



Native AI across the entire software
development lifecycle

Ricardo Amarilla
Senior Solutions Architect



There has been an
explosion of
AI coding agents
entering the market...



Balancing speed and security is hard

Markets are competitive

66%

of companies are releasing 2x fast or faster

**GitLab DevSecOps Report*

Speed introduces risk

55%

Security teams say vulns are discovered after code is merged

**GitLab DevSecOps Report*

Apps are causing breaches

8/10

Top breaches last year stem from attacks on the application layer

**CrowdStrike State of Application Security Report*



The Problem

**AI is now a must have
for organizations
to compete and hire
top talent**

78%

of organizations said they
are currently using AI in
software development or
plan to in the next 2 years

(Up from 64% in 2023)



The Problem

The problem isn't code creation

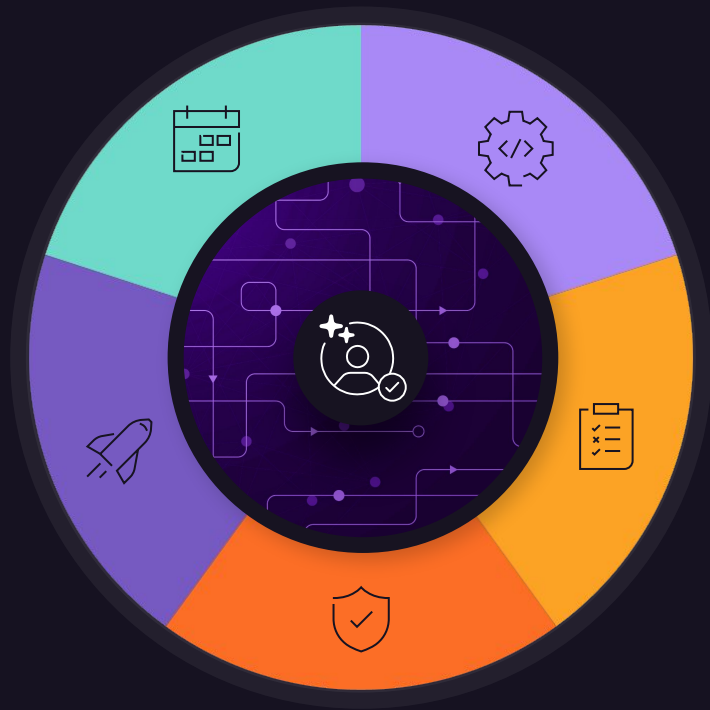
AI coding agents can boost the volume of code created however the overall efficiency is hindered by other software development steps that AI agents do not understand.



The Solution

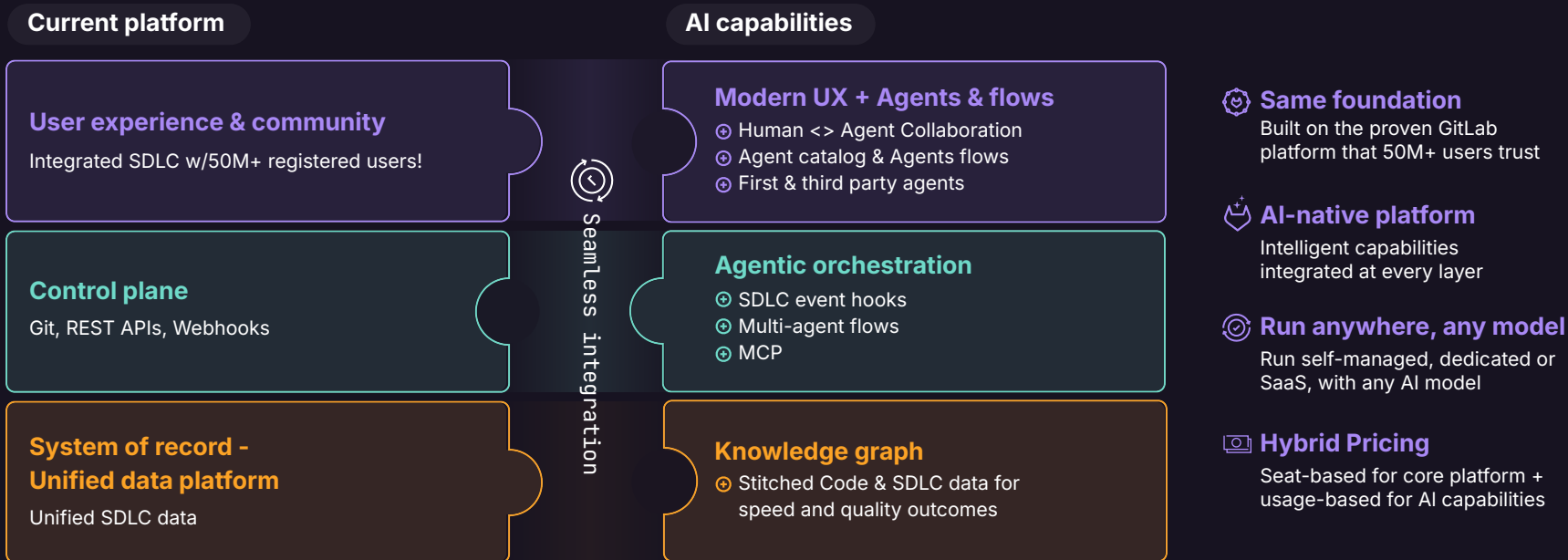
AI agents need full project context

AI agents that have full context of the software project, from planning to coding, building to securing, and deploying to monitoring, can boost efficiency across the SDLC unlocking delivering secure software faster.



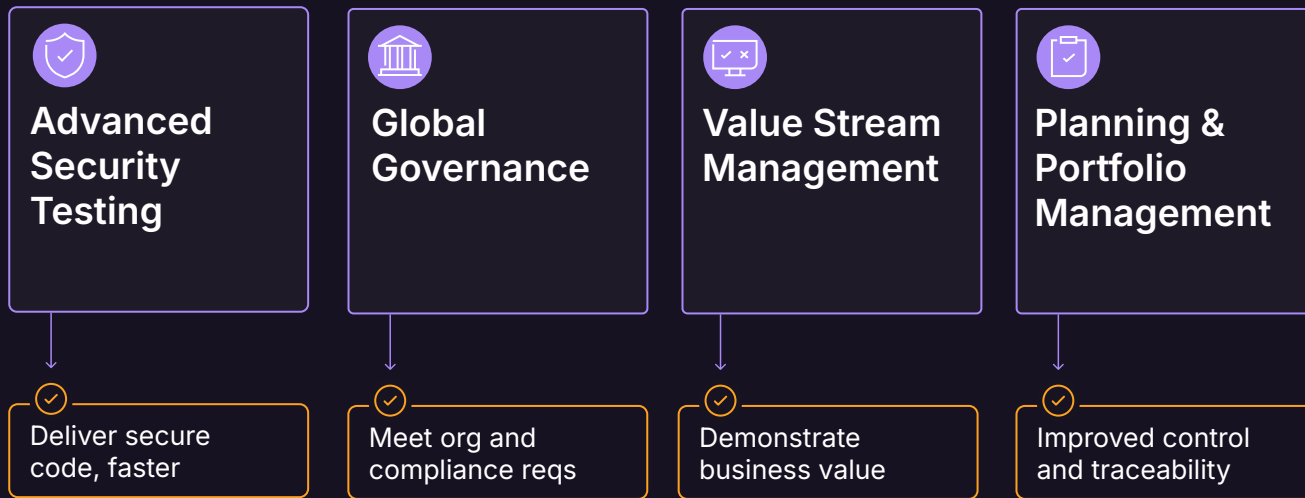
GitLab AI native transformation

Agentic orchestration across the software lifecycle



Security by Design

Organization wide security, compliance and planning



AI-Assisted Workflows

Ship more secure software faster with AI

Enterprise Agile Planning Add-On

One workflow for everyone involved in the software development lifecycle

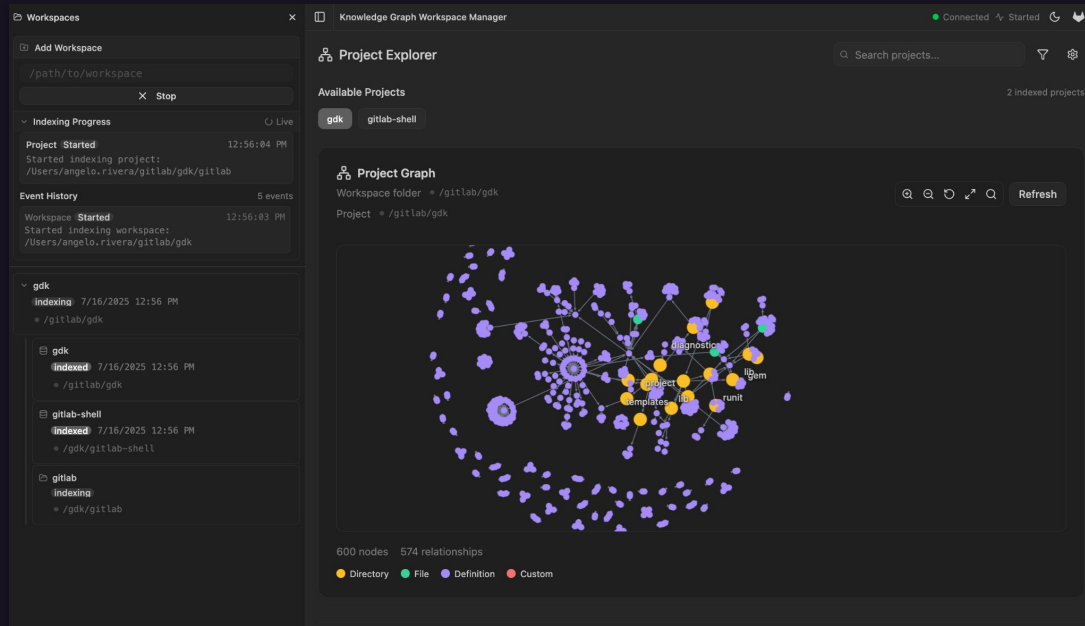


Knowledge Graph w/ visualizations

Map files and dependencies across the codebase, increase accuracy and speed of AI agents

Q3 (Public beta*) / Q4 (GA)

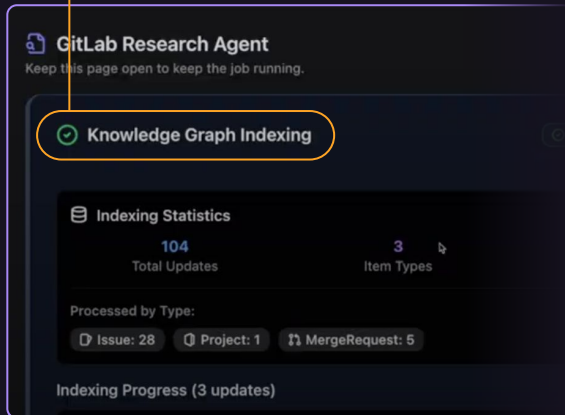
- ✔ Allows Duo agents to quickly understand relationships across development environment
- ✔ Unlock faster, more precise answers to complex questions



Deep research leveraging Knowledge Graph

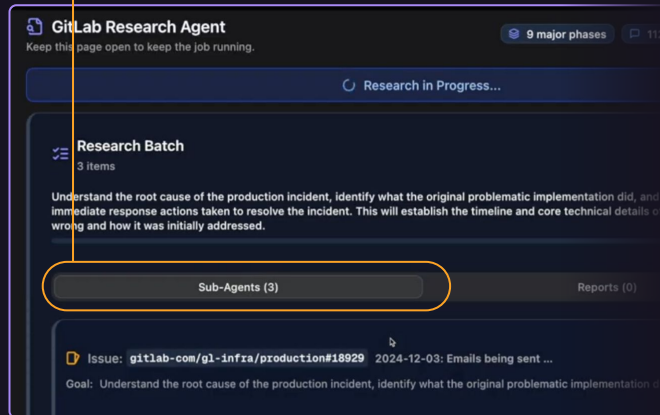
- **GitLab has your engineering context in one place**

Knowledge Graph maps this context, and is updated in real time to incorporate changes.



- **The value of agent orchestration**

The flow orchestrates multiple research agents, which working in parallel while humans pursue bigger challenges.



Agentic Chat

- Powered by Workflow Engine
- Multi-Source Search - Finds and update issues, MRs, files across projects
- File Operations - Creates/edits/reads multiple files locally
- Cross-Platform - GitLab UI, VS Code, JetBrains, Visual Studio
- Custom Rules - Project-specific instructions via `.gitlab/duo/chat-rules.md`
- MCP Support - Connects to external data sources and tools
- Git & Shell - Commits/pushes (UI), shell commands (IDEs)
- Context Synthesis - Combines multiple sources for answers
- Example [Video](#)

 do i have any open jira issues?

Tool `atlassian_getAccessibleAtlassianResources` requires approval. Please confirm if you want to proceed.

Duo would like to execute a tool. Do you want to proceed?

 `atlassian_getAccessibleAtlassianResources`

Request

No parameters will be sent with this request.

 Deny

 Approve



Evolving from single agents to agent orchestration

Key components to know

Agents

Mirrors familiar team roles. Collaborate them like you would other colleagues

- **Specialists**
Ex: software testing agent
- **Works with tools (GitLab objects)**
Ex: create_merge_request
- **Can invoke other agents**

Flows

Orchestrates multiple agents, guiding them in completing complex tasks

- **Steps sequentially through phases**
- **Started by triggers**
Ex: "create a merge request from the sql_change branch..."

GitLab Intelligent DevSecOps Platform



Agent



Agent



Agent



Agent



Agent



Agent

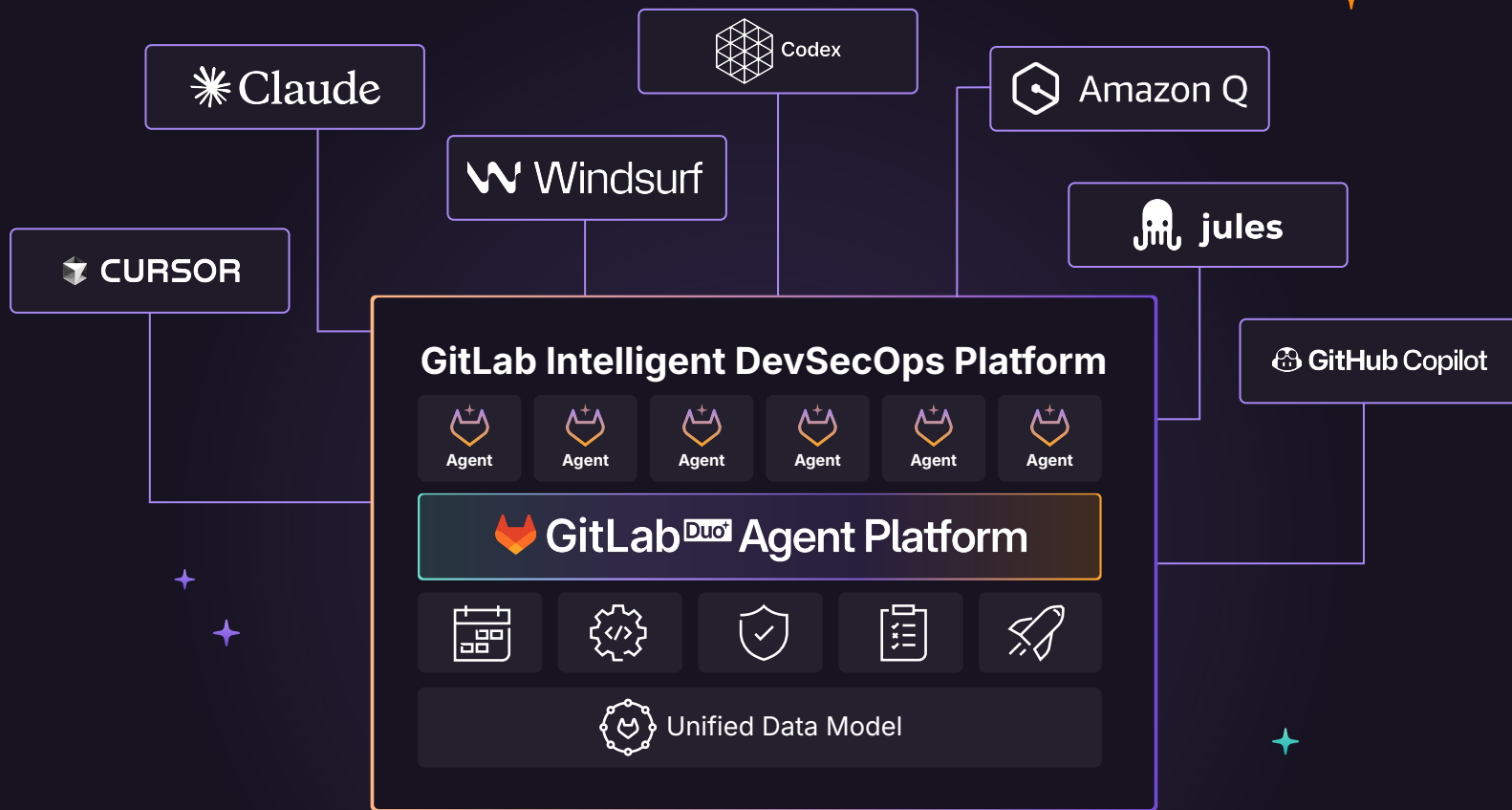


GitLab Duo Agent Platform



Unified Data Model



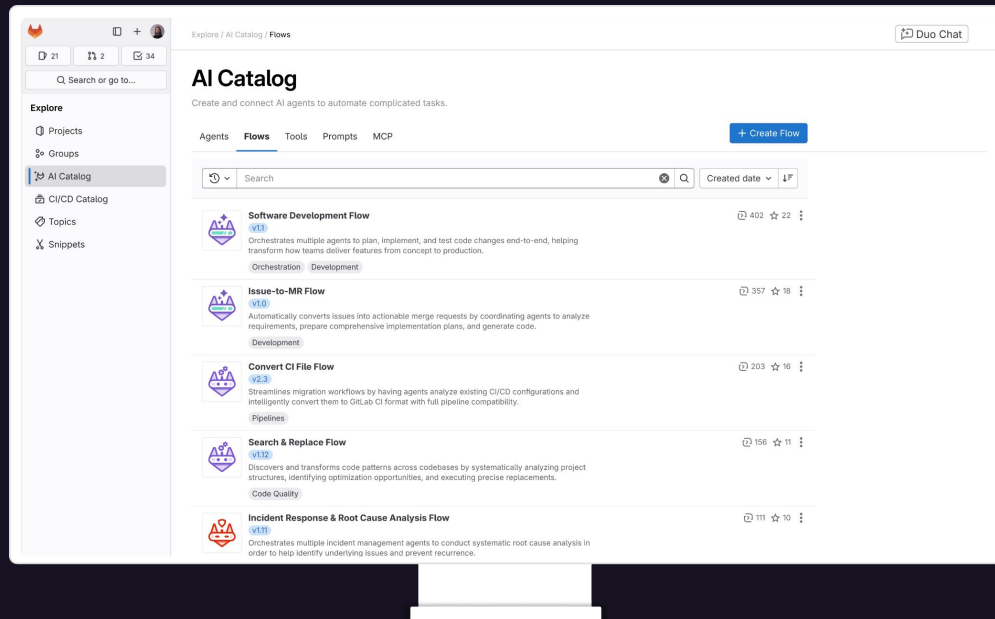


Why GitLab Duo Agent Platform

Duo AI Catalog

Agents can be shared via a centralized, discoverable AI Catalog. This catalog includes community publishing tools as well as a public marketplace where you can share and use custom agents created by the broader GitLab community.

Agents within the AI Catalog can be shared with customizable options or locked to specific tasks or skills.





AI across the SDLC

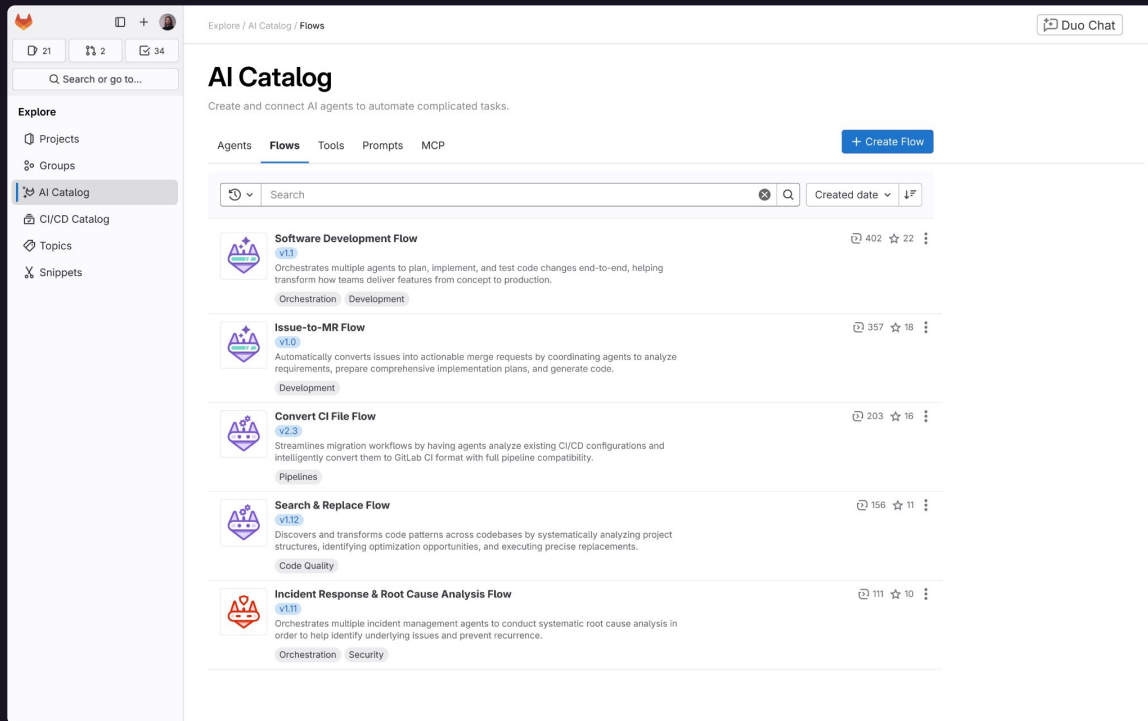
 **GitLab Duo** Agent Platform

AI Catalog

Extend Duo Agent Platform with
out-of-the-box agents and Flows

Q3 (Public beta*) / Q4 (GA)

- ✓ Share and use agents and Flows within your organization
- ✓ Create custom agents and flows

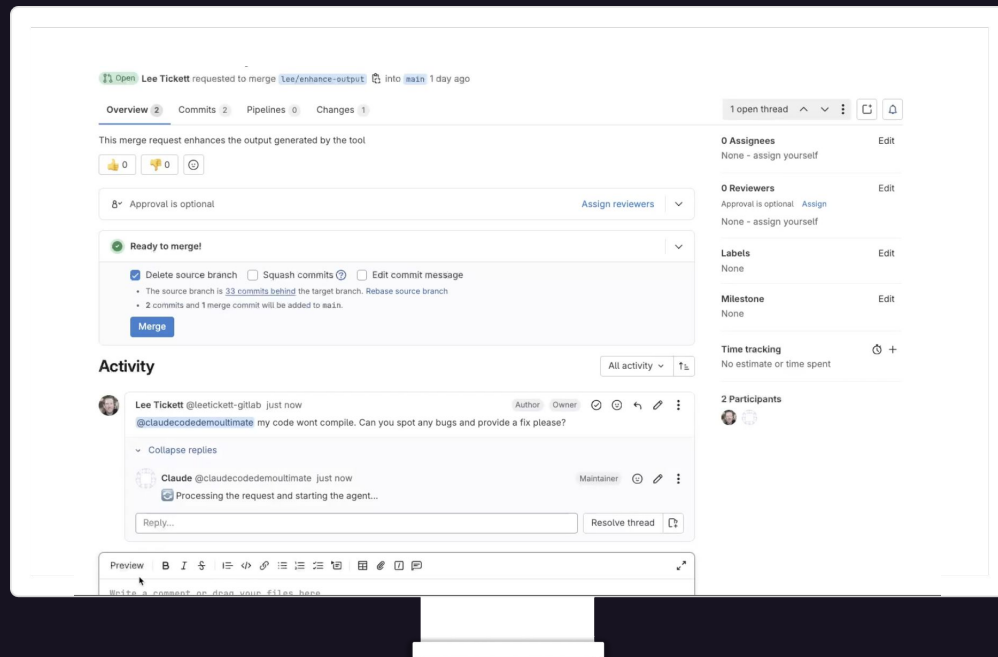


CLI Agent Support

Public beta (18.3) / GA Q4

✓ **Customer problem:** Teams waste time on routine development tasks and lack efficient ways to leverage AI assistance within their existing GitLab workflows, forcing context switching between different tools and interfaces.

✓ **Solution:** Native AI integrations that let developers @ mention AI assistants (Claude Code, Codex, OpenCode, Amazon Q, Google Gemini) directly in issues or merge requests, automatically providing context-aware code changes or comments while maintaining GitLab's permissions and audit trails.



[Demo Video](#)

Flows orchestrate agents to complete complex tasks

Example: Software Development Flow - Create an MR

User input

Triggers the Flow

I want you to create a merge request from the sqlchange branch

- Check which files are changed
- Assign kmoretal if any Go files changed - assign edmondchan if any database files changed
- any other code change assign to kmcknight you can extract their user IDs from our other merge request data

Software Development Flow

① Content building phase

Gather information needed to complete the request
↳ *Context Building agent*

② Clarification phase

Ask for clarification, if needed

③ Planning phase

Create a structured plan to complete the request
↳ *Planning agent*

④ Execution phase

Complete the actual merge request
↳ *Executor agent*

Result

Merge request assigned

✓ Merge Request Created Successfully

Merge Request #26: "SQL Changes: Update server port and query ordering"

- URL: https://gitlab.com/gi-demo-ultimate-eburugwall/eburugdemo/ai-demo/customer-analytics-dashboard-migration/-/merge_requests/26
- Source Branch: `sql_change`
- Target Branch: `main`

Files Changed Analysis

Based on the commit analysis, the following files were modified:

1. `main.go` (Go file)
 - Changed server port from `:8080` to `:9080`
2. `queries/analytics_queries.sql` (Database file)
 - Modified query ordering from `DESC` to `ASC` for `signup_date`

👥 Assignees Applied



Issue to MR Flow

 Generate MR with Duo

- GitLab UI
- CI Pipeline Execution allows scalable execution
- Issue-Driven - Analyzes issue description and requirements
- Draft MR Creation - Opens linked MR automatically
- Plan & Implement - Creates development plan and code structure
- Session Monitoring - Track progress via Automate > Sessions

Example Use Cases:

- Feature Addition: "Add a rating box for our users for Feedback" [Example](#)
- API Pagination: "Add pagination to GET /users endpoint with page and per_page params"
- Bug Fix: "Fix authentication error when session expires during checkout"
- Performance: "Implement caching layer for frequently accessed product data"
- Security: "Add rate limiting to login endpoint (5 attempts per minute)"
- Refactoring: "Extract email notification logic into separate service class"



Convert to GitLab CI

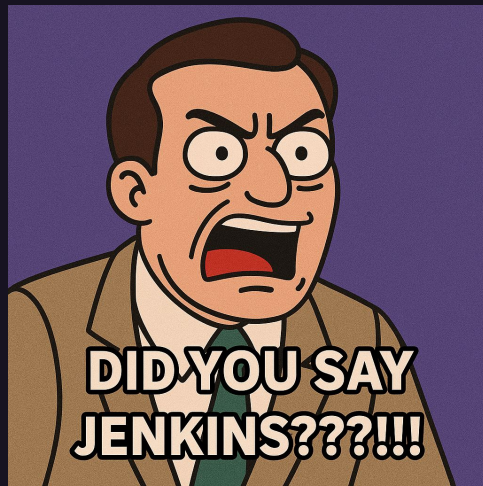


Convert to GitLab CI/CD Flow:

- GitLab UI Only - Triggered from Jenkinsfile view
- Jenkins Migration - Converts Jenkins pipelines to GitLab CI/CD
- Syntax Translation - Jenkins DSL to GitLab YAML
- Smart Mapping - Stages, triggers, artifacts, conditions
- MR Creation - Opens "Duo Workflow: Convert to GitLab CI"
- Plugin Guidance - Suggests GitLab equivalents for Jenkins plugins

Example Use Cases:

- Basic Pipeline: Convert multi-stage build/test/deploy Jenkins pipeline
- Parallel Jobs: Migrate Jenkins parallel execution to GitLab parallel jobs
- Conditional Deployments: Convert branch-based deployment conditions
- Build Artifacts: Translate artifact management and dependencies
- Complex Triggers: Convert cron, webhook, and parameterized builds
- Plugin Migration: Replace Jenkins plugins with GitLab native features
- [Demo Video](#) [Demo Repo](#)
- [JenkinsFile Examples](#) (for Demos)

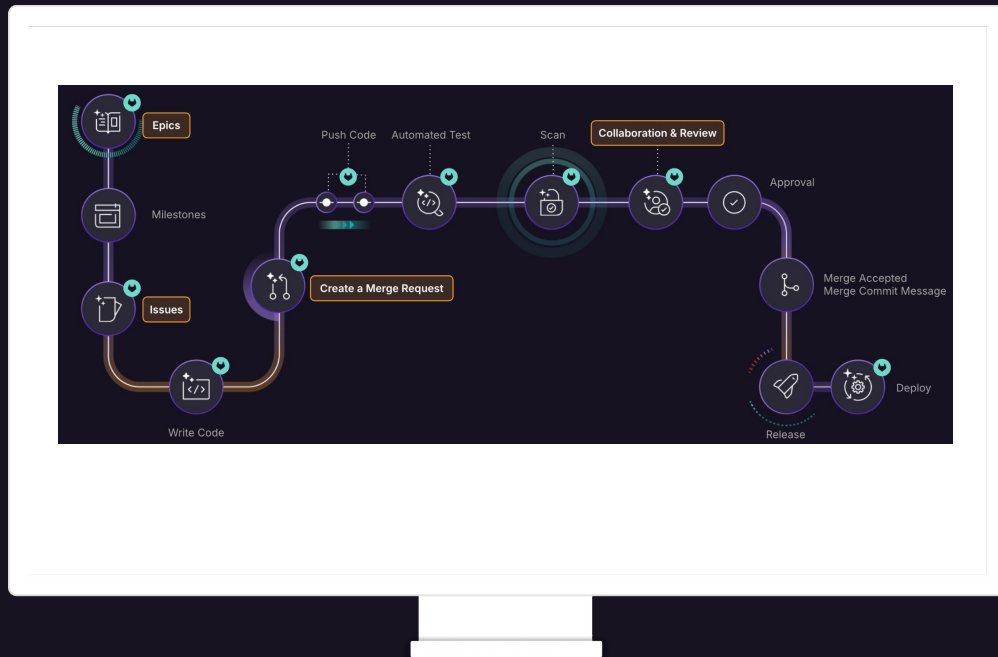




Fix Failed Pipelines Flow

Public beta (18.4) / GA Q4

- 🟢 **Customer problem:** When a pipeline fails, developers often lose valuable time investigating logs, identifying root causes, and implementing fixes. Even simple issues or configuration mistakes can still block progress, requiring manual troubleshooting and slowing delivery.
- 🟢 **Solution:** The Fixed Failed Pipelines Flow automatically detects failed jobs, analyzes the root cause, and generates a fix directly in a new commit and merge request.



GitLab Duo^{AI} Agent Platform

Customize agents and flows

Create, customize, and share agents and flows tailored to your organization

Q3 (Public beta*) / Q4 (GA)

- ✓ Create and edit agents and flows via GUI interface within Catalog
- ✓ Edit via declarative configuration file

Agent Orchestration

Duo Chat

AI Catalog

Create and connect specialized AI agents that automate complicated tasks.

Agents Agent Flows **Flow Builder** Tools

Agent Flow Builder

Name

Agent Flow

✦ Triggers

An agent flow starts running when a trigger condition is met.

Trigger @Duo assigned to a merge request

Manual Trigger Type /MR_doc_review in a merge review comment

+ Trigger

🧩 Agent Nodes

Nodes run sequentially.

1. Agent Search documentation

2. Agent Review MR changes

3. Agent Write documentation

4. Agent Create MR with changes

5. Agent Request review

+ Node

Configuration

Explain Verify setup

```
# workflow_definition.yaml
apiVersion: duo/v1
kind: Workflow
metadata:
  name: agent-workflow-here
  displayName: ""
  description: ""
  version: "1.0.0"
  tags: ["automation", "development"]

spec:
  # Trigger configuration
  triggers:
    - type: manual
      locations:
        - issue_sidebar
        - issue_quick_action # /duo-solve

    - type: event
      events:
        - issue.labeled
      conditions:
        - label: "duo:implement"

    - type: schedule
      cron: "0 9 * * MON" # Weekly on Mondays
      filters:
        - issue.label: "ready-for-implementation"

  # Input parameters
  inputs:
    issue_id:
      type: string
      required: true
      source: trigger.issue_id
```



DEMO



Agent Platform

A day in the life of a
developer with GitLab
Duo Agent Platform





Start shipping secure, better software faster

See what your team can do with the
intelligent DevSecOps platform.





Thank you