THE Open GROUP
Making standards work®

Guide to Security Architecture in TOGAF ADM

A White Paper developed by:

The Open Group Security Forum and
Members of The Open Group Architecture Forum

November, 2005

Guide to Security Architecture in TOGAF ADM

Document No.: W055

Published by The Open Group, November, 2005

Any comments relating to the material contained in this document may be submitted to:

The Open Group
44 Montgomery St. #960
San Francisco, CA 94104

or by email to:

ogspecs@opengroup.org

Table of Contents

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

Executive Summary

The Open Group Architecture Framework (TOGAF) is now well-established in the IT Architecture industry. The core of TOGAF is its Architecture Development Method (ADM). Development of TOGAF ADM has matured and extended in its coverage over many years now. The latest published version is TOGAF8.

Coverage of information security considerations in TOGAF ADM has, for several years, been acknowledged as a significant omission. In late 2004, The Open Group Security Forum undertook a collaborative project with members of the Architecture Forum to correct this omission. The result is this White Paper.

The goal of this White Paper is to explain what security considerations need to be addressed in the TOGAF ADM for the guidance of enterprise architects and system designers. Its primary purpose is as input to the Architecture Forum, for integrating security considerations into their development of the next version of TOGAF (designated TOGAF9). We also anticipate it will be of value to system architects and designers who include information security considerations in their designs.

The work in developing this White Paper is based on the existing published TOGAF-8, so the Security Forum and our collaborators from the Architecture Forum appreciate that there may be new considerations arising in TOGAF9 development that we will want to re-visit when TOGAF9 becomes sufficiently stable.

It is significant to note that during this development project, several security-related architecture and design issues arose which the information security experts wanted to include in this White Paper, but which were ruled as out of scope of the coverage of TOGAFADM. Follow-on work will aim at a minimum to capture these issues and make them available from the Security Forum web site (www.opengroup.org/security).

Introduction to Security Architecture Guidance in the ADM

ADM Overview

The TOGAF Architecture Development Method (ADM) is the result of continuous contributions from a large number of architecture practitioners. It describes a method for developing an enterprise architecture, and forms the core of TOGAF. It integrates elements of TOGAF as well as other available architectural assets, to meet the business and information technology needs of an organization.

The TOGAF Architecture Development Method (ADM) forms the core of TOGAF. It is a method for developing an enterprise architecture to meet the business and information technology needs of an organization, utilizing the other elements of TOGAF described in this document, and other architectural assets available to the organization.

Architectural development in the process of the ADM is iterative in nature, in that as the development of the architecture progresses, many areas of concern are revisited but at a finer-grained level of examination. Through the several phases the reader might see topics repeated, or in an earlier phase a topic might be treated at a higher level than the reader might expect. Architecture development methods are also tools in the hands of the practitioner to be used as best fits the practitioner's experience. The guidance included here is intended to help practitioners avoid missing a critical security concern. It is expected that elements included by the authors in specific phases will be modified and shifted according to the practitioner's experience.

This guide is not intended to be a Security Architecture Development Methodology. It is intended for the enterprise architect deploying TOGAF ADM, to inform the enterprise architect of what the security architect will need to carry out their security architecture work. It is also intended as a guide to help the enterprise architect avoid missing a critical security concern.

Discussion of security architecture has the tension of being separate from the remainder of enterprise architecture development and at the same time needing to be fully integrated in it. The focus of the security architect is enforcement of security policies of the enterprise, which at times can be seen as inhibiting advancement of projects undertaken by the enterprise architect and application development team. Security architects spend a good deal of effort proving the negative.

Characteristics of Security Architecture

- Security architecture has its own methods. These methods might be the basis for a discreet security methodology.
- Security architecture composes its own discrete view and viewpoints.
- Security architecture addresses non-normative flows through systems and among applications.
- Security architecture introduces its own normative flows through systems and among applications.
- Security architecture introduces unique, single-purpose components in the design.
- Security architecture calls for its own unique set of skill requirements in the IT architect.

Guidance on Security for the Architecture Domains

Pervasively throughout the architectural domains and in all phases of the architecture development, security concerns of the enterprise need to be accounted for. Security is called out separately because it is infrastructure that is rarely visible to the business function being added to the target architecture to derive value. Its fundamental purpose is to protect the value of the systems and information assets of the enterprise. The nature of security in the enterprise is that it is deemed successful if nothing happens that is visible to the user or other observer, and no damage or losses occur. That is, if the enterprise retains the use and value of its information assets, the goals of security in the enterprise have been met. These assets might be obvious – like the data in a customer records database – or intangible – like not having the company name appear in an article in the news saying that its data systems had been compromised.

While security architecture does have its own single-purpose components, security is experienced as a quality of systems in the architecture. As salt is a separate seasoning, it becomes part of the quality of the dish once it is used.

The security view of the architecture calls out its own unique building blocks, collaborations, and interfaces. These security-unique elements must interface with the business systems in a balanced and cost-effective way, so as to maintain the security policies of the enterprise, but not interfere with system operations and functions. It is least costly and most effective to plan for and implement security-specific function in the target architecture as early as possible in the development cycle to avoid costly retrofit or rework because required building blocks for security were not added or used during systems development and deployment. The approach of the IT architect operating in the security domain is also different from IT architects operating in other architecture domains. The security architect considers not only the normal flow of the application, but also the abnormal flows, failure modes, and ways the systems and applications can be interrupted. Put differently, the IT architect tends to focus mostly on how a system will work while the security architect focuses primarily on how the system might fail.

All groups of stakeholders in the enterprise will have security concerns. These concerns might not be obvious as security-related concerns unless there is special awareness on the part of the IT architect. It is desirable to bring a security architect into the project as early as possible. Throughout the phases of the ADM, guidance will be offered on security-specific information which should be gathered, steps which should be taken, and artifacts which should be created. Architectural decisions related to security, like all others, should be traceable to business and policy decisions, which should derive from a risk analysis. The generally accepted areas of concern for the security architect are:

- Authentication

  The substantiation of the identity of a person or entity related to the system in some way.

- Authorization

  The definition and enforcement of permitted capabilities for a person or entity whose identity has been established.

- Audit

  The ability to provide forensic data attesting that the system was used in accordance with stated security policies.

- Assurance

  The ability to test and prove that the system has the security attributes required to uphold the stated security policies

- Availability

  The ability of the system to function without service interruption or depletion despite abnormal or malicious events.

- Asset Protection

  The protection of information assets from loss or unintended disclosure, and resources from unauthorized and unintended use.

- Administration

  The ability to add and change security policies, add or change how policies are implemented in the system, and add or change the persons or entities related to the system.

- Risk Management

  The organization's attitude and tolerance for risk. (This risk management is different from the special definition found in financial markets and insurance institutions that have formal risk management departments.)

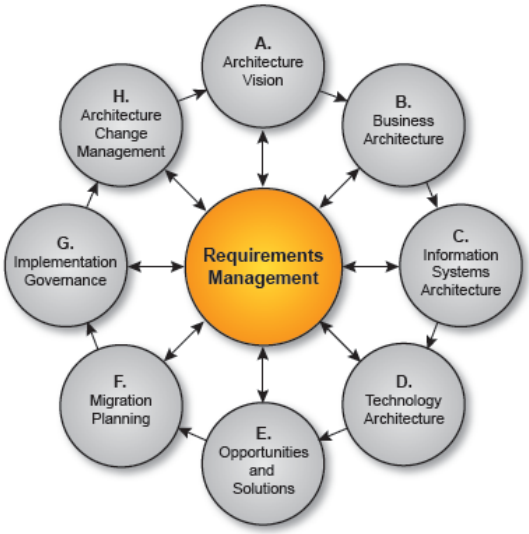
Sample Artifacts

1. Business rules regarding handling of data/information assets
2. Written and published security policy
3. Codified data/information asset ownership and custody
4. Risk analysis documentation
5. Data classification policy documentation

Security Guidance for ADM Architecture Requirements Management

Objective

To define a process whereby requirements for enterprise architecture are identified, stored, and fed into and out of the relevant ADM phases.



Approach

As indicated by the "Requirements Management" circle at the center of the ADM graphic, the ADM is continuously driven by the requirements management process.

It is important to note that the "Requirements Management" circle denotes, not a static set of requirements, but a dynamic process whereby requirements for enterprise architecture and subsequent changes to those requirements are identified, stored, and fed into and out of the relevant ADM phases.

The ability to deal with changes in requirements is crucial. Architecture is an activity that by its very nature deals with uncertainty and change – the "grey area" between what stakeholders aspire to and what can be specified and engineered as a solution. Architecture requirements are therefore invariably subject to change in practice. Moreover, architecture often deals with drivers and constraints, many of which by their very nature are beyond the control of the enterprise (changing market conditions, new legislation, etc.), and which can produce changes in requirements in an unforeseen manner.

Note also that the requirements management process itself does not dispose of, address, or prioritize any requirements: this is done within the relevant phase of the ADM. It is merely the process for managing requirements throughout the overall ADM.

Resources

The world of requirements engineering is rich with emerging recommendations and processes for requirements management. TOGAF does not mandate or recommend any specific process or tool: it simply states what an effective requirements management process should achieve (i.e., the "requirements for requirements", if you like).

- Business Scenarios

  One effective technique that is described in TOGAF itself is business scenarios, which are an appropriate and useful technique to discover and document business requirements, and to articulate an architectural vision that responds to those requirements. Business scenarios are described in detail in TOGAF Part IV: Resource Base, Business Scenarios.

- Volere Requirements Specification Template

  Architecture requirements is very much a niche area within the overall requirements field. One useful resource is the Volere Requirements Specification Template, available from the Volere web site hosted by the Atlantic Systems Guild. While not designed with architecture requirements in mind, this is a very useful requirements template, which is freely available and may be modified or copied (for internal use, provided the copyright is appropriately acknowledged).

  One interesting item in this template is the "waiting room", which is a hold-all for requirements in waiting. There are often requirements identified which, as a result of the prioritization activity that forms part of the requirements management process (see below), are designated as beyond the planned scope, or the time available, for the current iteration of the architecture. The waiting room is a repository of future requirements. Having the ability to store such requirements helps avoid the perception that they are simply being discarded, while at the same time helping to manage expectations about what will be delivered.

- Requirements Tools

  There is a large, and increasing, number of Commercial Off-The-Shelf (COTS) tools available for the support of requirements management, albeit not necessarily designed for architecture requirements. The Volere web site has a very useful list of leading requirements tools.

Security Guidance

The security policy and security standards become part of the enterprise requirements management process. Security policy is established at an executive level of the business, is long-lived, and resistant to whimsical change. Security policy is not tied to any specific technology. Once the security policies are established, they can be referred to as requirements for all architectural projects.

Security standards change more frequently and state technology preferences used to support security policies. New technologies that support the implementation of security policies in a better way can be adopted as needed. The improvements can be in reduced costs or increased benefits. Security standards will manifest themselves as security-related building blocks in the Enterprise Continuum. Security patterns for deploying these security-related building blocks are referred to in the Security Guidance to Phase E.

New security requirements arise from many sources:

1. A new statutory or regulatory mandate
2. A new threat realized or experienced
3. A new IT architectural initiative discovers new stakeholders and/or new requirements

In the case where 1 and 2 above occur, these new requirements would be drivers for input to the change management system discussed in Phase H. A new architectural initiative might be launched to examine the existing infrastructure and applications to determine the extent of changes required to meet the new demands. In the case of 3 above, a new security requirement will enter the requirements management system.

Is our security good?

This question inevitably comes from management to the security architect. No security measures are ever perfect, and the potential exists for the amount of money and effort expended to become very

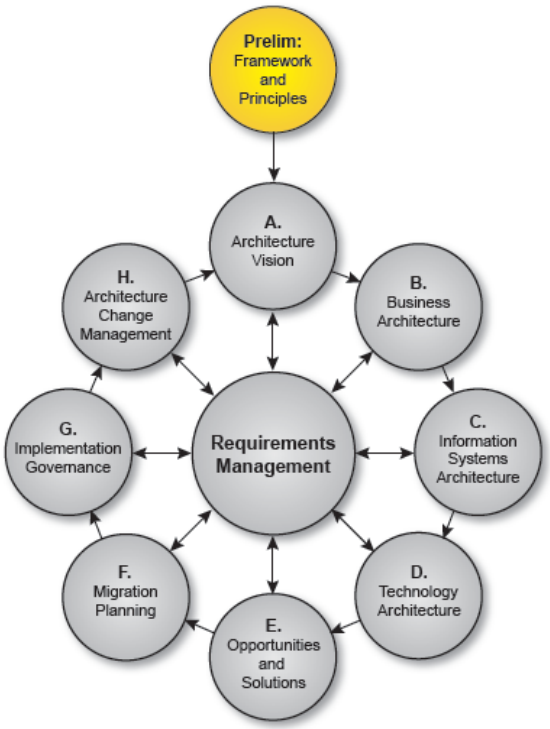
large for little additional return. Security assurance testing should be in place so that the security systems can be measured to ensure that they keep the security policies for which they were designed. Security policy audits should be held and might be mandatory by statute or regulation. These security audits and possible security policy changes are the exact reason why separation of policy enforcement from application code is so strongly emphasized.

Nothing useful can be said about a security measure outside the context of an application, or a system and its environment

The efficacy of a security measure is considered in relation to the risk it mitigates. An enterprise cannot determine how much it will be willing to spend on securing an asset until it understands the asset value. For example, the use of that asset in an application and the concomitant risk the asset is exposed to as a result will determine the true requirements for security. Additionally, the organization's tolerance for risk is a factor. In other words, the question asked should not be: “is it secure?” but rather: “is it secure **enough**?” The latter is ultimately a question to be answered by risk analysis.

Security Guidance for Preliminary Phase: Framework and Principles

Objectives



- To ensure that everyone who will be involved in, or benefit from the architectural process is committed its success
- To define the architecture principles that will inform the constraints on any architecture work
- To define the "architecture footprint" for the organization – the people responsible for performing architecture work, where they are located, and their responsibilities
- To define the scope and assumptions (particularly in a federated architecture environment)
- To define the framework and detailed methodologies that are going to be used to develop enterprise architectures in the organization concerned (typically, an adaptation of the generic ADM)
- To set up and monitor a process (normally including a pilot project) to confirm the fitness-for-purpose of the defined framework
- If necessary, to define a set of criteria for evaluating architecture tools (an example set of criteria is given in TOGAF Part IV: Resource Base), repositories, and repository management processes to be used to capture, publish, and maintain architecture artifacts

Approach

This Preliminary Phase is about defining "how we do architecture" in the enterprise concerned. There are two main aspects: defining the framework to be used; and defining the architecture principles that will inform any architecture work.

The enterprise's approach to re-use of architecture assets is a key part of both the framework definition and architecture principles. (Typically the principles will state the policy on re-use; and the framework will explain how re-use is effected.)

In federated architectures (see TOGAF Part II: ADM, Introduction to the ADM, Enterprise Scope/Focus), requirements from a higher-level architecture are often manifested as "principles" in lower-level architectures.

Security Guidance

Define and document applicable regulatory and security policy requirements

The framework and principles rarely change, and so the security implications called out in the Objectives of this phase should be fairly straightforward. A written security policy for the organization must be in place, and there should be regular notification and education established for employees. ISO/IEC 17799:2005[1] is a good place to start the formation of a security policy, and can be used to assess the security readiness of an organization. Without a written and published security policy, enforcement is difficult. Security policies refer to many aspects of security for the organization – such as physical premises security – that are remotely related to security of systems and applications. The security policy should be examined to find relevant sections, and updated if necessary. Architectural constraints established in the security policy must be communicated to the other members of the architecture team.

In a similar fashion, there may be regulatory requirements that specify obligations the system must fulfill or actions that must be taken. Whether the system will be subject to regulation will depend upon the functionality of the system and the data collected or maintained. In addition, the jurisdiction where the system or service is deployed, where the users reside, or under which the deploying entity is chartered or incorporated will inform this decision. It may be wise to obtain legal counsel regarding these obligations at the outset of activities.

Identify a security architect or security architecture team

Agreement on the role of the security architect in the enterprise architecture process and in the architecture and IT governance should also be established. Security considerations can conflict with functional considerations and a security advocate is required to ensure that all issues are addressed and conflicts of interest do not prevent explicit consideration of difficult issues. Executive policy decisions should be established at this point about what security policies can be negotiable and which policies must be enforced for regulatory or statutory reasons.

Identify first-order assumptions and boundary conditions

If the business model of the organization does encompass federation with other organizations, the extent of the security federation should be established at this point in the process. Contractual federation agreements should be examined for their security implications and agreements. It may be necessary to establish joint architectural meetings with other members of a federation to establish interfaces and protocols for exchange of security information related to federated identity, authentication, and authorization.

Security Inputs

- Written security policy
- Relevant statutes
- List of applicable jurisdictions

Security Outputs

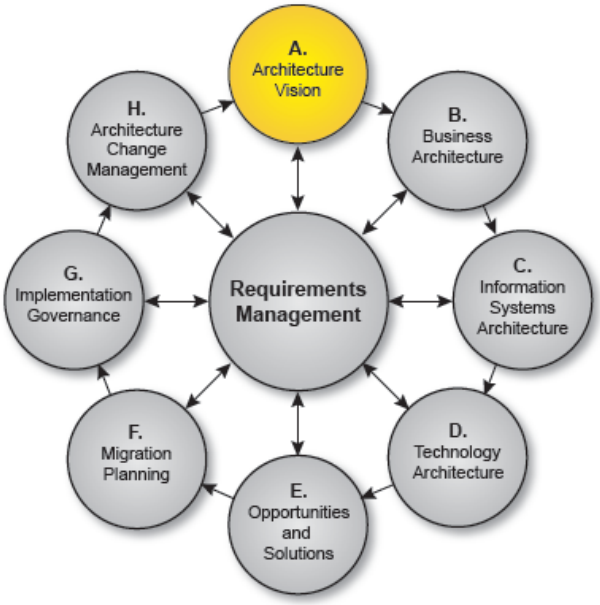
- List of applicable regulations
- List of applicable security policies

[1] ISO/IEC 17799:2005, Information Technology – Security Techniques – Code of Practice for Information Security Management.

- Security team roster
- List of security assumptions and boundary conditions

Security Guidance for Phase A: Architecture Vision

Objectives



- To ensure that this evolution of the architecture development cycle has proper recognition and endorsement from the corporate management of the enterprise, and the support and commitment of the necessary line management
- To validate the business principles, business goals, and strategic business drivers of the organization
- To define the scope of, and to identify and prioritize the components of, the current architecture effort
- To define the relevant stakeholders, and their concerns and objectives
- To define the key business requirements to be addressed in this architecture effort, and the constraints that must be dealt with
- To articulate an architectural vision that demonstrates a response to those requirements and constraints
- To secure formal approval to proceed
- To understand the impact on, and of, other enterprise architecture development cycles ongoing in parallel

Approach

Phase A starts with receipt of a Request for Architecture Work from the sponsoring organization to the architecture organization.

The issues involved in ensuring proper recognition and endorsement from corporate management, and the support and commitment of line management, are discussed in TOGAF Part IV: Resource Base, Architecture Governance.

Phase A also defines what is in and what is outside the scope of the architecture effort and the constraints that must be dealt with. Scoping decisions need to be made on the basis of a practical assessment of resource and competence availability, and the value that can realistically be expected to accrue to the enterprise from the chosen scope of architecture work. The issues involved in this are discussed in TOGAF Part II: ADM, Introduction to the ADM, Scoping the Architecture.

The constraints will normally be informed by the business principles and architecture principles, developed as part of the Preliminary Phase.

Normally, the business principles, business goals, and strategic drivers of the organization are already defined elsewhere in the enterprise. If so, the activity in Phase A is involved with ensuring that existing definitions are current, and clarifying any areas of ambiguity. Otherwise, it involves defining these essential items from scratch.

Similarly, the architecture principles that inform the constraints on architecture work will normally have been defined in the Preliminary Phase. The activity in Phase A is concerned with ensuring that the existing principles definitions are current, and clarifying any areas of ambiguity. Otherwise, it entails defining the architecture principles from scratch, as explained in TOGAF Part IV: Resource Base, Architecture Principles.

Security Guidance

Definition of relevant stakeholders, and discovery of their concerns and objectives will require development of a high-level scenario. Key business requirements will also be established through this early scenario work. The TOGAF ADM business scenario process may be useful here and at later stages.

Obtain management support for security measures

In similar fashion to obtaining management recognition and endorsement for the overall architectural project, so too endorsement of the security-related aspects of the architecture development effort should be obtained. Recognition that the project might have development and infrastructure impact that are not readily visible by looking solely at the systems in question should be made clear. Thorough consideration and mitigation of issues related to risk and security may be perceived as a waste of resources and time; the level of management support must be understood and communicated throughout the team.

Define necessary security-related management sign-off milestones of this architectural development cycle

The traceability of security-related architectural decisions should be documented and the appropriate executives and line management who need to be informed of security-related aspects of the project need to be identified and the frequency of reporting should be established. It should be recognized that the tension between delivery of new business function and enforcement of security policies does exist, and that a process for resolving such disputes that arise should be established early in the project. Such tensions often have the result of putting the security architect seemingly “in the way of completing the project”. It needs to be understood by management and the other architects involved that the role of the security architect is to safeguard the assets of the enterprise.

Determine and document applicable disaster recovery or business continuity plans/requirements

Any existing disaster recovery and business continuity plans must be understood and their relationship with the planned system defined and documented.

Identify and document the anticipated physical | business| regulatory environment(s) in which the system(s) will be deployed

All architecture decisions must be made within the context of the environments within which the system will be placed and operate. Physical environments that should be documented may include battlefield environments, commercial environments, outdoor environments, mobile environments, and

the like. In a similar fashion, the business environment must be defined. Potential business environments may include different assumptions regarding users and interfaces, and those users or interfaces may carry the onus of regulatory environments in which the system must operate (users under the age of thirteen in the US, for example).

Determine and document the criticality of the system: safety-critical | mission-critical | non-critical

Safety-critical systems place lives in danger in case of failure or malfunction. Mission-critical systems place money, market share, or capital at risk in case of failure. Non-critical systems have little or no consequence in case of failure.

Security Inputs

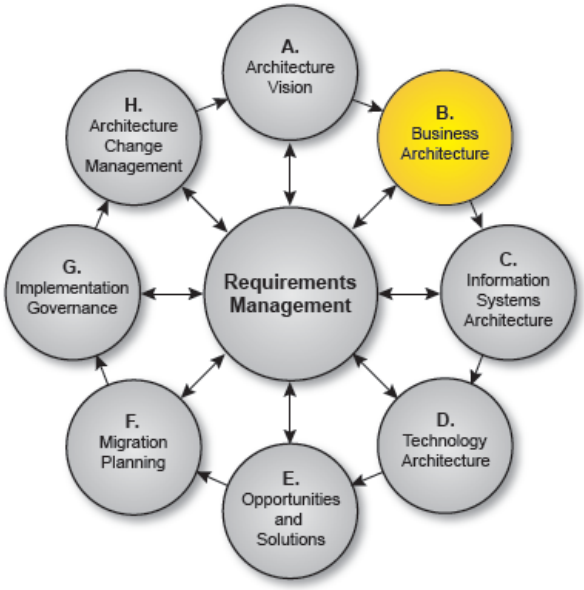
- List of applicable security policies
- List of applicable jurisdictions
- Complete disaster recovery and business continuity plans

Security Outputs

- Physical security environment statement
- Business security environment statement
- Regulatory environment statement
- Security policy cover letter signed by CEO or delegate
- List of architecture development checkpoints for security sign-off
- List of applicable disaster recovery and business continuity plans
- Systems criticality statement

Security Guidance for Phase B: Business Architecture

Objectives



- To describe the current Baseline Business Architecture
- To develop a Target Business Architecture, describing the product and/or service strategy, and the organizational, functional, process, information, and geographic aspects of the business environment, based on the business principles, business goals, and strategic drivers
- To analyze the gaps between the Baseline and Target Business Architectures
- To select the relevant architectural viewpoints that will enable the architect to demonstrate how the stakeholder concerns are addressed in the Business Architecture
- To select the relevant tools and techniques to be used in association with the selected viewpoints

Approach

A knowledge of the Business Architecture is a prerequisite for architecture work in any other domain (Data, Applications, Technology), and is therefore the first architecture activity that needs to be undertaken, if not catered for already in other organizational processes (enterprise planning, strategic business planning, business process re-engineering, etc.).

In practical terms, the Business Architecture is also often necessary as a means of demonstrating the business value of subsequent Technology Architecture work to key stakeholders, and the return on investment to those stakeholders from supporting and participating in the subsequent work.

The extent of the work in Phase B will depend to a large extent on the enterprise environment. In some cases, key elements of the Business Architecture may be done in other activities; for example, the enterprise mission, vision, strategy, and goals may be documented as part of some wider business strategy or enterprise planning activity that has its own lifecycle within the enterprise.

In such cases, there may be a need to verify and update the currently documented business strategy and plans, and/or to bridge between high-level business drivers, business strategy, and goals on the one hand, and the specific business requirements that are relevant to this architecture development effort. (The business strategy typically defines what to achieve – the goals and drivers, and the metrics for success – but not how to get there. That is the role of the Business Architecture.)

In other cases, little or no Business Architecture work may have been done to date. In such cases, there will be a need for the architecture team to research, verify, and gain buy-in to the key business

objectives and processes that the architecture is to support. This may be done as a free-standing exercise, either preceding architecture development, or as part of Phase A.

In both of these cases, the business scenario technique of the TOGAF ADM, or any other method that illuminates the key business requirements and indicates the implied technical requirements for IT architecture, may be used.

A key objective is to re-use existing material as much as possible. In architecturally more mature environments, there will be existing architecture definitions, which (hopefully) will have been maintained since the last architecture development cycle. Where existing architectural descriptions exist, these can be used as a starting point, and verified and updated if necessary (see TOGAF Part III: Enterprise Continuum, The Architecture Continuum).

Gather and analyze only that information that allows informed decisions to be made relevant to the scope of this architecture effort. If this effort is focused on the definition of (possibly new) business processes, then Phase B will necessarily involve a lot of detailed work. If the focus is more on the Target Architectures in other domains (data/information, application systems, infrastructure) to support an essentially existing Business Architecture, then it is important to build a complete picture in Phase B without going into unnecessary detail.

Security Guidance

Determine who are the legitimate actors who will interact with the product/service/process

Development of the business scenarios and subsequent high-level use cases of the project in concern will bring to attention the people actors and system actors involved. Many subsequent decisions regarding authorization will rely upon a strong understanding of the intended users, administrators, and operators of the system, in addition to their expected capabilities and characteristics. It must be borne in mind that users may not be humans; software applications may be legitimate users. Those tending to administrative needs, such as backup operators, must also be identified, as must users outside boundaries of trust, such as Internet-based customers.

Assess and baseline current security-specific business processes. (enhancement of existing objective)

The business process regarding how actors are vetted as proper users of the system should be documented. Consideration should also be made for actors from outside the organization who are proper users of the system. The outside entities will be determined from the high-level scenarios developed as part of Phase A.

Determine whom/how much it is acceptable to inconvenience in utilizing security measures

Security measures, while important, can impose burden on users and administrative personnel. Some will respond to that burden by finding ways to circumvent the measures. Examples include administrators finding ways to create “back doors” or customers choosing a competitor to avoid the perceived burden of your infrastructure. The trade-offs can require balancing security advantages against business advantages and demand informed judicious choice.

Identify and document interconnecting systems beyond project control

Every cybernetic or business system must rely upon existing systems beyond the control of your project. These systems possess advantages and disadvantages, risks and benefits. Examples include the Domain Name System (DNS) that resolves computer and service names to Internet addresses, or paper currency issued by the local treasury. The address returned by the host or service DNS may not always be trustworthy; paper currency may not always be genuine, and recourse will vary in efficacy between jurisdictions. These interfaces must be understood and documented.

Determine the assets at risk if something goes wrong – "What are we trying to protect?"

Assets are not always tangible and are not always easy to quantify. Examples include: loss of life, loss of customer good will, loss of a AAA bond rating, loss of market share.

Determine the cost (both qualitative and quantitative) of asset loss/impact in failure cases

It must be remembered that those assets most challenging to quantify can be the most valuable and must not be neglected. Even qualitative estimates will prove valuable in assessing comparative risks.

Identify and document the ownership of assets

Assets may be owned by outside entities, or by inside entities. Inside entities may be owned by individuals or by organizations.

Determine:

1. Where trust is assumed
2. How it is established
3. How it is communicated

Always trace it to the real world; i.e.:

- Assessment (credit searches, personal vouching)
- Liability (monetary damages, jail terms, sanctions)

All security decisions rely upon trust that has been established in some fashion. No trust assumptions have any value if they cannot be rooted in real-world assessment and liability. In most business environments, trust is established through contracts that define liability where the trust is breached. The onus for assessing trust is the responsibility of those choosing to enter into the contracts and their legal counsel. It is important to note that technology (e.g., digital certificates, SAML, etc.) cannot create trust, but can only convey in the electronic world the trust that already exists in the real world through business relationships, legal agreements, and security policy consistencies.

Determine and document appropriate security forensic processes

To be able to enforce security policies, breaches of security need to be properly captured so that problem determination and possible policy or legal action can be taken against the entity causing the breach. Forensic practices suitable to provide evidence where necessary need to be established and documented. Security personnel should be trained to follow the forensic procedures and training material regarding the need to collect evidence should be considered for the standard security education given to employees.

Identify the criticality of the availability and correct operation of the overall service

The risks associated with loss of availability may have already been adequately considered in the foregoing mission-critical/safety-critical assessment.

Determine and document how much security (cost) is justified by the threats and the value of the assets at risk

A risk analysis (an understanding of the value of assets at risk and the likelihood of potential threats) provides an important guideline for investments in mitigation strategies for the identified threats.

Reassess and confirm Architecture Vision decisions

Business analysis involves a number of rigorous thought exercises and may call into question the initial assumptions identified in the Architechture Vision.

Assess alignment or conflict of identified security policies with business goals

The security policies identified in the Preliminary Phase may have provisions that are difficult or impossible to reconcile with the business goals in light of the identified risks. Possible responses include alteration of aspects of the business environment, modification of the intended user population, or technical mitigation of risks (addressed in Phase C).

Determine "what can go wrong?"

Perform a threat analysis that identifies the high-level threats bearing upon the system and their likelihood.

Security Inputs

- Initial business and regulatory security environment statements
- List of applicable disaster recovery and business continuity plans
- List of applicable security policies and regulations

Security Outputs

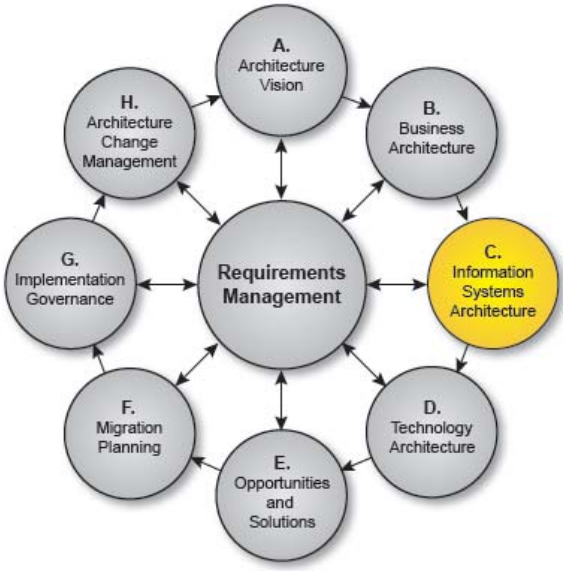
- List of forensic processes
- List of new disaster recovery and business continuity requirements
- Validated business and regulatory environment statements
- List of validated security policies and regulations
- List of target security processes
- List of baseline security processes
- List of security actors
- List of interconnecting systems
- Statement of security tolerance for each class of security actor
- Asset list with values and owners
- List of trust paths
- Availability impact statement(s)
- Threat analysis matrix

References

- NIST 800-18 Guide for Developing Security Plans for Information Technology Systems
- NIST 800-30 Risk Management Guide for Information Technology Systems

Security Guidance for Phase C: Information System Architectures

Objective



To develop Target Architectures covering either or both (depending on project scope) of the Data and Application Systems domains.

The scope of the business processes supported in Phase C is limited to those that are supported by IT, and the interfaces of those IT-related processes to non-IT-related processes.

Approach

Development

Phase C involves some combination of Data and Applications Architecture, in either order. Advocates exist for both sequences. For example, Spewak's Enterprise Architecture Planning (EAP) recommends a data-driven approach.

On the other hand, major applications systems – such as those for Enterprise Resource Planning (ERP), customer relationship management, etc. – often provide a combination of technology infrastructure and business application logic, and some organizations take an application-driven approach, whereby they recognize certain key applications as forming the core underpinning of the mission-critical business processes, and take the implementation and integration of those core applications as the primary focus of architecture effort (the integration issues often constituting a major challenge).

Implementation

Implementation of these architectures may not necessarily follow the same order. For example, one common implementation approach is top-down design and bottom-up implementation:

- Design:
  - Business Architecture design
  - Data (or Applications) Architecture design
  - Applications (or Data) Architecture design
  - Technology Architecture design
- Implementation:
  - Technology Architecture implementation
  - Applications (or Data) Architecture implementation
  - Data (or Applications) Architecture implementation
  - Business Architecture implementation

An alternative approach is a data-driven sequence, whereby application systems that create data are implemented first, then applications that process the data, and finally applications that archive data.

Security Guidance

Assess and baseline current security-specific architectural elements. (enhancement of existing objective)

A full inventory of architectural elements that implement security services must be compiled in preparation for a gap analysis.

Identify safe default actions and failure states

Every state change in any system is precipitated by some trigger. Commonly, an enumerated set of expected values of that trigger initiates a change in state. However, there are likely other potential trigger inputs that must be accommodated in non-normative cases. Additionally, system failure may take place at any point in time. Safe default actions and failure modes must be defined for the system informed by the current state, business environment, applicable policies, and regulatory obligations. Safe default modes for an automobile at zero velocity may no longer be applicable at speed. Safe failure states for medical devices will differ markedly from safe failure states for consumer electronics.

Identify and evaluate applicable recognized guidelines and standards

Standards are justly credited for reducing cost, enhancing interoperability, and leveraging innovation. From a security standpoint, standard protocols, standard object libraries, and standard implementations that have been scrutinized by experts in their fields help to ensure that errors do not find their way into implementations. From a security standpoint, errors are security vulnerabilities.

Revisit assumptions regarding interconnecting systems beyond project control

In light of the risk assessments performed, assumptions regarding interconnecting systems may require modification.

Determine and document the sensitivity or classification level of information stored/created/used

Information stored, created, or manipulated by the system may or may not be subject to an official classification that defines its sensitivity and the obligations to which the system and its owners are subject. The absence of any official classification does not necessarily absolve the onus on maintaining the confidentiality of data. Consideration must be made for different legislative burden that may hold jurisdiction over the system and the data stored.

Identify and document custody of assets

All assets of value are kept and maintained on behalf of the owner. The specific persons or organizations charged with this responsibility must be identified.

Identify the criticality of the availability and correct operation of each function

Presumably, in the event of system failure or loss of functionality, some value is lost to stakeholders. The cost of this opportunity loss should be quantified, if possible, and documented.

Determine the relationship of the system under design with existing business disaster/continuity plans

Existing business disaster/continuity plans may accommodate the system under consideration. If not, some analysis is called for to determine the gap and the cost if that gap goes unfilled.

Identify what aspects of the system must be configurable to reflect changes in policy | business environment | access control

No environment is static and systems must evolve to accommodate change. Systems architected for ready reconfiguration will better reflect that change and result in lower cost over the life of the system. Security is enhanced when security-related changes can be implemented inexpensively and are, hence, not sidelined. Security is also enhanced when changes require no changes to code; changes to code introduce bugs and bugs introduce security vulnerabilities.

Identify lifespan of information used as defined by business needs and regulatory requirements

Information maintained beyond its useful lifespan represents wasted resources and, potentially, business decisions based upon suboptimal data. Regulation, however, sometimes mandates the timetable for maintenance of information as archival data.

Determine approaches to address identified risks:

- Mitigate
- Accept
- Transfer
- Avoid

There are several standard ways to address identified and quantified risk. The list above is not intended to be exhaustive for all approaches.

Identify actions/events that warrant logging for later review or triggering forensic processes

Anomalous actions and states will outnumber planned actions and states. These transitions will warrant logging to reconstruct chains of events, facilitate root cause analysis, and, potentially, establish evidence for civil or criminal action. It must be borne in mind that logs must be regularly reviewed to be introduced as evidence into a court of law in some jurisdictions.

Identify and document requirements for rigor in proving accuracy of logged events (non-repudiation)

Since malicious tampering of systems is commonly accompanied by tampering of logged data to thwart investigation and apprehension. The ability to protect and establish the veracity of logs through cryptographic methods will remove uncertainty from investigations and bolster cases in legal proceedings.

Identify potential/likely avenues of attack

Thinking like an adversary will prepare the architect for creation of a robust system that resists malicious tampering and, providentially, malfunction arising from random error.

Determine "what can go wrong?"

Security Inputs

- Threat analysis matrix
- Risk analysis
- Documented forensic processes
- Validated business policies and regulations
- List of interconnecting systems
- New disaster recovery and business continuity requirements

Security Outputs

- Event log-level matrix and requirements
- Risk management strategy
- Data lifecycle definitions
- List of configurable system elements
- Baseline list of security-related elements of the system
- New or augmented security-related elements of the system
- Security use case models:
  - Normative models
  - Non-normative models
- List of applicable security standards:
  - Protocols
  - Object libraries
  - Others …
- Validated interconnected system list
- Information classification report
- List of asset custodians
- Function criticality statement
- Revised disaster recovery and business continuity plans
- Refined threat analysis matrix

References

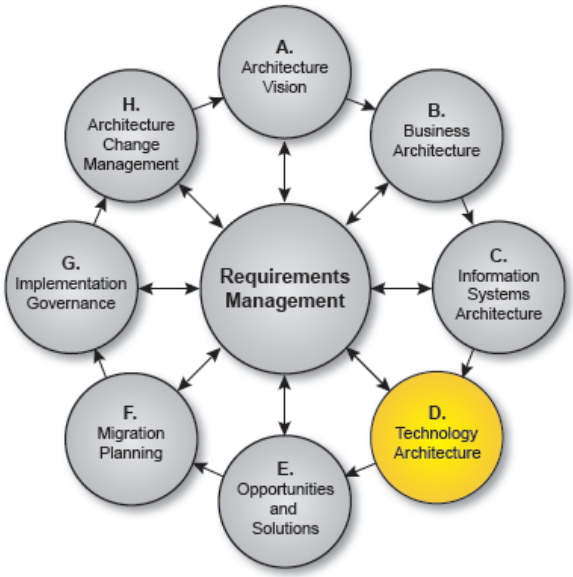
- NIST 800-18 Guide for Developing Security Plans for Information Technology Systems

Security Guidance for Phase D: Technology Architecture

Objective

To develop a Technology Architecture that will form the basis of the following implementation work.

For the security architect, this iteration is for creating the technical detail specifications for the systems that will carry out the security requirements defined and refined in the previous phases.



Approach

Detailed guidelines for Phase D including Inputs, Steps, and Outputs, are given in TOGAF Part II: Architecture Development Method (ADM), Phase D – Technology Architecture Detail.

Architecture Continuum

As part of Phase D, the architecture team will need to consider what relevant Technology Architecture resources are available in the Architecture Continuum.

Security Guidance

Assess and baseline current security-specific technologies (enhancement of existing objective)

Revisit assumptions regarding interconnecting systems beyond project control

Identify and evaluate applicable recognized guidelines and standards

Identify methods to regulate consumption of resources

Every system will rely upon resources that may be depleted in cases that may or may not be anticipated at the point of system design. Examples include network bandwidth, battery power, disk space, available memory, and so on. As resources are utilized approaching depletion, functionality may be impaired or may fail altogether. Design steps that identify non-renewable resources, methods that can recognize resource depletion, and measures that can respond through limiting the causative factors, or through limiting the effects of resource depletion to non-critical functionality, can enhance the overall reliability and availability of the system.

Engineer a method by which the efficacy of security measures will be measured and communicated on an ongoing basis

As systems are deployed and operated in dynamic environments, security measures will perform to varying degrees of efficacy as unexpected threats arise and as expected threats change in the environment. A method that facilitates ongoing evaluation of the value of security measures will inform ongoing changes to the system in response to changing user needs, threat patterns, and problems found.

Identify the trust (clearance) level of:

- All users of the system
- All administrators of the system
- All interconnecting systems beyond project control

Regulatory requirements, information classification levels, and business needs of the asset owners will all influence the required level of trust that all interactive entities will be required to fulfill to qualify for access to data or services.

Identify minimal privileges required for any entity to achieve a technical or business objective

Granting sweeping capabilities to any user, application, or other entity can simplify successful transation completion at the cost of complicating or precluding effective control and audit. Many regulatory obligations are more challenging to demonstrate compliance where privileges are sweeping and controls are loose.

Identify mitigating security measures, where justified by risk assessment

This objective is where the classic security services of identification, authentication, authorization, data confidentiality, data integrity, non-repudiation, assurance, and audit are brought into play, after their applicability is determined and the cost/value of protection has been identified.

Determine “what can go wrong?”

Security Inputs

- List of security-related elements of the system
- List of interconnected systems
- List of applicable security standards
- List of security actors
- Risk management strategy
- Validated security policies
- Validated regulatory requirements
- Validated business policies related to trust requirements

Security Outputs

- Baseline list of security technologies
- Validated interconnected systems list

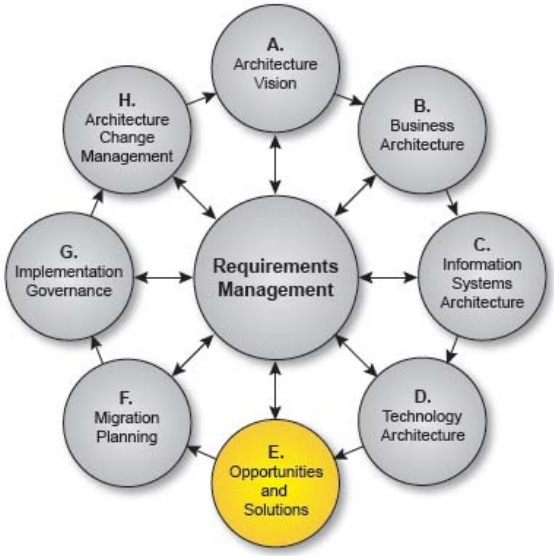
- Selected security standards list
- Resource conservation plan
- Security metrics and monitoring plan
- User authorization policies
- Risk management plan
- User trust (clearance) requirements

References

- NIST 800-18 Guide for Developing Security Plans for Information Technology Systems
- NIST 800-27 Engineering Principles for Information Technology Security (A Baseline for Achieving Security)

Security Guidance for Phase E: Opportunities and Solutions

Objective



- To evaluate and select among the implementation options identified in the development of the various Target Architectures (for example, build *versus* buy *versus* re-use options, and sub-options within those major options)
- To identify the strategic parameters for change, and the top-level work packages or projects to be undertaken in moving from the current environment to the target
- To assess the dependencies, costs, and benefits of the various projects
- To generate an overall implementation and migration strategy and a detailed Implementation Plan

Approach

Phase E identifies the parameters of change, the major phases along the way, and the top-level projects to be undertaken in moving from the current environment to the target. The output of Phase E will form the basis of the Implementation Plan required to move to the Target Architecture. This phase also attempts to identify new business opportunities arising from the architecture work in previous phases.

Sometimes the process of identifying implementation opportunities allows a business to identify new applications, and in this case it may be necessary to iterate between Phase E and previous phases. Iteration must be limited by time or money to avoid wasting effort in the search for a perfect architecture.

Phase E is the first phase which is directly concerned with implementation. The task is to identify the major work packages or projects to be undertaken.

An effective way to do this is to use the gap analysis on the business functions between the old environment and the new, created in Phase D. Any functions appearing as "new" items will have to be implemented (developed or purchased and deployed).

Slightly harder to identify are the projects required to update or replace existing functions which must be done differently in the new environment. One of the options to be considered here is leaving an existing system in place and coexisting with the new environment.

During this final step in the specification of building blocks it must be verified that the organization-specific requirements will be met. Key to this is reason checking against the business scenario driving the scope of the project. It is important to note that the ensuing development process must include

recognition of dependencies and boundaries for functions and should take account of what products are available in the marketplace. An example of how this might be expressed can be seen in TOGAF Part IV: Resource Base, Building Blocks, Building Blocks Example.

Coexistence appears on the surface to be easy. After all, the original system is left in place, largely unchanged. Unfortunately, it is not always as easy as it looks. The main problems with coexistence are:

- **User interfaces**: Combining user interfaces to the old and new applications in a single unit on the users' desks can be difficult, if not impossible.
- **Access to data**: Often the new applications need to share some data with the old applications, and some kind of data sharing must be established. This can be difficult unless the old and new systems use the same database technology.
- **Connectivity**: This may involve expenditure on software and gateway equipment. In difficult cases, equipment simply may not be available in a useful timescale. Often this happens because the old system is simply too out-of-date for connectivity solutions to be still on the market.

The most successful strategy for Phase E is to focus on projects that will deliver short-term pay-offs and so create an impetus for proceeding with longer-term projects.

Security Guidance

Identify existing security services available for re-use

From the Baseline Security Architecture and the Enterprise Continuum, there will be existing security infrastructure and security building blocks that can be applied to the requirements derived from this architecture development engagement. For example, if the requirement exists for application access control external to an application being developed, and such a system already exists, it can be used again. Statutory or regulatory requirements may call for physical separation of domains which may eliminate the ability to re-use existing infrastructure. Known products, tools, building blocks, and patterns can be used, though newly implemented.

Engineer mitigation measures addressing identified risks

Having determined the risks amenable to mitigation and evaluated the appropriate investment in that mitigation as it relates to the assets at risk, those mitigation measures must be designed, implemented, deployed, and/or operated.

Evaluate tested and re-usable security software and security system resources

Since design, code, and configuration errors are the roots of many security vulnerabilities, taking advantage of any problem solutions already engineered, reviewed, tested, and field-proven will reduce security exposure and enhance reliability.

Identify new code | resources | assets that are appropriate for re-use

Having successfully engineered new solutions in the absence of existing re-usable solutions, it is appropriate to evaluate those new solutions for inclusion into any existing libraries, archives, or other repositories for future re-use.

Determine "what can go wrong?"

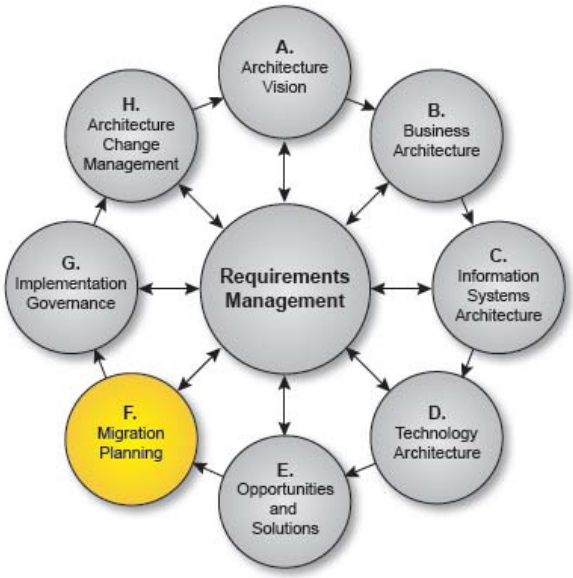
References

- NIST 800-18 Guide for Developing Security Plans for Information Technology Systems

Security Guidance for Phase F: Migration Planning

Objective

To sort the various implementation projects into priority order. Activities include assessing the dependencies, costs, and benefits of the various migration projects. The prioritized list of projects will go on to form the basis of the detailed Implementation Plan and Migration Plan.



Approach

There are some important questions to be asked before embarking on a migration exercise:

- What are the implications of this project on other projects and activities?
- What are the dependencies between this project and other projects and activities?
- What products are needed?
- What components must be developed?
- Does the organization have the resources needed to develop such components?
- What standards are the products or components built on?
- When will they be available?
- Will the products stand the test of time, both because of the technology they use and also because of the viability of the supplier?
- What is the cost of retraining the users?
- What is the likely cultural impact on the user community, and how can it be controlled?
- What is the total cost of the migration, and what benefits will it deliver? It is important to look at actual benefits, and not presumed benefits. Is the funding available?
- Is the migration viable?

Many things affect the answers to these questions, including the current and future architectures, the size of the organization and its complexity, and the value of technology to the core functions of the organization. Other things to consider are the asset value of the current systems, and the level of risk associated with changing the solution and/or the supplier.

Most organizations find that a change of architecture has too much impact on the organization to be undertaken in a single phase. Migration often requires consideration of a number of technical issues, not the least of which are those associated with the means of introducing change to operational systems.

Issues requiring special consideration may include:

- Parallel operations
- Choices of proceeding with phased migration by subsystem or by function
- The impact of geographical separation on migration

The decisions resulting from these considerations should be incorporated in the Implementation Plan.

There are a number of strategies for developing the Implementation Plan and Migration Plan.

The most successful basic strategy is to focus on projects that will deliver short-term pay-offs and so create an impetus for proceeding with longer-term projects.

One common approach is to implement business functions in a data-driven chronological sequence: i.e., create the applications and supporting technology that create data before those that process the data, before those that simply store, archive, or delete data.

For example, the following detailed description of this approach is taken from SPE 68794, Implementing Enterprise Architecture – Putting Quality Information in the Hands of Oil and Gas Knowledge Workers.[2]

1. Determine the future disposition of current systems. Each current system is classified as:
    - **Mainstream systems** – part of the future information system.
    - **Contain systems** – expected to be replaced or modified in the planning horizon (next three years).
    - **Replace systems** – to be replaced in the planning horizon.

      The current system disposition decisions should be made by business people, not IT people.
2. Applications should be combined or split into parts to facilitate sequencing and implementation. This rearrangement of applications creates a number of projects, a project being equivalent to an application or to combinations or parts of applications.
3. Develop the data sequence for the projects as described in the Data Architecture. Using the CRUD (Create/Read/Update/Delete) matrix developed as part of the Data Architecture, sequence the projects such that projects that create data precede projects that read or update that data.
4. Develop an estimated value to the business for each project. To do this, first develop a matrix based on a value index dimension and a risk index dimension. The value index includes the following criteria: principles compliance, which includes financial contribution, strategic alignment, and competitive position. The risk index includes the following criteria: size and complexity, technology, organizational capacity, and impact of a failure. Each of the criteria has an individual weight. The index and its criteria and weighting are developed and approved by senior management early in the project. It is important to establish the decision-making criteria before the options are known.

In addition, there will be key business drivers to be addressed that will also tend to dictate the sequence of implementation, such as:

- Reduction of costs
- Consolidation of services

[2] G.A. Cox, R.M. Johnston, SPE, & R.M. Palermo, Aera Energy LLC, Copyright 2001, Society of Petroleum Engineers Inc.

- Ability to handle change
- A goal to have a minimum of "interim" solutions (they often become long-term/strategic!)

Another, possibly complementary, approach is for the individual projects or work packages to be group-sorted into a series of plateaux, each of which can be achieved in a realistic time scale.

Security Guidance

Assess the impact of new security measures upon other new components or existing leveraged systems

In a phased implementation the new security components are usually part of the infrastructure in which the new system is implemented. The security infrastructure needs to be in a first or early phase to properly support the project.

Implement assurance methods by which the efficacy of security measures will be measured and communicated on an ongoing basis

During the operational phases, mechanisms are utilized to monitor the performance of many aspects of the system. Its security and availability are no exception.

Identify correct secure installation parameters, initial conditions, and configurations

Security of any system depends not on design and implementation alone, but also upon installation and operational state. These conditions must be defined and monitored not just at deployment, but also throughout operation.

Implement disaster recovery and business continuity plans or modifications

Determine "what can go wrong?"

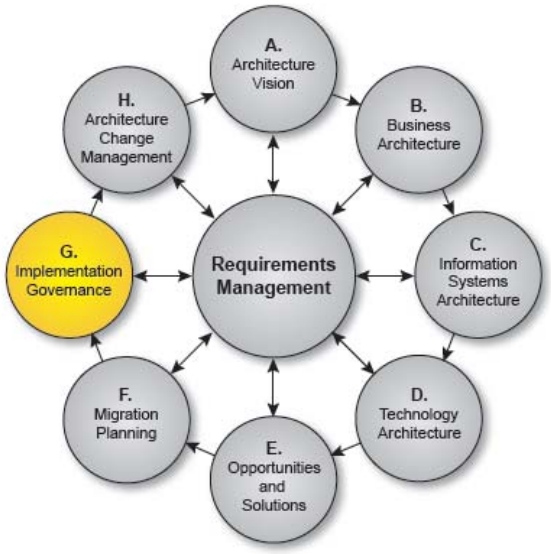
References

- NIST 800-18 Guide for Developing Security Plans for Information Technology Systems

Security Guidance for Phase G: Implementation Governance

Objective

- To formulate recommendations for each implementation project
- To construct an Architecture Contract to govern the overall implementation and deployment process
- To perform appropriate governance functions while the system is being implemented and deployed
- To ensure conformance with the defined architecture by implementation projects and other projects



Approach

It is here that all the information for successful management of the various implementation projects is brought together. Note that in parallel with Phase G there is the execution of an organizational-specific development process, where the actual development happens.

Phase G establishes the connection between architecture and implementation organization, through the Architecture Contract.

Project details are developed, including:

- Name, description, and objectives
- Scope, deliverables, and constraints
- Measures of effectiveness
- Acceptance criteria
- Risks and issues

Implementation governance is closely allied to overall Architecture Governance, which is discussed in TOGAF Part IV: Resource Base, Architecture Governance.

A key aspect of Phase G is ensuring compliance with the defined architecture(s), not only by the implementation projects, but also by other ongoing projects within the enterprise. The considerations involved with this are explained in detail in TOGAF Part IV: Resource Base, Architecture Compliance.

Security Guidance

Establish architecture artifact, design, and code reviews and define acceptance criteria for the successful implementation of the findings

Many security vulnerabilities originate as design or code errors and the simplest and least expensive method to locate and find such errors is generally an early review by experienced peers in the craft. Locating such errors, of course, is the first step and implementing corrections at an appropriate point in the development lifecycle is necessary to benefit from the investment. Follow-on inspections or formalized acceptance reviews may be warranted in high-assurance or safety-critical environments.

Implement methods and procedures to review evidence produced by the system that reflects operational stability and adherance to security policies

While planning and specification is necessary for all aspects of a successful enterprise, they are insufficient in the absence of testing and audit to ensure adherance to that planning and specification in both deployment and operation. Among the methods to be exercised are:

- Review system configurations with security impact which can be modified to ensure configuration changes have not compromised security design
- Audit the design, deployment, and operations against security policies
- Audit the design, deployment, and operations against business objectives
- Run test cases against systems to ensure the security systems have been implemented as designed
- Run disaster recovery tests
- Run business continuity tests

Implement necessary training to ensure correct deployment, configuration, and operations of security-relevant subsystems and components; ensure awareness training of all users and non-privileged operators of the system and/or its components

Training is not necessary simply to preclude vulnerabilities introduced through operations and configuration error, though this is critical to correct ongoing secure performance. In many jusrisdictions, proper training must be performed and documented to demonstrate due diligence and substantiate corrective actions or sanctions in cases where exploits or error compromise business objectives or to absolve contributory responsibility for events that bring about harm or injury.

Determine “what has gone wrong?”

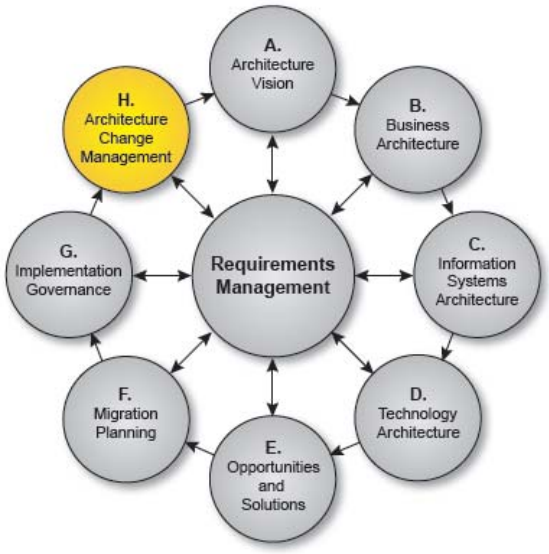
The very purpose of governance is the establishment of a feedback loop that determines the efficacy of plan execution and implements corrections, where required. It must be borne in mind that the imperfections in plans executed are rooted both in human processes and cybernetic processes.

Security Guidance for Phase H: Architecture Change Management

Objective

To establish an architecture change management process for the new enterprise architecture baseline that is achieved with completion of Phase G. This process will typically provide for the continual monitoring of such things as new developments in technology and changes in the business environment, and for determining whether to formally initiate a new architecture evolution cycle.

Phase H also provides for changes to the framework and principles set up in the Preliminary Phase.



Approach

The goal of an architecture change management process is to ensure that changes to the architecture are managed in a cohesive and architected way, and to establish and support the implemented enterprise architecture as a *dynamic* architecture; that is, one having the flexibility to evolve rapidly in response to changes in the technology and business environment.

The change management process once established will determine:

- The circumstances under which the enterprise architecture, or parts of it, will be permitted to change after implementation, and the process by which that will happen
- The circumstances under which the enterprise architecture development cycle will be initiated again to develop a new architecture

The architecture change management process is very closely related to the architecture governance processes of the enterprise, and to the management of the Architecture Contract between the architecture function and the business users of the enterprise.

In Phase H it is critical that the governance body establish criteria to judge whether a change request warrants just an architecture update or whether it warrants starting a new cycle of the ADM. It is especially important to avoid "creeping elegance", and the governance body must continue to look for changes that relate directly to business value.

Guidelines for establishing these criteria are difficult to prescribe, as many companies accept risk differently, but as the ADM is exercised, the maturity level of the governance body will improve, and criteria will become clear for specific needs.

Drivers for Change

There are many technology-related drivers for architecture change requests. For example:

- New technology reports
- Asset management cost reductions
- Technology withdrawal
- Standards initiatives

This type of change request is normally manageable primarily through an enterprise's change management and architecture governance processes.

In addition there are business drivers for architecture change, including:

- Business-as-usual developments
- Business exceptions
- Business innovations
- Business technology innovations
- Strategic change

This type of change request often results in a complete re-development of the architecture, or at least in an iteration of a part of the architecture development cycle, as explained below.

The Change Management Process

The change management process needs to determine how changes are to be managed, what techniques are to be applied, and what methodologies used. The process also needs a filtering function that determines which phases of the architecture development process are impacted by requirements. For example, changes that affect only migration may be of no interest in the architecture development phases.

There are many valid approaches to change management, and various management techniques and methodologies that can be used to manage change; for example, project management methods such as PRINCE 2, service management methods such as ITIL, management consultancy methods such as Catalyst, and many others. An enterprise that already has a change management process in place in a field other than architecture (for example, in systems development or project management) may well be able to adapt it for use in relation to architecture.

The following describes an approach to change management, aimed particularly at the support of a dynamic enterprise architecture, which may be considered for use if no similar process currently exists.

The approach is based on classifying required architectural changes into one of three categories:

- **Simplification change**: A simplification change can normally be handled via change management techniques.
- **Incremental change**: An incremental change may be capable of being handled via change management techniques, or it may require partial re-architecting, depending on the nature of the change. See below for guidelines.
- **Re-architecting change**: A re-architecting change requires putting the whole architecture through the architecture development cycle again.

Another way of looking at these three choices is to say that a simplification change to an architecture is often driven by a requirement to reduce investment; an incremental change, by a requirement to derive additional value from existing investment; and a re-architecting change, by a requirement to increase investment in order to create new value for exploitation.

To determine whether a change is simplification, incremental, or re-architecting, the following activities are undertaken:

1. Registration of all events that may impact the architecture
2. Resource allocation and management for architecture tasks
3. The process or role responsible for architecture resources assessment of what should be done
4. Evaluation of impacts

Guidelines for Maintenance versus Architecture Re-Design

A good rule-of-thumb is:

- If the change impacts two stakeholders or more, then it is likely to require an architecture re-design and re-entry to the ADM.
- If the change impacts only one stakeholder, then it is more likely to be a candidate for change management.
- If the change can be allowed under a dispensation, then it is more likely to be a candidate for change management.

For example:

- If the impact is significant for the business strategy, then there may be a need to redo the whole enterprise architecture – thus a re-architecting approach.
- If a new technology or standards emerge, then there may be a need to refresh the Technology Architecture, but not the whole enterprise architecture – thus an incremental change.
- If the change is at an infrastructure level – for example, ten systems reduced or changed to one system – this may not change the architecture above the physical layer, but it will change the baseline description of the Technology Architecture. This would be a simplification change handled via change management techniques.

In particular, a refreshment cycle (partial or complete re-architecting) may be required if:

- The Foundation Architecture needs to re-aligned with the business strategy.
- Substantial change is required to components and guidelines for use in deployment of the architecture.
- Significant standards used in the product architecture are changed which have significant end-user impact; e.g., regulatory changes.

If there is a need for a refreshment cycle, then a new Request for Architecture Work must be issued (to move to another cycle).

Security Guidance

As stated in the Requirements Management section, change is driven by new requirements. Changes in security requirements are often more disruptive than a simplification or incremental change. Changes in security policy can be driven by statute, regulation, or something that has gone wrong.

Changes in security standards are usually less disruptive since the trade-off for their adoption is based on the value of the change. However, standards changes can also be mandated. Similar approaches to these changes as mentioned above are good rules of thumb for security as well. However, security changes are often infrastructure changes, and can have a greater impact. A seemingly small security requirement change can easily trigger a new architecture development cycle.

Determine "what has gone wrong?"

Good security forensics practices in conjunction with a written published security policy make determination of what has gone wrong possible. Further, they make enforcement possible. As the guidance above suggests, minor changes can be made in the context of change management and major changes will require a new architectural effort.

Incorporate security-relevant changes to the environment into the requirements for future enhancement (enhancement of existing objective)

Changes that arise as a result of a security problem or new security technology will feed into the Requirements Management process.

About The Open Group

The Open Group is a vendor-neutral and technology-neutral consortium, whose vision of Boundaryless Information Flow™ will enable access to integrated information within and between enterprises based on open standards and global interoperability. The Open Group works with customers, suppliers, consortia, and other standards bodies. Its role is to capture, understand, and address current and emerging requirements, establish policies, and share best practices; to facilitate interoperability, develop consensus, and evolve and integrate specifications and Open Source technologies; to offer a comprehensive set of services to enhance the operational efficiency of consortia; and to operate the industry's premier certification service, including UNIX® system certification. Further information on The Open Group can be found at www.opengroup.org.