CONTEXTBRIDGE FOR ATLASSIAN FORGE

# Incident-to-PR Context Graph & Rovo / MCP Copilot

Connecting Jira incidents, historical Confluence postmortems, pull requests, and service runbooks into a deterministic multi-hop Context Graph inside Jira, Rovo, Teamwork Graph, and external MCP clients.

### Apps for Software Teams

Jira Issue Panel with live SVG graph & blast radius KPIs

### Best Rovo Agent

4 grounded actions for lineage, drift audit & ADF RCA

### Best Atlassian MCP

Typed JSON-RPC 2.0 MCP server + companion `skill.md`

### Teamwork Graph

Cross-product ARI ingestion & decision lineage edges

# Incident Context Fragmentation Slows Every SEV-1 Response

Critical engineering context is scattered across four Atlassian & Git surfaces while responders race the clock.

### 1. Siloed Cross-Product Trails

During an outage, responders manually tab-hop between:

- Jira incident tickets ( `INC-402` )
- Recent GitHub/Bitbucket PRs ( `PR #1847` )
- Historical Confluence postmortems ( `PM-291` )
- Service ownership & Compass dependencies

### 2. Silent Runbook Drift

Runbooks rot as code evolves:

- Recovery docs still reference deprecated flags (e.g. `REDIS_SINGLE_NODE_URL` ).
- Engineers waste precious minutes following obsolete steps before discovering recent PR changes.
- No automated drift scoring links PRs back to Confluence runbooks.

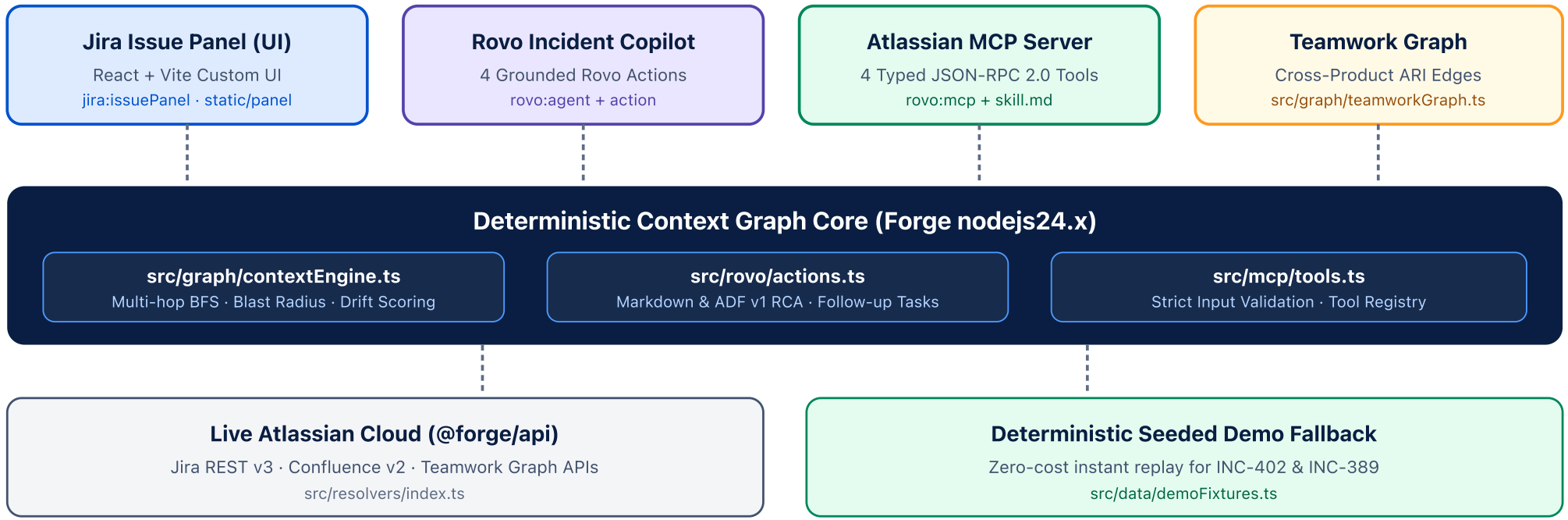
### 3. Ungrounded AI & Blank-Page RCAs

Generic LLM chats lack graph grounding:

- Risk of hallucinated issue keys or fabricated PR numbers.
- Postmortem RCAs are written from scratch hours later, losing causal decision lineage.
- External AI coding agents lack typed MCP access to incident context.

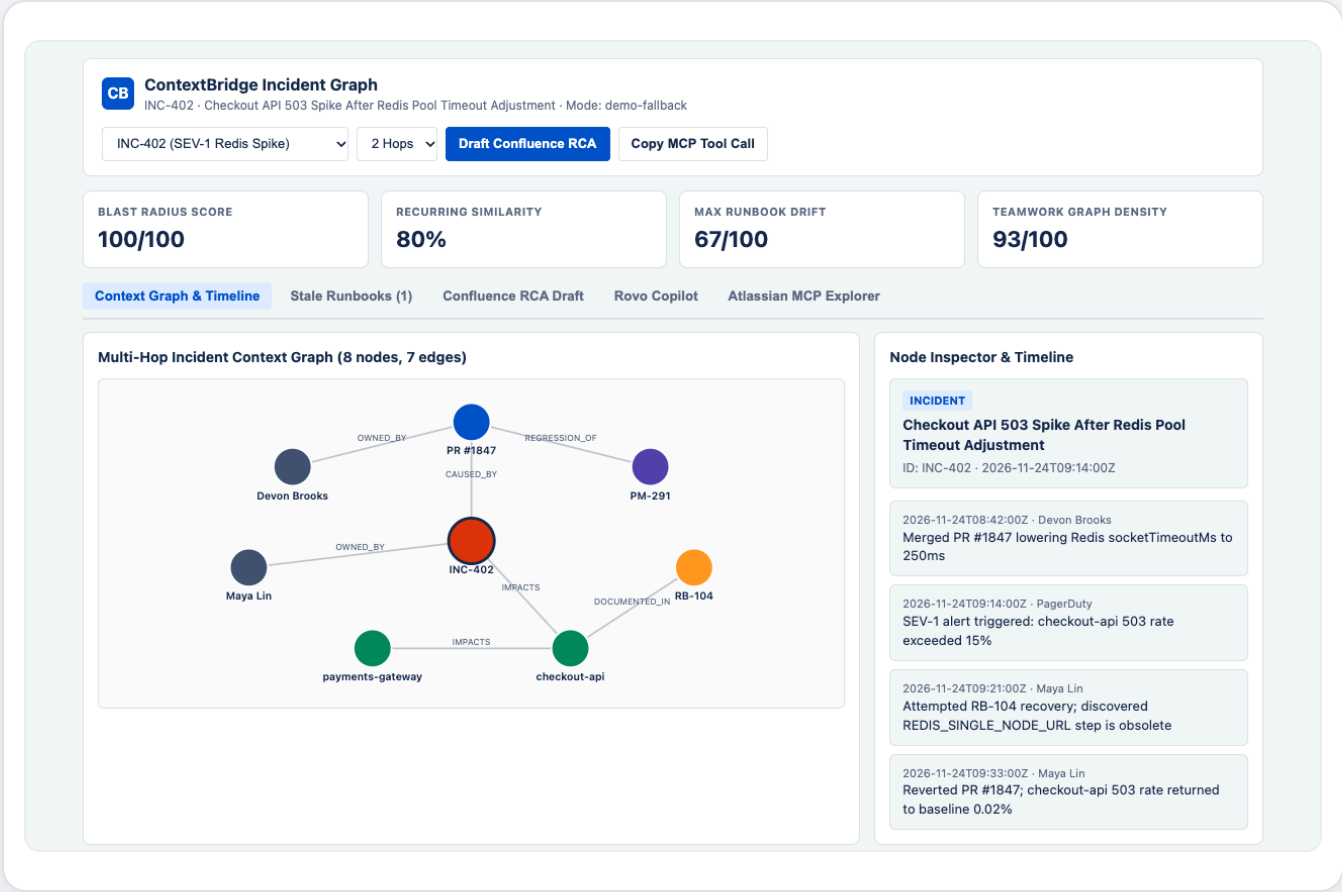
# Four-Layer Forge + Rovo + MCP + Teamwork Graph Architecture

Deterministic graph traversal powers every surface—Jira UI, Rovo Agent, Teamwork Graph, and external MCP clients—from one shared TypeScript core.



# Live Interactive Demo: 5-Tab Jira Issue Panel

Judges can explore the full Custom UI directly on GitHub Pages with zero backend setup.




### 1. Context Graph & Timeline

Interactive SVG node inspector (1–3 hops) + chronological outage timeline for `INC-402` and `INC-389`.

### 2. Stale Runbooks Detector

Flags `RB-104` (drift `67/100`) where `PR #1847` invalidated documented recovery steps.

### 3. Confluence RCA & Rovo Copilot

Generates blameless Markdown + ADF v1 RCAs and runs grounded Rovo starter prompts interactively.

### 4. Atlassian MCP Explorer

Executes all 4 typed MCP tools live in-browser and copies ready-to-use JSON-RPC 2.0 payloads.

# ContextBridge Incident Copilot ( `rovo:agent` + 4 Actions)

Strictly grounded in Context Graph nodes and Teamwork Graph edges—never fabricates issue keys or PRs.

### Agent Persona & Guardrails

- **Manifest Key:** `contextbridge-incident-copilot`
- **Grounding Rule:** Every response cites verified graph nodes, drift scores, and causal edges.
- **4 Built-in Starters:** Blast radius, stale runbooks, blameless RCA, and decision lineage.
- **Blameless Output:** Produces both Markdown and Atlassian Document Format ( `ADF v1` ) payloads ready for Confluence v2 APIs.

| ROVO ACTION KEY | VERB | GROUNDED OUTPUT |
|---|---|---|
| `get-incident-context-graph` | GET | Multi-hop nodes, edges, blast radius & similarity score |
| `detect-stale-runbooks` | GET | Runbooks exceeding drift threshold + concrete patch advice |
| `draft-confluence-rca` | CREATE | Structured blameless RCA in Markdown + ADF v1 |
| `create-followup-tasks` | CREATE | Prioritized Jira remediation tasks mapped to owners |

# Atlassian Forge MCP Server ( `rovo:mcp` ) & `skill.md`

Exposes incident intelligence to external AI coding agents via strictly typed JSON-RPC 2.0 tools.

| MCP TOOL NAME | KEY INPUTS & BEHAVIOR |
| --- | --- |
| `list_incident_context` | `issueKey` , `includeTeamworkGraph` → full graph + ARIs |
| `trace_decision_lineage` | `issueKey` , `targetType` → hop-by-hop causal chain |
| `audit_stale_runbooks` | `issueKey` , `driftThreshold` → ranked drift findings |
| `draft_postmortem_rca` | `issueKey` , `spaceKey` → Markdown + ADF v1 + tasks |

### Companion Open-Source `skill.md`

Ships with a portable agent skill defining the 4-step SRE investigation workflow:

1. Call `list_incident_context` to measure blast radius.
2. Call `trace_decision_lineage` for historical postmortems.
3. Call `audit_stale_runbooks` to catch PR-induced drift.
4. Call `draft_postmortem_rca` for grounded ADF & follow-ups.

# Verified Engineering Quality & Zero Paid LLM Overhead

Every module is covered by deterministic unit tests and headless browser QA.

VITEST SUITE

**21 / 21**

All 5 test suites passing (manifest, graph, Rovo, MCP, Teamwork Graph)

HEADLESS CHROME QA

**0 Errors**

0 console errors, 0 page errors, 0 failed network requests across all 5 tabs

SECURITY & HYGIENE

**0 Leaks**

Verified clean via `gitleaks` secret scan and strict TypeScript check

FORGE LLM COST

**$0.00**

Deterministic graph & template engine; zero paid external LLM API keys needed

## Test Coverage Breakdown

`tests/manifest.test.ts` (4) · `tests/contextEngine.test.ts` (5) · `tests/rovoActions.test.ts` (5) · `tests/mcpTools.test.ts` (5) · `tests/teamworkGraph.test.ts` (2)

# Codegeist 2026 Track Alignment & Live Links

Purpose-built for the primary Software Teams track and all three bonus tracks.

| CODEGEIST 2026 TRACK | CONTEXTBRIDGE IMPLEMENTATION | KEY ARTIFACT |
|---|---|---|
| **Apps for Software Teams** (Main) | Jira Issue Panel with interactive multi-hop SVG graph, timeline & drift KPIs | `static/panel/src/App.tsx` |
| **Best Rovo Agent** (Bonus) | ContextBridge Incident Copilot with 4 grounded actions & ADF v1 RCA | `src/rovo/actions.ts` |
| **Best App with Atlassian MCP** (Bonus) | Forge `rovo:mcp` JSON-RPC 2.0 server + 4 typed tools + `skill.md` | `src/mcp/tools.ts` |
| **Best Teamwork Graph Integration** (Bonus) | Cross-product ARI connector linking Jira, Confluence, PRs & Compass | `src/graph/teamworkGraph.ts` |

**Public GitHub Repository**
https://github.com/alphonshhhomenicmtu/contextbridge-codegeist

**Live Interactive Demo (GitHub Pages)**
https://alphonshhhomenicmtu.github.io/contextbridge-codegeist/



MIT License · Built on Atlassian Forge