

Onboarding Guide

Everything your team needs to go from zero to first query. Most partners complete this in under an hour.

Before you start

- Your organization has a **Premium or Ultimate** subscription.
- You have **Owner access** on the top-level group you want to index.
- An MCP/CLI compatible agent such as Claude Code or Codex for external agent testing.

Self-Managed or Dedicated? Steps 1 and 2 below describe the GitLab.com self-serve path. Self-Managed deployment is documented at docs.gitlab.com/orbit/self-managed; follow it to stand Orbit up in your environment, then pick this guide up at Step 3 once your graph is live. Dedicated deployments are operated by GitLab.

Step 1 - Enable Orbit on your group

Turn it on

A top-level group Owner visits gitlab.com/dashboard/orbit/explore and enables Orbit for the namespace. Enablement is at the top-level group level and covers all projects within it.

Indexing starts automatically

Indexing begins as soon as you enable and typically completes quickly. You can watch progress from the Orbit dashboard.

Indexing covers the whole namespace. Excluding individual projects within a group is not supported today, so pick the group you enable deliberately.

Step 2 - Each developer opts in

User preferences

Every developer who wants Orbit-powered answers goes to gitlab.com/-/profile/preferences, finds the Orbit in GitLab Duo section, checks **Use Orbit in GitLab Duo**, and enables the sub-features their team uses.

Step 3 - Verify it's working

- Check indexing status on the Orbit dashboard for your group.
- Ask a graph-shaped question in Duo and/or your external agent, for example "which projects in our group had the most failed pipelines this month?" A grounded, specific answer means the graph is live.

Step 4 - Connect an external agent

Get started with the glab CLI

The `glab` CLI is the recommended way to give agents like Claude Code or Codex access to Orbit: authenticate `glab` once and your agent can explore the graph schema and run queries from the command line. Setup is copy-paste from the docs: docs.gitlab.com/orbit/remote/access/glab.

Or connect over MCP

Secondarily, any MCP/CLI compatible agent can connect through Orbit's MCP server for schema exploration and graph queries: docs.gitlab.com/orbit/remote/access/mcp.

Query results always respect the authenticated user's existing GitLab permissions. There is no privilege escalation through the graph.

Step 5 - Run your first real queries

Open the [Starter Prompt Pack](#) and run the six scenarios against your own codebase. Then move to the three business use cases you committed to in your application.

If something doesn't work

- Bring it to your weekly sync, or flag it to your program contact directly. We respond within one business day.
- Indexing gaps, query errors, and surprising answers are exactly the feedback we want. Screenshots and the query you ran are enough.