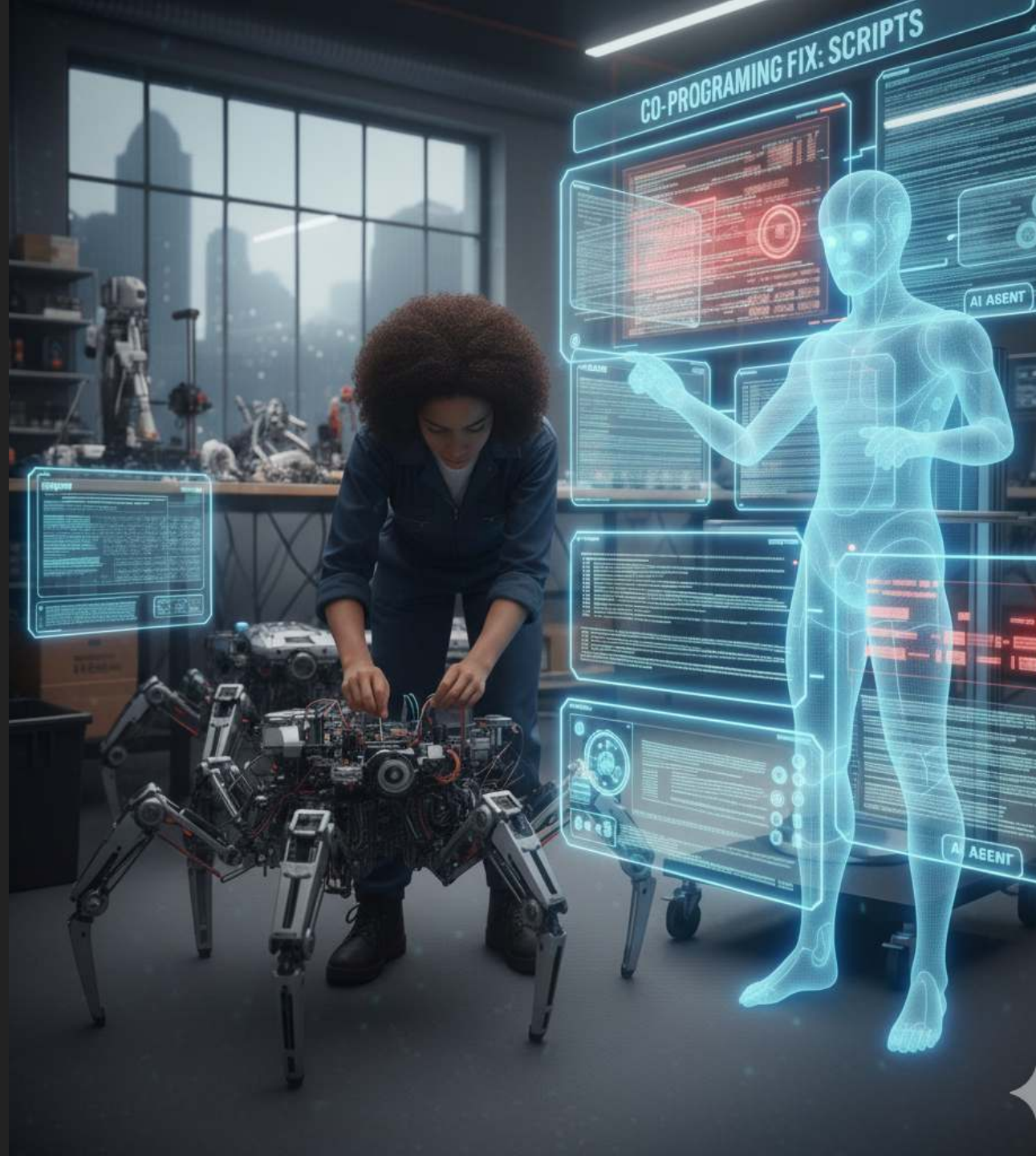
Agentic Robotics Engineering

We're building the first Agentic AI that lives in your robotics stack.

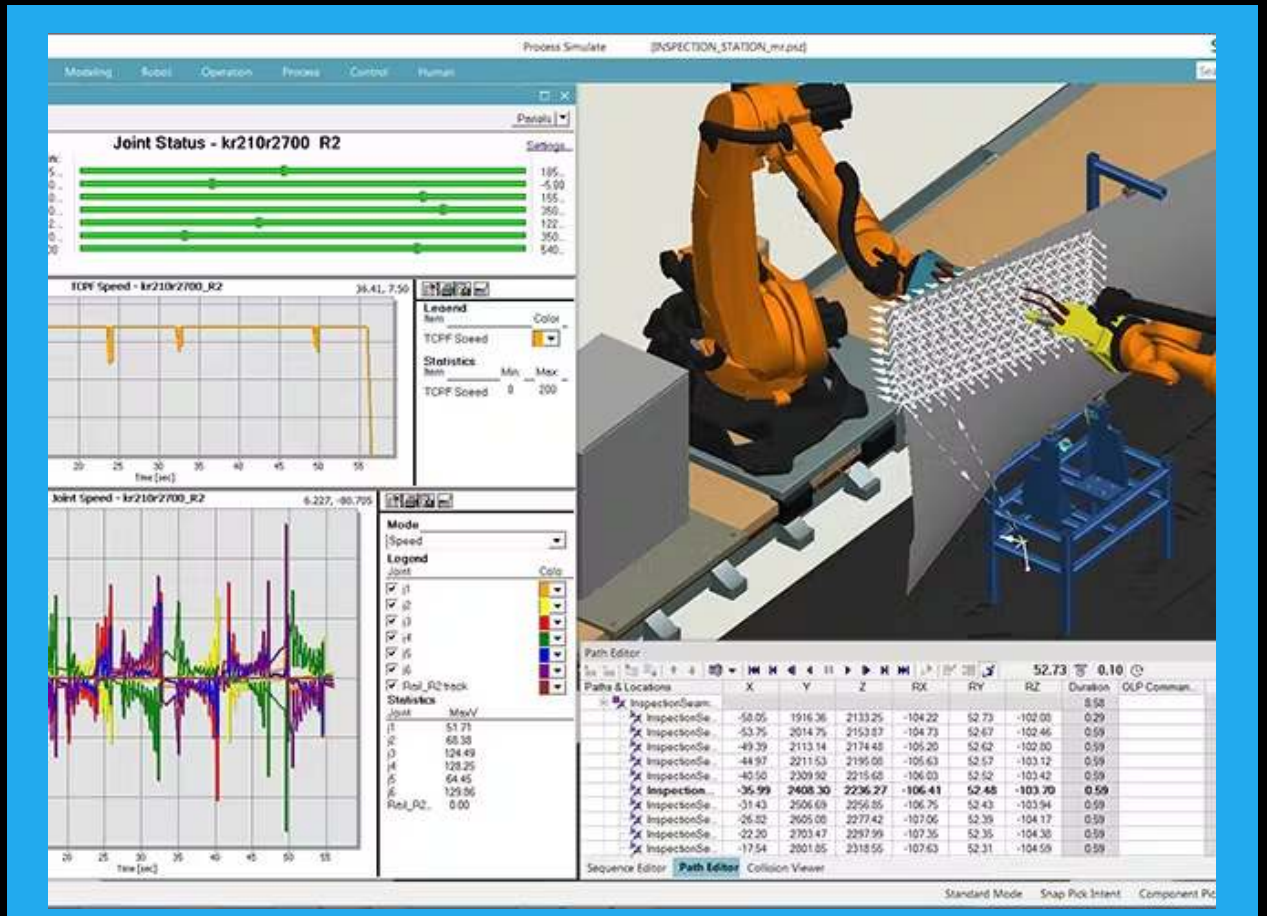
Temp AI

Your friendly agentic co-robotics engineer observing your stack to detect, explain, predict, and resolve failures in real time.



The Problem

Robotics development is becoming increasingly complex, with modern systems consisting of hundreds of interdependent nodes, topics, and services. When a robot fails, engineers often spend hours or even days manually sifting through logs, monitoring topics, and reproducing errors to find the root cause, which is a time consuming and expensive process. Often having to rely upon the intuition of experienced robotics engineers.




```
1607700411.4432 [DEBUG] Time jumped forward by [0.021882] for timer of period [0.0
1607700411.4432 [DEBUG] Time jumped forward by [0.021880] for timer of period [0.0
1607700411.4432 [DEBUG] Time jumped forward by [0.021880] for timer of period [0.0
1607700411.4434 [DEBUG] Error transforming from frame 'base_link' to frame 'map':
1607700411.4747 [DEBUG] Time jumped forward by [0.021540] for timer of period [0.0
1607700411.4748 [DEBUG] Time jumped forward by [0.021543] for timer of period [0.0
1607700411.4748 [DEBUG] Time jumped forward by [0.021546] for timer of period [0.0
1607700411.4749 [DEBUG] Error transforming from frame 'base_link' to frame 'map':
1607700411.5067 [DEBUG] Time jumped forward by [0.021968] for timer of period [0.0
1607700411.5067 [DEBUG] Time jumped forward by [0.021977] for timer of period [0.0
```

The Solution

Agentic AI system that serves as the intelligent watchdog for your ROS architecture. It moves beyond simple alerts to provide deep root cause analysis and allows your developers to converse with the system to solve problems faster than ever.

Codebase Context-Aware Real-time System Diagnosis

Our platform moves beyond simple metrics by integrating directly with your codebase to understand the *intent* behind every node. It provides a holistic, real-time diagnosis of your entire robotics architecture, knowing not just *what* failed, but also what that component was *supposed* to be doing.

Conversational System Cartography

Transform your complex system into an interactive map. Our agent allows any developer to ask plain-language questions like, "Show me the data flow for the navigation stack." It instantly generates explanations and diagrams, drastically reducing onboarding time and serving as living, always-updated documentation.

Intelligent Fault Cascade Analysis

This is our AI detective. It analyzes the stream of system-wide data to uncover the true origin of a failure. By connecting seemingly unrelated events—a minor sensor lag, a brief CPU spike, and a warning log—it traces the entire chain reaction back to the root cause, distinguishing the symptom from the source.

Error Prompting, Conversational Deep-Dive into Problems

Turn any error report into an interactive dialogue. Once an issue is flagged, developers can ask our AI follow-up questions to rapidly explore the context around the failure. This transforms a multi-hour manual investigation into a focused, minutes-long conversation to pinpoint the solution.

How It Works

Architectural Blueprinting Engine

Our system begins with a deep, static analysis of your source code. Using advanced parsing techniques to deconstruct your codebase to create a foundational blueprint. This blueprint meticulously maps out every key element: ROS nodes, their publishers and subscribers, service definitions, and critically the context of developer-written logs, which reveal the intended state of the application.

Dynamic Interaction Graph Synthesis

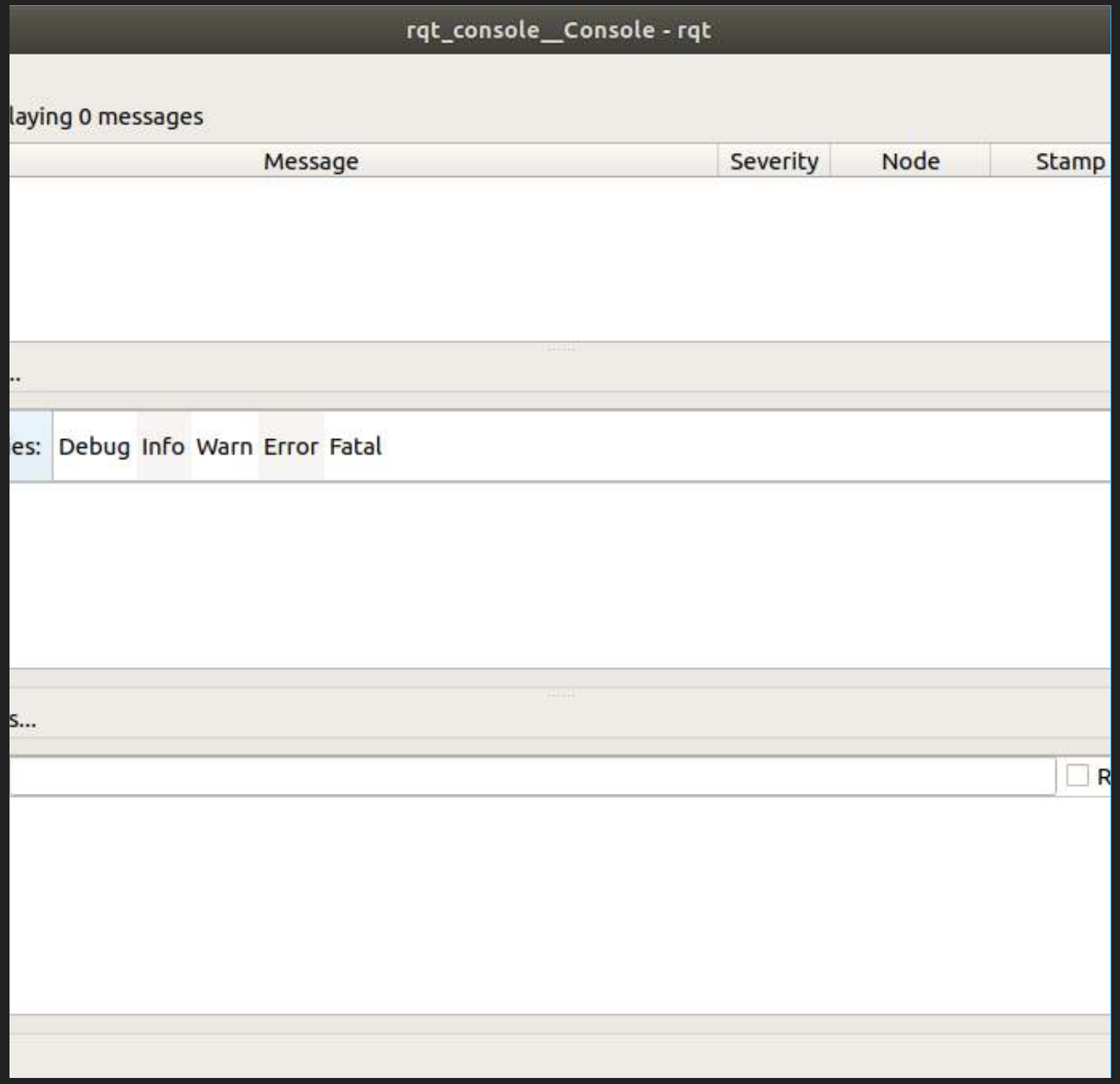
This architectural blueprint is then synthesized into a dynamic interaction graph. This isn't just a static diagram; it's a machine-readable model of your system's nervous system. The graph maps ROS nodes as vertices and the flow of data through topics and services as directed edges, creating a comprehensive and digestible model of all potential component interactions.

The Semantic Reasoning Layer

Here, our first layer of intelligence interprets the interaction graph to understand the system's *intended functionality*. It infers the purpose of each node and the significance of their relationships. Most importantly, it uses this understanding to strategically identify the most critical topics and developer-defined logs to monitor for diagnosis, while intelligently filtering out the noise from the underlying ROS 2 framework.

The Diagnostic Inference Engine

Our Diagnostic Inference Engine takes the crucial context and monitoring strategy from the Semantic Layer and fuses it with a real-time window of live log and message data from the running system. By cross-referencing this live data against the system's known architecture and intent, it performs a rapid root cause analysis, delivering a precise and actionable explanation of the failure.



The MVP

Our initial product is a powerful, AI-driven log monitoring and anomaly detection tool. It securely ingests all `rosout` streams from your ROS system, uses advanced Natural Language Processing (NLP) models to understand the intent and context of log messages, and applies anomaly detection techniques to flag unusual patterns that precede a failure.

Market Opportunity

- **Total Addressable Market (TAM)**

  The Global Robotics Market, valued at about $50 billion in 2025 with an estimated CAGR of about 14%. The market is estimated to be about $111 billion by 2030.

- **Serviceable Addressable Market (SAM)**

  The market for robotics software, currently is placed at about $20 billion and is growing with a CAGR of 22.4%. With long term goal to build generalisable RobOps (DevOps for Robotics) platforms that integrates realtime AI solutions, every robotics software platform can be a potential target.

- **Serviceable Obtainable Market (SOM)**

  Our initial target is high-growth startups and enterprise R&D labs in logistics, automation, and agriculture robotics, representing a multi-billion dollar entry market. Post market acquisition of such entities we plan to develop scaled RobOps solutions.

Business Model

SaaS for Robotics Teams

We will use a tiered subscription model (SaaS):

Developer (Free)

For individual developers and academic use. Basic log analysis for a single machine.

Team ($/seat/month)

For professional teams. Includes advanced real-time monitoring, integrations (Slack, Jira), and historical data analysis.

Enterprise (Custom Pricing)

For companies deploying fleets of robots. On-premise deployment options, fleet-wide analytics, predictive maintenance alerts, and premium support.

The Team

Aditya Kovilur

MS Computer Science @ Texas A&M
Prev, researcher @ High Performance Computing Lab
Undergrad, IIT Madras

Aayush Agrawal

MS Robotic Systems Development @ CMU
Prev, Co-founder @ TelebortiX
Undergrad, IIT Madras

Mehul Menon

MS Robotics @ CMU
Prev, researcher @ NUS
Undergrad, NIT Durgapur

In summary, our AI-powered ROS diagnostics platform presents a transformative solution to the pressing challenges facing robotics development. By providing real-time, context-aware insights, we empower engineering teams to focus on innovation rather than spend countless hours debugging complex systems. With a massive market opportunity and a talented team of robotics experts, we are poised to lead the future of intelligent robotics monitoring. Join us in building the AI co-pilot that every robot deserves.