Towards Self-Healing Software: Understanding Failures with AI Agents

Marcello Martini

Politecnico di Milano / NECSTLab

Who I Am

Research interests

- **Agentic AI & self-healing**
- **Cloud-native failure diagnosis**
- **System & operational knowledge**

Education

`2026-now` **PhD in Information Technology**
Politecnico di Milano / NECSTLab

`2026` **MSc Computer Science & Engineering**
Politecnico di Milano / Artificial Intelligence

`2023` **BSc Electronic & Computer Engineering**
University of Ferrara



Marcello Martini

PhD student
Politecnico di Milano / NECSTLab

marcello.martini@polimi.it

marcellomartini.tech

Why is failure diagnosis difficult?

01

Distributed complexity

Symptoms and causes can span multiple services and dependencies.

02

Cognitive load

Engineers must connect logs, metrics, and traces to test possible causes.

03

Diagnosis bottleneck

Collecting and interpreting evidence takes time during incident response.

More telemetry does not automatically make failures easier to explain.

[1] Google SRE Book, "Being On-Call", ch. 11. [2] J. Lewis and M. Fowler, "Microservices", 2014.

POLITECNICO MILANO 1863
POLITECNICO MILANO 1863 NECST laboratory

AIOps aims to streamline IT operations

The application of AI to streamline IT environments,
shifting from manual monitoring to data-driven incident management. [3]

Observe

Collect and correlate operational telemetry.

→

Diagnose

This talk: diagnose failures.

→

Recover

Long-term goal: guide safe recovery.

[3] IBM, What is AIOps? [4] Chen et al., AIOpsLab (2025). [5] Wang et al., EviRCA (2026, preprint).

How does agent architecture shape diagnostic effectiveness and investigation cost?

Compare agent workflows for Kubernetes RCA, keeping GPT-5-mini fixed.

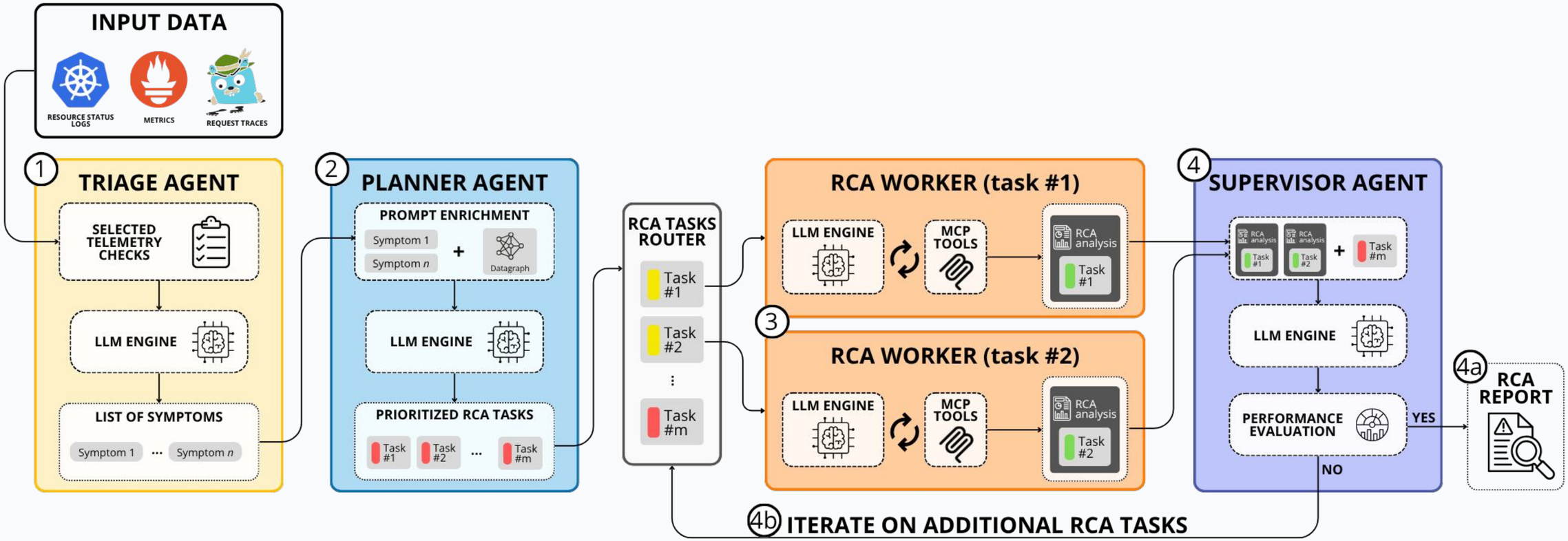
Diagnostic effectiveness

Detection · Localization
False alarms

Investigation cost

Tokens
Execution time

F — Reference workflow



[6] S. Yao et al., ReAct, arXiv:2210.03629.

O — Agent-directed evidence collection

Compared with F: change how the initial evidence is collected.

Triage

ReAct agent chooses evidence tools

CHANGED: COLLECTION

→

Planner

Topology context
LLM task planning

Same stage as F

→

Workers

Parallel ReAct investigations

Same stage as F

→

Supervisor

Synthesize findings
Report or continue

Same stage as F

Design choice Let the agent choose evidence rather than prescribe the checks.

POLITECNICO MILANO 1863
POLITECNICO MILANO 1863 NECST laboratory

P — Symptom-driven tasks, no Planner LLM

Compared with F: change task construction and initial dispatch.

Triage

Programmed checks
LLM symptoms

Same stage as F

→

Planner

One task / symptom
No Planner LLM

NO DEPENDENCY ENRICHMENT

→

Workers

All tasks start
in the first batch

CHANGED: DISPATCH

→

Supervisor

Synthesize findings
Report or continue

Same stage as F

Design choice Investigate symptoms directly rather than create a joint, topology-enriched plan.

POLITECNICO MILANO 1863
POLITECNICO MILANO 1863 NECST laboratory

N — No topology supplied to Planner

Compared with F: remove supplied Planner dependency context.

Triage

Programmed checks
LLM symptoms

Same stage as F

→

Planner

Evidence-based plan
No topology input

CHANGED: CONTEXT

→

Workers

Parallel ReAct
investigations

Graph tools retained

→

Supervisor

Synthesize findings
Report or continue

Same stage as F

Design choice Supply topology to the Planner or leave it to investigation-time tool access.

POLITECNICO MILANO 1863
POLITECNICO MILANO 1863 NECST laboratory

M — Single ReAct investigator

Architecture contrast: one investigation history instead of four stages.



Collect evidence → investigate → report
One agent, one investigation history

Design choice

Use one investigator rather than divide work among specialized agents.

How we test the agents

MODEL

GPT-5-mini

Diagnostic model fixed

APPLICATIONS

3

Hotel / Social / Astronomy

FAULT SCENARIOS

29

One execution per cell

CONTROLS

3 / variant

No-fault scenarios

Test environment

AIOpsLab
Deploys apps; injects faults

KinD / Kubernetes
Local cluster

Agent execution

LangGraph
Agent orchestration

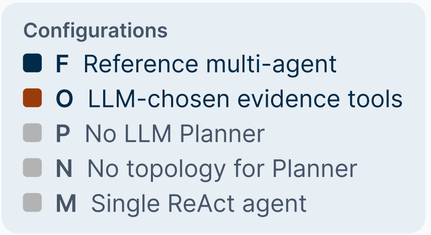
MCP tools
Prometheus / Jaeger / Kubernetes API

Run tracing

LangSmith
Agent run traces

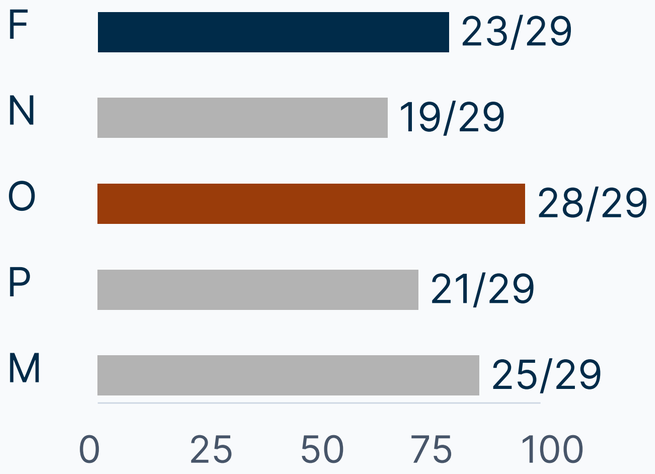
[4] Y. Chen et al., AIOpsLab, MLSys, 2025.

More detections do not guarantee better localization



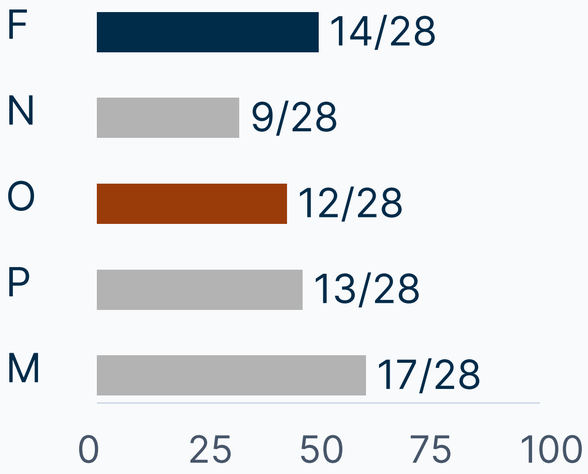
Fault detection

Positive verdicts / 29



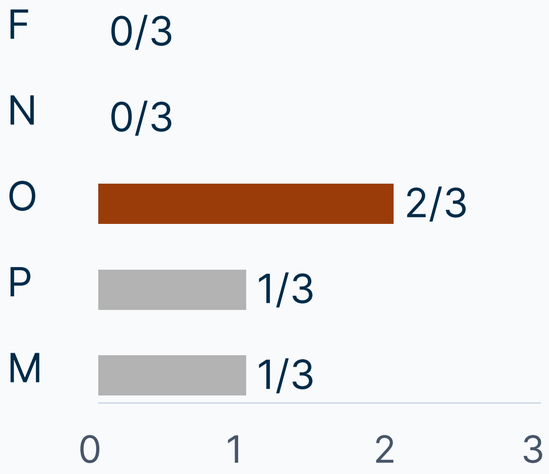
Component localization

Restrictive matches / 28



Control alarms

Alarms / 3 no-injection controls



O: 28/29 detected, 12/28 localized, 2/3 control alarms.

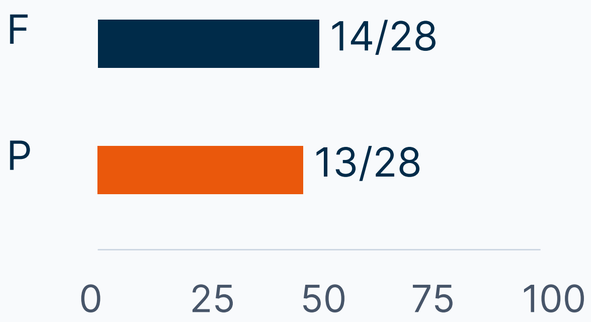
POLITECNICO MILANO 1863
NECST laboratory

Similar totals hide different successes

Configurations
- **F** Reference multi-agent
- **O** LLM-chosen evidence tools
- **P** No LLM Planner
- **N** No topology for Planner
- **M** Single ReAct agent

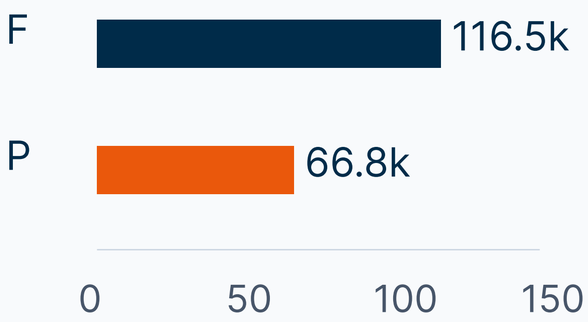
Component matches

Restrictive matches / 28



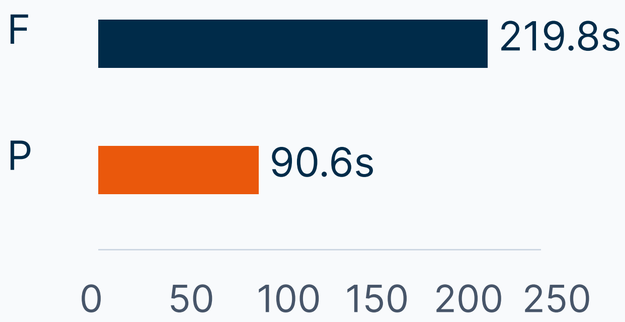
Mean tokens

Recorded tokens / fault (k)



Mean time

Diagnostic time / fault (s)



Component coverage across the same 28 cases



P vs F: -42.7% tokens / -58.8% diagnostic time.

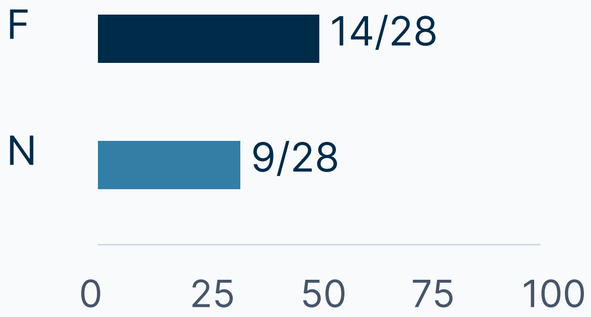
POLITECNICO MILANO 1863
NECST laboratory

Planner context: five more matches at roughly twice the cost

Configurations
- F Reference multi-agent
- O LLM-chosen evidence tools
- P No LLM Planner
- N No topology for Planner
- M Single ReAct agent

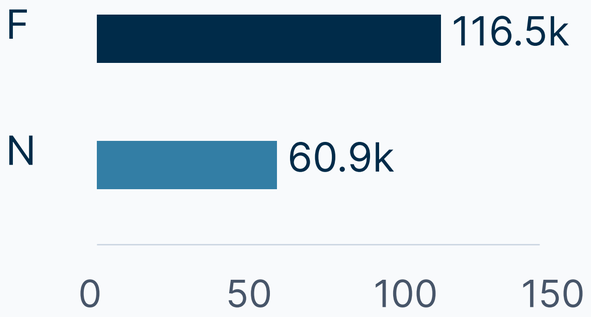
Component matches

Restrictive matches / 28



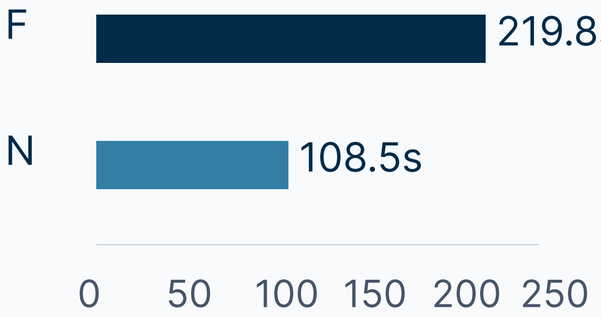
Mean tokens

Recorded tokens / fault (k)



Mean time

Diagnostic time / fault (s)



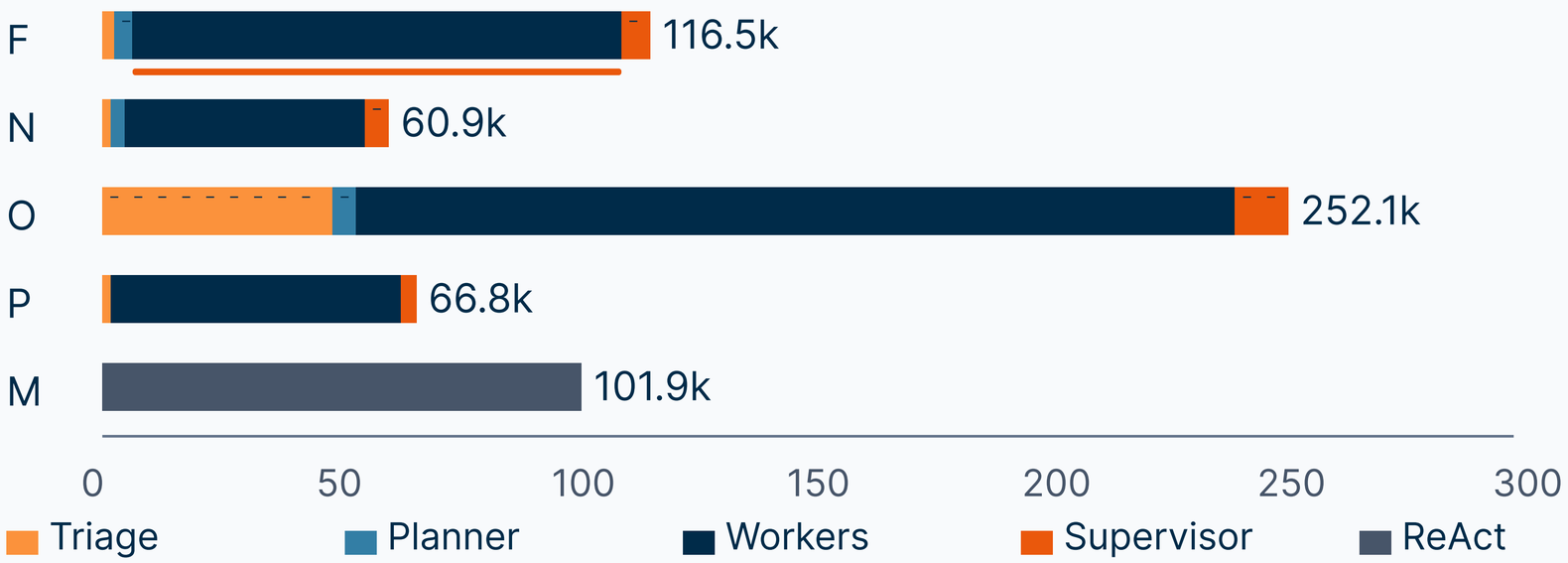
F: dependencies supplied to the Planner

N: no Planner context; Workers keep graph tools

N vs F: -47.7% tokens / -50.7% diagnostic time.

Workers use 89.2% of the reference tokens

Mean recorded tokens per fault (k)



89.2%

F tokens in Workers

3.3%

F tokens in Planner

Does the multi-agent workflow justify its cost?

Configurations
- **F** Reference multi-agent
- **O** LLM-chosen evidence tools
- **P** No LLM Planner
- **N** No topology for Planner
- **M** Single ReAct agent

	F / Reference multi-agent	M / Single ReAct
Component localization	14/28	17/28
Mean tokens / fault run	116,464	101,873
Mean diagnostic time / fault run	220 s	66 s

Cost means include all 29 fault runs per workflow.

A multi-agent design must justify its additional cost.

What matters for diagnosis and cost?

RCA quality is more than one score

Compare detection, localization, false alarms, and investigation cost; gains may bring trade-offs.

Plans shape the work that follows

Evaluate downstream investigation, not only the Planner call.

Dependency access is not dependency use

N keeps Worker graph tools, but its Planner receives no topology.

Research direction

What knowledge should we provide, at which stage, and when?

What's next?

Specialized models

Deep-learning models that process telemetry and extract meaningful diagnostic signals.



Complementary research direction

Operational knowledge

Capture knowledge about enterprise systems and engineers’ experience to guide investigations.





Next: enterprise systems and tribal knowledge

Knowledge for RCA in enterprise and proprietary systems

System knowledge and experience beyond public documentation

Enterprise & proprietary systems

Custom platforms and internal services
Company-specific behavior and dependencies

Engineers' tribal knowledge

Experience from incidents
Practical workarounds and practices
Often missing from runbooks

Research focus: extract, link, provide at runtime

Extract system and tribal knowledge; link it to operational data.
Provide relevant knowledge to the agent during investigation.

Where I'd like to go next

My starting point

Agent workflow design
Cloud systems experience
Evidence-based evaluation

Open research question

How can we extract and link operational knowledge, and provide it to RCA agents at runtime?

How does this question connect with your group's research?

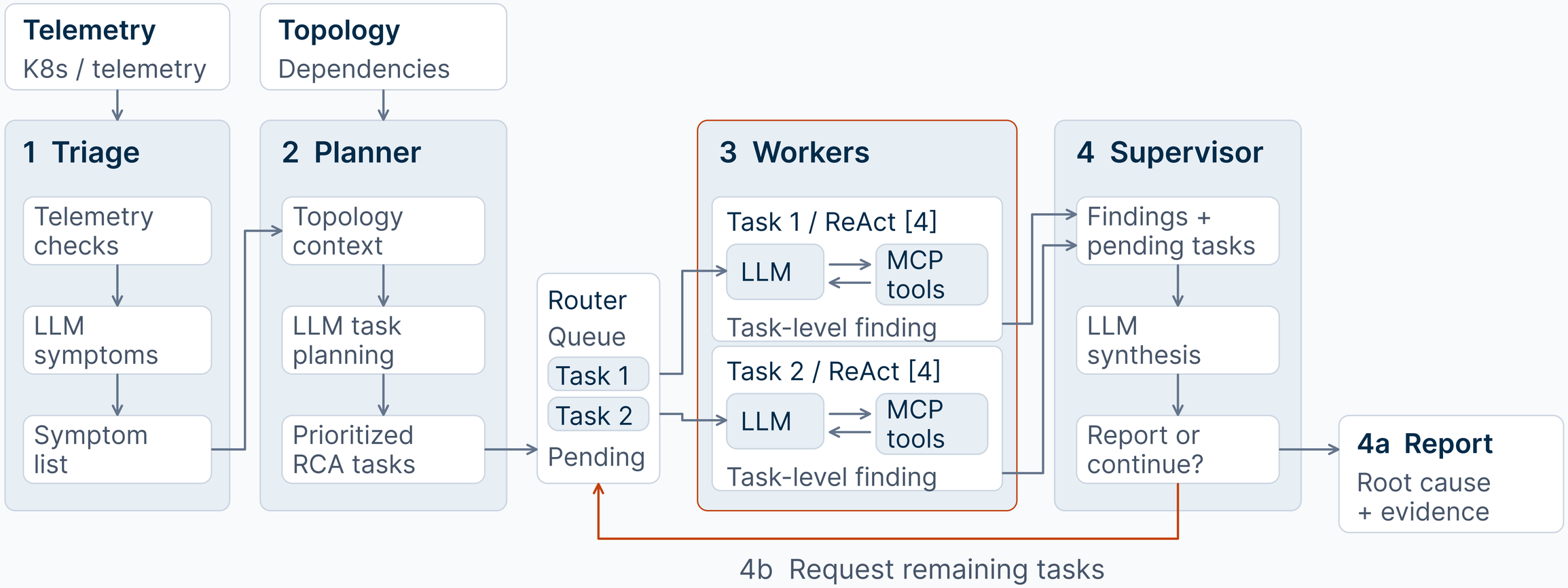
Thank You

Marcello Martini
marcello.martini@polimi.it
marcellomartini.tech

Research links

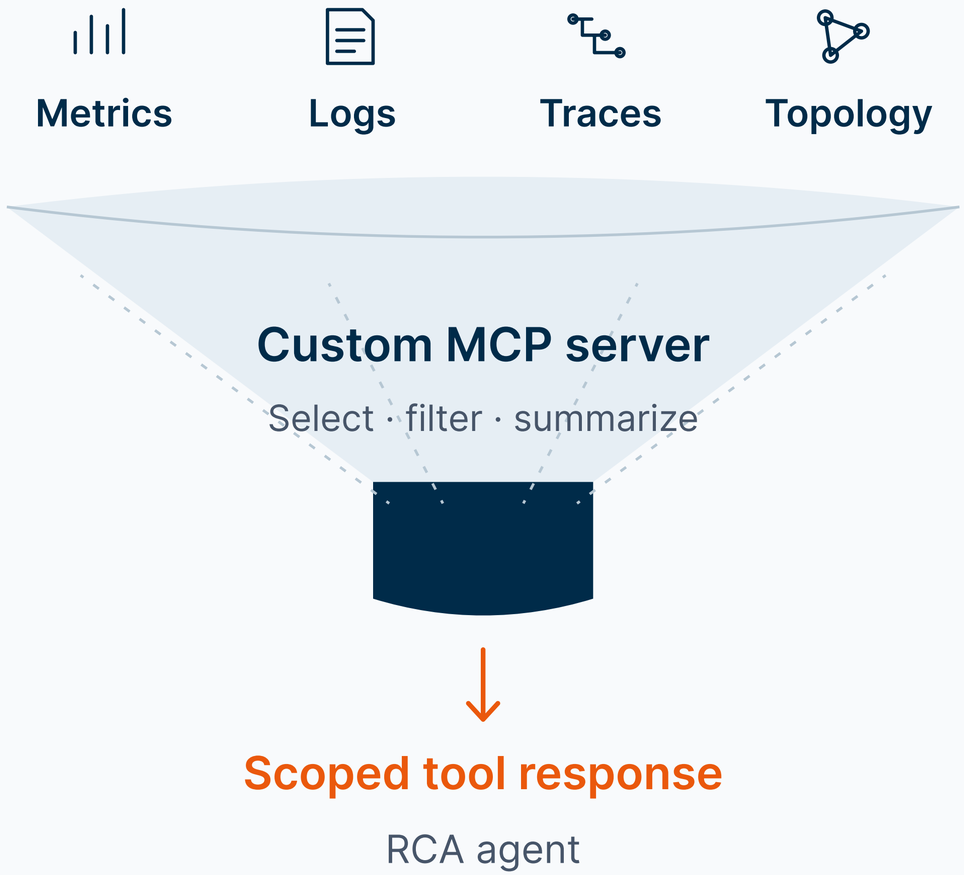
link.marcellomartini.tech/
boston-meeting/

Diagnostic Workflow and Tool Access



[6] S. Yao et al., ReAct, arXiv:2210.03629.

MCP: from operational data to agent context



Predefined metrics

Pod / service scope
Current values or time range

Log filtering

Recent-line limit
Optional keyword filter

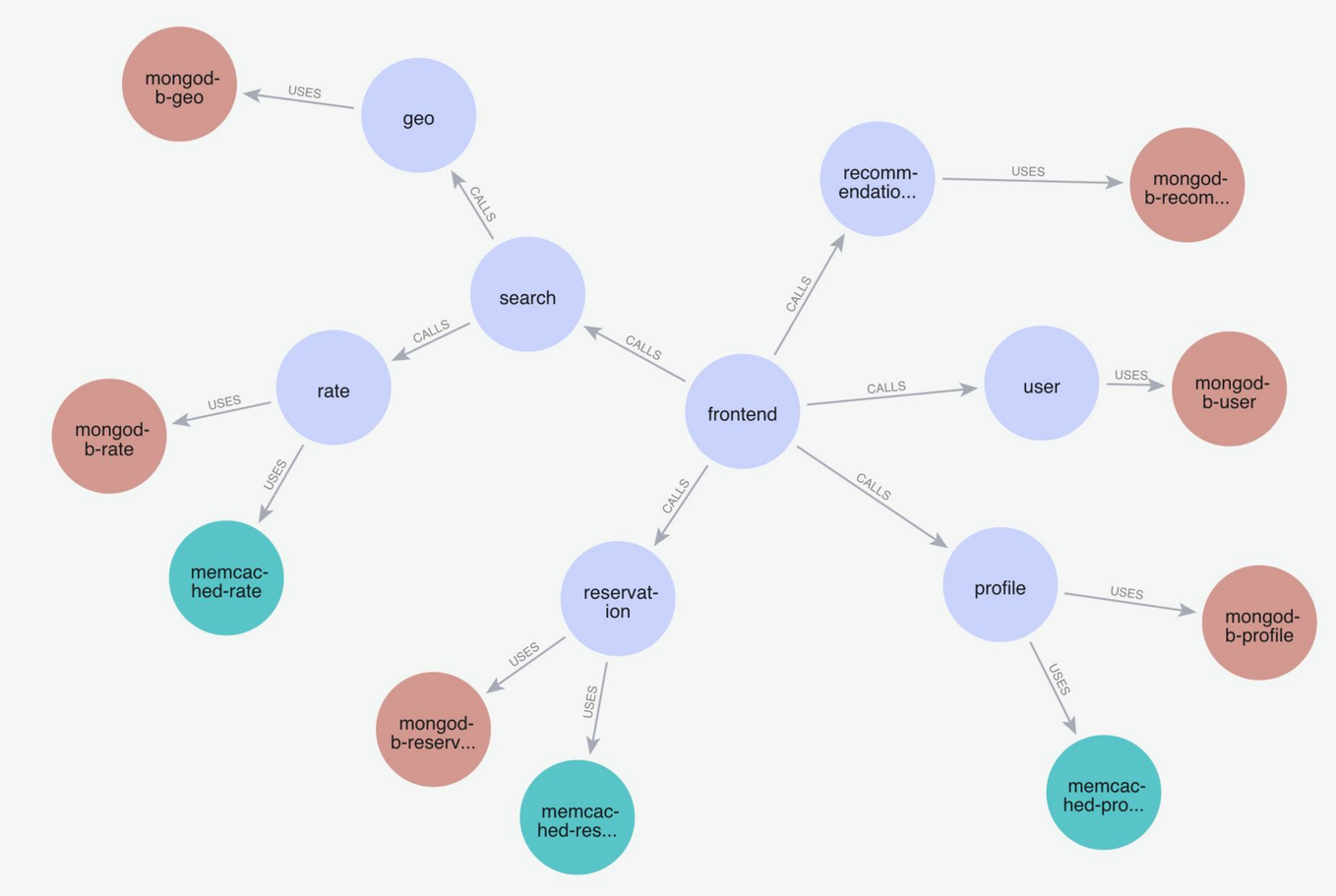
Trace summaries

Optional errors-only filter
Trace-ID drill-down

Dependency selection

Direct service calls
Infrastructure dependencies

Datagraph: service calls and infrastructure dependencies



Hotel Reservation

CALLS / service interactions
USES / infrastructure dependencies

Configured Neo4j topology, loaded from application-specific definitions.

This service-topology graph is distinct from the proposed operational-knowledge graph.

POLITECNICO MILANO 1863
POLITECNICO MILANO 1863 NECST laboratory

Programmed checks select evidence before the LLM

Pod status

Waiting containers
Nonzero exit code
More than 3 restarts

Trace selection

Error traces
Slow traces: 2 s minimum
Error-trace fallback logic

Metric flags

Thread ratio > 95%
CPU load average > 10
Network error/drop totals > 1

Fixed checks → selected evidence → LLM interpretation → structured symptoms

Heuristics, not a complete health test. Missing symptoms can prevent investigation from starting.

Thresholds apply when metrics are available; network counters are lifetime totals, not rates.

Testbed: applications, telemetry and agent execution

Kubernetes on KinD

AIOpsLab application deployment + fault scenarios
Hotel Reservation / Astronomy Shop / Social Network

Telemetry + topology

Prometheus metrics
Kubernetes logs / resources
Jaeger traces
Neo4j service graph

→

Diagnosis + run traces

Custom MCP connector
Python / LangGraph workflows
GPT-5-mini
LangSmith agent traces

Selected faults per workflow: Hotel 12 / Astronomy 11 / Social 6

Experiment protocol and interpretation limits

1 / Scenario
Deploy application and introduce fault

→

2 / Context
Collect telemetry and topology

→

3 / Diagnosis
Run workflow with GPT-5-mini

→

4 / Record
Save reports, tokens and time

→

5 / Evaluate
Detection, localization and control alarms

5 workflows × 29 fault scenarios + 3 no-fault controls per workflow

Localization: 28 component cases. Cost: all 29 selected fault runs per workflow.

One execution per workflow/scenario: descriptive trade-offs, not statistical equivalence or isolated causal effects.

The Five Evaluated Configurations

Same diagnostic model and report schema; compare concrete workflow alternatives.

Workflow	Model	Collection	Planning	Context	Notes
F	GPT-5-mini	Programmed	Joint LLM	Supplied	Fan-out 2; guidance 7
N	GPT-5-mini	Programmed	Joint LLM	Not supplied	Workers retain graph tools
O	GPT-5-mini	LLM-directed	Joint LLM	Supplied	Triage guidance 7
P	GPT-5-mini	Programmed	Symptom-wise	Not enriched	All symptom tasks start first
M	GPT-5-mini	Tool-using	Self-directed	On demand	One ReAct history; guidance 14

Budgets guide tool use; they are not hard limits. N omits Planner context, not Worker tool access.

POLITECNICO MILANO 1863
POLITECNICO MILANO 1863 NECST laboratory

Metrics, Populations, and Comparison Limits

DETECTION

Positive verdicts / 29 fault runs

F 23/29 | N 19/29 | O 28/29 | P 21/29 | M 25/29
A positive verdict can concern an incidental anomaly.

CONTROL ALARMS

Alarms / 3 no-injection controls

F 0/3 | N 0/3 | O 2/3 | P 1/3 | M 1/3
One selected control per application.

COMPONENT LOCALIZATION

Restrictive matches / 28 component cases

F 14/28 | N 9/28 | O 12/28 | P 13/28 | M 17/28
Nonempty predictions inside accepted targets; not top-1.

RESOURCE COST

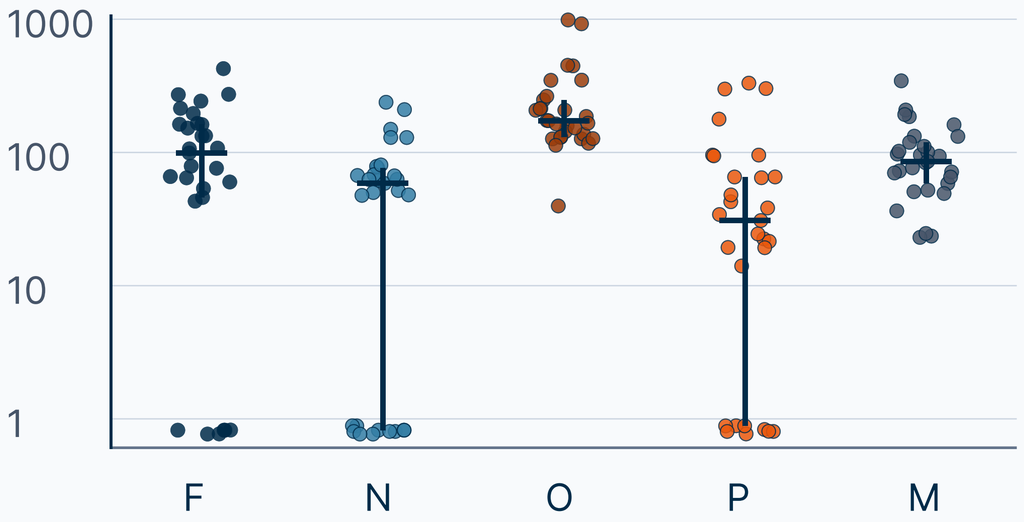
Tokens and diagnostic wall-clock time

Means over 29 faults; all-run distributions retained.
Excludes setup, fault injection, and later evaluation.

All-Run Token and Time Distributions

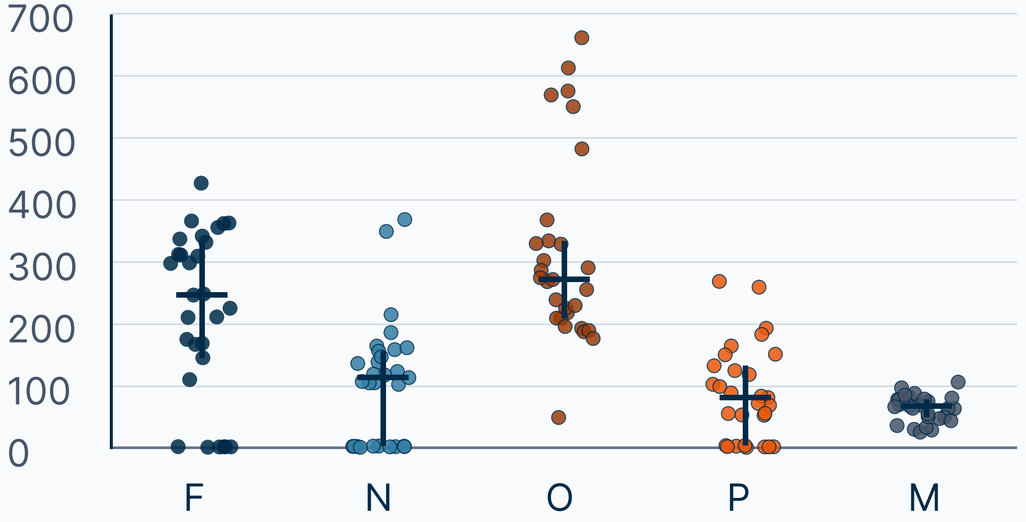
Tokens per fault (k)

Base-10 logarithmic axis



Execution time per fault (s)

Linear axis, starting at zero



One point per fault-scenario execution. Black marks: median and scenario IQR, not confidence intervals. Horizontal jitter only; seed 20261024. The M Kafka run is included despite pending tracing status.

POLITECNICO MILANO 1863
POLITECNICO MILANO 1863 NECST laboratory

Component Localization by Scenario

Hotel Reservation

	F	N	O	P	M
Container Kill	×	×	×	×	●
Geo misconfiguration	●	●	●	×	●
Network delay	●	×	×	×	×
Network loss	●	×	×	×	×
Pod failure	●	●	△	●	●
Pod kill	×	●	△	●	●
Revoke auth (Geo)	●	●	●	●	●
Revoke auth (Rate)	●	●	●	●	●
Unregistered user (Geo)	●	●	●	●	●
Unregistered user (Rate)	●	●	△	●	●
Wrong bin usage	×	×	●	×	●

Social Network

	F	N	O	P	M
Nonexistent node	×	×	×	×	●
MongoDB TLS	△	△	×	●	●
Port: post-storage	●	×	●	×	●
Port: text	●	×	●	●	●
Port: user	●	×	△	×	●
Scale to zero	×	×	●	×	●

Astronomy Shop

	F	N	O	P	M
Ad failure	×	×	×	●	×
Ad high CPU	×	×	×	×	×
Ad manual GC	●	×	×	●	×
Cart failure	×	●	●	×	×
Image slow load	×	●	●	×	×
Kafka queue problems	×	×	×	●	×
Load flood homepage	×	×	×	×	×
Payment failure	●	×	●	●	●
Payment unreachable	×	×	×	×	×
Product catalog failure	●	×	●	●	●
Recommendation cache	×	×	×	×	×

● Restrictive match △ Any-target only × No accepted target

28 component scenarios; controls and namespace-wide redeployment excluded. Pending M Kafka outcomes are included.

POLITECNICO MILANO 1863 NECST laboratory

Cart Failure: A Saved Evidence Path

F / WITH PLANNER TOPOLOGY

OBSERVED PATH
Proxy/frontend investigations

→

OBSERVED PATH
Report names frontend + checkout

→

OBSERVED PATH
Cart component not matched

N / NO PLANNER TOPOLOGY; WORKER GRAPH TOOLS RETAINED

01
Cart/storage investigation

→

02
Graph lookup: no relations returned

→

03
Deployment inspection

→

04
Valkey configuration acquired

→

05
Report names cart: component matched

Incoming symptoms differ: this is not a fixed-input causal comparison. Empty lookup does not prove no dependency.

POLITECNICO MILANO 1863
POLITECNICO MILANO 1863 NECST laboratory

Component Matches and Explanation Quality

Saved ordinal assessments on 29 fault runs per configuration

Variant	No analysis	1	2	3	4	5	Low score / localized
F	6	12	4	3	1	3	7 / 14
N	10	10	2	2	1	4	2 / 9
O	1	16	2	3	1	6	3 / 12
P	8	8	5	0	4	4	7 / 13
M	0	14	2	1	2	10	4 / 17

23 / 65 restrictive component matches have saved scores 1–2.

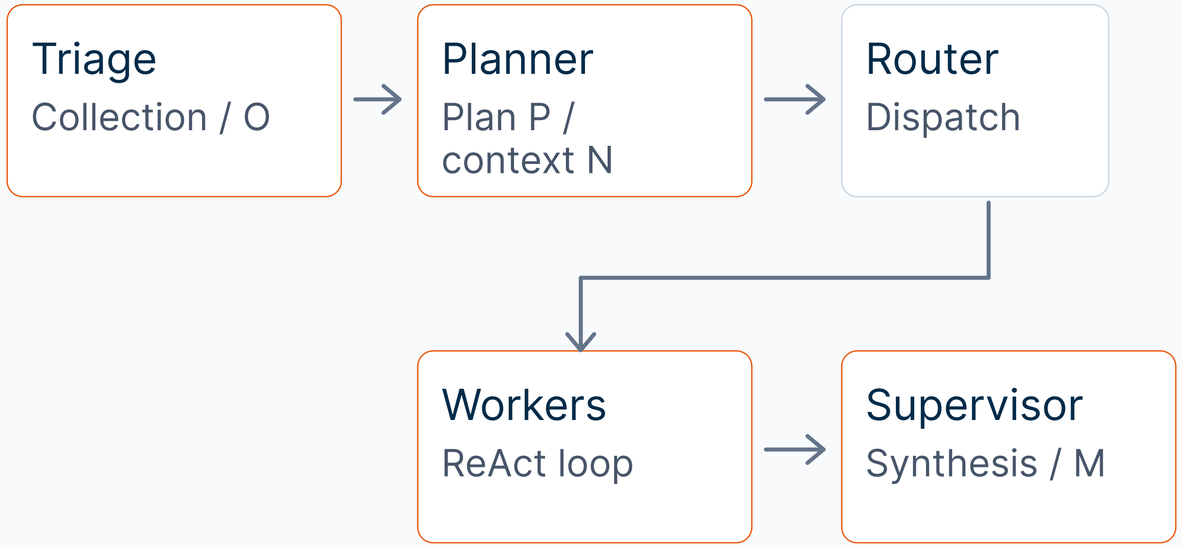
No analysis: automatic score 1 without a judge call. Scores are auxiliary, ordinal, and not human-validated.

[7] L. Zheng et al., Judging LLM-as-a-Judge, NeurIPS, 2023.

POLITECNICO MILANO 1863
POLITECNICO MILANO 1863 NECST laboratory

What changes in each configuration?

Reference workflow F



Why each alternative?

F	Reference multi-agent workflow
O	Model-directed collection
P	Tasks directly from symptoms
N	No supplied Planner topology
M	Single ReAct investigator

Exploratory analysis: the A–L design space

Historical exploration — separate from the current F / O / P / N / M contrasts.

A / B / C	Plain ReAct	P = 2 / 3 / 5; B = 7
D / E	Conservative investigation prompts	P = 2 / 3; B = 7
F / G	Tool-free planning	P = 2 / 3; B = 7
H / I	Certainty-enforcing Supervisor	P = 2 / 3; B = 7
J / K / L	Specialized prompts	J/K: P2/3, B7; L: P2, B15

P = initial task fan-out. B = prompt-level tool guidance, not a hard limit.

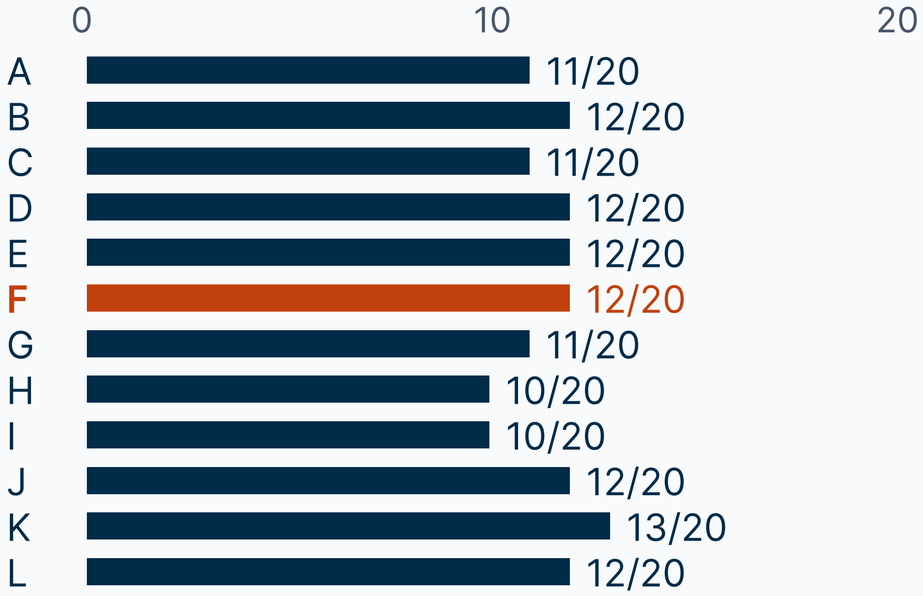
353 archived records; incomplete coverage. F is a reference, not a proven optimum.

POLITECNICO MILANO 1863
POLITECNICO MILANO 1863 NECST laboratory

Exploratory results: matched scenarios, mixed trade-offs

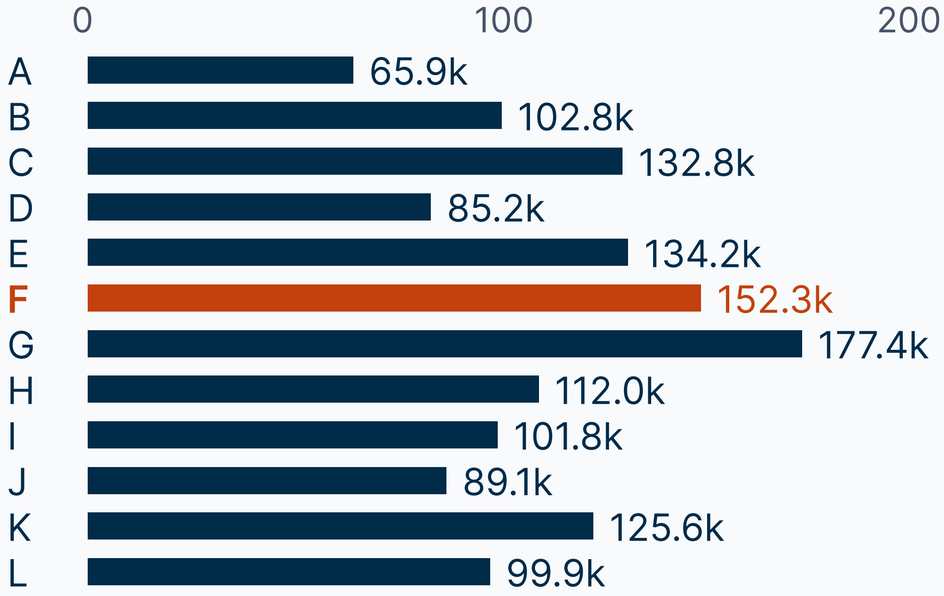
Component matches / 20 common fault scenarios

Nonempty restrictive localization; controls excluded



Mean tokens / 19 common fault scenarios (k)

One common scenario excluded for missing cost



Descriptive, one retained run per configuration/scenario; no inferred optimum. F = reference.

Historical cohort. Cost omits Social / MongoDB TLS. Current recorded evaluations, not old summaries.

POLITECNICO MILANO 1863
POLITECNICO MILANO 1863 NECST laboratory

Linking operational knowledge for RCA

Operational sources

Telemetry and events
Code, configuration and dependencies
Past incidents and guides

Engineers' tribal knowledge

Experience, workarounds and conditions

→

Extract and link

Connect knowledge to services,
evidence and conditions.
Preserve source and scope.

→

Provide at runtime

Relevant knowledge for the RCA agent's current investigation

Research question: Does this help RCA beyond document search?

References & Useful Links

[1] Google SRE Book. "Being On-Call", chapter 11.

sre.google/sre-book/being-on-call/

[2] J. Lewis and M. Fowler. "Microservices." 25 March 2014.

martinfowler.com/articles/microservices.html

[3] IBM. "What is AIOps?" IBM Think. Accessed 1 November 2026.

ibm.com/think/topics/aiops

[4] Y. Chen et al. "AIOpsLab: A Holistic Framework to Evaluate AI Agents for Enabling Autonomous Clouds." Proceedings of Machine Learning and Systems, 2025.

AIOpsLab publisher record

[5] Y. Wang et al. "EviRCA: Decoupling Evidence Extraction from Reasoning for Microservice Root-Cause Analysis." arXiv:2609.19825, 2026. Preprint.

arxiv.org/abs/2609.19825

[6] S. Yao et al. "ReAct: Synergizing Reasoning and Acting in Language Models." arXiv:2210.03629.

arxiv.org/abs/2210.03629

[7] L. Zheng et al. "Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena." NeurIPS 2023 Datasets and Benchmarks Track.

arxiv.org/abs/2306.05685

Research profile

marcellomartini.tech/about/

CV

marcellomartini.tech/CV_MarcelloMartini.pdf

Study artifacts (public)

Public results and materials

POLITECNICO MILANO 1863

POLITECNICO MILANO 1863 NECST laboratory