

Use Case Menu

The application asks you to commit to three business use cases. These six are where design partners consistently find value, with example questions to help you decide. Your own use case is just as welcome.

How to pick well: choose use cases where (1) you can name the team that owns the problem today, (2) a correct answer would change a real decision, and (3) you can judge whether the answer is right. Avoid use cases nobody on your side has time to verify.

1. CI pipeline health across the org

FOR PLATFORM AND DEVOPS TEAMS

Pipeline health today is visible one project at a time. The graph sees all of it at once.

- *"Which files correlate most with pipeline failures across our group?"*
- *"Which projects have the slowest or most retried pipelines this quarter?"*
- *"Where are failures concentrated: which jobs, which projects, which teams?"*

2. Blast radius / impact analysis

FOR TEAMS CHANGING SHARED CODE

Know what breaks before you change it, across every repo, not just the one that's open.

- *"Show me every caller of this function across all repos."*
- *"If we change this shared library, which services are affected?"*
- *"I'm deprecating this endpoint. Find all internal callers."*

3. Onboarding acceleration

FOR GROWING TEAMS AND ROTATING ENGINEERS

Help new engineers understand the codebase without months of reading.

- *"Give me a high level map of this system: key files, dependencies, call chains."*
- *"Where does a request enter this service and what does it touch?"*
- *"Who are the main contributors to this area and what changed recently?"*

4. Multi-repo dependency intelligence

FOR MICROSERVICE ORGANIZATIONS

Dependency mapping across hundreds of projects, beyond what anyone can track manually.

- *"Map the dependencies between the services in this group."*
- *"Which projects consume this internal package?"*
- *"Before we build a new service for X, do we already have something similar?"*

5. Find and deprecate legacy pipelines

FOR ORGANIZATIONS WITH THOUSANDS OF PROJECTS

Old pipelines accumulate for years. Finding them all is slow manual work.

- *"Which projects haven't had a successful pipeline in six months?"*
- *"Which projects still run this legacy job or stage?"*
- *"Rank our projects by pipeline activity so we can retire the dead ones."*

6. Vulnerability tracing and service maps

FOR SECURITY AND RESILIENCE TEAMS

Connect security findings to the code and services they actually affect, and build the service maps that disaster recovery and migration planning need.

- *"Which projects have open critical vulnerabilities, and what depends on them?"*
- *"Trace this finding to the services and owners affected."*
- *"Build a service map of this group for our DR planning."*

Orbit Design Partner Program · Orbit is in Beta · Bring your own use case: describe it in the application and we will build the test plan with you.