

Minimum Requirements for a Project Charter

Module 4: Use Cases

Topic 3: Scope Definition and Boundaries



Why In Scope / Out of Scope Matters

A use case without clear boundaries is impossible to evaluate for feasibility, and impossible for your own team to know when it's "done."

Charter Section 4 expects an explicit In Scope / Out of Scope split.

For anything you're unsure whether to include, ask:

"Would completing this use case still count as a success if this particular piece were left out?"

- If yes, it's probably a nice-to-have that belongs in Out of Scope or a future use case.
 - If the use case would be meaningless without it, it belongs In Scope.

What Belongs In Scope

The core deliverable

The research workflow or capability being built or connected, and the outcome that marks the use case as delivered

The services involved

The services from each participating Node that are used, and the data, standards, or interfaces employed

Stakeholder Support

The types of users and stakeholders this use case is designed to support (a research domain, a type of institution), named explicitly

Integration

How this use case connects with other EOSC services and Nodes, including which interoperability standards and FAIR principles it follows

What Belongs Out of Scope

Excluded Activities

Specific services, activities, or tools your organisation will not be responsible for within this use case

Limitations

Any limits on the support, geographic reach, or target audience (e.g., this covers North Atlantic data only)

Dependencies

Activities this use case relies on but expects another Node or external partner to handle (e.g. if it depends on EOSC AAI being in place, that's a dependency here, not part of this use case's own scope)

Beyond these three, it's also worth excluding anything aspirational that would be nice to include but hasn't been agreed with the other participating Node(s).

Scope Boundary Principles



Too broad: “This use case will improve data interoperability between Nodes.”

This has no boundary at all — it could never be definitively finished, and it doesn’t say which Nodes, which data, or which specific interoperability problem.



Properly scoped: “In scope: building a metadata translation layer between the Marine Science Node’s sensor data catalogue and the Regional Climate Node’s model input format, covering temperature, salinity, and pH readings only. Out of scope: extending this translation layer to additional sensor types, additional Nodes, or historical data predating 2020 — these may form the basis of a future use case.”

The second version tells an evaluator exactly what “done” looks like, and just as importantly, reassures them that the team understands its own limits — a use case with no stated boundaries often signals a team that hasn’t thought through what’s actually achievable.

In Scope

Core deliverable and services involved:

Stakeholder Support (who is this use case designed to support):

Integration (interoperability standards, FAIR principles, other Nodes/services connected to):

Out of Scope

Excluded Activities (services/tools your organisation won't be responsible for):

Limitations (on support, geographic reach, or audience):

Dependencies (handled by another Node or external partner, not by you):

Thank you!