

# Use an architecture review to resolve a decision.

Bring a specific proposal, its alternatives and the evidence needed to challenge the important assumptions.

## Review session structure

| Discussion    | Expected output                                   |
|---------------|---------------------------------------------------|
| Context       | The business task and the constraints that matter |
| Options       | A comparable view of feasible approaches          |
| Failure modes | Controls, recovery and unresolved risks           |
| Decision      | An owner, next action and review condition        |

## Invite the people who own the consequences

Include application, data and operating perspectives where the decision crosses those boundaries. A review without the future operator can miss a costly constraint.

## Finish with a decision record

Capture what was agreed, what remains uncertain and who will obtain the missing evidence. A meeting summary without actions does not move the design forward.

## Choose a question the session can resolve

An architecture session works best when it has one decision to advance. Examples include where customer identity should live, how a new integration will recover from failure or whether a workflow needs an AI component. A broad request to review everything makes it difficult to prepare useful evidence.

Before the session, share a short description of the current system, the problem and any fixed constraints. Include a simple diagram if one exists, but do not delay the conversation to produce a polished presentation. Invite the people who operate the workflow as well as those who build it. They often know failure cases that are absent from the diagrams.

## **Work through one normal case and one failure**

Start by following a real transaction through the system. Identify where information changes, where access is checked and where a person makes a decision. Then change one condition, such as an unavailable provider, a duplicate request or a user losing access part way through the process.

Use the failure case to expose assumptions. If nobody can say which system owns a partially completed order, the design has an unresolved boundary. Record the uncertainty in plain language and decide what evidence would resolve it. The session should produce a clearer decision, not simply a more detailed picture of an uncertain system.

## **Turn discussion into an accountable next step**

Close by recording the chosen direction, alternatives considered and unresolved questions. Assign an owner to each follow-up and agree when the decision needs to be revisited. A diagram without those details can look complete while leaving the original disagreement intact.

Useful outputs may include a decision record, a small validation experiment or a first delivery increment with acceptance criteria. Keep those outputs accessible to the team that will implement and maintain the change. This page describes an architecture session format rather than a calendar of scheduled events. Availability, participants and commercial arrangements need to be agreed for an actual engagement.

## **Are these scheduled public events?**

No. This page describes an architecture review engagement. There is no public event calendar or booking commitment.

## **Read the current resource online**

<https://cobnex-review.rgcsekaraa.workers.dev/events/>