

Make a technology decision reviewable.

Prepare a compact brief that allows a reviewer to challenge the assumptions and understand the proposed tradeoff.

Architecture review agenda

Question	Material to bring
What changes?	A current-state map and the affected business workflow
Why this option?	Comparable alternatives and their constraints
What can fail?	Failure scenarios, ownership and recovery paths
How will we know?	Acceptance checks and a decision owner

Use the same criteria for each option

Compare delivery effort, operability, data boundaries and exit cost consistently. Avoid a detailed analysis of the preferred option beside vague descriptions of alternatives.

Record a trigger for reconsideration

A decision may be sound at today's workload and wrong later. State which change in demand, regulation or operating capacity should reopen the discussion.

Frame one decision clearly

State the decision needed, why it matters now and who can approve it. "Improve the platform" is too broad to review. "Choose how the customer portal reads order status without overloading the ERP" identifies a concrete boundary. Describe the existing workflow, observed constraint and consequence of leaving it unchanged.

Separate requirements from preferences. A recovery requirement or restriction on external processing is different from familiarity with a framework. Record the source of important requirements and the person who can clarify them. Where evidence is missing, identify an investigation rather than presenting an assumption as a settled fact. The brief should give business and technical reviewers the same starting point.

Compare credible options

Use criteria relevant to the decision. For portal order status, options might include direct reads, a cached view or an event-fed read model. Compare freshness, dependency load, operating complexity and recovery. Avoid listing technologies that were never realistic candidates.

Make the recommendation concrete enough to challenge. Include a simple request flow, affected systems and work needed from other teams. State its principal disadvantage and how that will be managed. Reviewers should understand what the choice gives up as well as what it promises. Supporting measurements can sit in an appendix, while the main explanation remains readable by the people approving the business change.

Specify evidence and the next action

Describe how the proposal will be validated. A small experiment may establish dependency capacity, data freshness or feasibility of access controls. Define the observation that would cause the team to reconsider. Include the rollout or rollback boundary where it affects the decision.

Record the review outcome, participants and unresolved conditions. Link the brief to implementation work and update it if the decision changes. Keep detailed evidence available without requiring every reader to reconstruct it. A good brief ends with a bounded next step, an accountable owner and a clear statement of remaining uncertainty. Approval of a direction should not silently become approval of unlimited implementation scope.

What is the right length for a brief?

Enough to explain the decision and its evidence. Link detailed diagrams and measurements rather than burying the decision inside a large document.

Read the current resource online

<https://cobnex-review.rgcsekaraa.workers.dev/resources/architecture-briefs/>