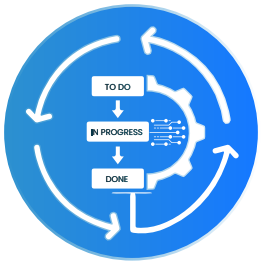
Workflow Bottleneck & Insight – Clovity AI Copilot for Jira



Introduction

Modern software projects often suffer from invisible delays. Work may appear to be “in progress” but is stuck in a review stage, waiting for approvals, or moving back and forth between statuses. Traditional Jira dashboards provide raw data, but they are difficult to configure and often fail to give teams clear, actionable insights.

The AI Workflow Analyzer addresses this challenge. Built as a Jira app, it integrates directly into Jira Cloud, automatically extracts issue changelogs, and analyzes workflows. The results are presented in a conversational, chat-style interface. This design ensures that project managers, team leads, developers, and executives can all ask questions in plain English and receive instant insights about workflow health.

The app helps you answer critical questions:

- How long do issues take to move from start to finish?
- Where do issues get stuck?
- What is the average time spent in each workflow stage?
- How many issues are being completed each month or sprint?
- Are bottlenecks improving or worsening over time?

System Capabilities

The AI Workflow Analyzer delivers five main categories of insights:

- **Cycle Time** – The total time an issue takes from creation to completion.
- **Time-in-Status** – The average time issues spend in each workflow stage.
- **Throughput** – The number of issues resolved in a given timeframe.
- **Bottleneck Identification** – Detection of statuses where issues spend excessive time.
- **Timelines** – A chronological breakdown of how issues moved across statuses.

These insights allow teams to **visualize process efficiency, identify delays,** and **improve predictability** in project delivery.

How the Application Works

The application is built on **Atlassian Forge,** which ensures secure communication with Jira. Here's the behind-the-scenes process:

- **Data Extraction** – The app fetches Jira project issues, including their changelogs .
- **Event Processing** – Status changes are extracted and ordered chronologically.
- **Timeline Construction** – Each issue gets a timeline showing when it entered and exited each status.
- **Aggregation** – Timelines are combined into project-wide metrics such as average cycle time and bottlenecks.
- **Result Delivery** – Results are returned to you in the chat interface and uploaded securely to a backend service for advanced analysis.

Accessing and Launching the Application

Prerequisites

- Jira Cloud access.
- At least one active Jira Software project.
- Application installed by your Jira administrator.
- Knowledge of project keys .

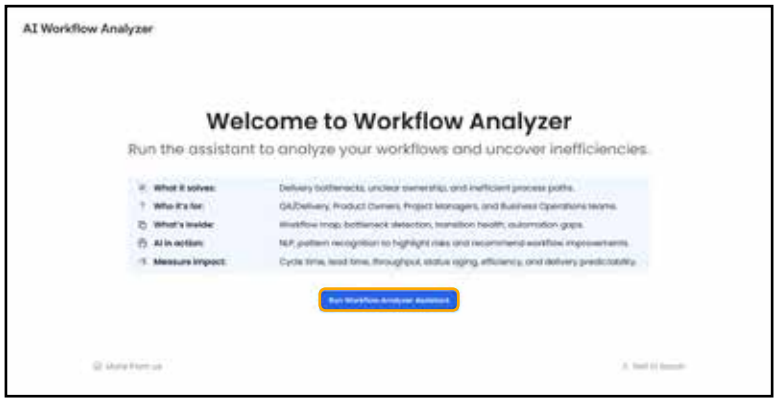
Launching

- Log in to Jira Cloud.
- From the left-hand **Apps sidebar**, click on **AI Workflow Analyzer.**
- The application will open with a **chat interface** where you can start entering queries.

Application Buttons

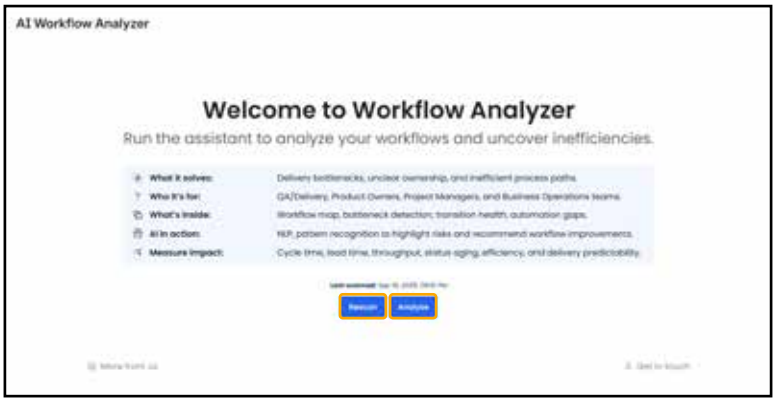
When you first open the application, the available options depend on whether you are using it for the **first time** or are a **returning user.**

- **Run Workflow Analyzer Assistant(first-time users)**



- Appears only when the app is launched for the first time on a Jira site.
- Clicking this button performs the initial scan of your Jira data.
- Once complete, the Rescan and Analyze buttons will be unlocked for ongoing use.

- **Rescan (returning users)**



- Refreshes and synchronizes Jira workflow data.
- Ensures the most current information is available for analysis.
- Displays the last scan time, so users know whether a new scan is needed.

- **Analyze (returning users)**
  - Opens the chat interface for workflow analysis.
  - Allows users to ask natural-language questions such as:
    - “What is the average cycle time?”
    - “Where do issues get stuck most often?”
    - “How many issues were completed in the past 30 days?”

The Chat Interface

The chat interface is designed to be intuitive and natural. Instead of navigating multiple dashboards, you simply type questions or instructions.

Basic Commands

- **List available projects** – Returns all Jira Software projects accessible to you.
- **Analyze a project** – For example, "Analyze project HRM for the past 60 days."
- **Ask targeted questions** – After running analysis, you can ask:
  - What is the average cycle time?
  - Where do issues get stuck?
  - How many issues were completed in the last 30 days?
  - Show me average time spent in each workflow stage.

Example Interaction

- You: "List projects"
- App: "Found projects: HRM, OPS, FIN"
- You: "Analyze HRM for the last 60 days"
- App: "Processing HRM... Analysis complete. Cycle time: 10.2 days. Bottleneck: In Review (4.5 days). Throughput: 45 issues completed."
- You: "Where do issues get stuck?"
- App: "40% of time is spent in the 'In Review' status."

Understanding the Results

When you run an analysis, the app provides three primary metrics:

Cycle Time

- **Definition:** The average number of days it takes for an issue to move from “To Do” to “Done.
- **Why it matters:** Shorter cycle times mean faster delivery. Long cycle times indicate delays.

Time-in-Status

- **Definition:** How long issues spend in each workflow status.
- **Why it matters:** Helps pinpoint where work is slowing down. For example, a long “In Review” stage suggests insufficient reviewers.

Throughput

- **Definition:** The number of issues resolved in the chosen timeframe.
- **Why it matters:** Reflects productivity and delivery volume.

Bottlenecks

- **Definition:** Workflow stages consuming the most time.
- **Why it matters:** Identifies where intervention is needed to accelerate delivery.

Issue Timelines

- **Definition:** A chronological record of how individual issues transitioned between statuses.

- **Why it matters:** Useful for detailed audits or explaining delays to stakeholders.

Role-Based Training

Project Managers

- **Objective:** Track delivery performance, plan better, and communicate with stakeholders.
- **How to use:**
  - Run 90-day analyses for high-level performance reviews.
  - Focus on cycle time and throughput for planning.
  - Use bottleneck reports to allocate resources where needed.
- **Example use case:** Discover that “In Review” delays are causing missed sprint goals, then assign additional reviewers.

Team Leads and Developers

- **Objective:** Improve daily workflow efficiency and remove blockers.
- **How to use:**
  - Run shorter analyses (30–60 days) for up-to-date insights.
  - Ask specifically about time-in-status.
  - Use bottleneck detection to highlight where tickets are waiting too long.
- **Example use case:** If tickets spend excessive time in “Blocked,” escalate issues quickly to management.

Agile Coaches and Scrum Masters

- **Objective:** Drive continuous improvement in team processes.
- **How to use:**
  - Analyze throughput trends to track delivery stability.
  - Monitor cycle time fluctuations between sprints.
  - Present bottleneck insights during retrospectives.
- **Example use case:** Show that average cycle time dropped by two days after adding daily stand-up reviews.

Executives and Stakeholders

- **Objective:** Gain quick, digestible summaries of project health.
- **How to use:**
  - Run a 90-day project analysis.
  - Focus on top-level metrics: cycle time, bottlenecks, and throughput.
  - Use results for reporting and decision-making.
- **Example use case:** Demonstrate that a project delivers 40 issues per quarter with a 12-day average cycle time, supporting investment decisions.

Best Practices

- Run analyses regularly — at least once per sprint.
- Use 30-day windows for operational insights, 90-day windows for strategic insights.

- Always compare multiple metrics together (cycle time, throughput, bottlenecks).
- Encourage teams to use the chat interface directly for transparency.
- Include results in retrospectives and quarterly reviews.

Troubleshooting

- “projectKey required” – Add the project key to your query.
- “cloudId required” – Contact your Jira admin to ensure configuration is correct.
- Analysis taking too long – Use shorter timeframes.
- Upload failed – Ensure the API key is configured correctly by your admin.

Security Considerations

- All communication runs through Atlassian Forge with strict authentication.
- Only issue metadata and workflow events are processed; sensitive data such as comments and attachments are excluded.
- Data is uploaded securely to Clovity’s backend for further processing.

Training Exercise

Here’s a hands-on training scenario to practice:

- Open the AI Workflow Analyzer.
- Ask the app to list available projects. Note down the keys.
- Pick one project and request a 60-day analysis.

- Review the summary. Write down the cycle time, bottleneck, and throughput.
- Ask a follow-up question: “Where do issues get stuck?”
- Based on the answer, discuss with your team how to address the bottleneck.

Conclusion

The AI Workflow Analyzer is more than just a reporting tool. It transforms Jira data into conversational insights, empowering every role in the organization to understand and improve workflows. By using it consistently, teams can:

- Deliver faster.
- Reduce bottlenecks.
- Improve predictability.
- Align stakeholders around transparent, data-driven insights.