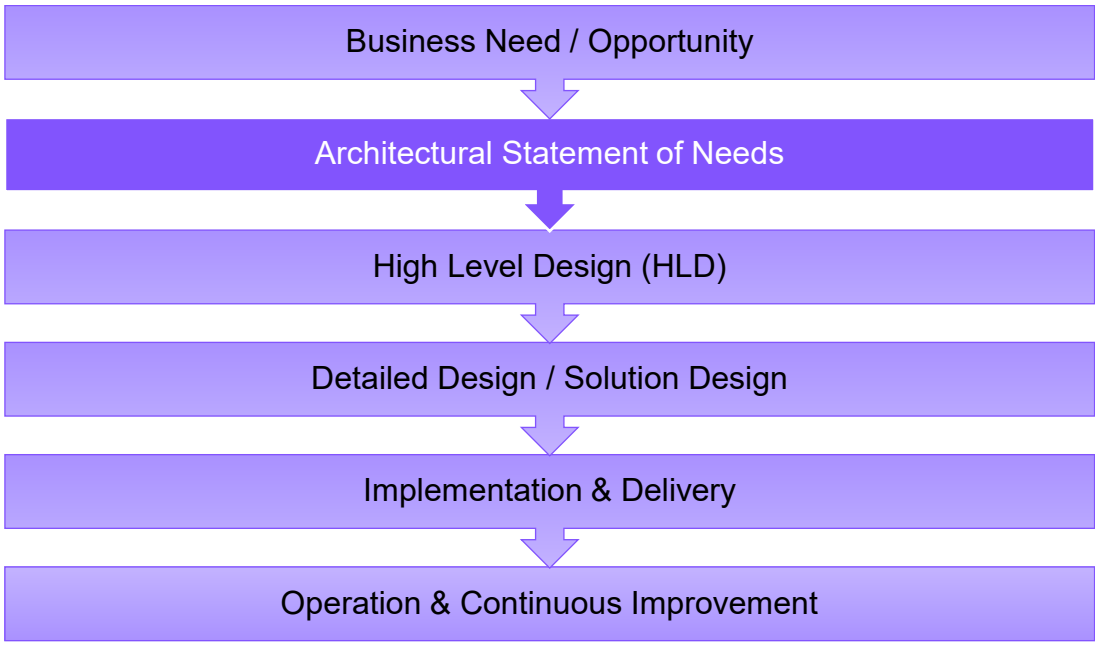
Architectural Statement of Needs

1 Architectural Artefact Flow

The Architectural Statement of Needs is an early-stage artefact that defines needs and requirements, which are then addressed by a High-Level Design (HLD), followed by more detailed design and implementation.

Figure 1.1: Position of the Architectural Statement of Needs in the overall Architecture Process



The Architectural Statement of Needs is produced after a business need or opportunity is identified. It defines the context, drivers, principles, and requirements for the solution.

The High-Level Design (HLD) is developed in response to the Statement of Needs, translating requirements into architectural solutions. The HLD is then elaborated into detailed designs, which guide implementation and delivery.

The process concludes with operation and ongoing improvement.

2 Completion Guidance for Authors

KEEP THE GUIDANCE SECTIONS IN WORKING DRAFTS. BEFORE SUBMITTING AS A FINAL ARTEFACT, REMOVE THEM —UNLESS THE INTENDED AUDIENCE SPECIFICALLY VALUES HAVING THEM INCLUDED.

This template is designed to help you capture the architectural needs for a solution in a clear, structured, and agile aligned manner. Please read and follow these instructions before completing each section:

2.1.1 Purpose and Intent

The Architectural Statement of Needs is intended to define the context, drivers, principles, and requirements for a proposed solution. Its purpose is to provide a comprehensive foundation that will be satisfied by the subsequent production of a High-Level Design (HLD). Authors should focus on what is needed and why, rather than how it will be implemented—the HLD will address the “how”.

2.1.2 Progressive Detail

Begin with high-level summaries in early sections (e.g., Problem Statement, Architectural Vision) and elaborate with more detail in later sections (e.g., Solution Intent, Enablers). This helps readers understand the context before diving into specifics.

2.1.3 Cross-Referencing

Some information (such as business drivers, constraints, or principles) may be relevant to multiple sections. To avoid unnecessary repetition, reference earlier sections where appropriate (e.g., “See Architectural Principles for compliance requirements”).

2.1.4 Consistency

Use consistent terminology and definitions throughout the document. If you introduce a concept or requirement in one section, ensure it is described in the same way elsewhere.

2.1.5 Clarity and Brevity

Aim for clear, concise language. Use bullet points, tables, and diagrams where helpful. Avoid jargon unless it is widely understood by all stakeholders.

2.1.6 Traceability

If a requirement, principle, or constraint is critical, it is acceptable for it to appear in more than one section. This ensures visibility for stakeholders who may only review specific parts of the document.

2.1.7 Iterative Updates

This document is expected to evolve as the solution matures. Update sections as new information becomes available, and clearly indicate version changes.

2.1.8 Risk Management

If there are known risks that need to be avoided or accommodated, the most appropriate section to use is the Initial Solution Intent. This section is designed to capture design

constraints, non-functional requirements (NFRs), models, and assumptions—including risks that may impact the solution.

How to include risks:

- List risks as part of the constraints and assumptions.
- Clearly state whether each risk should be avoided, mitigated, or accommodated.
- If a risk is critical and influences architectural principles or the overall approach, you may also reference it in the Architectural Principles section.

If risks are closely tied to business drivers or strategic alignment, you may briefly mention them in the Problem Statement or Architectural Vision, but the main detail should be in Initial Solution Intent.

2.1.9 Section Guidance

Each section contains prompts to help you focus your input. If a prompt asks for information already covered elsewhere, provide a brief summary and reference the relevant section.

By following these guidelines, you will help ensure the document is clear, actionable, and easy to maintain, while minimising unnecessary duplication. The ultimate goal is for the Statement of Needs to be addressed and fulfilled by the High-Level Design that follows.

3 Architectural Statement of Needs

3.1 Problem Statement

3.1.1 Purpose

Clearly articulate the business drivers, pain points, and the context for the solution.

3.1.2 Guidance

- Describe the business challenge or opportunity.
- Identify who is impacted and why change is needed.
- State measurable outcomes or objectives.
- Example prompts:
    - What business process or capability is being improved?
    - What risks or inefficiencies exist today?
    - What will success look like?

3.2 Architectural Vision

3.2.1 Purpose

Set out the high-level goals and strategic alignment of the solution.

3.2.2 Guidance

- Summarise the desired future state.
- Align with organisational strategy, digital transformation, or portfolio themes.
- Example prompts:
    - What is the target state for this capability?
    - How does this solution support business strategy or strategic themes?
    - What are the guiding values or principles?

3.3 Architectural Principles

3.3.1 Purpose

Identify and explain the architectural principles, standards, and directives that must be followed.

3.3.2 Guidance

- List relevant internal principles (e.g., interoperability, resilience, security).
- Reference applicable policies, procedures, or standards (internal and external).
- Explain why each principle is important for this solution.
- Example prompts:
    - Which enterprise architecture principles apply?
    - Are there regulatory or compliance standards to consider?
    - What lessons from previous projects should be applied?

3.4 Initial Solution Intent

3.4.1 Purpose

Capture design constraints, non-functional requirements (NFRs), models, and assumptions.

3.4.2 Guidance

- Outline key constraints (e.g., technology, budget, timeline).
- List NFRs (e.g., performance, scalability, security, usability).
- Include initial models or diagrams if available.
- State assumptions and known unknowns.
- Identify any known risks that could affect the solution, such as technical limitations, compliance concerns, or resource constraints. For each risk, specify whether it should be avoided, mitigated, or accommodated, and describe any planned actions or contingencies.
- Include a table or bullet list of identified risks, their impact, and planned mitigation or accommodation strategies.
- Example prompts:
    - What are the critical NFRs for this solution?
    - Are there technology or vendor constraints?
    - What assumptions are being made at this stage?
    -

3.5 Architectural Runway

3.5.1 Purpose

Describe existing infrastructure, capabilities, and gaps that affect the solution.

3.5.2 Guidance

- Identify current systems, platforms, or services to be leveraged.
- Highlight gaps or technical debt that may impact delivery.
- Example prompts:
    - What reusable assets or platforms exist?
    - What infrastructure gaps must be addressed?
    - Are there dependencies on other projects or teams?

3.6 Enablers

3.6.1 Purpose

Define technical stories or activities needed to explore or implement architectural elements.

3.6.2 Guidance

- List enabler stories (e.g., proof-of-concept, integration test).
- Explain how each enabler supports architectural runway or solution intent.
- Example prompts:
    - What technical investigations are required?
    - Are there prototypes or pilots needed?
    - What integration or migration activities are planned?

3.7 Stakeholders and Roles

3.7.1 Purpose

Identify who contributes to and who decides on architectural matters.

3.7.2 Guidance

- List key stakeholders (business, IT, external partners).
- Define roles and responsibilities (e.g., architect, product owner, sponsor).
- Example prompts:
- Who are the decision-makers?
- Who provides input or review?
- Are there external dependencies?

3.8 Roadmap Alignment

3.8.1 Purpose

Show how the architecture supports upcoming features, releases, or business milestones.

3.8.2 Guidance

- Link architectural needs to the product or solution roadmap.
- Highlight alignment with Project, Demand or other need.
- Example prompts
    - What features or capabilities are enabled by this architecture?
    - How does this solution fit into the overall delivery timeline?
    - Are there critical milestones or deadlines?