Architecture Work Template

Table of Contents

Change Log

TIP The purpose of the change log is to keep track of changes made to the document. It should be updated each time a change is made, with the date, author, and a brief description of the change.

When?	Who?	What?
date of the change	author of the change	description of the change

Introduction

TIP The purpose of the introduction is to provide context for the architecture work. It should include the purpose of the document, definitions of key terms, and any relevant background information.

Purpose

TIP Explain why it is important to perform the architecture work. For instance, it may be related to a change request that is significant or can be implemented in different ways, requiring the approval of stakeholders. Therefore, the architecture work is needed to clarify the options and support decision-making.

Definitions

TIP Define any key terms that will be used throughout the document. This will help ensure that everyone has a common understanding of the terms used in the architecture work.

Architecture Vision

TIP This is where the artefacts of the first action of the Quick Technical Architecture Method are documented.

Stakeholders optional

TIP The stakeholder map helps identify the stakeholders involved in the architecture work. It should include their drivers, concerns, and any other relevant information that will help to model the Motivation View.

TIP Drivers are the motivations of the stakeholders. They can be business drivers, technical drivers, or any other type of driver that influences the architecture work. We can also see them like concerns or needs of the stakeholders.

TIP Goals are the objectives that the stakeholders want to achieve through the architecture work. They shall be related to the drivers. If a goal exists without a driver, either the driver is missing or the goal is may be not relevant.

TIP Outcomes are the results that the stakeholders expect from the fulfilment of the goals. They are the benefits that the stakeholders will gain from the architecture work. It is highly related to the concept of "value" in the business architecture.

TIP Requirements are the constraints that the stakeholders impose on the architecture work. They can be functional or non-functional requirements.

TIP Principles are the guidelines that the stakeholders want to follow in the architecture work. They can be related to the design, implementation, or any other aspect of the architecture work. They are often derived from the requirements and help to ensure that the architecture work is aligned with the stakeholders' needs and expectations.

Table 1. Architecture Vision: Stakeholder Map

Stakeholder	Drivers	Goals	Outcome	Requirements	Principles
the name of the stakeholder	the drivers of the stakeholder	the goals of the stakeholder	the outcomes expected by the stakeholder	the requirements imposed by the stakeholder	the principles followed by the stakeholder

Motivation View

TIP The motivation view provides the context for the architecture work. It should include the business drivers, constraints, and any other relevant information that helps stakeholders understand the need for and challenges of the architecture work.

TIP The output of the stakeholder map can help define the motivation view. However, some concepts like requirements and principles may not be directly related to the stakeholders. They can be derived from business drivers, goals, and outcomes, or defined by the architects to ensure that the architecture work is aligned with stakeholders' needs and expectations.

include the diagram there

Strategy viewpoint optional

TIP The strategy viewpoint enables the Business Architect to model a high-level overview

> of the enterprise’s strategies, supported capabilities, value streams, resources, and intended outcomes.

include the diagram there

Solution Concept diagram optional

> **TIP** The solution concept diagram provides a high-level overview of the architecture work. It should include the main components, their relationships, and any other relevant information that helps stakeholders understand the outcome of the architecture work.

include the diagram there

Objective of the Architecture Work

> **TIP** The objective of the architecture work is to define the scope and purpose of the architecture work. It should be clear, concise, and aligned with the stakeholders' needs and expectations.

Table 2. Architecture Vision: SMART Criteria Assessment

Criteria	Description	Content	Self-Assessment
Specific	The objective should be clear and unambiguous, detailing exactly what is to be accomplished.	type the object content matching the criteria	[ ] Yes the objective is Specific!
Measurable	There should be criteria for measuring progress and success. It answers the question: How will you know when it’s done?	type the object content matching the criteria	[ ] Yes the objective is Measurable!
Achievable	The objective should be realistic, considering available resources and constraints.	type the object content matching the criteria	[ ] Yes the objective is Achievable!
Relevant	The objective should align with stakeholder expectations and practitioner(s) skills.	type the object content matching the criteria	[ ] Yes the objective is Relevant!
Time-bound	The objective should have a defined timeline or deadline to create urgency and focus.	type the object content matching the criteria	[ ] Yes the objective is Time-bound!

include the full objective there

🏢 Business Architecture

TIP This is where the artefacts of the second action of the Quick Technical Architecture Method are documented

Information Map

TIP A collection of information concepts and their relationships, reflecting the business vocabulary (e.g., client, account, product). Mapping begins by identifying key elements important to the business and defining them in business terms.

Baseline Information Map

include the baseline diagram there

Target Information Map

include the target diagram there

Gaps

highlight the high-level differences between the baseline and target diagrams

Organization Viewpoint optional

TIP The organization viewpoint focuses on the internal structure of a company, department, or network. It can be modeled using nested block diagrams or traditional organizational charts and helps identify competencies, authority, and responsibilities.

Baseline Organization Viewpoint

include the baseline diagram there

Target Organization Viewpoint

include the target diagram there

Gaps

highlight the high-level differences between the baseline and target diagrams

Product Viewpoint optional

TIP The product viewpoint shows the value a product offers to customers or external parties. It details the product's composition using services and related contracts or agreements. It can also show delivery channels and related events. This viewpoint supports product development by guiding the design of new or existing services and informing business process and ICT design.

Baseline Product Viewpoint

include the baseline diagram there

Target Product Viewpoint

include the target diagram there

Gaps

highlight the high-level differences between the baseline and target diagrams

Gap Analysis

TIP The gap analysis identifies the differences between the current state and the desired future state of the architecture. It helps to identify the areas that need to be addressed in order to achieve the objectives of the architecture work.

Table 3. Business Architecture: Gap Analysis

Gap	Description	Notes
the name of the gap	a breif description of the gap, baseline vs target	additional notes or comments if needed

Information Systems Architecture

TIP This is where the artefacts of the third action of the Quick Technical Architecture Method are documented.

TIP Alternative architectures can pop up during the Information Systems Architecture analysis. Drawing a target diagram for each can be time-consuming, so it is recommended to, as much as possible, represent the target alternatives within the same diagram. This way, the Baseline and Target Architectures can be compared side by side, highlighting the differences and gaps.

Application Structure Viewpoint

TIP The application structure viewpoint illustrates the structure of one or more applications or components. It helps design or understand their architecture and related data, such as breaking down a system or identifying legacy components for migration or integration.

Baseline Application Structure Viewpoint

include the baseline diagram there

Target Application Structure Viewpoint

include the target diagram there

Gaps

highlight the high-level differences between the baseline and target diagrams

C4 Model diagrams optional

TIP The C4 model diagrams provide a hierarchical view of the architecture, from the context diagram to the component diagram. They help to visualize the architecture and its components, their relationships, and interactions.

TIP The C4 model diagrams are optional, but they can be useful to provide a more detailed view of the architecture. They can be used to complement the application structure viewpoint and provide a more comprehensive view of the architecture. They can also just replace the Application Structure Viewpoint.

TIP The Container Diagram may be too confusing in the Application Architecture section because of its technological nature. It is recommended to use it in the Technology Architecture section instead, where it is more appropriate.

TIP Drawing a C4 model diagram for the Baseline and another for the Target Architecture can be time-consuming. Therefore, it is recommended to, as much as possible, represent the Baseline and Target Architectures in the same diagram.

TIP The purpose of the modeling is to identify and highlight gaps, so model the strict minimum.

Solution context diagram

TIP The solution diagram is not defined in the C4 model. Usually, architecture work is about a system or a set of systems within an ecosystem that has defined boundaries.

Most of the time, the boundary is the limit of a so-called "solution." Therefore, the diagram starts with the solution and shows persons and systems or other solutions consuming its services, as well as systems or other solutions serving it. This way, in one view, we can see who depends on the solution and who the solution depends on. Then, when we zoom into the Solution box, we find the Container Diagram, which describes the systems composing the solution.

include the solution context diagram there

highlight the high-level differences between the baseline and target diagrams

System context diagram

TIP

The system context diagram is the first diagram of the C4 model. It provides a high-level overview of the system and its interactions with external entities, such as users and systems. It helps to understand the context in which the system operates and its relationships with external entities.

include the system context diagram there

highlight the high-level differences between the baseline and target diagrams

Information Structure Viewpoint

TIP

The application usage viewpoint shows how applications support business processes and interact with other applications. It helps design applications by identifying needed services, or design processes by showing available services. It also highlights process dependencies, aiding operational management.

Baseline Information Structure Viewpoint

include the baseline diagram there

Target Information Structure Viewpoint

include the target diagram there

Gaps

highlight the high-level differences between the baseline and target diagrams

Application/Data Matrix optional

TIP

The Application/Data matrix shows the relationship between applications and the data entities they access or modify. It maps how applications create, read, update, or delete data—such as a CRM system managing customer data. Data entities may include master, reference, transactional, content, or historical data, and are handled by various types of applications like transactional, information management, or business warehouse systems.

Table 4. Information Systems Architecture: Application/Data Matrix

Data → Applications ▯	Data#A	Data#B
App#A	read	
App#B		write
App#C	read	write

Gap Analysis

TIP

The gap analysis identifies the differences between the current state and the desired future state of the architecture. It helps to identify the areas that need to be addressed in order to achieve the objectives of the architecture work.

Table 5. Information Systems Architecture: Gap Analysis

Alternative	Gap	Description	Notes
the name of the alternative architecture, if any	the name of the gap	a breif description of the gap, baseline vs target	additional notes or comments if needed

Technology Architecture

TIP

This is where the artefacts of the fourth action of the Quick Technical Architecture Method are documented.

TIP

Alternative architectures can pop up during the Technology Architecture analysis. Drawing a target diagram for each can be time-consuming, so it is recommended to, as much as possible, represent the target alternatives within the same diagram. This way, the Baseline and Target Architectures can be compared side by side, highlighting the differences and gaps.

Technology Viewpoint

TIP

The technology viewpoint includes the software and hardware elements that support the Application Layer, such as physical devices, networks, operating systems, databases, and middleware.

Baseline Technology Viewpoint

include the baseline diagram there

Target Technology Viewpoint

include the target diagram there

Gaps

highlight the high-level differences between the baseline and target diagrams

C4 Model diagrams optional

TIP The C4 model diagrams provide a hierarchical view of the architecture, from the context diagram to the component diagram. They help to visualize the architecture and its components, their relationships, and interactions.

TIP Guidance on the C4 model diagrams is provided in the Information Systems Architecture section.

Container Diagram

TIP In the technology architecture section, the Container Diagram is the most relevant C4 model diagram. It provides a high-level overview of the technology architecture, showing the containers (applications, databases, etc.) and their relationships.

include the container diagram there

highlight the high-level differences between the baseline and target diagrams

Physical Viewpoint

TIP The physical viewpoint represents equipment that creates, uses, stores, moves, or transforms materials, along with its connections via distribution networks and any assigned active elements.

Baseline Physical Viewpoint

include the baseline diagram there

Target Physical Viewpoint

include the target diagram there

Gaps

highlight the high-level differences between the baseline and target diagrams

C4 Model deployment diagrams optional

TIP The C4 model deployment diagrams focus on the physical deployment of the technology architecture. They show how the containers are deployed on the physical infrastructure, including servers, networks, and other hardware or cloud components.

TIP Guidance on the C4 model diagrams is provided in the Information Systems Architecture section.

Deployment Diagram

include the deployment diagram there

highlight the high-level differences between the baseline and target diagrams

Codebase Analysis optional

TIP Some changes may require a codebase analysis to understand the current state of the code and its dependencies. This can help identify potential issues and risks associated with the architecture work.

include the codebase analysis there

highlight the high-level differences between the baseline and target about the codebase

Threat Modeling optional

TIP Threat modeling is used to identify and address potential security threats in an architecture. Techniques like STRIDE and DREAD help analyze and mitigate these risks.

include the data flow diagram (DFD) here

include the STRIDE analysis here

include the DREAD analysis here

Gap Analysis

TIP The gap analysis identifies the differences between the current state and the desired future state of the architecture. It helps to identify the areas that need to be addressed in order to achieve the objectives of the architecture work.

Table 6. Technology Architecture: Gap Analysis

Alternative	Gap	Description	Notes
the name of the alternative architecture, if any	the name of the gap	a breif description of the gap, baseline vs target	additional notes or comments if needed

Work Packages

TIP This is where the artefacts of the fifth action of the Quick Technical Architecture Method are documented.

Gap Consolidation

TIP This list consolidate the gaps identified in the previous sections. It provides a high-level overview of the work packages that need to be addressed in order to achieve the objectives of the architecture work.

Table 7. Work Packages: Gap Consolidation

Alternative	Gap References	Work Package	Description	Notes
the name of the alternative architecture, if any	the list of gaps related to the work package	the name of the work package	a brief description of the work package	additional notes or comments if needed

Work Package Grouping

TIP This section groups the work packages into logical groups. It helps to organize the work packages and identify the dependencies between them.

Table 8. Work Packages: Work Package Grouping

Alternative	Work Package References	Group	Description	Dependencies	Notes
the name of the alternative architecture, if any	the list of work packages in the group	the name of the group	a brief description of the group	the list of dependencies between the work packages	additional notes or comments if needed

Action List

TIP This section provides a list of actions that need to be taken in order to implement the work packages. It helps to identify the tasks that need to be completed and the resources required to complete them.

Table 9. Work Packages: Action List

Alternative	Group	Work Package	Actions	Effort	Notes
the name of the alternative architecture, if any	the name of the group	the name of the work package	the list of actions that need to be taken to implement the work package	the estimated effort required to complete the actions	additional notes or comments if needed

Course of Action

TIP This is where the artefacts of the sixth action of the Quick Technical Architecture Method are documented.

Implementation Phases

TIP Identifying the implementation phases is crucial for planning and executing the architecture work. It helps to break down the work into manageable phases. The unit of work for the phase is the group of work packages.

TIP The concept of wave letter help to identify dependencies between the phases. For instance, if a phase is dependent on the completion of another phase, it can be assigned a letter that comes after the letter of the dependent phase. This way, it is easy to see which phases can be executed in parallel and which ones need to be executed sequentially.

Table 10. Course of Action: Phases

Alternative	Group References	Phase	Wave	Deliverables	Effort	Description	Notes
the name of the alternative architecture, if any	the list of work package groups related to the phase	the name of the phase	the wave or iteration of the phase	the list of deliverables for the phase	the estimated effort required to complete the phase, based on estimated actions	a brief description of the phase	additional notes or comments if needed

Implementation and Migration Viewpoint

TIP The implementation and migration viewpoint links programs and projects to the architecture elements they implement. It models the scope of these initiatives in terms of realized plateaus or affected elements, with annotations showing how the elements are impacted.

TIP The "Course Of action: Phases" table shall provide most of the information needed to fill this section.

include the implementation and migration viewpoint diagram here

Gant Chart optional

TIP The Gantt chart provides a visual representation of the implementation phases and their dependencies. It helps to plan and schedule the work packages and actions, ensuring that the architecture work is executed in a timely manner.

include the Gantt chart here

Risks optional

TIP List the risks associated with the implementation phases. It helps to identify potential issues that may arise during the implementation of the architecture work and to plan for their mitigation.

include the risks, their likelihood, impact, and mitigation strategies here

Implementation Factors optional

TIP This section provides a list of factors that need to be considered during the implementation. It helps the implementation team providing guidance on how to implement the architecture work and to ensure that it is aligned with the stakeholders' needs and expectations.

TIP The notes column from the other tables can be used to note the implementation factors when performing the architecture work. This way, it is easy to keep track of the factors that need to be considered during the implementation and just consolidate them in this section.

Table 11. Course of Action: Implementation Factors

Factor	Description	Notes
the name of the factor	a brief description of the factor	additional notes or comments if needed

License