

Give every operating responsibility an owner.

Clarify who can change, observe, recover and retire the application and its dependencies.

Operational ownership register

Responsibility	Information to record
Application changes	Repository, release authority and review process
Cloud environment	Account owner, access path and billing responsibility
Data	Business owner, retention decisions and recovery procedure
Incidents	Coverage, escalation contact and decision authority

Distinguish access from accountability

Possessing an administrator account does not establish responsibility for the service. Name the owner of each decision and the limits of delegated access.

Plan for a change of supplier or team

Keep source code, configuration and operating information available to the business. Review access and continuity when an engagement changes.

Name the owners of the running system

Operational ownership covers the application, its data, environments and dependencies. Record who approves releases, who maintains access and who can make a decision when the system fails. Include business ownership of the workflow as well as technical administration. A supplier managing infrastructure does not automatically own the correctness of business data.

Map responsibilities across internal teams and external providers. For a portal connected to an ERP, identify who investigates a missing order update and who can correct the authoritative record. Make subscription, domain and certificate ownership visible. These details are easy to overlook during implementation and difficult to resolve during an incident if the original team is unavailable.

Keep routine work reviewable

Create a practical schedule for dependency updates, access reviews, backup checks and other maintenance appropriate to the application. Record the expected evidence and how failures enter the backlog. Avoid relying on a statement that a platform is managed without identifying what management includes.

A release record should connect the change to its approval, checks and observed result. A backup record should support a restoration exercise, including configuration and access needed by the application. Documentation should be available to the people carrying out these tasks. Review whether another authorised maintainer can follow the instructions without using a personal account belonging to the original developer.

Make boundaries part of the agreement

Document the systems covered by support, agreed service hours and escalation arrangements. Distinguish defect correction, routine maintenance and new development. State how third-party incidents are handled and what information a provider needs to investigate them. Do not infer availability or response commitments from a general description of managed services.

Review ownership when the application, team or commercial arrangement changes. A new integration may introduce a responsibility that nobody previously held. A handover may require access transfer and practical rehearsal. Clear ownership allows the business to make informed operating decisions and reduces dependence on informal relationships that may not survive staff or supplier changes.

What belongs in an exit handover?

Repositories, deployment configuration, dependency inventory, data export paths, runbooks and an explicit record of outstanding risks.

Read the current resource online

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