Lifecycle by Design

# ARCHITECTURE AUTHORITY

## MATURITY GUIDE

Lifecycle by Design Guides

**From Sales Engineering to Institutional Solution Architecture**

# Table of Contents



---

## Introduction: Why Architecture Maturity Matters

Architecture plays a central role in how services organizations translate client needs into solutions that can actually be delivered. It defines how technologies are assembled, how platforms integrate, and how services ultimately operate within client environments.

In most services organizations, architecture begins as an individual capability rather than a structured organizational function. Skilled engineers and solution designers work directly with clients to shape solutions and often remain involved through implementation. In smaller organizations this approach works well. The same people who design the solution frequently deliver it, ensuring that architectural decisions remain grounded in real operational experience.

As organizations grow, however, the environment changes. More opportunities emerge simultaneously, delivery teams expand, and services begin to cover broader technology domains. Architecture becomes involved in more deals, more projects, and more technical decisions across the organization.

At this point, architecture must evolve.

The challenge is not the importance of architecture itself, but **where it sits within the organization**. In many services companies, architects remain positioned primarily within presales or sales engineering functions. While this structure supports deal development, it often prevents architecture from fully benefiting from delivery experience or contributing to the development of repeatable service patterns.

Mature services organizations gradually reposition architecture so that it supports two critical functions:

- **Bid & Proposal**, where solutions are assembled into structured service offerings
- **Practices**, where delivery experience is captured and converted into repeatable architectural patterns

When architecture matures into these roles, it becomes a bridge between client intent, service design, and delivery capability. Solutions become easier to assemble, delivery becomes more predictable, and architectural learning begins to compound across engagements.

This guide explores how services organizations evolve from **sales-driven architecture toward architecture that supports service assembly and pattern development**.

---

## The Hidden Problem in Services Architecture

Most architecture challenges in services organizations begin with **where architecture sits within the organization**.

In many companies, architects are positioned primarily within the sales or presales organization. Their primary responsibility is to help win deals by designing solutions that align with client requirements and available technologies. These architects play an essential role in translating client needs into viable solutions.

However, this structure creates an important limitation.

When architecture is positioned mainly to support sales activity, it often becomes disconnected from two of the most important sources of learning in a services organization:

- **Delivery experience**
- **Service pattern development**

As a result, architects frequently design solutions that are technically sound but not fully informed by how services are actually implemented at scale. Delivery teams then adapt those designs during implementation, and valuable lessons learned during delivery remain confined to individual projects rather than being captured systematically.

This separation produces several common outcomes:

Solutions may be designed repeatedly from scratch rather than assembled from established service patterns.

Delivery teams may encounter architectural challenges only after implementation begins.

Architectural learning becomes fragmented across projects instead of contributing to the continuous improvement of service offerings.

These issues rarely appear as obvious failures. Most organizations continue delivering successful projects through the skill and effort of their engineers and architects. But over time the lack of structural alignment slows proposal development, increases delivery complexity, and prevents the organization from institutionalizing its architectural knowledge.

As services organizations mature, architecture gradually migrates toward roles that support **Bid & Proposal solution assembly and Practice-led pattern development**. This shift allows architectural expertise to incorporate delivery feedback, accelerate proposal development, and continuously refine how services are designed and delivered.

The phases that follow describe how this evolution typically unfolds.

## Architecture Maturity Model

**Phase 1 — Sales-Owned Architecture**

In early-stage services organizations, architecture often emerges from the sales organization itself.

Sales engineers, technical account managers, or senior engineers work directly with clients to design solutions during the sales process. These individuals frequently possess deep technical expertise and may also be responsible for implementing the solution once the deal is won.

In smaller organizations this model works effectively. The same experts who design the solution often remain involved throughout delivery, ensuring that architectural decisions align closely with implementation reality.

However, architecture authority at this stage resides primarily in **individual expertise rather than institutional structure**.

Characteristics of this phase include:

- Architecture performed by sales engineers or senior technical staff
- Solution design closely tied to specific deals
- Limited formal documentation or reference patterns
- Sales, architecture, and delivery responsibilities often overlapping

While effective in small environments, this model becomes difficult to scale as organizations grow and the number of simultaneous engagements increases.

---

**Phase 2 — Presales Solution Architecture**

As services organizations expand, architecture responsibilities often move into a **dedicated presales function**.

Solution architects support account teams by translating client requirements into structured technical designs. These architects begin to standardize documentation, improve technical rigor, and provide greater consistency across proposals.

Architecture becomes more specialized and professionalized, but its primary objective remains **supporting the sales process**.

Characteristics of this phase include:

- Dedicated presales solution architects supporting multiple deals
- Increased technical rigor in proposal development
- Early architecture documentation standards
- Greater separation between sales and delivery teams

While solution quality improves, architecture is still largely **deal-centric**, with designs created individually for each engagement.

---

**Phase 3 — Delivery-Informed Architecture**

As organizations mature further, delivery teams begin influencing solution design earlier in the sales cycle.

Delivery architects or experienced engineers participate in proposal development to ensure that proposed solutions reflect operational realities. Lessons learned from previous implementations start informing architectural decisions.

This stage marks the beginning of **organizational learning within architecture**.

Characteristics include:

- Delivery architects contributing to presales design reviews
- Feedback loops emerging between implementation teams and architects
- Early architectural patterns forming from repeated delivery experiences
- Reduced implementation surprises after project kickoff

Architecture begins shifting from purely presales design toward **delivery-informed solution development**.

---

**Phase 4 — Bid & Proposal Service Assembly**

At higher levels of maturity, architecture authority increasingly shifts into the **Bid & Proposal function**.

Rather than designing solutions from scratch for every engagement, proposal teams assemble solutions using predefined service components, architectural models, and delivery patterns.

Architecture becomes less about invention and more about **structured service assembly**.

Characteristics of this phase include:

- Bid & Proposal teams coordinating solution architecture
- Standard solution frameworks emerging across services
- Faster proposal development through reusable architectural models
- Improved alignment between architecture and delivery models

The focus of architecture moves from designing unique solutions toward assembling services from **proven building blocks**.

---

**Phase 5 — Practice Pattern Architecture**

As services organizations mature, architecture increasingly resides within **practices**.

Practices formalize the architectural patterns that define how services are delivered. These patterns are informed by delivery experience and refined over time through repeated implementation.

Practice architects define:

- Reference architectures
- Integration models
- Deployment patterns
- Standard design frameworks

Bid & Proposal teams assemble client solutions using these patterns, ensuring alignment between presales design and delivery capability.

Architecture at this stage becomes **institutional knowledge rather than individual expertise** and now functions as a **learning system**, where every engagement contributes to improving the organization's design capabilities.

## Architecture as the Preservation of Solution Integrity

Architecture plays an essential role in preserving the integrity of the solution envisioned during the sales process.

When a services organization engages with a client, the conversation begins with a set of needs, constraints, and desired outcomes. Architects help translate those needs into a technical approach—defining how technologies will be assembled, how platforms will integrate, and how the solution will operate within the client environment.

As the opportunity progresses from early discussions into formal proposals and eventually into delivery, that original design intent must remain coherent. Without architectural continuity, solutions can easily fragment as different teams interpret the design in different ways.

Architecture provides the technical continuity that connects these stages.

This does not mean that solutions remain unchanged. Client environments evolve, constraints emerge, and delivery teams frequently adapt designs during implementation. What architecture preserves is not rigidity, but **the integrity of the overall solution approach**.

In mature services organizations, architecture ensures that the technical vision established during the sales process survives the transition into proposal development and ultimately into delivery.

---

## The Structural Relationship Between Sales, Bid & Proposal, Practices, and Delivery

Architecture operates across several key functions within a services organization.

Sales teams engage with clients to understand business challenges and identify potential opportunities. Architects frequently participate in these discussions, helping translate client needs into potential technical approaches.

As opportunities mature, Bid & Proposal teams coordinate the development of formal proposals. During this stage, architects support the assembly of the solution, ensuring that the proposed approach aligns with the organization's service capabilities.

Practices define the patterns that underpin these services. They maintain reference architectures, establish deployment models, and capture lessons learned from delivery engagements.

Delivery teams ultimately implement the solution within the client environment. During this phase, delivery architects often adapt architectural patterns to accommodate the specific realities of the client environment while maintaining the overall design intent.

When these groups operate independently, architectural learning becomes fragmented. Solutions may be repeatedly designed from scratch, delivery teams may encounter architectural surprises, and lessons learned during implementation may never influence future designs.

In mature organizations, architecture acts as a connective capability that supports all stages of the lifecycle. Sales introduces client intent, Bid & Proposal assembles the solution, Practices define patterns, and Delivery implements the work. Architecture helps maintain alignment across these stages.

## The Two Dimensions of Architecture Maturity

Architecture maturity evolves along two complementary dimensions.

The first dimension concerns **how solutions are assembled during the sales and proposal process**.

Early in an organization's development, architects design solutions largely from scratch for each engagement. This approach relies heavily on individual expertise and often results in highly customized solutions.

As the organization matures, solution design becomes increasingly structured. Bid & Proposal teams assemble solutions using defined service offerings and architectural patterns, allowing proposals to be developed more quickly and with greater consistency.

The second dimension concerns **how architectural knowledge evolves through delivery experience**.

In less mature organizations, architectural learning remains tied to individual projects. Engineers and architects gain valuable experience, but those insights are rarely captured in a way that benefits future engagements.

Over time, mature organizations capture this knowledge through practices that define reference architectures, integration models, and deployment patterns. Architects contribute to these patterns and refine them based on real-world implementation experience.

Together, these dimensions allow architecture to support both **efficient service assembly and continuous improvement of service design**.

---

## What Mature Architecture Maturity Looks Like in Practice

In mature services organizations, architecture is embedded throughout the services lifecycle rather than confined to a single organizational function.

Architects participate in early sales discussions to help frame possible solution approaches. As opportunities progress, they support Bid & Proposal teams in assembling solutions that align with established service offerings.

Practices maintain the architectural patterns that guide these services. Architects contribute to the development and refinement of these patterns based on insights gained from both presales solution design and delivery experience.

During implementation, delivery architects adapt these patterns to the specific realities of the client environment. They ensure that technical decisions made during delivery remain aligned with the original design intent.

This structure allows the organization to balance flexibility and consistency. Solutions can be adapted to meet client requirements while still benefiting from established patterns and prior experience.

Architecture becomes a lifecycle capability that connects opportunity development, proposal assembly, and delivery execution.

## A Note on Pricing Delivery Architects Correctly

Delivery architects perform some of the most complex technical work within a services organization.

They are responsible for adapting architectural patterns to the specific realities of client environments, resolving integration challenges, and ensuring that complex systems function as intended once deployed.

Despite the importance of this work, delivery architects are frequently priced incorrectly within services engagements. In some organizations they are treated as extended engineering resources rather than as senior technical designers responsible for maintaining the integrity of the solution.

This can create several problems.

Underpricing delivery architects makes it difficult to staff engagements appropriately and may discourage organizations from involving architects during critical stages of implementation. When architectural expertise is minimized during delivery, technical decisions can drift away from the original solution design.

Mature services organizations recognize the importance of delivery architects and price them accordingly. Their work protects delivery outcomes, preserves solution integrity, and contributes valuable insights that improve future architectural patterns.

---

## Conclusion: From Hero Architecture to Institutional Design

Most services organizations begin with what might be described as **hero architecture**.

A small number of highly skilled individuals design solutions, support sales conversations, and often deliver the work themselves. Their expertise allows the organization to succeed even without formal architectural structures.

As organizations grow, however, this model becomes increasingly difficult to sustain. The number of engagements increases, delivery teams expand, and architectural decisions must be made across multiple opportunities and projects simultaneously.

Architecture must therefore evolve from an individual capability into a structured component of the services lifecycle.

Architects support sales conversations, help Bid & Proposal teams assemble solutions, contribute to practice-led architectural patterns, and guide delivery teams during implementation.

Over time, architectural knowledge becomes institutional rather than individual. Patterns emerge, proposals become easier to assemble, and delivery outcomes become more predictable.

The organization moves from hero-driven design toward **institutional design systems that allow services knowledge to scale across the business**.