THE Open GROUP

World-Class EA: Governors' Approach to Developing and Exercising an Enterprise Architecture Governance Capability

A White Paper by:

Dave Hornford, Conexiam
Nathan Hornford, Conexiam
Sriram Sabesan, Conexiam
Chris Searle, Conexiam
Ken Street, Conexiam
Samantha Toder, Conexiam

June 2017

World-Class EA: Governors' Approach to Developing and Exercising an Enterprise Architecture Governance Capability

This White Paper is an informational document and does not form part of the TOGAF documentation set. Readers should note that this document has not been approved through the formal Open Group Standards Process and does not represent the formal consensus of The Open Group Architecture Forum.

World-Class EA:
Governors' Approach to Developing and Exercising an Enterprise Architecture Governance Capability
Document No.: W178

Published by The Open Group, June 2017.
Any comments relating to the material contained in this document may be submitted to:
The Open Group, Apex Plaza, Forbury Road, Reading, Berkshire, RG1 1AX, United Kingdom
or by email to:
ogpubs@opengroup.org

Table of Contents

Boundaryless Information Flow™
achieved through global interoperability
in a secure, reliable, and timely manner

Executive Summary

This White Paper is intended to provide additional guidance to using TOGAF®, a standard of The Open Group, for Enterprise Architecture Governance. This paper puts forward current thinking on developing, maintaining, and using an Enterprise Architecture Governance Capability based upon the established best practice contained within the TOGAF standard.

The TOGAF framework is set apart from every other Enterprise Architecture framework because it contains three central parts: a Method, a Content Framework, and an EA Capability Framework. By design, it is scalable and configurable. This is always troubling for new architecture governors who want a cookbook. The TOGAF standard does not provide a cookbook; it provides the essential scaffolding that Enterprise Architecture teams use to build their Enterprise Architecture Governance Capability.

This White Paper is structured to provide the context, content, and rationale behind choices and steps in developing and sustaining an Enterprise Architecture

Governance Capability.[1] In short, this paper is intended to guide the architecture governor to use the TOGAF essential scaffolding and this paper to deliver an actionable Enterprise Architecture Governance designed to deliver and protect ongoing value to an Enterprise.

[1] This paper strictly adheres to the definition of governance in ISO/IEC 38500:2015 (Governance of IT for the Organization) and ISO 19600:2014 (Compliance Management Systems) where governance will focus on the activities of direction and control. Any managing or monitoring activities will happen in other governance or management functions.

This guidance is for Enterprise Architecture Governance that is specifically set up to ensure that measurable value is delivered during any architecture-specified change and that measurable value continues to be delivered over time. This is not a guide for IT Governance; however, any EAGC should be integrated with other corporate governance structures, including IT Governance.

Definitions

To share a clear understanding with the authors, a few terms need to be defined distinctly from common English usage. The terms below are distinctly defined, and Capitalized wherever found. The authors mean exactly these definitions and nothing else in the content of this document.

We find that defining a term without an explanation tends to create a definition that must be broad enough to ensure that all reasonably conceivable edge cases are included. As a result, the definition quickly loses meaning. In this paper where a formal definition doesn't provide pragmatic guidance, we will provide a pragmatic explanation, and leave the argument on semantic purity to those whose day-job pays for semantic purity.

Enterprise

The highest level of description of an organization used to identify the boundary encompassed by the EA and EA Capability.

Note: This definition is deliberately flexible and not associated with an organization's legal or functional boundaries. It must cover monolithic organizations and extended organizations that include separate organizations connected by a mission or supply chain, as well as operating entities within an organization. Consider an organization that uses outsourced partners to provide manufacturing, logistics, and support; a multi-national peacekeeping force; and a multi-billion-dollar division of a Fortune 50 firm, all are Enterprises.

Enterprise Architecture (EA)

For convenience, we routinely use EA as a shorthand for Enterprise Architecture.

The two best definitions in our view that can be used are from Gartner and DoDAF.

Gartner[2] defines EA as: "*the process of translating business vision and strategy into effective Enterprise change by creating, communicating, and improving the key principles and models that describe the Enterprise's future state and enable its evolution*".

DoDAF defines architecture as: "*a set of abstractions and models that simplify and communicate complex structures, processes, rules, and constraints to improve understanding, implementation, forecasting, and resourcing*".

While many in the EA profession find distinguishing the terms "architecture" and "architecture description" useful, we do not see a need in this document. We think it is a distinction without a difference for a Practitioner.

[2] Refer to: www.gartner.com/doc/740712/gartner-clarifies-definition-term-enterprise.

Enterprise Architecture Governance Capability (EAGC)

For convenience, we use EAGC throughout this document as a shorthand for Enterprise Architecture Governance Capability.

Many use their EAGC to oversee many of the functions of IT decision-making or IT operations as well as many other functions such as managing IT innovations. The historic entanglement of EA and IT has created confusion in both the scope of EA, its role, and positioning in an Enterprise. For this paper, we clearly distinguish between EA and IT, and suggest readers review the TOGAF® Leader's Guide to Establishing and Evolving an EA Capability for purpose and organizational alignment patterns of an EA Capability.

We are not saying that IT governance is not important, or that it does not need attention. We are saying that a high-functioning EA Capability will be engaged well outside of the IT function, and when a high-functioning EA Capability exists it will be used to address all corporate functions, including IT. We only focus on the governance of EA.

We limit the use of governance to ensuring that the intended benefits and values expected with business-IT transformational investments are realized. For this EAGC discussion we will be strictly adhering to the ISO/IEC 38500:2015 definitions of *governance* & *governance framework*. Governance is the "system of directing and controlling". The governance framework includes "strategies, policies, decision-making structures, and accountabilities" through which the organization's governance arrangements operate.

Management of change initiatives, IT operational management and decision-making, and other business function operational management and decision-making are outside the scope of an EAGC.

Further, we limit the use of governance to ensuring that the intended benefits and values expected with business transformational investments are realized.

An EAGC is expressed in four ways:

- Governance over the creation and evolution of the target architecture
- Governance over the implementation[3] of the target architecture
- Governance over the realization of value of the target architecture
- Governance over the EA practice

[3] We have observed that a common trap is getting into efforts to fix terminology by using a different synonym. This is always done when people have added meaning, or special conditions, to a word. Implementation means "the process of putting a decision or plan into effect". Feel free to substitute transformation, change, program execution, deployment, or a different synonym if these words align with your language preferences.

Introduction

This White Paper provides specific, concise, and consistent guidance on using the TOGAF framework to develop, maintain, and govern an EA. This paper is a companion to the TOGAF framework and is intended to bring the concepts and generic constructs in the TOGAF framework to life.

This paper puts forward current thinking around the development, maintenance, and use of an EAGC that:

- Aligns to a set of requirements
- Aligns to the expectations of the Stakeholders
- Enables predictable value creation

The EAGC achieves this by enabling the Auditor and Stakeholder roles, ensuring that the architecture work and its implementation provide ongoing value to the Enterprise.

This White Paper is divided into three sections, as follows.

Part 1: Guidance on EA Governance Roles

This section addresses:

- Architecture governance roles
- Architecture governance decision rights
- Architecture governance board
- Architecture governance traceability

Part 2 Guidance on EA Governance

This section addresses:

- Governing the target architecture
- Governing the target architecture implementation
- Governing the delivery of value of the target architecture
- Governing the architecture practice

Part 3: Guidance on Maintaining an EAGC

This section addresses:

- Architecture governance repository
- Governance information management
- Architecture governance reporting

Governance Overview

An EA is developed for one very simple reason: the guidance of effective change that produces a value to the Enterprise.

An EAGC is employed to ensure that the changes to an Enterprise are well thought out and deliberate, and that the intended values and benefits are delivered as a result of the change.

Guidance on effective change will take place during the change activities to realize an approved target. During implementation, the EAGC is used by the Stakeholders to govern those changes. The first role of governance is to direct change activity – to align the change with the optimal path to realizing the expected value. The second role of governance is to control the change activity – ensuring the change stays on the optimal path.

It is often assumed that EA is used for initiatives with a substantial scope. Nothing can be further from the truth. Here we need to make a distinction between engaging the EA staff and using the EA. The EA, the "set of abstractions and models that simplify and communicate the complexity of the Enterprise to improve understanding, implementation, forecasting, and resourcing", should be used anytime there is a change to ensure that the value of the change is protected.

The same concepts, methods, techniques, and frameworks can readily be used to address the end state, preference trade-offs, and ensure value realization for big and little questions. A well established EAGC can be used to ensure that the appropriate level of detail is present in the architecture to attain required confidence in the answer provided, whether it be a large or small question. Essentially, the scope of the system varies; the detail in description of elements and properties vary; all the concepts remain the same.

Governance over the target[4] ensures that the right Stakeholders have been identified and that their concerns are addressed. It is necessary to have the target architecture approved by appropriate Stakeholders, otherwise all you have is an opinion, and an unapproved opinion cannot be subject to governance.

To ensure that the intended benefits and value are delivered during implementation it is necessary for an EAGC to maintain vigilance over the implementation of the target. Most emphatically, this does not mean that architects need to attend every implementation meeting. In fact, in a high-functioning EAGC most implementation governance activity does not involve the architecture team. It does mean that the EAGC needs to establish the cross-organizational processes and information to ensure that any deviations from the architecture during implementation that reduce the value delivered are identified and acted upon.

In many cases an implementation project will end well before value can be realized. The impact of learning curves to absorb and internalize the change or adjustments to changing operational dynamics could negatively impact the value immediately after completion of the change project. The EAGC guides the Enterprise to make ongoing adjustments to the portfolio to assure the value of the target architecture is well protected.

[4] We use target as a short-hand for the description of the Enterprise at the end of a planning cycle, or implementation. For the EAGC, it includes the changes required, constraints on implementing the change, and the expected value.

Governance over the EA Capability is a critical aspect of good governance and is needed to protect the target architectures value. The EA Capability must evolve in order to support the Enterprise, and the EAGC governs changes to the EA Capability.

To find out more on how to develop governance as part of an evolving EA Capability see the TOGAF® Leader's Guide to Establishing and Evolving an EA Capability.

When following the TOGAF method to create architecture for a purpose (to support strategy, portfolio, project, or solution delivery) the entire path is governed. The objective of this governance is to provide confidence to decision-makers that architecture change recommendations are well considered, well grounded, and that the required work was completed. Decision-makers and Implementers should also be confident that the architecture is well connected with other architectures in the EA landscape and that the architecture also considers time and recency impacts.

The activities an architecture governor performs are dependent on two things: the expression the governance is performed at and the purpose for which the architecture was created (see the context tables in Enterprise Architecture Governance Context on page 17).

The key role of an architecture governor is that of an Auditor, whose function is to ensure that the architecture suits the intended purpose and that any change work has been completed to the right level of detail. In short, the architecture governor is the guardian of the value EA delivers to the organization.

It is important to exercise care when extending the roles and decision authorities in your EAGC to ensure that puffing or, worse, roadblocks to your Enterprise's ability to achieve value realization are not being introduced.

How to Use this White Paper with the TOGAF Standard

This paper is written directly for the person who needs to govern using EA. The person who is not worried about the theory, who is not worried about how to structure, develop, or maintain an EA Capability. The governor is deeply concerned with realizing value:

- Does the current target architecture describe the desired value, and the path to realizing it?
- Is the current implementation activity on the path to enabling the Enterprise to realize the value expected by the target architecture?
- Are the Enterprise's operations able to achieve the value expected by the implementation?
- Is the EA Capability aligned to the current need of the Enterprise to obtain value?

While this paper assumes no detailed knowledge of the TOGAF framework, it explores the core concepts of the TOGAF 9.1 standard. It places these concepts together in the context of using them to govern an EA. This includes guidance on how an EAGC is used to protect the value of business transformation investments.

We strongly recommend learning the TOGAF conceptual structures in the context of using them; using the TOGAF standard as a framework rather than as an educational topic. This paper follows that approach. This paper assumes that you have established an EA Capability and have customized the TOGAF framework for your Enterprise.[5]

For a complete interpretation of the TOGAF standard, we suggest reading this White Paper in conjunction with the White Paper: World-Class EA: A Leader's Approach to Establishing and Evolving an EA Capability and the White Paper: World-Class EA: A Practitioners' Approach to Developing Enterprise Architecture Following the TOGAF® ADM. Together with these White Papers, this paper supports the exploration of the TOGAF framework to establish an EA Capability and how that established EA Capability develops, maintains, governs, and uses an EA to ensure value delivery.

Referenced Techniques

We reference techniques and key literature created by thought leaders that we actively use in our work. We have a limited reference to materials freely available through standards organizations and academic publications. We do not promote or reference any commercial techniques, even our own. There is often commercial material available for topics discussed in this paper. It is up to the reader to seek them.

References to key literature and their techniques are intended to be representative. The reader is expected to read and assimilate referenced publications for a full understanding of these related topics.

This White Paper does not suggest that any of the referenced tool, techniques, and literature are definitive. Other tools, techniques, and literature can readily be substituted. In fact, the referenced material is part of a body of knowledge that continuously evolves, and the reader is advised to explore updates to literature and techniques referenced in this paper.

[5] For assistance customizing the TOGAF framework, we recommend The Open Group Guide: The TOGAF® Leader's Guide to Establishing and Evolving an EA Capability, which provides in-depth commentary and guidance for executing the TOGAF ADM Preliminary Phase.

Governance Expressions and Authorities

The governance expression ensures that the architecture being developed is done to the right level of detail and that the Stakeholder is confident that the expected value will be delivered. Although there is always more architecture work to be done than there will ever be time, money, or resources to accomplish, it is important that the architecture completed is focused and provides value. The governor should always be crystal clear on the purpose of the architecture being developed. The governor's primarily role is to ensure the architecture remains focused on its purpose.

The four expressions of an EAGC are:

- Governance over the target architecture
- Governance over implementation
- Governance over the architecture practice
- Governance over value realization

Governance over the Target Architecture

This expression addresses the creation of a new target architecture. Governance will focus on:

- The purpose for which the target is being created
- Compliance with existing superior architecture
- Approval of the new target

Neither the governor nor an architecture governing board has any decision authority over the target architecture. The target architecture is developed for Stakeholders, and it is the Stakeholder who has the decision rights.

Without approval by the Stakeholders of the target, no implementation governance is possible, nor is governance of more detailed architecture. Without approval, the Practitioner has only a documented opinion. Subject Matter Experts, Implementers, and decision-makers also have opinions. Stakeholders have decision rights.

The architecture needs to be timely and in sync with the organization's business cycle; architecture delivered the day after a decision is made is valueless.

Governance over the Implementation

This expression addresses the implementation work to perform the changes necessary to reach the target or a transition state. The target will identify gaps that must be filled and constraints that limit design choice. Constraints will typically include an approach, or strategy, for a work package, and architecture specifications that limit design choice. Governance will focus on:

- Compliance of design work with the target architecture

- Compliance of implementation work with the design, and target

Neither the governor nor an architecture governing board has any decision authority over the implementation of the target architecture. Implementation is performed for a Stakeholder, and it is the Stakeholder who has the decision rights.

In its simplest form, implementation governance has two functions via which it provides value. The first is to ensure that the solution design conforms to the target, and second that the final deliverable conforms to the design.

Within the constraints and expectation of the target, Implementers have decision rights as to how the value will be delivered. They decide on the solution, implementation method, resourcing, and all other implementation decisions. They must comply with the approved architecture and identify any deviations for Stakeholder approval.

It is the governor's responsibility to identify any compliance issue, or variance, to the Stakeholder so the Stakeholder can make an informed decision regarding the impact to value any implementation compliance issue will have.

Governance over Value Realization

This expression addresses realization of the value the target was designed to deliver. Typically, a target will be achieved over time after an implementation provides part of a capability and after several changes occur in the Enterprise. No single project will be expected to deliver the value. Governance will focus on:

- Tracing value realization expected in the target architecture
- Focusing attention on expected value

To be able to govern value realization, a Stakeholder, or value owner, who cares about the value to be delivered needs to be identified.[6] The Practitioner does need to identify this risk to value.

The Practitioner is responsible for ensuring that the value realization processes and required metrics are established to inform the value owner. This is necessary so the value can be proved and that the ongoing value is realized. All managing or monitoring functions are performed by the value owner. The architecture governance test: did the Practitioner complete the architecture work and did the Implementers deliver it so that a value owner can monitor and manage the value realization over time.

Value delivery should be monitored during the whole architecture project or architecture endeavor, so value that is delivered in the early stages of the process should deliver ongoing value in the later stages. If this is not accomplished, the gap should be identified, and the portfolio corrected and then tested again using the corresponding metrics previously defined in the architecture.

[6] If a value owner cannot be identified, or the Stakeholders determine that one is not needed, there is no reason to pursue governance over the value realization. These events should be fed into improvement in the EA Capability.

Governance over the Enterprise Architecture Practice

An EA Capability focused on supporting decision-making for strategy will use a different set of work products than an EA Capability chartered to support governance of projects. This is a critical distinction. The EA Capability must be developed to align with its expected value.

The governor needs to ensure that the EA Capability, whether a single organization or a complex organization, is structured, staffed, resourced, and engaged to align to purpose. A central focus of the TOGAF® Leader's Guide was to highlight that an EA Capability will have distinct structure, staffing, skills, process touch-points, and resources, and will deliver different artifacts to different purposes (architecture to support strategy, portfolio, project, and solution delivery; see Architecture for Purpose on page 16).

Governance on this expression will focus on ensuring the leader of the EA Capability is:

- Focused on consistently delivering value to the Enterprise
- Enabled for success in that endeavor

Architecture for Purpose

A purpose-based EA Capability model identifies four purposes that typically frame the planning horizon, depth, and breadth of an architecture project and the contents of the EA repository. The purpose-based EA Capability model used in this White Paper was introduced in the White Paper: World-Class Enterprise Architecture and refined in the White Paper: World-Class EA: A Leader's Approach to Establishing and Evolving an EA Capability.

The four broad purposes of an EA Capability are shown in Figure 1.

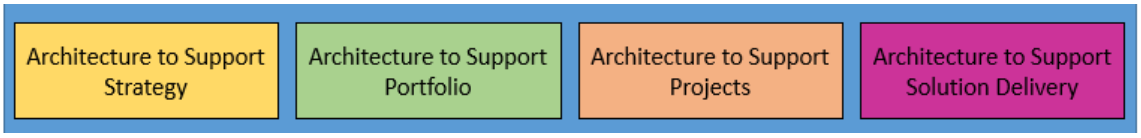

Figure 1: Purposes of Enterprise Architecture

EA to Support Strategy

Deliver EA to provide an end-to-end target architecture, and develop roadmaps of change over a three to ten-year period. An architecture for this purpose will typically span many change programs or portfolios. In this context, architecture is used to identify change initiatives and supporting portfolio and programs. Set terms of reference, identify synergies, and govern the execution of strategy via portfolio and programs.

EA to Support Portfolio

Deliver EA to support cross-functional, multi-phase, and multi-project change initiatives. An architecture for this purpose will typically span a single portfolio. In this context, architecture is used to identify projects, and set their terms of reference, align their approaches, identify synergies, and govern their execution of projects.

EA to Support Project

Deliver EA to support the Enterprise's project delivery method. An architecture for this purpose will typically span a single project. In this context, the architecture is used to clarify the purpose and value of the project, identify requirements to address synergy and future dependency, assure compliance with architectural governance, and to support integration and alignment between projects.

EA to Support Solution Delivery

Deliver EA that is used to support the solution deployment. An architecture for this purpose will typically be a single project or a significant part of it. In this context, the architecture is used to define how the change will be designed and delivered, identify constraints, controls, and architecture requirements to the design, and, finally, act as a governance framework for change.

Enterprise Architecture Governance Context

In the tables below, we present the context, thought process, and related set of actions needed to define the required governance activities to support value realization. Not all activities listed in the context tables are performed by any one role. Most emphatically, this is not a list for an EA Practitioner. The value owner is a Stakeholder class that describes an individual who cares about or is assigned to care about the monitoring and managing of the delivered and ongoing value realization.

The tables contain examples of the governance activities to be considered.

The activities an architecture governor performs are dependent on the intersection of the governance expression and the architecture purpose.

Governance for Target Architecture

	Architecture Purpose			
	Architecture to Support Strategy	**Architecture to Support Portfolio**	**Architecture to Support Projects**	**Architecture to Support Solution Delivery**
Governance for Target Architecture	Activities here support the alignment of the organizational goals and objectives with the architecture.	Activities here support the selection of what work needs to be done to deliver the required change(s).	Activities here support budget approval.	Activities here ensure that the design and implementation teams are appropriately constrained.
	Enterprise Context:	**Enterprise and Portfolio Context:**	**Portfolio Context:**	**Delivery Context:**
	Were the right views developed?	Were the work packages completed?	Do the work packages properly describe the change work needed?	Are the change cost estimations approved?
		Were the priorities and dependencies identified and approved?	Are architecture specifications completed?	Are architecture specifications completed?
		Were the trade-off discussions finalized?	Were the trade-off discussions finalized?	Are the solution partners chosen and do they understand the value that needs to be delivered?

		Do the solution delivery notebooks describe the change for more detailed target architecture?	Do the solution delivery notebooks describe the change for an implementation leader to deliver on?	Do the solution delivery notebooks describe the change for an implementation team to deliver on?
	Is the architecture at the right level of completeness and confidence? Did the Stakeholder approve the architecture?	Is the architecture at the right level of completeness and confidence? Did the Stakeholder approve the architecture?	Is the architecture at the right level of completeness and confidence? Did the Stakeholder approve the architecture?	Is the architecture at the right level of completeness and confidence? Did the Stakeholder approve the architecture?

Governance for Target Architecture Implementation

	Architecture Purpose			
	Architecture to Support Strategy	**Architecture to Support Portfolio**	**Architecture to Support Projects**	**Architecture to Support Solution Delivery**
	Activities here support developing constraints to align changes to organizational goals, objectives, and value.	Activities here are aligned to understanding the intended benefits and values of the change.	Activities here are in support of developing the required programs and projects.	Activities here ensure that the solution design and the delivery of the change comply with the architecture.
Governance for Target Architecture Implementation	**Enterprise Context:**	**Enterprise and Portfolio Context:**	**Portfolio Context:**	**Program and Project Context:**
	Risk and control reviews completed.	All initiative values are understood and traceable?	Ensure alignment and integration between projects.	Are the solution delivery notebooks developed?
	Did the Stakeholder approve the architecture roadmap and value traceability?	Do the Implementers understand the value assessment?	Do the Implementers understand the value expectation and traceability?	Is the governance reporting and monitoring developed and in place?
		Ensure alignment with the roadmap.	Ensure alignment with the roadmap.	Ensure alignment with the roadmap, transition architecture, and work packages.

		Did the Stakeholder approve the architecture roadmap and value traceability?	Did the Stakeholder approve the architecture roadmap and value traceability?	Did the Stakeholder approve the architecture roadmap and value traceability?

Governance for Value Realization

	Architecture Purpose			
	Architecture to Support Strategy	**Architecture to Support Portfolio**	**Architecture to Support Projects**	**Architecture to Support Solution Delivery**
	Activities here identify that value owners are identified and are assigned accountability.	Activities here are to collaborate with assigned value owners and develop the value metrics and processes.	Activities here are for the implementation team to deliver the value metrics and processes.	Activities here are for the value owner to monitor and manage the value metrics.
Governance for Value Realization	**Enterprise Context:**	**Portfolio Context:**	**Program and Project Context:**	**Delivery Context:**
	Stakeholders understand the value to be delivered and have approved.	Metrics to be monitored are identified and approved.	Metrics to be monitored are developed.	Compliance to architecture is monitored.
		The processes used to monitor are identified and approved.	The processes used to monitor are developed.	Value metrics are reviewed with Stakeholders after implementation and then operationalized for ongoing value delivery.

Governance over the Architecture Practice

	Architecture Purpose			
	Architecture to Support Strategy	**Architecture to Support Portfolio**	**Architecture to Support Projects**	**Architecture to Support Solution Delivery**
Governance over the Architecture Practice	**Enterprise Architecture Governance Capability Monitors**			
	EA Capability purpose & priorities			
	EA team (skills, resources, & organizational alignment)			

	Architecture viewpoints
	Architecture standards
	EA repository (meta-model, analytic & modeling tool)
	Architecture templates
	Architecture deliverables for Enterprise processes
	Compliance and governance processes

Part 1: Guidance on Architecture Governance Roles

Architecture Governance Roles

In order to define architecture governance roles, we focus on context, thought process, and the action(s) taken by the individual to drive the distinction between roles. For the purpose of this discussion, a distinction must be drawn between the *role* and the person *performing the role*.

During interactions, even before completing an architecture conversation, an individual may switch between multiple roles, often making the transition implicitly in their head and not necessarily being cognizant of the transitions. When an individual is not cognizant of role switches, mistakes can be made in exercising authority. Indeed the most common failure pattern is role confusion. Facilitating effective communication requires us to make a distinction between the communities who are interested in the architecture.

The primary roles involved in an effective EAGC are as follows:[7]

- Stakeholder – owner of the architecture

  Provides priority, preference, and direction. All decision rights regarding the target architecture, and any relief from and enforcement of the target, are vested in the Stakeholders.

- Stakeholder Agent – representative of the Stakeholder

- Subject Matter Expert – possesses specialized knowledge about some aspect of the Enterprise or the environment in which it operates

  Provides knowledge, advice, and validation of interpretation.

- Implementer – responsible for performing all change activity

  The scope of change is not relevant. Transformative capital projects and incremental operational changes are changes performed by an Implementer. All decision rights about proposed implementation choices, such as design, product selection, and change sequence, are vested with the Implementer.

- Practitioner (Architect) – developer of the target architecture

  Provides recommendations when non-compliance with the target is determined.

- Auditor – performs systematic reviews of both the target and implementation

  Best performed at multiple stages to capture errors before the cost of correction exceeds potential value realization. All decision rights about compliance during the development of the architecture and implementation are vested with the Implementer. Auditing can be performed within a formal structure such as an architecture governing board or by a peer reviewer. Auditing can also be self-performed but the role being performed needs to be clear in the mind of the individual and that they are acting in accordance with the role.

[7] See the White Paper: World-Class EA: A Practitioners' Approach to Developing Enterprise Architecture Following the TOGAF® ADM.

When architecture development drives deep into the EA landscape, the expanding level of detail causes the level of scrutiny to increase exponentially. As the issues are abstracted for Stakeholders and decision-makers, quantity of data and level of detail shrinks dramatically. The Practitioner must pay attention to what is being governed at what level and hence the information needs.

In many organizations, the Practitioner will fill the role of Stakeholder Agent, Subject Matter Expert, and Implementer. This typically occurs when the organization does not use architecture to direct and control change. Instead, the organization attempts to use skilled thoughtful individuals to make tactical decisions. The value is illusionary.

When the Practitioner is reflecting the implications of an architecture specification, they are playing the role of a Stakeholder Agent. While clarifying the impact of a solution to related projects, they play the role of the Subject Matter Expert, and while discussing rationalization of implementation choices to the architecture, the role of the Practitioner. Decisions and guidance provided as a Stakeholder Agent are to be treated as absolute and resolved. Whereas, decisions made playing the other two roles are subject to resolution, relief, or appeal to a higher authority in the governance chain. Likewise, a sponsor can be playing the role of a decision-maker or a Stakeholder or that of a sponsor. Directions provided by a sponsor are subject to relief requests, while the other two are not.

Stakeholder

Owner of the architecture. Provides priority, preference, and direction. All decision rights about the target architecture, and any relief from and enforcement of the target are vested in the Stakeholders.

We follow the ISO/IEC/IEEE 42010 guidance on Stakeholders which focuses our attention on those whose concerns are fundamental to the architecture, or architecturally significant.[8]

A Stakeholder holds approval rights on the target and the implementation. Stakeholders own the architecture and the value preference and priority the architecture is expected to enable.

Best practice governance has the Architect demonstrate that the views produced for the Stakeholders, along with any related constraints and guidance, are derived from the architecture. Stakeholders approve views, not architecture descriptions.

[8] The term Stakeholder is one of the most baggage-laden terms. The correct Stakeholders need to be identified, and since not everyone can take on that role, a careful Stakeholder assessment needs to be made. ISO/IEC/IEEE 42010 guidance on Stakeholders in the context of an architecture description is useful. Stakeholders are those whose "concerns are considered fundamental to the architecture, or architecturally significant".

The TOGAF 9.1 definition is: "an individual, team, or organization (or classes thereof) *with interests in, or concerns relative to, the outcome of the architecture*. Different Stakeholders with different roles will have different concerns".

The PMI definition is: "an individual, group, or organization, *who may affect, be affected by, or perceive itself to be affected* by a decision, activity, or outcome of a project".

The ISO/IEC/IEEE 42010 definition is: "an individual, team, organization, or classes thereof, having an interest in an Enterprise or system".

Stakeholder Agent

A Stakeholder typically is unavailable to perform all the activities required of a Stakeholder. In practice, Stakeholders will formally, or more often informally, assign an agent to represent them. It can be and often is that the Practitioner is assigned the role of Stakeholder Agent. The Practitioner is then wearing two hats and must be clear in their mind which role they are performing.

Auditors should pay close attention when the same individual Practitioner routinely acts as a Stakeholder Agent and Practitioner.

Subject Matter Expert

Also, known as a domain expert. A person with *bona fide* expert knowledge about what it takes to do a job. A person with a special knowledge or skills in an area of endeavor or topic. A person who understands a subject better than most other people; what's core and what's peripheral in that knowledge domain; is familiar with the most recent developments in the field; and, on the scarier side, can be summoned by a court to provide testimony. Provides knowledge, advice, and validation of interpretation. The Subject Matter Expert has no decision authorities.

For the purposes of this White Paper, the team or agency building the solution is not of interest. Their interests, concerns, and challenges are represented to the architecture effort by the Solution Architect or another Implementer performing this role.

This role is often performed by a Solution Architect.[9]

Auditors should pay close attention when the same individual performs the Practitioner and Subject Matter Expert roles; and investigate when the same individual performs the Subject Matter Expert and Stakeholder roles. Subject Matter Experts are typically poorly disposed temperamentally to undertake the trade-offs and to take the proper actions necessary to satisfy architecture approval.

Implementer

Responsible for performing all change activity. The scope of change is not relevant. Transformative capital projects and incremental operational changes are changes performed by an Implementer. All decision rights about proposed implementation choices, such as design, product selection, and change sequence are vested with the Implementer.

[9] A Solution Architect is someone who is well versed in one or more domain architectures. A Solution Architect focuses on converting requirements into the architecture and design that ultimately constitute the blueprint for the solution. In that process, the Solution Architect usually relies on design patterns from their previous engagements, published reference architectures, and on guidance from Enterprise Architecture. (Derived from Aligning Enterprise, System, and Software Architectures, Mistrik et al.)

Solution Architects play an important role in ensuring that the solution architecture aligns with the roadmaps established by the Enterprise Architecture, and that it adheres to the EA principles. Solution Architects are both a consumer and contributor to EA collateral. Often, the patterns and guidance Solution Architects develop becomes reusable in a broader EA context. (Derived from Breaking Down Software Development Roles.)

Auditors should investigate when the same individual performs the Implementer and any other role. Implementers are typically poorly disposed to undertake the trade-offs and to take the proper actions necessary for architecture.

Practitioner (Architect)

The person tasked to develop, maintain, and use an Enterprise Architecture. The Practitioner is the developer of the target architecture, guides effective change, and protects the Stakeholder's interest in the value that is to be realized from the implementation of the target architecture. Practitioners are best served when they identify Stakeholders who have approval rights and separate them from other Stakeholders who only need communications about the Enterprise Architecture. Practitioners provide recommendations when non-compliance with the target is identified; an Architect must be clear that as a Practitioner they have no decision rights. However, a Practitioner often has a dual role, the other being the role of Stakeholder Agent. This is a trusted advisor role and the Architect is acting on behalf of a Stakeholder and must make decisions from the Stakeholder's perspective and not that of a Practitioner.

During the course of developing the target architecture the Practitioner will provide trade-off implications and recommendations to the Stakeholder. The Practitioner can act as the Stakeholder Agent and hold the Stakeholder's decision rights by proxy. Outside of performing the role of Stakeholder Agent, the Practitioner has no decision rights.

When a non-compliance with the target is determined during an implementation, when required, the Practitioner must present the non-compliance to the Stakeholder with the recommendation to:

- Provide relief to the Implementer
- Ensure the Implementer conforms to the target
- End the implementation

Practitioners will often fulfill multiple roles in the architecture development and change processes.

Auditor

The individual who examines the records presented and certifies their conformance with process, procedures, standards, and guidelines. This individual produces a compliance or conformance report that is used by the governance body to assess and decide next steps. Normally performed by the Enterprise Architect or the Solution Architect.

In organizations where the EA team is embedded within an IT organization there is typically an overlap between IT and EA activity. In many IT organizations, the individual performing the Auditor role also has an IT decision-making or IT operational role. This is a result of the historic entanglement of EA and IT that has created confusion in both the scope of EA and its role and positioning in an Enterprise. Blurring the lines between auditing and any operational decision-making roles is poor practice.

When the Auditor role is entangled within IT operations, the most common problem is the development of an architecture governance board that appropriates decision-making authority. Whatever governance structure is in place, the only decision rights available are to the Stakeholder.

Blurring the Auditor and Practitioner role is common practice, and we have never seen the roles blurred without a mistaken appropriation of decision-making rights. We have never seen this practice succeed. Auditing can be self-performed, but the role that the individual is performing needs to be clear and they must be acting in accordance with the role.

The Auditor performs systematic reviews of both the target and implementation. Auditing is best performed at multiple stages to capture errors before the cost of correction exceeds potential value realization.

Auditors report to Stakeholders, with an architecture board existing to manage the process.

Architecture Governance Board

Setting up the Board

Architecture governance is practiced in most (if not all) Enterprises with an EA Capability irrespective of the EA Capability's maturity level. It may be being done formally or informally, by a governance board or an individual. When establishing a formal governance process the trick is to determine what is working well and what needs to be changed; a typical failure path during the set-up of an architecture governance board is to fix things that are not broken and that are actually functioning well and providing value.

An architecture board should be neither heavy-handed nor a bureaucratic juggernaut; it should be fit for purpose and it should enable the EA practice to provide value. There is no "right way" to set up an architecture governing board; it does not need to be excessively formal and it may be that existing informal processes are functioning very well. It may only be necessary to improve information flows and put some rigor around the existing processes; e.g., document the board's findings and ensure those that need the information have access to it. Always bear in mind that an architecture governing board has no decision rights; it merely exercises the architecture audit process and reports its findings to the Stakeholders.

This EA governance audit function can be performed by simply using checklists (see Target Governance Checklist on page 34 and Implementation Governance Checklist on page 36). The purpose is to ensure that the architecture has been done to the right level of detail and that the Stakeholder is confident that the architecture presented will deliver value.

It is equally as important for the board to determine whether too much architecture work is being done as it is to determine if not enough was done. It is preferable to determine whether the architecture is going into too much detail prior to the completion of the work so as to ensure that an overspend of architecture resources does not occur. An iterative approach to architecture development is good practice and can aid in achieving the appropriate level of detail. Frequent peer reviews or reviews by an architecture community of practice can accomplish this and can make a governance board's job considerably more straightforward.

When setting up an architecture governing board, once the board's purpose, accountabilities, and role in value realization are determined, some basic information flows and processes will be required in order to ensure that four primary responsibilities are performed:

- Maintaining links to Stakeholders
- Establishing, approving, and delivering a target architecture that delivers measurable value
- Assuring effective EA performance by doing only what is required to achieve the level of confidence required
- Ensuring that measurable value is delivered over time by ensuring performance indicators are in place for post-implementation value realization

Setting up the Governance Process

All organizations have existing change processes. The EA team needs to be aligned with the organization's planning, budgeting, operational, and change processes. The governor must understand that a theoretically perfect world where the EA team is engaged in all change cannot be expected. In practice, the scope of the EA team will be limited to some purposes, or will only be engaged in some changes.

For most organizations, the budget cycle controls change in the organization. Pragmatically, the EA team will be aligned to the budget cycle. Figure 2 shows a timeline view, depicting an alignment of key decisions made during a business cycle and the purpose architectures. EA for Strategy, Portfolio, and Projects needs to be completed before key milestones for budget decisions are made. EA for Solution Delivery is a continuous operation around budget control. The key takeaway is architecture *before* the decision.

Governance processes should align to the business cycle and assure that the architecture is delivered *before* the decision. To deliver architecture before the decision, the governance process should be able to certify the work and compliance to defined processes.

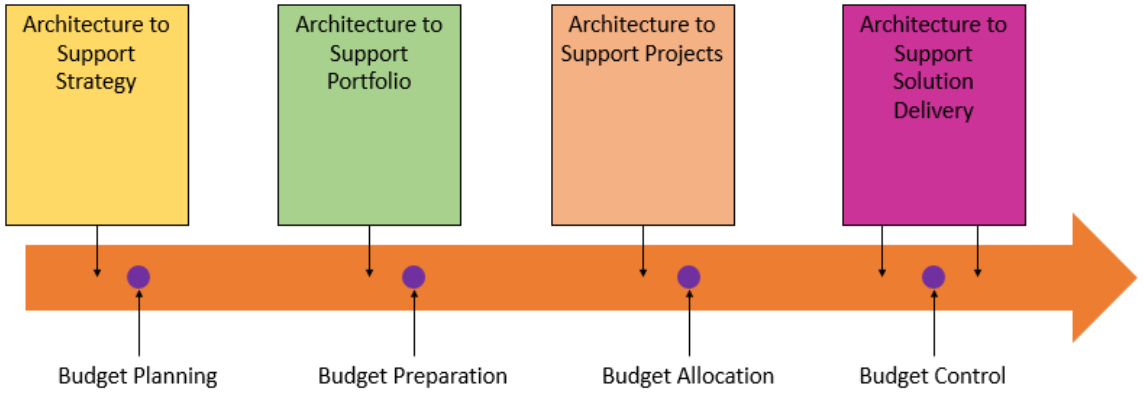

Figure 2: Business Cycle and Architecture by Purpose

We tie everything to the budget cycle to highlight the importance of good EA on guiding and constraining the change decisions. When there is no practical input from a good EA team before the decision an organization needs to take is made, the decision is still made. It might even be a good choice. But it was a less informed choice.

Keep in mind that in all EA, the Stakeholders, decision-makers, and Implementers require effective support ahead of the decision. Good architecture that informs decisions is infinitely more valuable than perfect architecture that follows decision and execution.

At all levels of the governance process, it is essential that measurements, metrics, and rationale for relief are defined in business terms. Governing a portfolio by number of machines eliminated does not relate itself to a business outcome. Translate to something like cost optimization for the same operational capacity.

Governance often results in a change, either to current effort or future efforts. Organizational and architecture change management should account for triggers and provide a timeline to implement the change from governance decisions. Imagine opening a faucet for hot water in the morning. Other control mechanisms sense the opening of the faucet, and it takes a while for the hot water to start flowing out of the faucet –

flushing out the cold water in the line. Governance operates in a similar way at times, and its process should also account for long lead times for corrective actions to take effect.

Traceability for Architecture Governance

The governance process does not have to be a heavyweight bureaucracy. It is simply based on demonstrating sufficient traceability that the organization can have confidence in the target being the best path to reaching the Enterprise's preferences. With confidence, the Enterprise will enforce the target in deliberate change activity.

Without traceability from your Enterprise goals, objectives, and value propositions to your business transformational investments, value cannot be delivered to the organization. The investments made to organizational changes and the architecture will have no way to demonstrate that value was delivered, even if there was value delivered. See Figure 3 below.

To prove that the work you do as an Architect is useful, you must be able to prove that you are the protector of value. If you can't, why would your organization need you?

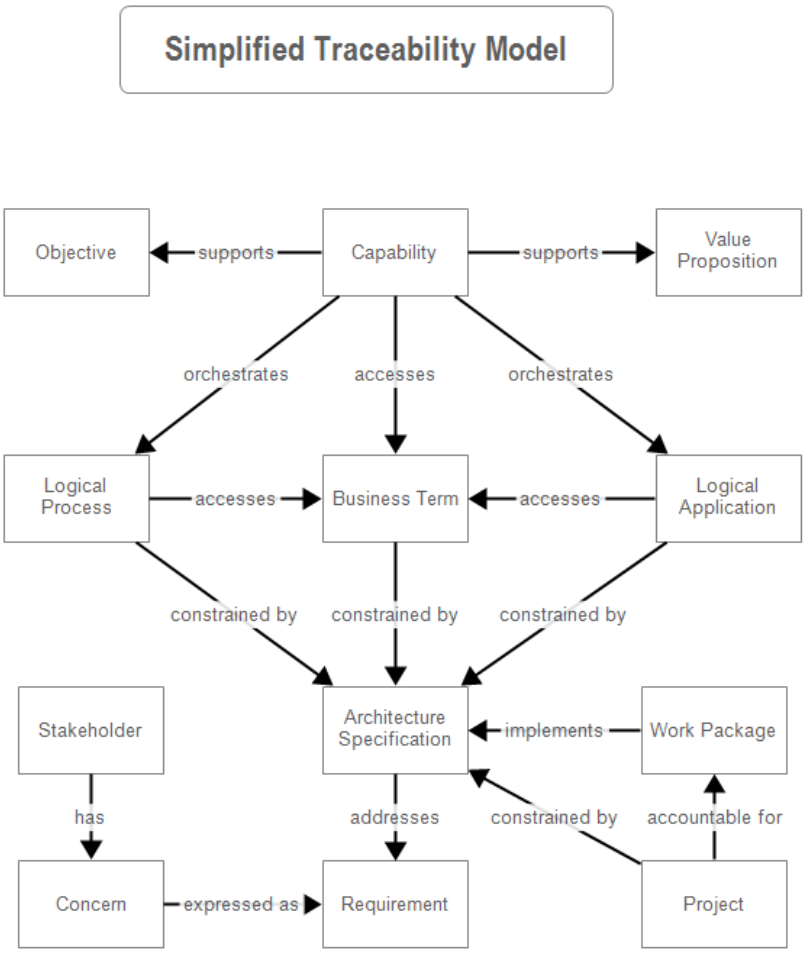

Figure 3: Simple Traceability Model across Architecture Components (Example)

Governance over Architecture Completeness and Confidence

The EAGC can be the thermometer of architecture value by helping to determine when enough work has been done to provide value. There are two aspects that can be used to determine how much work is required to provide value: completeness and confidence. As an example, if you can get to a level of 80% confidence with only doing 20% of the architecture (completeness), then stop. Doing more may start to deliver diminishing value returns.

Completeness comes in two forms. The first is to ensure you get to the right level of work for your architecture to have the right level of confidence. You do not need to do more work than necessary to provide value; in fact, the less work you do the better. Many Architect teams deliver way past the point of delivering value. You will never have enough time, money, or resources to do a complete architecture.

The other role of completeness is to ensure that every implementation provides value. Far too many projects build metaphorical half-bridges; building everything but the last piece to cross the obstacle. The justification is usually to "make progress". Bluntly, an organization is not making progress when it embarks on a change it will not finish. The organization is simply wasting resources. If you are not going to get to a value delivery point with your change initiative, then do not start.

Architecture Completeness

Completeness of an architecture indicates the level of architecture work that has been done compared to an architecture that has been brought to an end state and does not require any further work.

Completeness is required to be able to understand that the architecture will provide value and that the value will be delivered.

Completeness is an audit and management assertion concept. It is derived from two parameters: inclusion of all disclosures and significant transactions being recorded. The Auditor should be able to assert that there are no misstatements and no liabilities exist. Not including a Stakeholder in discovery or subsequent communication is a liability to the stability and validity of the architecture. From a governance point of view, the Architect should be able to demonstrate readiness for audit by showing engagement of appropriate Stakeholders, identifying the processes, systems, or control improvements, and recommendations for appropriate allocation of resources.

Architecture Confidence

Confidence in an architecture is the belief in your architecture to deliver value. This is done from the perspective of your Stakeholder that is approving your target architecture. This often must be expressed by the Architect acting as an agent on behalf of the approving Stakeholder.

Confidence is another audit and management assertion concept. It is based on two factors: the belief of the Stakeholder that the architecture will deliver value, and the reduction in the incremental value between two iterations of elaborating the architecture. When the elaborations take too long or the value measurements are conducted infrequently, the accuracy of the measurements or the parameters cannot be trusted to be stable. The Architect should be able to demonstrate the Stakeholder's belief and that measurements were taken in relatively short succession.

Part 2: Guidance on Architecture Governance

Governing the Target

Without approval by the Stakeholders, no implementation governance is possible, and no governance of more detailed architecture is possible. Without approval, the Practitioner only has a documented opinion. Stakeholders, Subject Matter Experts, Implementers, and decision-makers also have opinions.

Real architecture approval is and should be complex. The Practitioner is assisting their organization to select the best possible path against a set of competing preferences over time. The Practitioner needs to take the time to explore options and impacts.

With an approved target architecture, traceability to the objective is available, and trade-off has been performed. Good architecture trade-off explores options, cost, and benefits to arrive at the optimal answer for an organization. Often that answer is a compromise between competing interests.

Strategic Architecture

Getting to value requires that your architecture has complete traceability from your strategic to the implementation components of your architecture. A strategic architecture is a vital component to ensure value realization.

The EAGC can be used to determine where on the strategic architecture continuum you are and where your organization needs to be to deliver the value of your change initiatives. The two ends of the strategic architecture continuum indicate whether your strategic architecture is inferred or whether it is exhaustive. As Practitioners, we often tell ourselves that we have not been told what the strategies are or that the business won't engage with us. This is when you need to do your job and go and infer your organization's strategic architecture by talking to Stakeholders, looking at the annual reports, and even visiting the company website. The Enterprise's strategies are not hard to find.

The ideal is the other end of the continuum where the Enterprise is engaged and works with the Architects to develop an exhaustive Enterprise strategic architecture. The commonality with either approach is that the Architect must write it down and ensure it can be used to trace to value. The difference in the two approaches is that if the Enterprise strategic architecture is inferred, then the inferences must be continually tested by validating with Stakeholders during the architectural development.

The EAGC is required to ensure that an audit is performed and that the Enterprise strategic architecture is at the right level of completeness and confidence (see Governance over Architecture Completeness and Confidence on page 31).

Target Architecture

With an approved target architecture, traceability to the objective is available, and trade-off has been performed. Good architecture trade-off explores options, cost, and benefits to arrive at the optimal answer for an organization. Often that answer is a compromise between competing interests.

Target Governance Checklist

The role of Auditor is easily performed with the use of checklists. The checklist ensures that the right level of completeness has been performed by the Architect to get the necessary level of confidence in the architecture.

Here is an example of a target architecture checklist:

- Were the correct Stakeholders identified: Y/N

  If yes, proceed.

  If no, direct the Architect to engage with the Stakeholders appropriate to the scope of the architecture being developed.

- Were constraints and guidance from previous approved architectures (superior architectures) considered: Y/N?

  If yes, proceed.

  If no, direct the Practitioner to perform their job and take into account guidance and constraints from a superior architecture. Where the Practitioner identifies a conflict, obtain a recommendation on whether to grant relief from the superior architecture or enforce the superior architecture. This decision must be made by the superior architecture Stakeholders.

- Do appropriate Subject Matter Experts agree with the facts and interpretation of the facts in the architecture: Y/N?

  If yes, proceed.

  If no, the Practitioner has to do their job and engage with the Subject Matter Experts. Where the Practitioner identifies a conflict with, or between, Subject Matter Experts develop a recommendation for the Stakeholders that they should have limitations in confidence.

- Are all the constraints or guidance produced in the architecture reflected in the views produced for the Stakeholders as well as any underpinning architecture models, reference models, in-flight projects, and organizational landscapes: Y/N?

  If yes, proceed.

  If no, the Practitioner needs to do their job and develop appropriate views that are consistent with a complete architectural perspective.

- Do the Stakeholders understand the value, and any uncertainty in achieving the value, provided by reaching the target state: Y/N?

  If yes, proceed.

  If no, the Practitioner needs to do their job and develop appropriate views, and other work products, then return to the Stakeholders.

- Do the Stakeholders understand the work necessary to reach the target state and any uncertainty (risk) in successfully accomplishing the work: Y/N?

  If yes, proceed.

If no, the Practitioner needs to do their job and develop appropriate work products and return to the Stakeholders.

- Do the Stakeholders understand any limitations in confidence they should have in the target architecture: Y/N?

  If yes, proceed.

  If no, the Practitioner needs to do their job and develop appropriate guidance on the limitations in confidence and return to the Stakeholders.

- Have the Stakeholders approved the views: Y/N?

  If yes, proceed.

  If no, the Practitioner needs to do their job and update the views and return to the Stakeholder for approval.

Governing the Implementation

In its simplest form, implementation governance has two functions to ensure value is being delivered:

- Ensure that the solution design conforms to the approved architecture
- Ensure that the final deliverable conforms to the solution design

As discussed earlier in this paper, that governance is about directing and controlling; however, monitoring of the implementation is required to ensure that value does not get side-tracked. This is where the need for cross-organization information is required; the monitoring function is best left to the Implementer if the necessary information is being fed to the EAGC. Implementation information needs to be reviewed and action taken if required. When a compliance issue is determined during an implementation, the Practitioner must present the compliance issue to the Stakeholder with the recommendation to:

- Provide relief to the Implementer
- Ensure the Implementer conforms to the target
- End the implementation

Implementation monitoring is best performed at multiple stages to capture errors before the cost of correction exceeds potential value realization.

Solution Delivery Notebook

A view of the architecture for the implementation teams is critical to the delivery of value. It provides the guidance, constraints, and architecture specifications for the Implementer to be governed too.

Architecture to support solution delivery is directly aligned with work to implement effective change. In the business cycle, budget control provides ongoing financial control and benefits realization. Architecture to support solution delivery is directly aligned to the governance of the implementation project. Enabling direct association of spend with benefits realization is the contribution to the budget cycle.

Architecture to support solution delivery is dependent on traceability through the EA landscape. The definition of acceptable boundaries for the design and implementation, as well as boundaries for design and delivery, facilitate procurement and third-party contracting.

We are aware of repeated efforts to draw distinctions between "Enterprise Architecture" and "Solution Architecture", which seems to be driven by some attempts to associate EA with big thoughts and big initiatives. We believe in practice it is a distinction that drives no changes in an effective EA team's organization and approach. We treat it as a distinction without a practical difference.

Implementation Governance Checklist

The role of implementation Auditor is easily performed with the use of checklists. The checklist ensures that the right level of completeness has been performed by the Architect to get the necessary level of confidence in the guidance being given to the Implementer.

Here is an example of an implementation checklist:

- Did the change being implemented reasonably interpret the target architecture's guidance and constraints: Y/N?

  If yes, their interpretation should be accepted as compliance and any issues addressed through a change to the architecture. This is a key point. Good architecture can have multiple implementation choices, and the Implementer is not required to adhere to opinion. If the implementation choice is a reasonable interpretation, it should be judged compliant.

  If no, proceed.

- Do appropriate Subject Matter Experts agree with the facts and interpretation of the facts in the impact assessment: Y/N?

  If yes, proceed.

  If no, the Practitioner must do their job and engage with the Subject Matter Experts. Where the Practitioner identifies a conflict with, or between, Subject Matter Experts, develop a report for the Stakeholders identifying what limitations in confidence they should have in the impact assessment.

- Do appropriate Subject Matter Experts agree with the recommendation to enforce the target, grant time-bound relief, or change the architecture: Y/N?

  If yes, proceed.

  If no, the Practitioner must do their job and engage with the Subject Matter Experts. Where the Practitioner identifies a conflict with, or between, Subject Matter Experts, develop a report identifying what limitations in confidence the Stakeholder should have in the compliance recommendation.

- Do the views and other materials produced for the Stakeholders reflect the impact assessment and reflect any underpinning architecture models and analysis: Y/N?

  If yes, proceed to the Stakeholders for approval.

  If no, the Practitioner must do their job.

- Do the Stakeholders understand the impact on prior expected value, and any change in certainty in achieving the value, provided by reaching the target state: Y/N?

  If yes, proceed.

  If no, the Practitioner must do their job and provide the appropriate work products that highlight the impact on expected value, and on uncertainty in reaching the expected value and return to the Stakeholders.

Governing the Architecture Practice

Alignment of EAGC activity on the purpose of the EA Capability, the viewpoint library, and the meta-model provides the foundation of directing and controlling the entire EA Capability. The process integration, required skills, staffing, resources, and engagement all flow from the purpose, viewpoint library, and meta-model.

An EA Capability focused on supporting decision-making for strategy will use a different set of work products than an EA Capability chartered to support implementation governance. This is a critical distinction. The meta-model should be adjusted to align with the charter of the EA Capability. As a rule-of-thumb, the more high-level decision-making the EA Capability supports, the less detail is required in documentation and supporting information. The more it focuses on implementation governance and solution delivery activity, the more detail and consistency are required in supporting documentation and information.

A successful high-functioning EA Capability will maintain a viewpoint library that identifies the questions they are expected to have answers for, and the information they must have to answer these questions. The information they must have, and the information they should have ready access to, in turn define the content meta-model and repository approach.

Minimize the information the EA Capability must maintain and focus on the purpose for which the EA Capability was formed. Address just those key questions. Take comfort in the fact that development of the content meta-model and viewpoint library will feed the evolution of each other. Every component that is added to the Enterprise's content meta-model comes with relationships that must be maintained and comes with attributes that must be tracked.

Consider what minimum information the EA Capability must have at hand, and what information it will need to gather upon demand. The information required at hand is the mandatory minimum. For the other information, ensure that there is a consistent way to gather and relate it to the mandatory minimum. This allows for traceability across more aspects of the Enterprise.

Directing and controlling the Stakeholder classes and what their common concerns are, how to address their concerns, and what information must be known to answer their concerns is the foundation of effectively governing the EA Capability.

Development of architectures is constrained by the list of components, connections between the components, and the component and connection properties. Views created for communication and describing the architecture are constrained by the viewpoint. Hence, governing the viewpoint library and meta-model drives operational and communication consistency across the EA Capability.

Since the viewpoint library and meta-model are key to the description of the EA and changes to it are key to that description and can have widespread serious and significant effects, the viewpoint library and meta-model must be tightly governed to contain those effects.

Architecture Viewpoint Library

Unless the architecture work is understood by a Stakeholder, the architecture work has no value. A library of viewpoint templates is a useful tool and helps the Practitioner. The viewpoint library requires governance to ensure the viewpoint meets its purpose in helping the Stakeholder reach understanding and approval of the architecture.

The viewpoint does not have to be weighty or present a burden to the governor; it only needs to ensure that the Practitioner understands the purpose of the viewpoint and that the viewpoint clearly communicates the architecture in a way the Stakeholder will understand. This governance activity can easily be completed by using a governor's checklist of general viewpoint questions, as the example below illustrates.

A set of viewpoints in a library will not always meet the needs of a Stakeholder in understanding, or the needs of a Practitioner to help explain the architecture to a Stakeholder. It is often required for an *ad hoc* view to be created. In this case, it is important for the Practitioner to be clear on the viewpoint's purpose and to ensure it meets the needs of the purpose. It is always good practice to have the viewpoint peer reviewed prior to presentation to a Stakeholder.

Viewpoint Governance Items

Viewpoints, Templates

General Viewpoint Questions

- What is the purposes of this viewpoint?
- What are the benefits of this viewpoint?
- What are drawbacks of this viewpoint?
- If this viewpoint was not allowed, is there an existing viewpoint in the library?
- Was this viewpoint peer reviewed?
- Could the *ad hoc* viewpoint become a library template?
- Should the *ad hoc* viewpoint become a library template?

Architecture Meta-Model

This is the one area the Auditor may defer to a specialized work group, although the governance board retains overall accountability. The responsibilities should fall to those with the skill set to understand your meta-model. It is also the one area where the architecture governance board has decision rights.

- For each governance item, indicate what operation is proposed: *add*, *remove*, *replace*, or *change*
- As it may be important, indicate the order of the changes: properties, standards, {components, connections}, viewpoints, templates, and then structure in that order
- Identify any required changes to permissions which may include defining a new user group to support the meta-model change

The checklist for the meta-model does not require yes or no answers like the checklists for target architecture governance and implementation governance above. The answers typically require an explanation as to what has been done to satisfy the question. Governing the meta-model is easily performed with the use of processes and checklists. The processes and checklist ensure that the right level of completeness has been performed by the Architect to get the necessary level of confidence that the meta-model changes are necessary to support the delivery of value.

An example of a target architecture question list is given below.

Meta-Model Governance Items

Components, Connections, Properties, Standards, Viewpoints, Templates

General Meta-Model Questions

- What are the purposes of this change?
- What are the benefits of this change?
- What are the drawbacks of this change?
- If this change was not allowed, is there another way to resolve the issue?
- If there are several ways to do what is being proposed, please enumerate these ways

Specific Meta-Model Questions

- Why is this new property needed?
- Are there any similar properties already?
- What is the difference between this and similar properties?
- Why was the name chosen?
- Was it from the standard list?
- If not, why do you need a new one?
- Why was the property type chosen?
- What is its default value and why?
- Are there alternative default values acceptable?
- What concern(s) is (are) being addressed by the viewpoint?
- What Stakeholders currently have those concerns?
- Why is this template named the way it is?
- Why is this template needed?

- Is this template based on a standard or design pattern? If so what is the reference material for it?
- Why were the components and connections chosen as they were?

Governing Value Realization

Governing value realization is the central purpose of TOGAF Phase H (Architecture Change Management). The central element is monitoring expected value and realized value.

Until now we have been talking about delivering value in a very linear fashion; we have a checklist for governing a target architecture that is being developed and an implementation governance for ensuring compliance to the target architecture. The real world, however, is not ever so nicely packaged. A portfolio is delivered over a period of years and it must ensure that value is continually delivered over time, which could be years after the initial implementation.

When a business transformation is delivered, it is expected that the intended values will be delivered. To prove that those values were delivered, at the time of delivery, metrics and the processes to monitor them need to be part of the delivery. A Stakeholder needs to be assigned to review the metrics and agree that the expected value was delivered.

A value that is delivered in the early stages of the portfolio of a business transformation needs to continue to deliver its value as the rest of the portfolio is delivered. If the ongoing value fails to be delivered as intended, the value gap needs to be identified to a Stakeholder and any correction required applied to the portfolio.

A Stakeholder is accountable for monitoring the delivery of value and the ongoing value realization. It is necessary for the Architect to ensure that value realization metrics and the processes to collect and report on them are defined and operationalized. The role of Stakeholder in monitoring is critical for ongoing value to be realized; if a Stakeholder is not identified, that risk needs to be identified to the Stakeholder accountable for the portfolio as a risk to ongoing value delivery.

It is the responsibility of the Stakeholder to determine when the monitoring of value is no longer required. This is an essential function of the Architect to assist the value owner in determining when the value owner needs to change or desist from value monitoring activities. There are several reasons for ceasing monitoring and auditing value realization. The value change is embedded into the organizational DNA and is self-perpetuating. The value delivered was part of a transitional architecture which has since been replaced and other value monitoring processes are in place.

Realizing value over time is easily performed with the use of checklists. The checklist ensures that the right metrics are in place and that metric information is supplied to the appropriate process owners for review.

Implementation Value Realization Governance Checklist

Here is an example of an implementation value checklist:

- Was a value owner (Stakeholder) identified and agreed to perform the role of value owner: Y/N

  If yes, proceed.

  If no, identify the value owner gap and the associated risks to value to the appropriate Stakeholder and ask for:

  — Guidance on who should perform the role and get their support to engage

— An exception and continue without a value owner

- Were value metrics identified and included in the architecture: Y/N

  If yes, proceed.

  If no, continue work on the value metrics until the Stakeholder approves.

- Were value processes developed and included in the architecture: Y/N

  If yes, proceed.

  If no, continue work on the value processes until the Stakeholder approves.

Value Realization Post-Implementation

As governors, once we have established implementation governance and can prove that the intended benefits and value were delivered by the implementation team, value realization is only partially complete.

It is necessary to have in the approved architecture the processes and information that is required to report on the ongoing value realization of a business transformation investment. From the governor's perspective, we need to ensure that the Stakeholders will have what they need to measure and monitor value realization, not only at delivery of a change but ongoing as well. This means that the governor must ensure that the Practitioner has identified the right Stakeholder and has secured the Stakeholder's approval for the value realization processes and information.

For the IT-centric readers, it has been said many times by business and IT management that the business believes IT has not been helpful or that IT poorly delivers on value. It is just as important for the business to see ongoing value as it for IT management to prove they are delivering ongoing value. In the modern world value realization that does not rely on IT functions is rare. A central activity of the EAGC is alignment of IT investment, change, and operations with the complete set of the Enterprise's investments, change, and operations to deliver the value expected.

Ongoing Value Realization Governance Checklist

Here is an example of an ongoing value realization checklist:

- Was an ongoing value realization owner (Stakeholder) identified and did they agree to perform the role of value owner: Y/N

  If yes, proceed.

  If no, identify the ongoing value owner gap and the associated risks to the ongoing value realization to the appropriate Stakeholder and ask for:

  — Guidance on who should perform the role and get their support to engage

  — An exception and continue without a value owner

- Were ongoing value metrics identified and included in the architecture: Y/N

  If yes, proceed.

If no, continue work on the ongoing value metrics until the Stakeholder approves.

- Were ongoing value processes developed and included in the architecture: Y/N

  If yes, proceed.

  If no, continue work on the ongoing value processes until the Stakeholder approves.

Part 3: Guidance on Maintaining Enterprise Architecture Governance

Architecture Governance Maintenance

Your EAGC requires you to have and maintain four governance aspects:

- Architecture Governance Repository
- Value Processes
- Value Information (metrics)
- Governance Reporting

Architecture Governance Repository

An EAGC repository can exist within an architecture tool or an Excel® spreadsheet. It is not important what is used as a repository, but that you have one and that you record the properties about your architecture so that an audit may be performed. An EAGC repository is not for decision logs, meeting minutes, or architecture work-in-progress – these things are for the management of an architecture capability. We view the EAGC repository as where the governance properties of an architecture are recorded and maintained. The key purpose for EAGC properties is to ensure that the required information to ensure value delivery is being preserved.

Value Processes

Every delivery of a business transformation requires that there are processes a Stakeholder or value owner will use to prove that value at the time of the implementation of the transformation, as well as continuing to prove value over time.

The Practitioner is accountable for defining and including in the architecture specification the Stakeholder approved processes.

It then becomes the responsibility of the Implementer to ensure that these processes are developed and delivered with the business transformation.

Value Information (Metrics)

Along with the delivery of the value processes, it is required that the necessary metrics are also developed and delivered. There may be a different set of metrics required at the time of delivery than the set needed for the ongoing measurement of value after implementation.

The Practitioner is accountable for defining and including in the architecture specification the Stakeholder approved metrics.

It then becomes the responsibility of the Implementer to ensure that these metrics are developed and delivered with the business transformation.

Inputs to the Implementation Team

There needs to be a complete view of the architecture that will support a change initiative; one such view can be the Solution Delivery Notebook (SDN). The SDN is a generic term for a living Stakeholder view that can be used to communicate the architecture to a design and delivery team. It is the bridge between architecture, design, and delivery.

An SDN communicates the rationale behind choices made and the constraints defined for the implementation team. A good architecture provides creativity and freedom of choice for the implementation team, but must bound and constrain them when necessary to protect the value and benefits of the change.

Unless the Practitioner communicates all the contents of the required architecture in the SDN, the Practitioner is failing to set the implementation team for success. Without a clear view of the architecture expressed in an SDN, the Practitioner is unknowingly creating undue work burden on themselves during the implementation cycle. Spending their time on day-to-day activities with the implementation team dramatically reduces available time to build the architecture to influence change in the next business cycle. The Practitioner is not preparing for success either.

It is not practical to daisy chain all architecture work packages together. The Practitioner must make choices to initiate concurrent project efforts, thereby introducing some unavoidable cross-project dependencies. It is imperative that the Practitioner lists the dependencies across initiatives and projects, along with triggers and potential directions for mitigation. Before finalizing the funding for the implementation project, the leaders of the implementation project and the Stakeholder should understand these conditions.

The list of details provided by the Practitioner may include conditions that should prompt:

- A review of the schedule and resources of the implementation project
- A review of standards, patterns, and building blocks
- Scheduled or *ad hoc* compliance reviews, confidence, and completeness assessments
- Top-down and bottom-up environment changes, like end-of-life announcements for products employed
- Change to vision of the solution project and development of alternatives, like accept changes to value to be realized or a change in the target architecture (stop the project or modify the project)

A high-functioning organization maintains a repository of building blocks or shared solutions to frequently occurring problems and issues. Though assumed to exist and well defined, relevance of a solution delivery method for the implementation project, balancing speed of delivery of different components, and integration approach should be examined.

It is likely that the solution delivery project may involve more than one supplier. Each supplier can have their own method of development and delivery of solutions. The architecture specification defines criteria to decide whose method should win or should the implementation team take care of the method at all.

The final input that supplements the SDN is approach to trade-off, what value to protect, where sub-optimizations can be performed *in lieu* of future benefits, and how to update the future effort that measures and reports value realization.

By providing all these details, the Practitioner effectively guides the change and protects the value the architecture needs to deliver. It also simplifies the number of items the implementation team needs to reach back to the architecture team for clarity of the architecture.

Inputs from the Implementation Team

Each Enterprise has its own approach to the toolset used for architecture development and solution delivery. The most common integration approach we observed is that implementation teams have to create part of the architecture work products within the solution delivery toolset/ecosystem. The implementation team either asks for onetime investment to improve the integration between the architecture development toolset and the solution delivery toolset, or defines a simple verification stage to ensure nothing was lost in translation.

Governance Reporting

The appropriate level of reporting is what provides the Practitioner and the Auditor with the flexibility to execute on a principles-based framework, which all EA activities are. Governance reporting plays a significant role in building confidence for the decision-maker. Governance reporting is a real opportunity to reap the benefits of the good practice that exists within companies.

Characteristics of good governance reports are:

- Inclusion of a standard set of disclosures – related to events of interest, risks, or challenges faced by the architecture effort and the Enterprise
- Not customizing the needs of a specific Stakeholder or a decision-maker, but addressing the concerns of the Enterprise and inter-dependent efforts
- Dealing with sensitive aspects of the Enterprise context, interests of individuals, and board concerns
- Parameters that drives completeness and confidence
- Looking ahead at issues that could impact the target architecture or achievement of the transition/target state or realizing value

Governance reporting is very similar to sending out an investor relationship letter. It must present facts, be objective, and gain the confidence of the Stakeholders and decision-makers to continue investing in the architecture effort. Set a cadence, keep it short and crisp to maintain interest, and enable attention to what is pertinent and relevant.

References

The following documents are referenced directly in various sections of this White Paper or the concepts discussed in this paper are derived from these works.

(Please note that the links below are good at the time of writing but cannot be guaranteed for the future.)

- Aligning Enterprise, System, and Software Architectures, Ivan Mistrik, Antony Tang, Rami Bahsoon, Judith A. Stafford, Business Science Reference (2013)
- Breaking Down Software Development Roles, an Internet.com Developer eBook, Jupitermedia Corporation (2006)
- ISO 19600:2014: Compliance Management Systems – Guidelines; refer to: www.iso.org/standard/62342.html
- ISO/IEC 38500:2015: Information Technology – Governance of IT for the Organization; refer to: www.iso.org/standard/62816.html
- ISO/IEC/IEEE 42010:2011: Systems and Software Engineering – Architecture Description; refer to: www.iso.org/standard/50508.html
- IT Governance: How Top Performers Manage IT Decision Rights for Superior Results, Peter Weill, Jeanne Ross, Harvard Business School Press, 2004
- IT Savvy: What Top Executives Must Know to Go from Pain to Gain, Peter Weill, Jeanne Ross, Harvard Business School Press, 2009
- Reinventing Your Board: A Step-by-Step Guide to Implementing Policy Governance, John Carver, Miriam Carver, Jossey-Bass, 2006
- The TOGAF® Leader's Guide to Establishing and Evolving an EA Capability, an Open Group Guide (G168), December 2016, published by The Open Group; refer to: www.opengroup.org/bookstore/catalog/g168.htm
- TOGAF® Version 9.1, an Open Group Standard (G116), December 2011, published by The Open Group; refer to: www.opengroup.org/bookstore/catalog/g116.htm
- World-Class EA: A Leader's Approach to Establishing and Evolving an EA Capability, White Paper (W160), January 2016, published by The Open Group; refer to: www.opengroup.org/bookstore/catalog/w160.htm
- World-Class EA: A Practitioners' Approach to Developing Enterprise Architecture Following the TOGAF® ADM, White Paper (W174), February 2017, published by The Open Group; refer to: www.opengroup.org/bookstore/catalog/w174.htm
- World-Class Enterprise Architecture, White Paper (W102), April 2010, published by The Open Group; refer to: www.opengroup.org/bookstore/catalog/w102.htm

About the Authors

Dave Hornford, Conexiam

Dave Hornford is Conexiam's Managing Partner and leads Conexiam's Boston practice. He is the owner of Conexiam's Navigate™ and Navigate Atlases. Dave serves on the board of trustees of The SABSA® Institute. He is the former Chair of The Open Group Architecture Forum and was a key contributor to the TOGAF® 9 standard. Based in North America, he works in a variety of industries including financial services, oil and gas, technology, and capital-intensive industry. Typically, he helps clients develop and execute a roadmap to transform.

Nathan Hornford, Conexiam

Nathan Hornford is a management consultant and ABACUS Certified Architect and Designer. Nathan is based in Canada, and is a key member of Conexiam's Victoria Architecture & Analytics team. Nathan owns Conexiam's Navigate™ toolkit. He specializes in formal modeling and supports all architecture analysis, option analysis, and roadmap development. Nathan works with all Conexiam's practices to provide consistent architecture methods and tools and address the client's change needs.

Sriram Sabesan, Conexiam

Sriram Sabesan is an Open Group Distinguished Certified IT Architect (Open CA). Based in North America, he specializes in technology, manufacturing, telecommunication, and financial services industries. Sriram helps clients to develop and execute strategies in response to digital or economic disruptions. He is actively involved in development of different Open Group standards.

Chris Searle, Conexiam

Chris Searle is an Enterprise Architect. Chris is based in Canada and is a member of Conexiam's Victoria Architecture & Analytics team. He specializes in architecture analysis, option analysis, and roadmap development. Chris works with all Conexiam's practices to provide architecture development and analysis.

Ken Street, Conexiam

Ken Street is a management consultant. Ken is the current Chair of the Big Data project and is active within The Open Group IT4IT™, Architecture, and Open Platform 3.0™ Forums. Based in Canada, he leads Conexiam's Business & IT Alignment, Governance, and Big Data practices. His focus is on helping clients to develop their Enterprise Architecture Governance Capability, improve their Business/IT alignment, and execute architecture-driven change programs.

Samantha Toder, Conexiam

Samantha Toder is a management consultant and ABACUS Certified Architect and Designer. Sam is based in the US. Sam owns Conexiam's Navigate™ method. She helps clients to develop in-house Enterprise Architecture Capability and execute complex transformation programs in the financial services industry.

About The Open Group

The Open Group is a global consortium that enables the achievement of business objectives through IT standards. With more than 500 member organizations, The Open Group has a diverse membership that spans all sectors of the IT community – customers, systems and solutions suppliers, tool vendors, integrators, and consultants, as well as academics and researchers – to:

- Capture, understand, and address current and emerging requirements, establish policies, and share best practices
- Facilitate interoperability, develop consensus, and evolve and integrate specifications and open source technologies
- Operate the industry's premier certification service

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