability, conformance, and performance monitoring. These assessments will be reviewed and either accepted or rejected depending on the criteria defined within the governance framework.

Dispensation (also known as Waiver)

A Compliance Assessment can be rejected where the subject area (design, operational, service level, or technology) are not compliant. In this case the subject area can:

1. Be adjusted or realigned in order to meet the compliance requirements
2. Request a dispensation

Where a Compliance Assessment is rejected, an alternate route to meeting interim conformance is provided through dispensations. These are granted for a given time period and a set of identified service and operational criteria that must be enforced during the lifespan of the dispensation. Dispensations are not granted indefinitely, but are used as a mechanism to ensure that service levels and operational levels are met while providing a level of flexibility in their implementation and timing. The time-bound nature of dispensations ensures that they are a major trigger in the compliance cycle.

Monitoring and Reporting

Performance management is required to ensure that both the operational and service elements are managed against an agreed set of criteria. This will include monitoring against SLAs and OLAs, feedback for adjustment, and reporting.

Internal management information will be considered in [Environment Management](#).

Business Control

Business Control relates to the processes invoked to ensure compliance with the organization's business policies.

Environment Management

This identifies all the services required to ensure that the repository-based environment underpinning the governance framework is effective and efficient. This includes the physical and logical repository management, access, communication, training, and accreditation of all users.

All architecture artifacts, service agreements, contracts, and supporting information must come under governance through a formal process in order to register, validate, ratify, manage, and publish new or updated content. These processes will ensure the orderly integration with existing governance content such that all relevant parties, documents, contracts, and supporting information are managed and audited.

The governance environment will have a number of administrative processes defined in order to effect a managed service and process environment. These processes will include user management, internal SLAs (defined in order to control its own processes), and management information reporting.

3.2.2 Architecture Governance Framework — Organizational Structure

3.2.2.1 Overview

Architecture Governance is the practice and orientation by which Enterprise Architectures and other architectures are managed and controlled. In order to ensure that this control is effective within the organization, it is necessary to have the correct organizational structures established to support all governance activities.

An Architecture Governance structure for effectively implementing the approach described in this section will typically include the following levels, which may in practice involve a combination of existing IT governance processes, organizational structures, and capabilities. They will typically include the following:

- Global governance board
- Local governance board
- Design authorities
- Working parties

The architecture organization illustrated in [Figure 3-2](#) highlights the major structural elements required for an Architecture Governance initiative. While each enterprise will have differing requirements, it is expected that the basics of the organizational design shown in [Figure 3-2](#) will be applicable and implementable in a wide variety of organizational types.

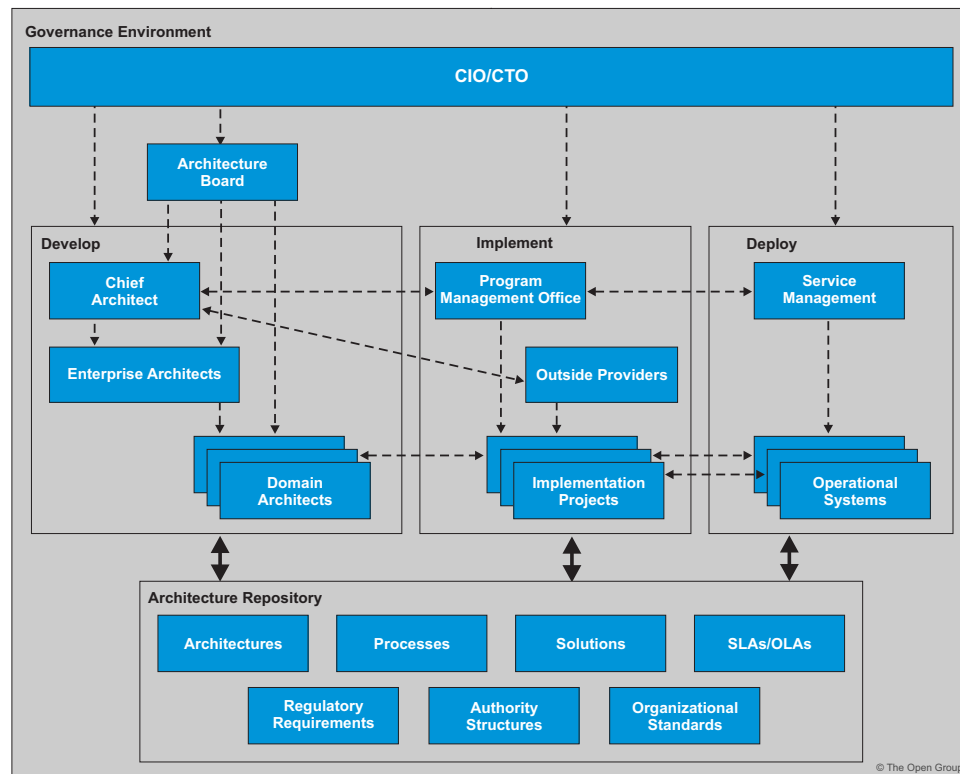


Figure 3-2 Architecture Governance Framework — Organizational Structure

3.2.2.2 Key Areas

Figure 3-2 identifies three key areas of architecture management: Develop, Implement, and Deploy. Each of these is the responsibility of one or more groups within the organization, while the Enterprise Continuum is shown to support all activities and artifacts associated with the governance of the architectures throughout their lifecycle.

The Develop responsibilities, processes, and structures are usually linked to the TOGAF ADM and its usage, while the Implement responsibilities, processes, and structures are typically linked to Phase G (see the TOGAF Standard — Architecture Development Method).

As mentioned above, the Architecture Governance Framework is integral to the Enterprise Continuum, and manages all content relevant both to the architectures themselves and to Architecture Governance processes.

3.2.2.3 Operational Benefits

As illustrated in Figure 3-2, the governance of the organization's architectures provides not only direct control and guidance of their development and implementation, but also extends into the operations of the implemented architectures.

The following benefits have been found to be derived through the continuing governance of architectures:

- Links IT processes, resources, and information to organizational strategies and objectives
- Integrates and institutionalizes IT best practices
- Aligns with industry frameworks such as COBIT (planning and organizing, acquiring and implementing, delivering and supporting, and monitoring IT performance)
- Enables the organization to take full advantage of its information, infrastructure, and hardware and software assets
- Protects the underlying digital assets of the organization
- Supports regulatory and best practice requirements such as auditability, security, responsibility, and accountability
- Promotes visible risk management

These benefits position the TOGAF Architecture Governance Framework as an approach, a series of processes, a cultural orientation, and a set of owned responsibilities, that together ensure the integrity and effectiveness of the organization's architectures.

3.3 Architecture Governance in Practice

This section provides practical guidelines for the effective implementation of Architecture Governance

3.3.1 Architecture Governance — Key Success Factors

It is important to consider the following to ensure a successful approach to Architecture Governance, and to the effective management of the Architecture Contract:

- Best practices for the submission, adoption, re-use, reporting, and retirement of architecture policies, procedures, roles, skills, organizational structures, and support services
- Organizational responsibilities and structures to support the Architecture Governance processes and reporting requirements
- Integration of tools and processes to facilitate the take-up of the processes, both procedurally and culturally
- Criteria for the control of the Architecture Governance processes, dispensations, compliance assessments, SLAs, and OLAs
- Internal and external requirements for the effectiveness, efficiency, confidentiality, integrity, availability, compliance, and reliability of all Architecture Governance-related information, services, and processes

3.3.2 Elements of an Effective Architecture Governance Strategy

3.3.2.1 Architecture Governance and Corporate Politics

An Enterprise Architecture imposed without appropriate executive support and corporate political alignment is bound to fail. In order to succeed, the Enterprise Architecture must reflect the needs of the organization. Enterprise Architects, if they are not involved in the development of business strategy, must at least have a fundamental understanding of it and of the prevailing business issues facing the organization. It may even be necessary for them to be involved in the system deployment process and to ultimately own the investment and product selection decisions arising from the implementation of the Technology Architecture.

There are three important elements of Architecture Governance strategy that relate particularly to the acceptance and success of architecture within the enterprise. While relevant and applicable in their own right apart from their role in governance, and therefore described separately, they also form an integral part of any effective Architecture Governance strategy.

- A cross-organizational Architecture Board (see [Chapter 4](#)) must be established with the backing of executive management to oversee the implementation of the Enterprise Architecture Governance strategy
- A comprehensive set of Architecture Principles (see the TOGAF Standard — ADM Techniques) should be established to guide, inform, and support the way in which an organization sets about fulfilling its mission through the use of IT
- An Architecture Compliance (see [Chapter 6](#)) strategy should be adopted — specific measures (more than just a statement of policy) to ensure compliance with the architecture, including Project Impact Assessments, a formal Architecture Compliance review process, and possibly including the involvement of the architecture team in product procurement

Chapter 4: Architecture Board

This chapter provides guidelines for establishing and operating an Enterprise Architecture Board.

4.1 Role

A key element in a successful Architecture Governance strategy (see [Chapter 3](#)) is a cross-organization Architecture Board to oversee the implementation of the strategy. This body should be representative of all the key stakeholders in the architecture, and will typically comprise a group of executives responsible for the review and maintenance of the overall architecture.

Architecture Boards may have global, regional, or business line scope. Particularly in larger enterprises, Architecture Boards typically comprise representatives from the organization at a minimum of two levels:

- Local (domain experts, line responsibility)
- Global (organization-wide responsibility)

In such cases, each board will be established with identifiable and articulated:

- Responsibilities and decision-making capabilities
- Remit and authority limits

4.2 Responsibilities

The Architecture Board is typically made responsible, and accountable, for achieving some or all of the following goals:

- Providing the basis for all decision-making with regard to the architectures
- Consistency between sub-architectures
- Establishing targets for re-use of components
- Flexibility of the Enterprise Architecture:
 - To meet changing business needs
 - To leverage new technologies
- Enforcement of Architecture Compliance
- Improving the maturity level of architecture discipline within the organization
- Ensuring that the discipline of architecture-based development is adopted

- Supporting a visible escalation capability for out-of-bounds decisions

Further responsibilities from an operational perspective should include:

- All aspects of monitoring and control of the Architecture Contract
- Meeting on a regular basis
- Ensuring the effective and consistent management and implementation of the architectures
- Resolving ambiguities, issues, or conflicts that have been escalated
- Providing advice, guidance, and information
- Ensuring compliance with the architectures, and granting dispensations that are in keeping with the technology strategy and objectives
- Considering policy (schedule, SLAs, etc.) changes where similar dispensations are requested and granted; e.g., new form of service requirement
- Ensuring that all information relevant to the implementation of the Architecture Contract is published under controlled conditions and made available to authorized parties
- Validation of reported service levels, cost savings, etc.

From a governance perspective, the Architecture Board is also responsible for:

- The production of usable governance material and activities
- Providing a mechanism for the formal acceptance and approval of architecture through consensus and authorized publication
- Providing a fundamental control mechanism for ensuring the effective implementation of the architecture
- Establishing and maintaining the link between the implementation of the architecture, the architectural strategy and objectives embodied in the Enterprise Architecture, and the strategic objectives of the business
- Identifying divergence from the architecture and planning activities for realignment through dispensations or policy updates

4.3 Setting Up the Architecture Board

4.3.1 Triggers

One or more of the following occurrences typically triggers the establishment of an Architecture Board:

- New CIO
- Merger or acquisition
- Organizational restructuring
- Consideration of a move to newer forms of computing
- Recognition that IT is poorly aligned to business

- Desire to achieve competitive advantage via technology
- Creation of an Enterprise Architecture program
- Significant business change or rapid growth
- Requirement for complex, cross-functional solutions

In many companies, the executive sponsor of the initial architecture effort is the CIO (or other senior executive). However, to gain broad corporate support, a sponsoring body has more influence. This sponsoring body is here called an Architecture Board, but the title is not important. Whatever the name, it is the executive-level group responsible for the review and maintenance of the strategic architecture and all of its sub-architectures.

The Architecture Board is the sponsor of the architecture within the enterprise, but the Architecture Board itself needs an executive sponsor from the highest level of the corporation. This commitment must span the planning process and continue into the maintenance phase of the architecture project. In many companies that fail in an architecture planning effort, there is a notable lack of executive participation and encouragement for the project.

A frequently overlooked source of Architecture Board members is the company's Board of Directors. These individuals invariably have diverse knowledge about the business and its competition. Because they have a significant impact on the business vision and objectives, they may be successful in validating the alignment of IT strategies to business objectives.

4.3.2 Size of the Board

The recommended size for an Architecture Board is four or five (and no more than ten) permanent members. In order to keep the Architecture Board to a reasonable size, while ensuring enterprise-wide representation on it over time, membership of the Architecture Board may be rotated, giving decision-making privileges and responsibilities to various senior managers. This may be required in any case, due to some Architecture Board members finding that time constraints prevent long-term active participation.

An organization may set up its Architecture Board using representational means, such that each Board member is assigned a set of stakeholders to represent in the meetings. It is then incumbent upon Board members to meet with stakeholders to gain their views on the various agenda items.

However, some continuity must exist on the Architecture Board, to prevent the corporate architecture from varying from one set of ideas to another. One technique for ensuring rotation with continuity is to have set terms for the members, and to have the terms expire at different times.

In the ongoing architecture process following the initial architecture effort, the Architecture Board may be re-chartered. The executive sponsor will normally review the work of the Architecture Board and evaluate its effectiveness; if necessary, the Architecture Compliance review process is updated or changed.

4.3.3 Board Structure

The TOGAF Architecture Governance Framework (see [Section 3.2](#)) provides a generic organizational framework that positions the Architecture Board in the context of the broader governance structures of the enterprise. This structure identifies the major organizational groups and responsibilities, as well as the relationship between each group. This is a best practice structure, and may be subject to change depending on the organization's form and existing structures.

Consideration must be given to the size of the organization, its form, and how the IT functions are implemented. This will provide the basis for designing the Architecture Board structure within the context of the overall governance environment. In particular, consideration should be given to the concept of global ownership and local implementation, and the integration of new concepts and technologies from all areas implementing against architectures.

The structure of the Architecture Board should reflect the form of the organization. The Architecture Governance structure required may well go beyond the generic structures outlined in the TOGAF Architecture Governance Framework (see [Section 3.2](#)). The organization may need to define a combination of the IT governance process in place and the existing organizational structures and capabilities, which typically include the following types of body:

- Global governance board
- Local governance board
- Design authorities
- Working parties

4.4 Operation of the Architecture Board

This section describes the operation of the Architecture Board particularly from the governance perspective.

4.4.1 General

Architecture Board meetings should be conducted within clearly identified agendas with explicit objectives, content coverage, and defined actions. In general, board meetings will be aligned with best practice, such as given in the COBIT framework (see [Section 3.1.4.1](#)).

These meetings will provide key direction in:

- Supporting the production of quality governance material and activities
- Providing a mechanism for formal acceptance through consensus and authorized publication
- Providing a fundamental control mechanism for ensuring the effective implementation of the architectures
- Establishing and maintaining the link between the implementation of the architectures and the stated strategy and objectives of the organization (business and IT)

- Identifying divergence from the contract and planning activities to realign with the contract through dispensations or policy updates

4.4.2 Preparation

Each participant will receive an agenda and any supporting documentation — e.g., dispensation requests, performance management reports, etc. — and will be expected to be familiar with the contents of each.

Where actions have been allocated to an individual, it is that person's responsibility to report on progress against these.

Each participant must confirm their availability and attendance at the Architecture Board meeting.

4.4.3 Agenda

This section outlines the contents of an Architecture Board meeting agenda. Each agenda item is described in terms of its content only.

Minutes of Previous Meeting

Minutes contain the details of previous Architecture Board meetings as per standard organizational protocol.

Requests for Change

Items under this heading are normally change requests for amendments to architectures, principles, etc., but may also include business control with regard to Architecture Contracts; e.g., ensure that voice traffic to premium numbers, such as weather reports, is barred and data traffic to certain websites is controlled.

Any request for change is made within agreed authority levels and parameters defined by the Architecture Contract.

Dispensations

A dispensation is used as the mechanism to request a change to the existing architectures, contracts, principles, etc. outside of normal operating parameters; e.g., exclude provision of service to a subsidiary, request for unusual service levels for specific business reasons, deploy non-standard technology or products to support specific business initiatives.

Dispensations are granted for a given time period and a set of identified services and operational criteria that must be enforced during the lifespan of the dispensation. Dispensations are not granted indefinitely, but are used as a mechanism to ensure that service levels and operational levels, etc. are met while providing a level flexibility in their implementation and timing. The time-bound nature of dispensations ensures that they are a trigger to the Architecture Compliance activity.

Compliance Assessments

Compliance is assessed against SLAs, Operational-Level Agreements (OLAs), cost targets, and required architecture refreshes. These assessments will be reviewed and either accepted or rejected depending on the criteria defined within the Architecture Governance Framework. The Architecture Compliance assessment report will include details as described.

Dispute Resolution

Disputes that have not been resolved through the Architecture Compliance and dispensation processes are identified here for further action and are documented through the Architecture Compliance assessments and dispensation documentation.

Architecture Strategy and Direction Documentation

This describes the architecture strategies, direction, and priorities and will only be formulated by the global Architecture Board. It should take the form of standard architecture documentation.

Actions Assigned

This is a report on the actions assigned at previous Architecture Board meetings. An action tracker is used to document and keep the status of all actions assigned during the Architecture Board meetings and should consist of at least the following information:

- Reference
- Priority
- Action description
- Action owner
- Action details
- Date raised
- Due date
- Status
- Type
- Resolution date

Contract Documentation Management

This is a formal acceptance of updates and changes to architecture documentation for onward publication.

Any Other Business (AOB)

Description of issues not directly covered under any of the above. These may not be described in the agenda but should be raised at the beginning of the meeting. Any supporting documentation must be managed as per all Architecture Governance documentation.

Schedule of Meetings

All meeting dates should be detailed and published.

Chapter 5: Architecture Contracts

This chapter provides guidelines for defining and using Architecture Contracts.

5.1 Role

Architecture Contracts are the joint agreements between development partners and sponsors on the deliverables, quality, and fitness-for-purpose of an architecture. Successful implementation of these agreements will be delivered through effective Architecture Governance (see [Chapter 3](#)). By implementing a governed approach to the management of contracts, the following will be ensured:

- A system of continuous monitoring to check integrity, changes, decision-making, and audit of all architecture-related activities within the organization
- Adherence to the principles, standards, and requirements of the existing or developing architectures
- Identification of risks in all aspects of the development and implementation of the architecture(s) covering the internal development against accepted standards, policies, technologies, and products as well as the operational aspects of the architectures such that the organization can continue its business within a resilient environment
- A set of processes and practices that ensure accountability, responsibility, and discipline with regard to the development and usage of all architectural artifacts
- A formal understanding of the governance organization responsible for the contract, their level of authority, and scope of the architecture under the governance of this body

The traditional Architecture Contract is an agreement between the sponsor and the architecture function or Information Systems (IS) department. However, increasingly more services are being provided by systems integrators, applications providers, and service providers, co-ordinated through the architecture function or IS department. There is therefore a need for an Architecture Contract to establish joint agreements between all parties involved in the architecture development and delivery.

Architecture Contracts may occur at various stages of the ADM; for example:

- The Statement of Architecture Work created in Phase A of the TOGAF Standard — Architecture Development Method is effectively an Architecture Contract between the architecting organization and the sponsor of the Enterprise Architecture (or the IT governance function)
- The development of one or more architecture domains (Business, Data, Application, Technology), and in some cases the oversight of the overall Enterprise Architecture, may be contracted out to systems integrators, applications providers, and/or service providers

Each of these arrangements will normally be governed by an Architecture Contract that

defines the deliverables, quality, and fitness-for-purpose of the developed architecture, and the processes by which the partners in the architecture development will work together.

- At the beginning of Phase G (Implementation Governance), between the architecture function and the function responsible for implementing the Enterprise Architecture defined in the preceding ADM phases; typically, this will be either the in-house systems development function, or a major contractor to whom the work is outsourced
 - What is being "implemented" in Phase G of the ADM is the overall Enterprise Architecture

This will typically include the technology infrastructure (from Phase D), and also those enterprise applications and data management capabilities that have been defined in the Application Architecture and Data Architecture (from Phase C), either because they are enterprise-wide in scope, or because they are strategic in business terms, and therefore of enterprise-wide importance and visibility. However, it will typically not include non-strategic business applications, which business units will subsequently deploy on top of the technology infrastructure that is implemented as part of the Enterprise Architecture.
 - In larger-scale implementations, there may well be one Architecture Contract per implementation team in a program of implementation projects
- When the finalized Architecture Definition Document is available (at the end of Phase F), an Architecture Contract will normally be drawn up between the architecting function (or the IT governance function, subsuming the architecting function) and the business users who will subsequently be building and deploying application systems in the architected environment

It is important to bear in mind in all these cases that the ultimate goal is not just an Enterprise Architecture, but a dynamic Enterprise Architecture; i.e., one that allows for flexible evolution in response to changing technology and business drivers, without unnecessary constraints. The Architecture Contract is crucial to enabling a dynamic Enterprise Architecture and is key to governing the implementation.

Typical contents of these three kinds of Architecture Contract are explained below.

5.2 Contents

5.2.1 Statement of Architecture Work

The Statement of Architecture Work is created as a deliverable of Phase A, and is effectively an Architecture Contract between the architecting organization and the sponsor of the Enterprise Architecture (or the IT governance function, on behalf of the enterprise).

The typical contents of a Statement of Architecture Work are as defined in the TOGAF Standard — Architecture Content.

5.2.2 Contract between Architecture Design and Development Partners

This is a signed statement of intent on designing and developing the Enterprise Architecture, or significant parts of it, from partner organizations, including systems integrators, applications providers, and service providers.

Increasingly, the development of one or more architecture domains (Business, Data, Application, Technology) may be contracted out, with the enterprise's architecture function providing oversight of the overall Enterprise Architecture, and co-ordination and control of the overall effort. In some cases even this oversight role may be contracted out, although most enterprises prefer to retain that core responsibility in-house.

Whatever the specifics of the contracting-out arrangements, the arrangements themselves will normally be governed by an Architecture Contract that defines the deliverables, quality, and fitness-for-purpose of the developed architecture, and the processes by which the partners in the architecture development will work together.

Typical contents of an Architecture Design and Development Contract are:

- Introduction and background
- The nature of the agreement
- Scope of the architecture
- Architecture and strategic principles and requirements
- Conformance requirements
- Architecture development and management process and roles
- Target architecture measures
- Defined phases of deliverables
- Prioritized joint workplan
- Time window(s)
- Architecture delivery and business metrics

The template for this contract will normally be defined as part of the Preliminary Phase of the ADM, if not existing already, and the specific contract will be defined at the appropriate stage of the ADM, depending on the particular work that is being contracted out.

5.2.3 Contract between Architecting Function and Business Stakeholders

When the Implementation and Migration Plan has been agreed (at the end of Phase F), an Architecture Contract may be drawn up between the architecting function and the business stakeholders who will subsequently be building and deploying the architected business solutions.

A business stakeholder's Architecture Contract may include:

- Introduction and background
- The nature of the agreement
- Scope
- Strategic requirements
- Architecture deliverables that meet the business requirements
- Conformance requirements
- Architecture adopters
- Time window
- Architecture business metrics
- SLA

This contract is also used to manage changes to the Enterprise Architecture in Phase H.

5.3 Relationship to Architecture Governance

The Architecture Contract document produced in Phase G of the ADM figures prominently in the area of Architecture Governance, as explained in [Chapter 3](#).

In the context of Architecture Governance, the Architecture Contract is often used as a means of driving architecture change.

In order to ensure that the Architecture Contract is effective and efficient, the following aspects of the governance framework may need to be introduced into Phase G:

- Simple processes
- People-centered authority
- Strong communication
- Timely responses and an effective escalation process
- Supporting organizational structures
- Status tracking of architecture implementation

Chapter 6: Architecture Compliance

This chapter provides guidelines for ensuring project compliance to the architecture.

6.1 Introduction

Ensuring the compliance of individual projects with the Enterprise Architecture is an essential aspect of Architecture Governance (see [Chapter 3](#)). To this end, the IT governance function within an enterprise will normally define two complementary processes:

- The **Architecture** function will be required to prepare a series of Project Architectures; i.e., project-specific views of the Enterprise Architecture that illustrate how the Enterprise Architecture impacts on the major projects within the organization (see ADM Phases A to F)
- The **Enterprise and IT Governance** functions will define a formal Architecture Compliance review process (see [Section 6.3](#)) for reviewing the compliance of all projects to the Enterprise Architecture

Apart from defining formal processes, the Architecture Governance function (see [Chapter 3](#)) may also stipulate that the architecture function should extend beyond the role of architecture definition and standards selection, and participate also in the technology selection process, and even in the commercial relationships involved in external service provision and product purchases. This may help to minimize the opportunity for misinterpretation of the Enterprise Architecture, and maximize the value of centralized commercial negotiation.

6.2 Terminology: The Meaning of Architecture Compliance

A key relationship between the architecture and the implementation lies in the definitions of the terms "conformant", "compliant", etc. While terminology usage may differ between organizations, the concepts of levels of conformance illustrated in [Figure 6-1](#) should prove useful in formulating an IT compliance strategy.

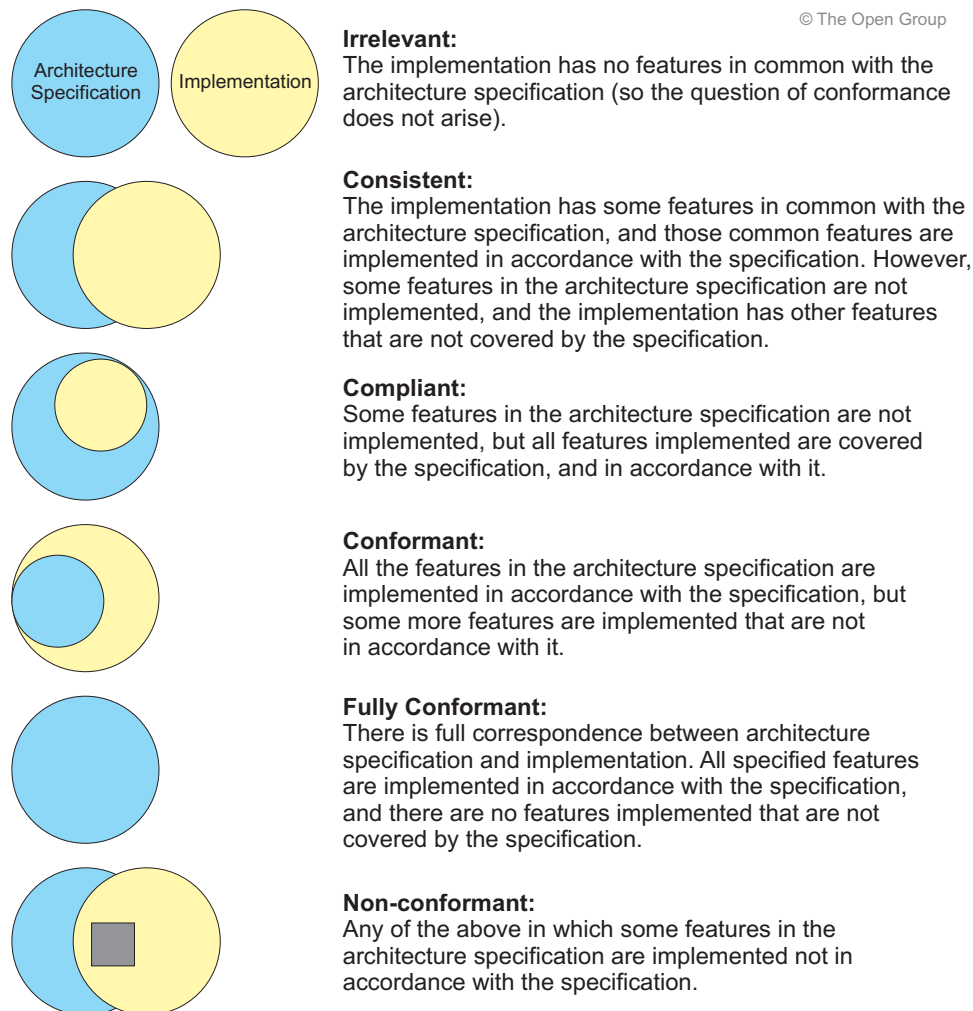


Figure 6-1 Levels of Architecture Conformance

The phrase "in accordance with" in [Figure 6-1](#) means:

- Supports the stated strategy and future directions
- Adheres to the stated standards (including syntax and semantic rules specified)
- Provides the stated functionality
- Adheres to the stated principles; for example:
 - Open wherever possible and appropriate
 - Re-use of component building blocks wherever possible and appropriate

6.3 Architecture Compliance Reviews

An Architecture Compliance review is a scrutiny of the compliance of a specific project against established architectural criteria, spirit, and business objectives. A formal process for such reviews normally forms the core of an Enterprise Architecture Compliance strategy.

6.3.1 Purpose

The goals of an Architecture Compliance review include some or all of the following:

- First and foremost, catch errors in the project architecture early, and thereby reduce the cost and risk of changes required later in the lifecycle

This in turn means that the overall project time is shortened, and that the business gets the bottom-line benefit of the architecture development faster.
- Ensure the application of best practices to architecture work
- Provide an overview of the compliance of an architecture to mandated enterprise standards
- Identify where the standards themselves may require modification
- Identify services that are currently application-specific but might be provided as part of the enterprise infrastructure
- Document strategies for collaboration, resource sharing, and other synergies across multiple architecture teams
- Take advantage of advances in technology
- Communicate to management the status of business and technical readiness of the project
- Identify key criteria for procurement activities; e.g., for inclusion in Commercial Off-The-Shelf (COTS) product Request for Information (RFI)/Request for Proposal (RFP) documents
- Identify and communicate significant architectural gaps to product and service providers

Apart from the generic goals related to quality assurance outlined above, there are additional, more politically-oriented motivations for conducting Architecture Compliance reviews, which may be relevant in particular cases:

- The Architecture Compliance review can be a good way of deciding between architectural alternatives, since the business decision-makers typically involved in the review can guide decisions in terms of what is best for the business, as opposed to what is technically more pleasing or elegant
- The output of the Architecture Compliance review is one of the few measurable deliverables to the executive management to assist in decision-making
- Architecture reviews can serve as a way for the architecture organization to engage with development projects that might otherwise proceed without involvement of the architecture function
- Architecture reviews can demonstrate rapid and positive support to the enterprise business community:

- The Enterprise Architecture and Architecture Compliance helps ensure the alignment of IT projects with business objectives
- Architects can sometimes be regarded as being deep into technical infrastructure and far removed from the core business
- Since an Architecture Compliance review tends to look primarily at the critical risk areas of a system, it often highlights the main risks for system owners

While compliance to architecture is required for development and implementation, non-compliance also provides a mechanism for highlighting:

- Areas to be addressed for realignment
- Areas for consideration for integration into the architectures as they are uncovered by the compliance processes

The latter point identifies the ongoing change and adaptability of the architectures to requirements that may be driven by indiscipline, but also allows for changes to be registered by faster moving changes in the operational environment. Typically, dispensations (see [Section 3.1.4](#)) will be used to highlight these changes and set in motion a process for registering, monitoring, and assessing the suitability of any changes required.

6.3.2 Timing

Timing of compliance activities should be considered with regard to the development of the architectures themselves.

Compliance reviews are held at appropriate project milestones or checkpoints in the project's lifecycle. Specific checkpoints should be included as follows:

- Development of the architecture itself (ADM compliance)
- Implementation of the architecture(s) (architecture compliance)

Architecture project timings for assessments should include:

- Project initiation
- Initial design
- Major design changes
- *Ad hoc*

The Architecture Compliance review is typically targeted for a point in time when business requirements and the Enterprise Architecture are reasonably firm, and the project architecture is taking shape, well before its completion.

The aim is to hold the review as soon as practical, at a stage when there is still time to correct any major errors or shortcomings, with the obvious proviso that there needs to have been some significant development of the project architecture in order for there to be something to review.

Inputs to the Architecture Compliance review may come from other parts of the standard project lifecycle, which may have an impact on timing.

6.3.3 Governance and Personnel Scenarios

In terms of the governance and conduct of the Architecture Compliance review, and the personnel involved, there are various possible scenarios:

- For smaller-scale projects, the review process could simply take the form of a series of questions that the project architects or project leaders pose to themselves, using the checklists provided below, perhaps collating the answers into some form of project report to management

The need to conduct such a process is normally included in overall enterprise-wide governance policies.

- Where the project under review has not involved a practicing or full-time architect to date (for example, in an application-level project), the purpose of the review is typically to bring to bear the architectural expertise of an Enterprise Architecture function

In such a case, the Enterprise Architecture function would be organizing, leading, and conducting the review, with the involvement of business domain experts. In such a scenario, the review is not a substitute for the involvement of architects in a project, but it can be a supplement or a guide to their involvement. It is probable that a database will be necessary to manage the volume of data that would be produced in the analysis of a large system or set of systems.

- In most cases, particularly in larger-scale projects, the architecture function will have been deeply involved in, and perhaps leading, the development project under review

(This is the typical TOGAF scenario.) In such cases, the review will be co-ordinated by the lead Enterprise Architect, who will assemble a team of business and technical domain experts for the review, and compile the answers to the questions posed during the review into some form of report. The questions will typically be posed during the review by the business and technical domain experts. Alternatively, the review might be led by a representative of an Architecture Board or some similar body with enterprise-wide responsibilities.

In all cases, the Architecture Compliance review process needs the backing of senior management, and will typically be mandated as part of corporate Architecture Governance policies (see [Chapter 3](#)). Normally, the enterprise CIO or Enterprise Architecture Board (see [Chapter 4](#)) will mandate architecture reviews for all major projects, with subsequent annual reviews.

6.4 Architecture Compliance Review Process

6.4.1 Overview

The Architecture Compliance review process is illustrated in Figure 6-2.

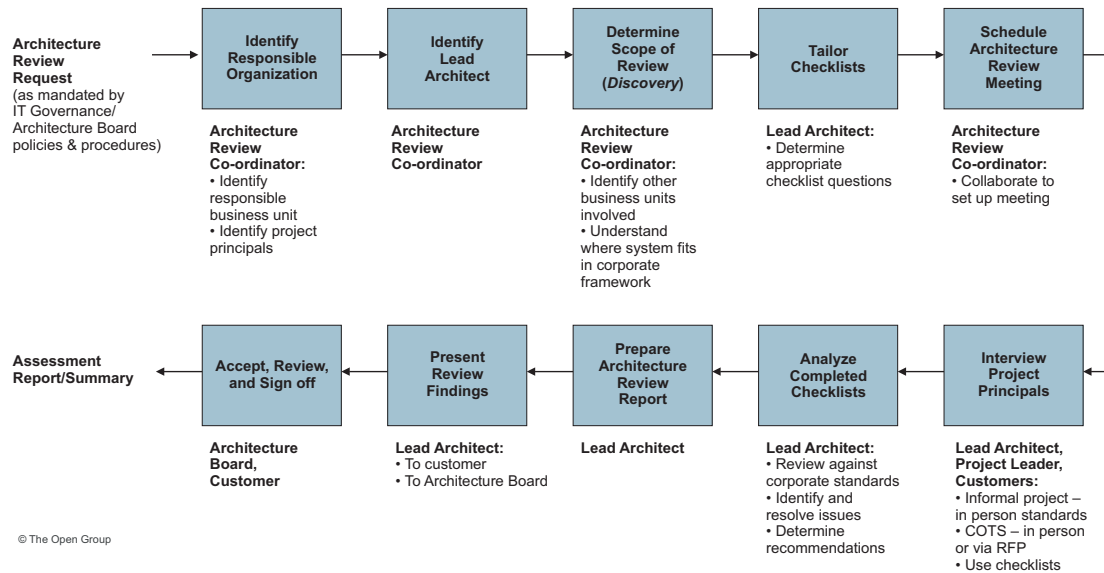


Figure 6-2 Architecture Compliance Review Process

Note: Another way to gain conformance checks in architecture is through the use of traceability in models and "drill-down" diagrams. In this technique, the Enterprise Architect creates a top-level view of an architecture which the Solution Architects further refine by tracing their elements to the Enterprise Architecture model. These refinements can be managed through so-called "drill-down" diagrams which contain all the refinement elements and their traceability. In this way, conformance can be reviewed and at least partially automated by tools (such as having a script that checks that all Enterprise Architecture elements have refinements in the Solution Architecture).

6.4.2 Roles

The main roles in the process are tabulated below.

No.	Role	Responsibilities	Notes
1	Architecture Board	To ensure that Enterprise Architectures are consistent and support overall business needs.	Sponsor and monitor architecture activities.
2	Project Leader (or Project Board)	Responsible for the whole project.	
3	Architecture Review Co-ordinator	To administer the whole architecture development and review process.	More likely to be business-oriented than technology-oriented.
4	Lead Enterprise Architect	To ensure that the architecture is business and technically coherent and future-proof.	An appropriate architecture specialist.
5	Architect	One of the Lead Enterprise Architect's assistants.	
6	Customer	To ensure that business requirements are clearly expressed and understood.	Manages that part of the organization that will depend on the success of the Enterprise Architecture.
7	Business Domain Expert	To ensure that the processes to satisfy the business requirements are justified and understood.	Knows how the business domain operates; may also be the customer.
8	Project Principals	To ensure that the architects have a sufficiently detailed understanding of the customer department's processes. They can provide input to the business domain expert or to the architects.	Members of the customer's organization who have input to the business requirements that the architecture is to address.

6.4.3 Steps

The main steps in the process are tabulated below.

No.	Action	Notes	Who
1	Request architecture review.	As mandated by governance policies and procedures.	Anyone, whether IT or business-oriented, with an interest in or responsibility for the business area affected
2	Identify responsible part of organization and relevant project principals.		Architecture Review Co-ordinator
3	Identify Lead Enterprise Architect and other architects.		Architecture Review Co-ordinator
4	Determine scope of review.	Identify which other business units/departments are involved. Understand where the system fits in the corporate architecture framework.	Architecture Review Co-ordinator
5	Tailor checklists.	To address the business requirements.	Lead Enterprise Architect
6	Schedule Architecture Review Meeting.		Architecture Review Co-ordinator with collaboration of Lead Enterprise Architect
7	Interview project principals.	To get background and technical information: <ul style="list-style-type: none"> ■ For internal project: in person ■ For COTS: in person or via RFP Use checklists.	Lead Enterprise Architect and/or Architect, Project Leader, and Customers
8	Analyze completed checklists.	Review against corporate standards. Identify and resolve issues. Determine recommendations.	Lead Enterprise Architect
9	Prepare Architecture Compliance review report.	May involve supporting staff.	Lead Enterprise Architect
10	Present review findings.	<ul style="list-style-type: none"> ■ To Customer ■ To Architecture Board 	Lead Enterprise Architect
11	Accept review and sign off.		Architecture Board and Customer

No.	Action	Notes	Who
12	Send assessment report/summary to Architecture Review Co-ordinator.		Lead Enterprise Architect

6.5 Architecture Compliance Review Checklists

The following review checklists provide a wide range of typical questions that may be used in conducting Architecture Compliance reviews, relating to various aspects of the architecture. The organization of the questions includes the basic disciplines of system engineering, information management, security, and systems management. The checklists are based on material provided by a member of The Open Group, and are specific to that organization. Other organizations could use the following checklists with other questions tailored to their own particular needs.

The checklists provided contain too many questions for any single review: they are intended to be tailored selectively to the project concerned (see [Section 6.6](#)). The checklists actually used will typically be developed/selected by subject matter experts. They are intended to be updated annually by interest groups in those areas.

Some of the checklists include a brief description of the Architecture Principle that provokes the question, and a brief description of what to look for in the answer. These extensions to the checklist are intended to allow the intelligent re-phrasing of the questions, and to give the user of the checklist a feel for why the question is being asked.

Occasionally the questions will be written, as in RFPs, or in working with a senior project architect. More typically they are expressed orally, as part of an interview or working session with the project.

The checklists provided here are designed for use in individual architecture projects, not for business domain architecture or for architecture across multiple projects. (Doing an architecture review for a larger sphere of activity, across multiple business processes and system projects, would involve a similar process, but the checklist categories and their contents would be different.)

6.5.1 Hardware and Operating System Checklist

1. What is the project's lifecycle approach?
2. At what stage is the project in its lifecycle?
3. What key issues have been identified or analyzed that the project believes will drive evaluations of hardware and operating systems for networks, servers, and end-user devices?
4. What system capabilities will involve high-volume and/or high-frequency data transfers?
5. How does the system design impact or involve end-user devices?
6. What is the quantity and distribution (regional and global) of usage, data storage, and processing?

7. What applications are affinitized with your project by similarities in data, application services, etc.? To what degree is data affinitized with your project?
8. What hardware and operating system choices have been made before the functional design of key elements of the system?
9. If hardware and operating system decisions were made outside of the project's control:
 - What awareness does the project have of the rationale for those decisions?
 - How can the project influence those decisions as system design takes shape?
10. If some non-standards have been chosen:
 - What are the essential business and technical requirements for not using corporate standards?
 - Is this supported by a business case?
 - Have the assumptions in the business case been subject to scrutiny?
11. What is your process for evaluating full lifecycle costs of hardware and operating systems?
12. How has corporate financial management been engaged in evaluation of lifecycle costs?
13. Have you performed a financial analysis of the supplier?
14. Have you made commitments to any supplier?
15. Do you believe your requirements can be met by only one supplier?

6.5.2 Software Services and Middleware Checklist

1. Describe how error conditions are defined, raised, and propagated between application components.
2. Describe the general pattern of how methods are defined and arranged in various application modules.
3. Describe the general pattern for how method parameters are defined and organized in various application modules. Are [in], [in/out], [out] parameters always specified in the same order? Do Boolean values returned by modules have a consistent outcome?
4. Describe the approach that is used to minimize the number of round-trips between client and server calls, particularly for out-of-process calls, and when complex data structures are involved.
5. Describe the major data structures that are passed between major system components.
6. Describe the major communication protocols that are used between major system components.
7. Describe the marshaling techniques that are used between various system components. Describe any specialized marshaling arrangements that are used.
8. Describe to what extent the system is designed with stateful and stateless components.

9. Describe how and when state is saved for both stateful and stateless components.
10. Describe the extent to which objects are created, used, and destroyed *versus* re-used through object pooling.
11. Describe the extent to which the system relies on threading or critical section coding.
12. Describe the approach and the internal documentation that is used internally in the system to document the methods, methods arguments, and method functionality.
13. Describe the code review process that was used to build the system.
14. Describe the unit testing that has been used to test the system components.
15. Describe the pre- and post-condition testing that is included in various system modules.
16. Describe the assertion testing that is included with the system.
17. Do components support all the interface types they need to support or are certain assumptions made about what types of components will call other components either in terms of language bindings or other forms of marshaling?
18. Describe the extent to which big-endian or little-endian data format problems need to be handled across different platforms.
19. Describe if numbers or strings need to be handled differently across different platforms.
20. Describe whether the software needs to check for floating-point round-off errors.
21. Describe how time and date functions manage dates so as to avoid improper handling of time and date calculation or display.
22. Describe what tools or processes have been used to test the system for memory leaks, reachability, or general robustness.
23. Describe the layering of the systems services software. Describe the general number of links between major system components. Is the system composed of a lot of point-to-point interfaces or are major messaging backbones used instead?
24. Describe to what extent the system components are either loosely coupled or tightly coupled.
25. What requirements does the system need from the infrastructure in terms of shared libraries, support for communication protocols, load balancing, transaction processing, system monitoring, naming services, or other infrastructure services?
26. Describe how the system and system components are designed for refactoring.
27. Describe how the system or system components rely on common messaging infrastructure *versus* a unique point-to-point communication structure.

6.5.3 Applications Checklists

6.5.3.1 Infrastructure (Enterprise Productivity) Applications

1. Is there need for capabilities that are not provided through the enterprise's standard infrastructure application products? For example:
 - Collaboration
 - Application sharing
 - Video conferencing
 - Calendaring
 - Email
 - Workflow management
 - Publishing/word processing applications
 - HTML
 - SGML and XML
 - Portable document format
 - Document processing (proprietary format)
 - Desktop publishing
 - Spreadsheet applications
 - Presentation applications
 - Business presentations
 - Image
 - Animation
 - Video
 - Sound
 - CBT
 - Web browsers
 - Data management applications
 - Database interface
 - Document management
 - Product data management
 - Data warehouses/mart
 - Program management applications
 - Project management
 - Program visibility

2. Describe the business requirements for enterprise infrastructure application capabilities that are not met by the standard products.

6.5.3.2 Business Applications

1. Are any of the capabilities required provided by standard products supporting one or more line-of-business applications? For example:
 - Business acquisition applications
 - Sales and marketing
 - Engineering applications
 - Computer-aided design
 - Computer-aided engineering
 - Mathematical and statistics analysis
 - Supplier management applications
 - Supply chain management
 - Customer relationship management
 - Manufacturing applications
 - Enterprise Resource Planning (ERP) applications
 - Manufacturing execution systems
 - Manufacturing quality
 - Manufacturing process engineering
 - Machine and adaptive control
 - Customer support applications
 - Airline logistics support
 - Maintenance engineering
 - Finance applications
 - People applications
 - Facilities applications
 - Information systems applications
 - Systems engineering
 - Software engineering
 - Web developer tools
 - Integrated development environments
 - Lifecycle categories
 - Functional categories

- Specialty categories
 - Computer-aided manufacturing
 - e-Business enablement
 - Business process engineering
 - Statistical quality control
2. Describe the process requirements for business application capabilities that are not met by the standard products.

6.5.3.3 Application Integration Approach

1. What integration points (business process/activity, application, data, computing environment) are targeted by this architecture?
2. What application integration techniques will be applied (common business objects [Object Request Brokers (ORBs)], standard data definitions [XML, etc.], common user interface presentation/desktop)?

6.5.4 Information Management Checklists

6.5.4.1 Data Values

1. What are the processes that standardize the management and use of the data?
2. What business process supports the entry and validation of the data? Use of the data?
3. What business actions correspond to the creation and modification of the data?
4. What business actions correspond to the deletion of the data and is it considered part of a business record?
5. What are the data quality requirements required by the business user?
6. What processes are in place to support data referential integrity and/or normalization?

6.5.4.2 Data Definition

1. What are the data model, data definitions, structure, and hosting options of purchased applications (COTS)?
2. What are the rules for defining and maintaining the data requirements and designs for all components of the information system?
3. What shareable repository is used to capture the model content and the supporting information for data?
4. What is the physical data model definition (derived from logical data models) used to design the database?
5. What software development and data management tools have been selected?
6. What data owners have been identified to be responsible for common data definitions, eliminating unplanned redundancy, providing consistently reliable, timely, and accurate information, and protecting data from misuse and destruction?

6.5.4.3 Security/Protection

1. What are the data entity and attribute access rules which protect the data from unintentional and unauthorized alterations, disclosure, and distribution?
2. What are the data protection mechanisms to protect data from unauthorized external access?
3. What are the data protection mechanisms to control access to data from external sources that temporarily have internal residence within the enterprise?

6.5.4.4 Hosting, Data Types, and Sharing

1. What is the discipline for managing sole-authority data as one logical source with defined updating rules for physical data residing on different platforms?
2. What is the discipline for managing replicated data, which is derived from operational sole-authority data?
3. What tier data server has been identified for the storage of high or medium-critical operational data?
4. What tier data server has been identified for the storage of type C operational data?
5. What tier data server has been identified for the storage of decision support data contained in a data warehouse?
6. What Database Management Systems (DBMSs) have been implemented?

6.5.4.5 Common Services

1. What are the standardized distributed data management services (e.g., validation, consistency checks, data edits, encryption, and transaction management) and where do they reside?

6.5.4.6 Access Method

1. What are the data access requirements for standard file, message, and data management?
2. What are the access requirements for decision support data?
3. What are the data storage and the application logic locations?
4. What query language is being used?

6.5.5 Security Checklist

1. **Security Awareness:** Have you ensured that the corporate security policies and guidelines to which you are designing are the latest versions? Have you read them? Are you aware of all relevant computing security compliance and risk acceptance processes? (Interviewer should list all relevant policies and guidelines.)

2. **Identification/Authentication:** Diagram the process flow of how a user is identified to the application and how the application authenticates that the user is who they claim to be. Provide supporting documentation to the diagram explaining the flow from the user interface to the application/database server(s) and back to the user. Are you compliant with corporate policies on accounts, passwords, etc.?
3. **Authorization:** Provide a process flow from beginning to end showing how a user requests access to the application, indicating the associated security controls and separation of duties. This should include how the request is approved by the appropriate data owner, how the user is placed into the appropriate access-level classification profile, how the user ID, password, and access is created and provided to the user. Also include how the user is informed of their responsibilities associated with using the application, given a copy of the access agreement, how to change password, who to call for help, etc.
4. **Access Controls:** Document how the user IDs, passwords, and access profiles are added, changed, removed, and documented. The documentation should include who is responsible for these processes.
5. **Sensitive Information Protection:** Provide documentation that identifies sensitive data requiring additional protection. Identify the data owners responsible for this data and the process to be used to protect storage, transmission, printing, and distribution of this data. Include how the password file/field is protected. How will users be prevented from viewing someone else's sensitive information? Are there agreements with outside parties (partners, suppliers, contractors, etc.) concerning the safeguarding of information? If so, what are the obligations?
6. **Audit Trails and Audit Logs:** Identify and document group accounts required by the users or application support, including operating system group accounts. Identify and document individual accounts and/or roles that have superuser type privileges, what these privileges are, who has access to these accounts, how access to these accounts is controlled, tracked, and logged, and how password change and distribution are handled, including operating system accounts. Also identify audit logs, who can read the audit logs, who can modify the audit logs, who can delete the audit logs, and how the audit logs are protected and stored. Is the user ID obscured in the audit trails?
7. **External Access Considerations:** Will the application be used internally only? If not, are you compliant with corporate external access requirements?

6.5.6 System Management Checklist

1. What is the frequency of software changes that must be distributed?
2. What tools are used for software distribution?
3. Are multiple software and/or data versions allowed in production?
4. What is the user data backup frequency and expected restore time?
5. How are user accounts created and managed?
6. What is the system license management strategy?
7. What general system administration tools are required?

8. What specific application administration tools are required?
9. What specific service administration tools are required?
10. How are service calls received and dispatched?
11. Describe how the system is uninstalled.
12. Describe the process or tools available for checking that the system is properly installed.
13. Describe tools or instrumentation that are available that monitor the health and performance of the system.
14. Describe the tools or process in place that can be used to determine where the system has been installed.
15. Describe what form of audit logs are in place to capture system history, particularly after a mishap.
16. Describe the capabilities of the system to dispatch its own error messages to service personnel.

6.5.7 System Engineering/Overall Architecture Checklists

6.5.7.1 General

1. What other applications and/or systems require integration with yours?
2. Describe the integration level and strategy with each.
3. How geographically distributed is the user base?
4. What is the strategic importance of this system to other user communities inside or outside the enterprise?
5. What computing resources are needed to provide system service to users inside the enterprise? Outside the enterprise and using enterprise computing assets? Outside the enterprise and using their own assets?
6. How can users outside the native delivery environment access your applications and data?
7. What is the life expectancy of this application?
8. Describe the design that accommodates changes in the user base, stored data, and delivery system technology.
9. What is the size of the user base and their expected performance level?
10. What performance and stress test techniques do you use?
11. What is the overall organization of the software and data components?
12. What is the overall service and system configuration?
13. How are software and data configured and mapped to the service and system configuration?
14. What proprietary technology (hardware and software) is needed for this system?

15. Describe how each and every version of the software can be reproduced and re-deployed over time.
16. Describe the current user base and how that base is expected to change over the next three to five years.
17. Describe the current geographic distribution of the user base and how that base is expected to change over the next three to five years.
18. Describe how many current or future users need to use the application in a mobile capacity or who need to work off-line.
19. Describe what the application generally does, the major components of the application, and the major data flows.
20. Describe the instrumentation included in the application that allows for the health and performance of the application to be monitored.
21. Describe the business justification for the system.
22. Describe the rationale for picking the system development language over other options in terms of initial development cost *versus* long-term maintenance cost.
23. Describe the systems analysis process that was used to come up with the system architecture and product selection phase of the system architecture.
24. Who besides the original customer might have a use for or benefit from using this system?
25. What percentage of the users use the system in browse mode *versus* update mode?
26. What is the typical length of requests that are transactional?
27. Do you need guaranteed data delivery or update, or does the system tolerate failure?
28. What are the up-time requirements of the system?
29. Describe where the system architecture adheres or does not adhere to standards.
30. Describe the project planning and analysis approach used on the project.

6.5.7.2 Processors/Servers/Clients

1. Describe the client/server Application Architecture.
2. Annotate the pictorial to illustrate where application functionality is executed.

6.5.7.3 Client

1. Are functions other than presentation performed on the user device?
2. Describe the data and process help facility being provided.
3. Describe the screen-to-screen navigation technique.
4. Describe how the user navigates between this and other applications.
5. How is this and other applications launched from the user device?

6. Are there any inter-application data and process sharing capabilities? If so, describe what is being shared and by what technique/technology.
7. Describe data volumes being transferred to the client.
8. What are the additional requirements for local data storage to support the application?
9. What are the additional requirements for local software storage/memory to support the application?
10. Are there any known hardware/software conflicts or capacity limitations caused by other application requirements or situations which would affect the application users?
11. Describe how the look-and-feel of your presentation layer compares to the look-and-feel of the other existing applications.
12. Describe to what extent the client needs to support asynchronous and/or synchronous communication.
13. Describe how the presentation layer of the system is separated from other computational or data transfer layers of the system.

6.5.7.4 Application Server

1. Can/do the presentation layer and application layers run on separate processors?
2. Can/do the application layer and data access layer run on separate processors?
3. Can this application be placed on an application server independent of all other applications? If not, explain the dependencies.
4. Can additional parallel application servers be easily added? If so, what is the load balancing mechanism?
5. Has the resource demand generated by the application been measured and what is the value? If so, has the capacity of the planned server been confirmed at the application and aggregate levels?

6.5.7.5 Data Server

1. Are there other applications which must share the data server? If so, identify them and describe the data and data access requirements.
2. Has the resource demand generated by the application been measured and what is the value? If so, has the capacity of the planned server been confirmed at the application and aggregate levels?

6.5.7.6 COTS (where applicable)

1. Is the vendor substantial and stable?
2. Will the enterprise receive source code upon demise of the vendor?
3. Is this software configured for the enterprise's usage?

4. Is there any peculiar A&D data or processes that would impede the use of this software?
 - Is this software currently available?
5. Has it been used/demonstrated for volume/availability/service-level requirements similar to those of the enterprise?
 - Describe the past financial and market share history of the vendor.

6.5.8 System Engineering/Methods & Tools Checklist

1. Do metrics exist for the current way of doing business?
2. Has the system owner created evaluation criteria that will be used to guide the project? Describe how the evaluation criteria will be used.
3. Has research of existing architectures been done to leverage existing work? Describe the method used to discover and understand. Will the architectures be integrated? If so, explain the method that will be used.
4. Describe the methods that will be used on the project:
 - For defining business strategies
 - For defining areas in need of improvement
 - For defining baseline and target business processes
 - For defining transition processes
 - For managing the project
 - For team communication
 - For knowledge management, change management, and configuration management
 - For software development
 - For referencing standards and statements of direction
 - For quality assurance of deliverables
 - For design reviews and deliverable acceptance
 - For capturing metrics
5. Are the methods documented and distributed to each team member?
6. To what extent are team members familiar with these methods?
7. What processes are in place to ensure compliance with the methods?
8. Describe the infrastructure that is in place to support the use of the methods through the end of the project and anticipated releases.
 - How is consultation and trouble-shooting provided?
 - How is training co-ordinated?
 - How are changes and enhancements incorporated and cascaded?

- How are lessons learned captured and communicated?
- 9. What tools are being used on the project? (Specify versions and platforms). To what extent are team members familiar with these tools?
- 10. Describe the infrastructure that is in place to support the use of the tools through the end of the project and anticipated releases.
 - How is consultation and trouble-shooting provided?
 - How is training co-ordinated?
 - How are changes and enhancements incorporated and cascaded?
 - How are lessons learned captured and communicated?
- 11. Describe how the project will promote the re-use of its deliverables and deliverable content.
- 12. Will the architecture designs "live" after the project has been implemented? Describe the method that will be used to incorporate changes back into the architecture designs.
- 13. Were the current processes defined?
- 14. Were issues documented, rated, and associated to current processes? If not, how do you know you are fixing something that is broken?
- 15. Were existing/planned process improvement activities identified and associated to current processes? If not, how do you know this activity is not in conflict with or redundant to other Statements of Work?
- 16. Do you have current metrics? Do you have forecasted metrics? If not, how do you know you are improving something?
- 17. What processes will you put in place to gather, evaluate, and report metrics?
- 18. What impacts will the new design have on existing business processes, organizations, and information systems? Have they been documented and shared with the owners?

6.6 Architecture Compliance Review Guidelines

6.6.1 Tailoring the Checklists

- Focus on:
 - High risk areas
 - Expected (and emergent) differentiators
- For each question in the checklist, understand:
 - The question itself
 - The principle behind it
 - What to look for in the responses
- Ask subject experts for their views

- Fix the checklist questions for your use
- Bear in mind the need for feedback to the Architecture Board

6.6.2 Conducting Architecture Compliance Reviews

- Understand clearly the objectives of those soliciting the review; and stay on track and deliver what was asked for. For example, they typically want to know what is right or wrong with the system being architected; not what is right or wrong with the development methodology used, their own management structure, etc. It is easy to get off-track and discuss subjects that are interesting and perhaps worthwhile, but not what was solicited. If you can shed light and insight on technical approaches, but the discussion is not necessary for the review, volunteer to provide it after the review.
- If it becomes obvious during the discussion that there are other issues that need to be addressed, which are outside the scope of the requested review, bring it up with the meeting chair afterwards. A plan for addressing the issues can then be developed in accordance with their degree of seriousness.
- Stay "scientific". Rather than: "We like to see large databases hosted on *ABC* rather than *XYZ*.", say things like: "The downtime associated with *XYZ* database environments is much greater than on *ABC* database environments. Therefore we don't recommend hosting type *M* and *N* systems in an *XYZ* environment."
- Ask "open" questions; i.e., questions that do not presume a particular answer.
- There are often "hidden agendas" or controversial issues among those soliciting a review, which you probably won't know up-front. A depersonalized approach to the discussions may help bridge the gaps of opinion rather than exacerbate them.
- Treat those being interviewed with respect. They may not have built the system "the way it should be", but they probably did the best they could under the circumstances in which they were placed.
- Help the exercise become a learning experience for you and the presenters.
- Reviews should include detailed assessment activities against the architectures and should ensure that the results are stored in the Enterprise Continuum.

Index

ADM	27
Architecture Board	21
Architecture Capability	3
Architecture Compliance	31
checklists	39
guidelines	51
review	33
review process	36
Architecture Contract	27
Architecture Governance	9, 12, 21
implementation	18
Architecture Governance Framework	9, 13
structure	13
COBIT	12
COBIT framework	24
corporate governance	9
COTS	33
dispensation	15
governance	
characteristics	10
ISACF	12
IT governance	9, 11
IT Governance Institute	12
OLA	15, 26
operational-level agreement	15, 26
ORB	44
service-level agreement	15
SLA	15, 22, 30
Statement of Architecture Work	29
technology governance	9, 11

