Hybrid Cloud Monitoring with Instana

IBM Z

IBM

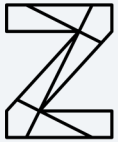
Disclaimer

Any reference to future plans are for planning purposes only. IBM reserves the right to change those plans at its discretion. Any reliance on such a disclosure is solely at your own risk. IBM makes no commitment to provide additional information in the future.

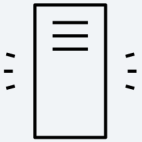
What we heard from you

Middleware is always blamed when something goes wrong

How to know that there is a problem with a specific service on z/TPF



z/TPF is always blamed when something goes wrong



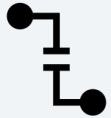
z/TPF needs to be regarded the same as other servers

“I want to quickly figure out *what* the problem is and connect with the right SME to fix it.”

“I don’t want to spend a lot of time proving my system *is not* the problem”

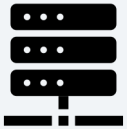
- Coverage programmer

z/TPF Specific Use cases



Messages are not getting to the z/TPF system

Services z/TPF applications call, are taking a long time to respond



The z/TPF system itself is ok, but there is a problem with a specific service

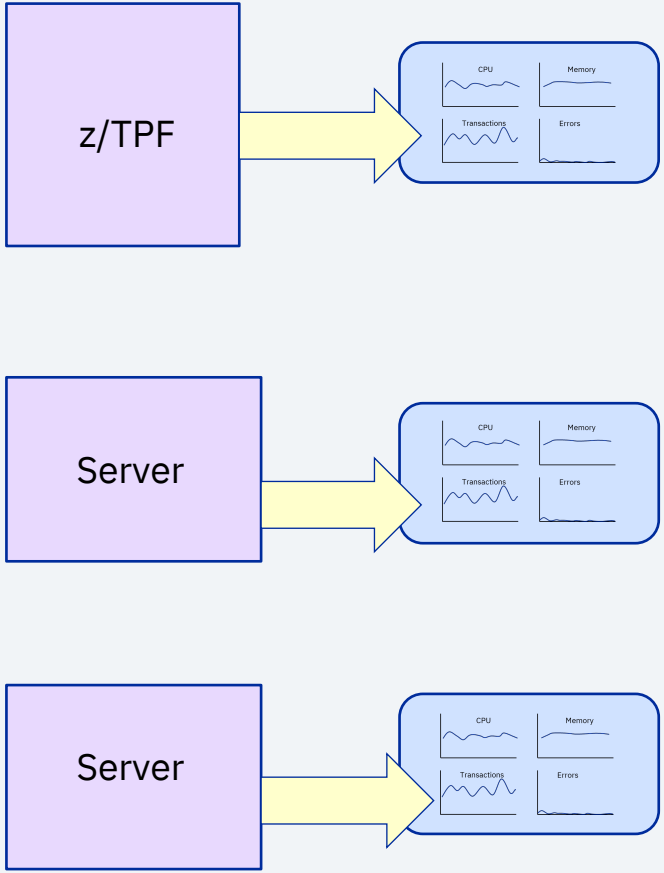
In all these cases, the problem surfaces as an impact to one or more end users

As-is monitoring

When the various components are monitored each within their own silo, a small problem may go unnoticed, or may not understand the impact it has on the end users

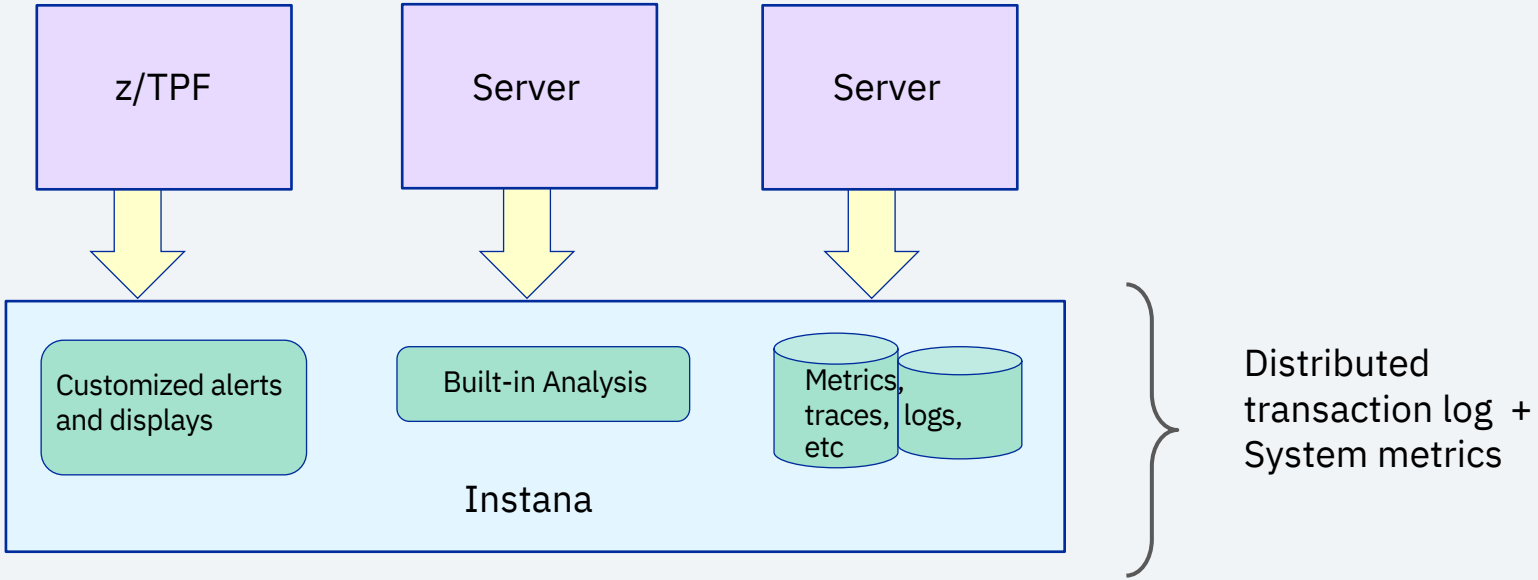
Example: one service is taking errors for a subset of calls

From a monitoring perspective the component where that service is running may seem to be operating normally.



To-be Monitoring

A single platform that is monitoring all systems that can correlate events across the various components, and provide an end-to-end view of specific transaction flows both at the service level as well as the machine level

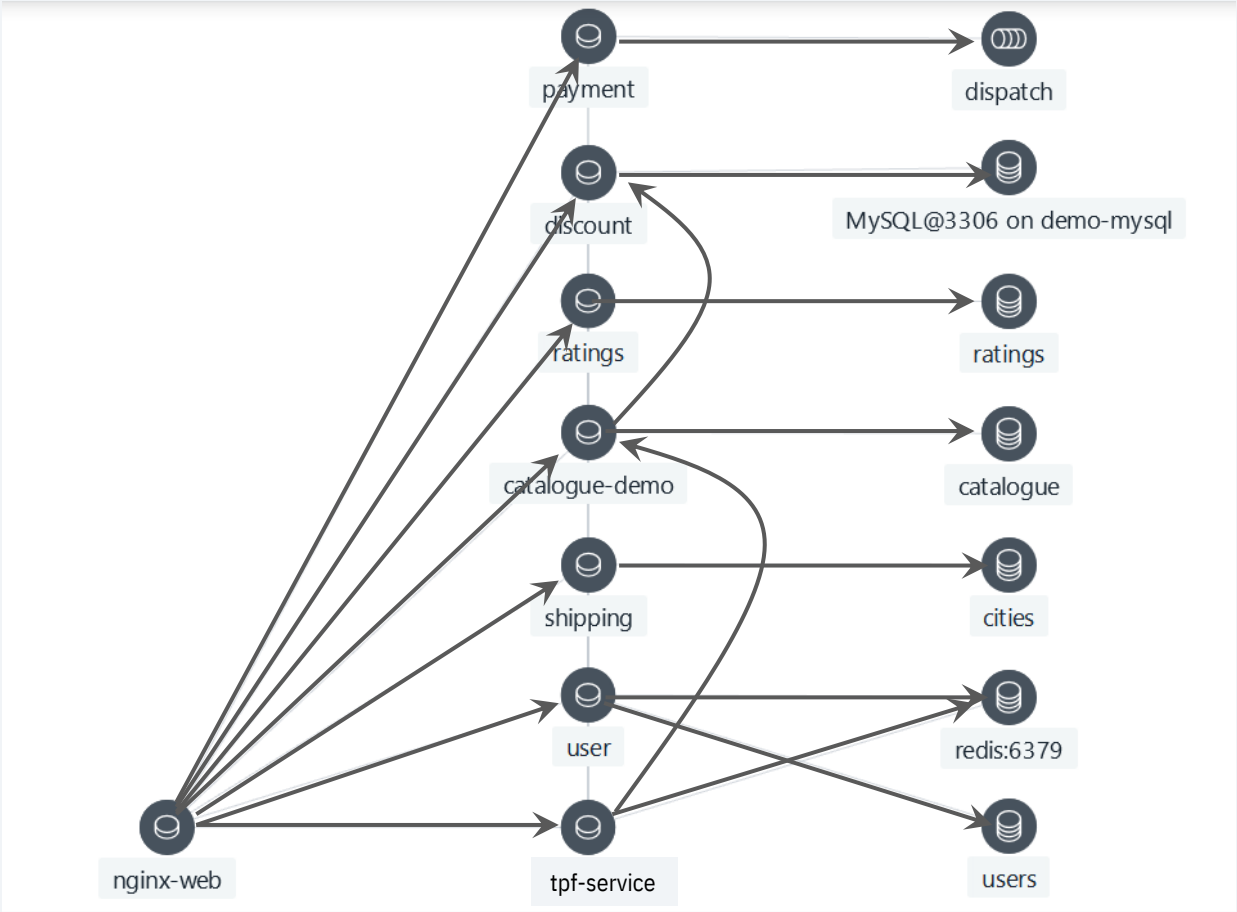


A Simple Example

Robot-Shop consists of a front end process that makes calls to several service calls, one of which calls a service on z/TPF which in turn makes outbound calls to another service.

All components are sending data to the monitoring platform where the information is analyzed and correlated across all services. Data collected includes latency and error rates.

Alerts have been set up to notify the Site Reliable Engineer (SRE) of any incidents.



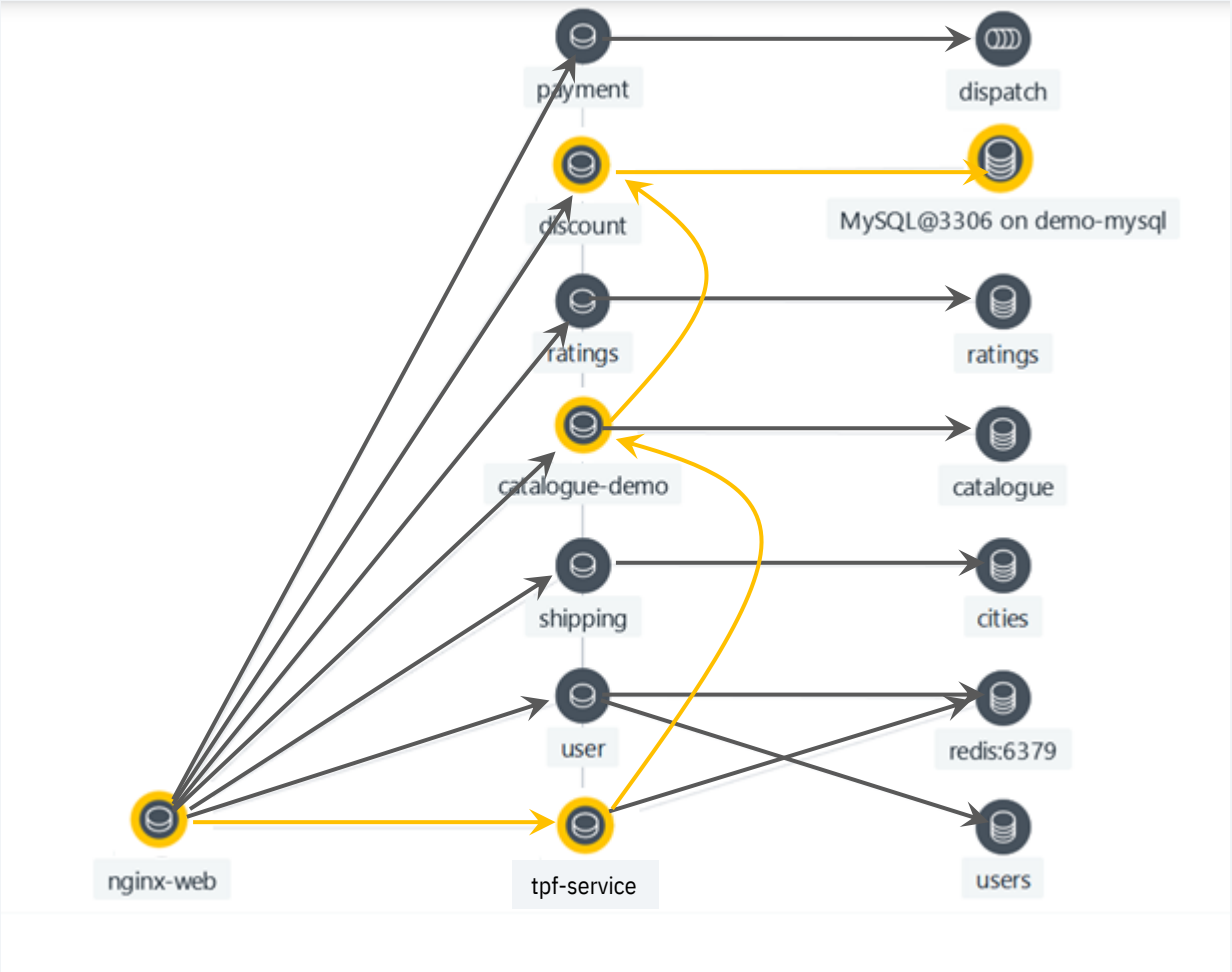
The Situation

The Site Reliability Engineer has set up configured incidents for the Robot-Shop to be sent to PagerDuty.

This will allow him to quickly respond if something in the environment is impacting the experience of the end users.

In our example, the termination of the backend service MySQL causes an impact to only a portion of the end user requests.

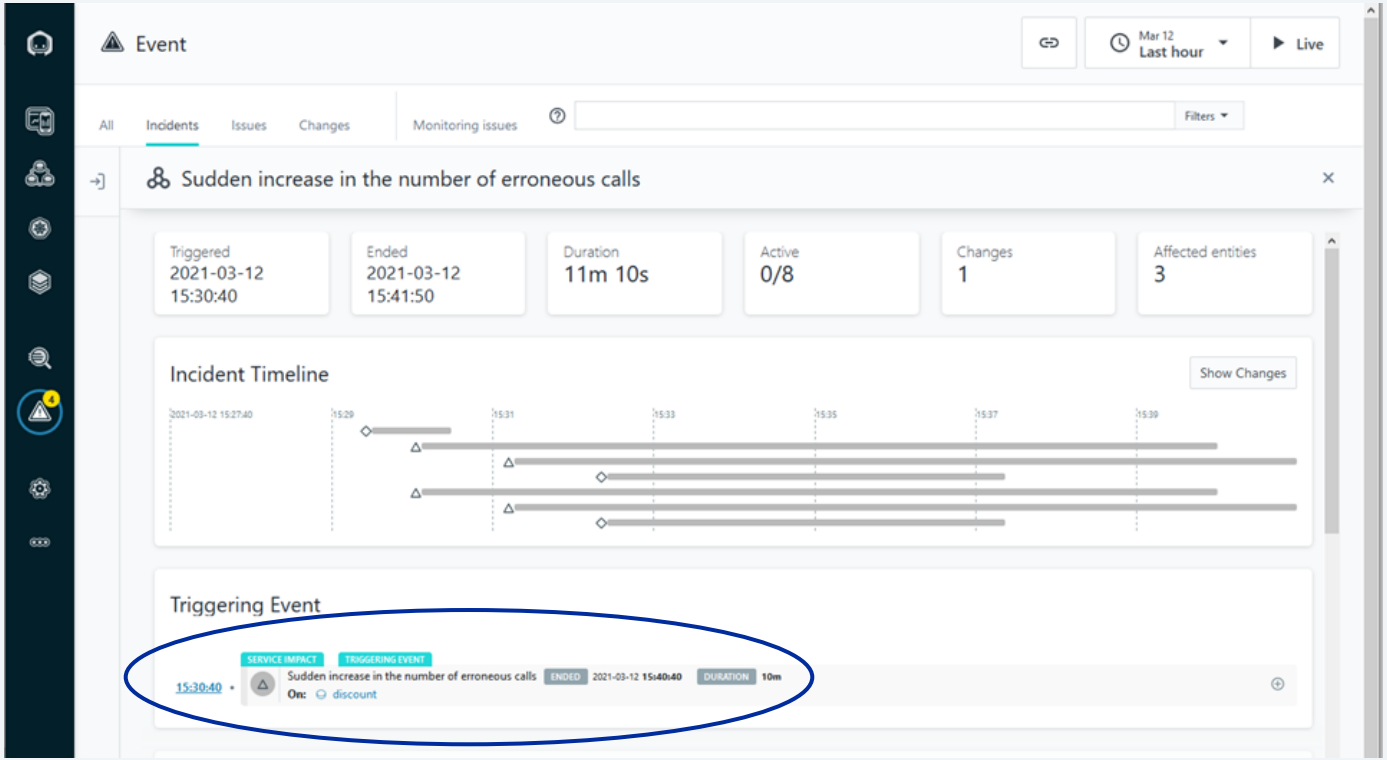
Let's see how the monitoring solution helps to quickly identify the problem.



Incident Alert

The SRE gets an incident alert and is taken to the event page to see what is the problem and when did it happen.

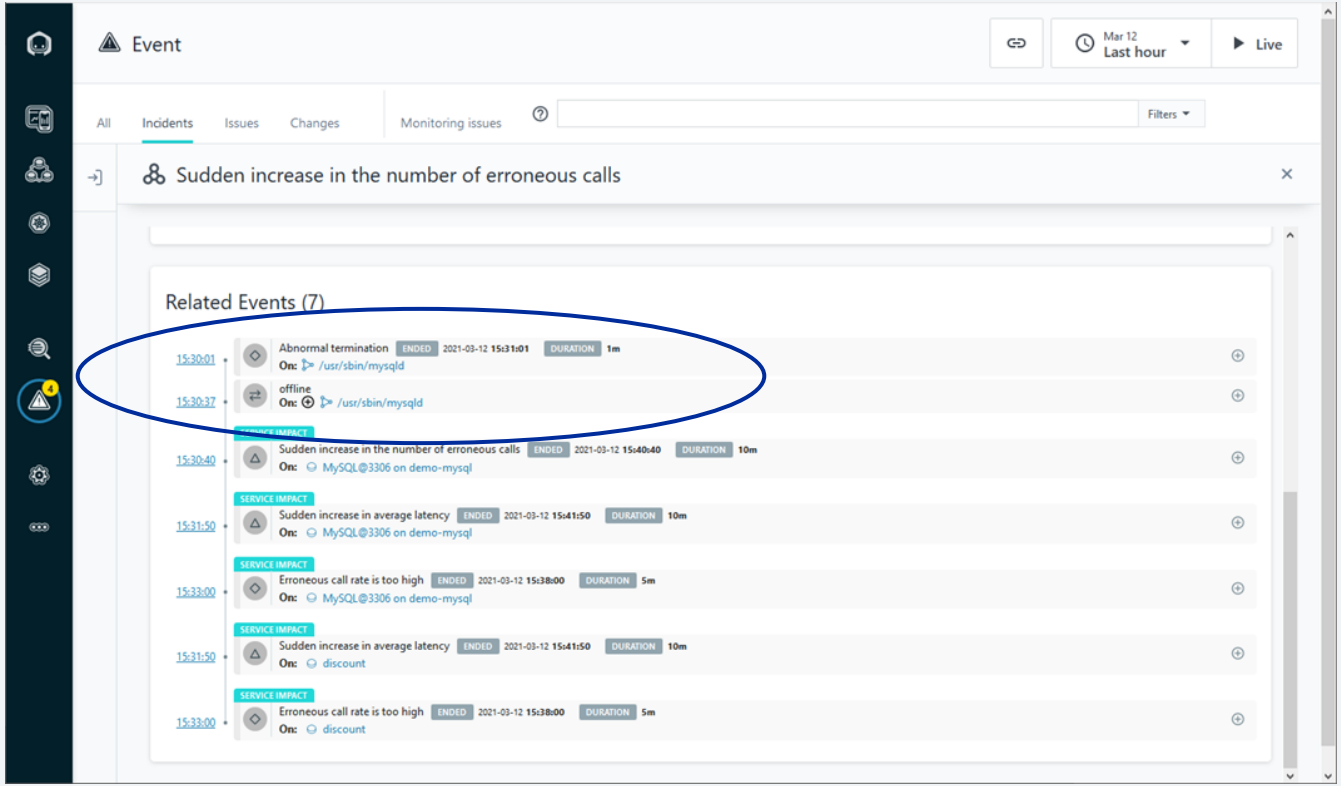
Triggering Event
Sudden increase in the number of erroneous calls
in service discount



Related Events

In addition to seeing the event that triggered the incident, all related events are displayed to bring the incident into context.

Related Events (7)
Abnormal termination
/usr/sbin/msqld

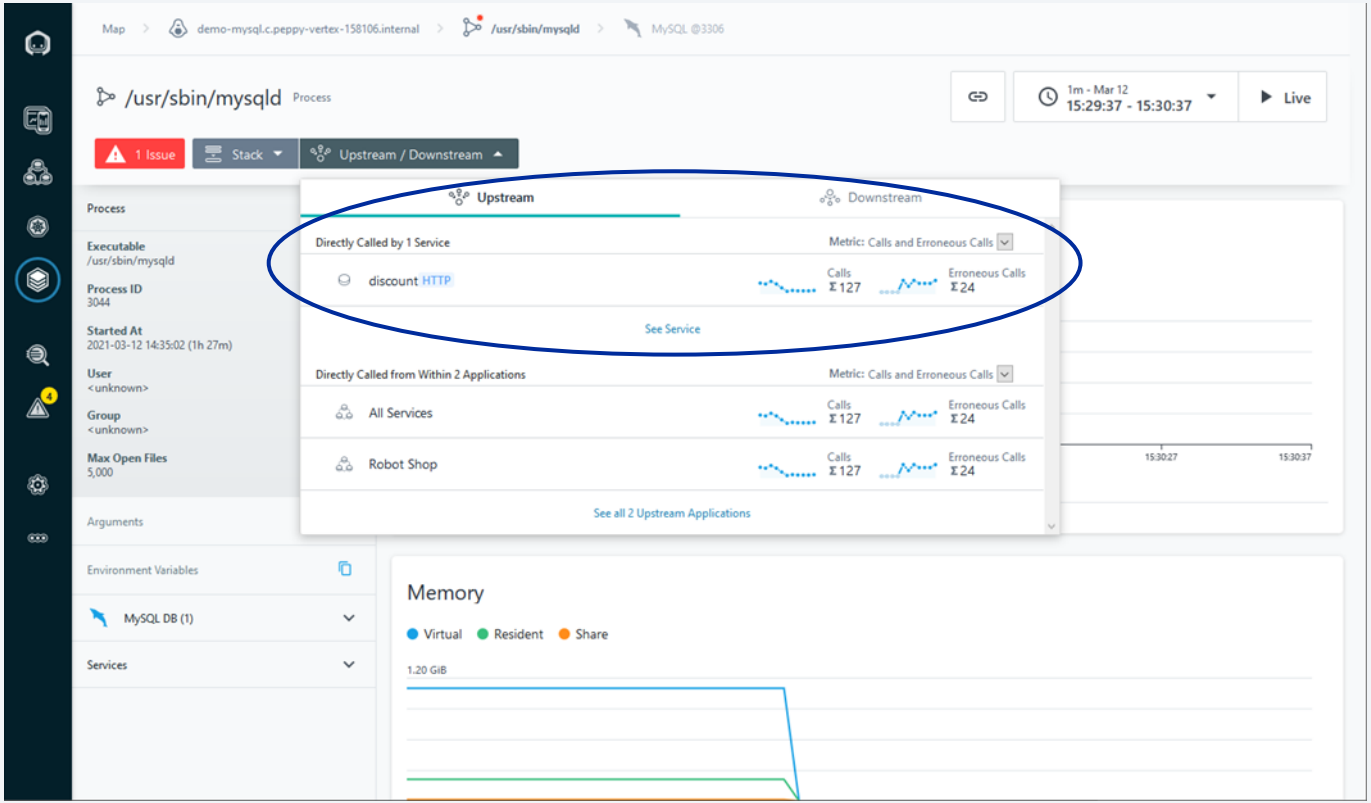


Upstream / Downstream Calls

Clicking on the abnormal termination event, the SRE can see both the upstream and downstream callers.

Upstream
Directly called by 1 Service
discount

This is the most likely cause of the problem

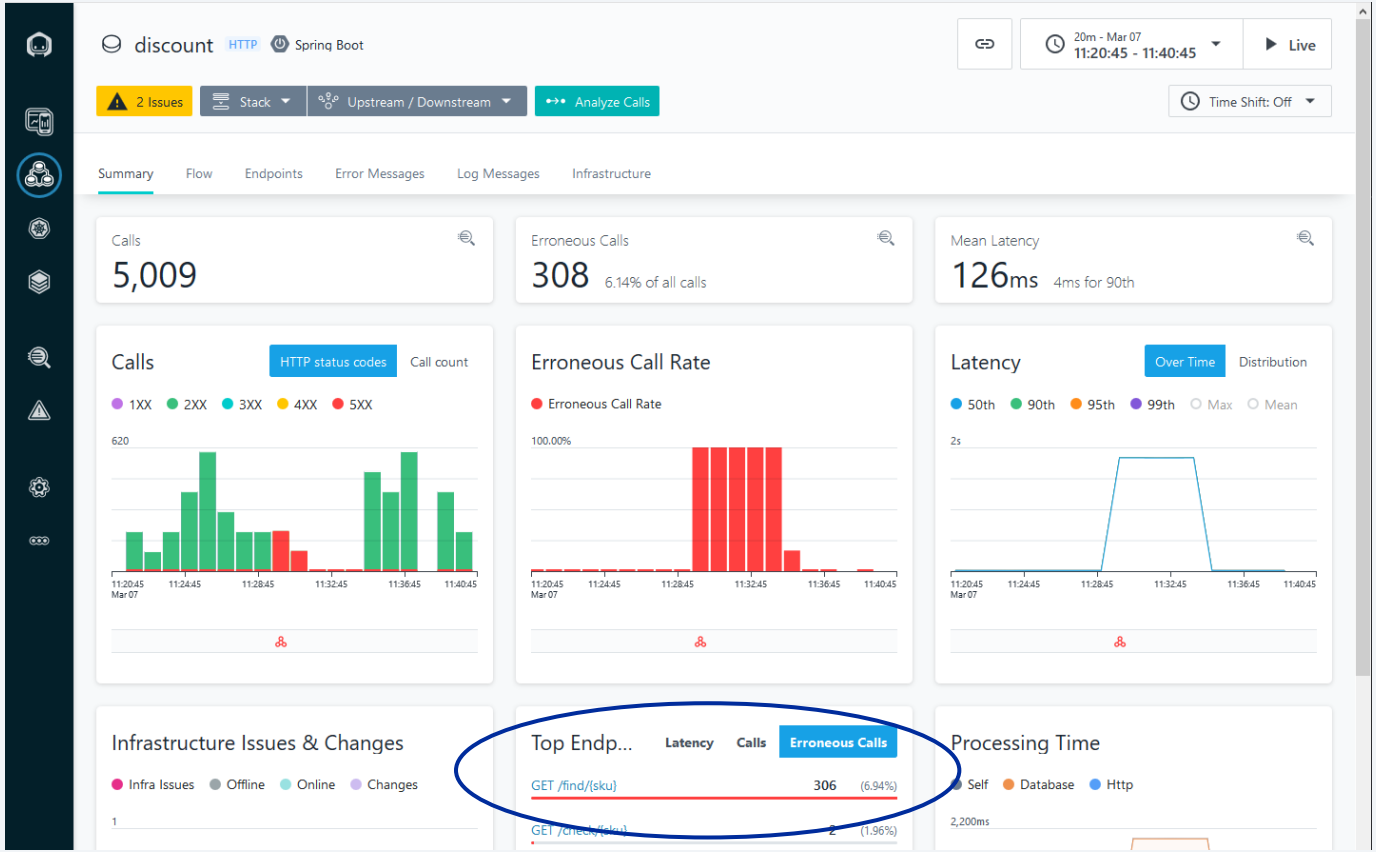


Service Details

To gather more information to confirm the issue can go into the details of the “discount” service.

Since the incident involved increase in erroneous calls, can see the top endpoints taking errors

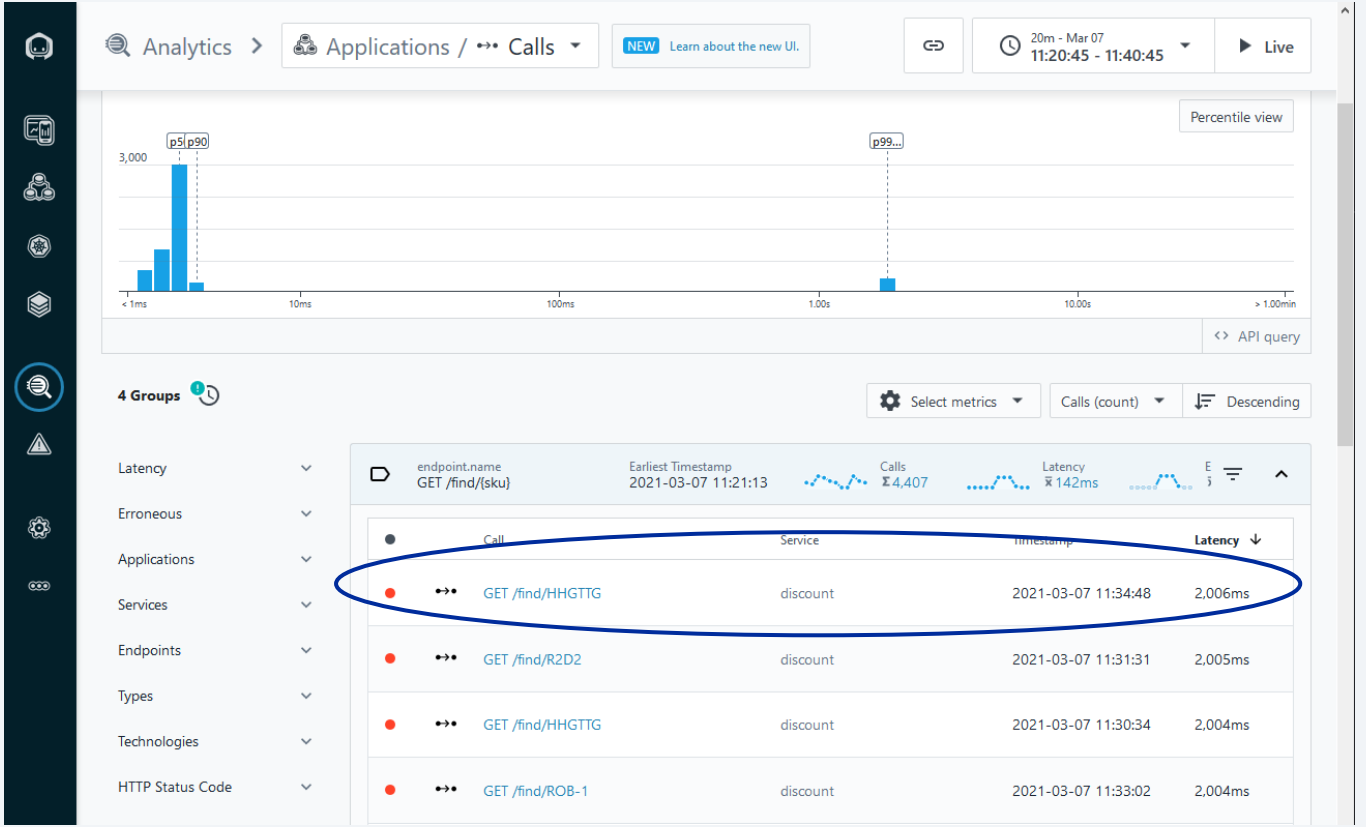
GET/find/{sku}



Analyze Calls

Shows the different APIs that make up the “discount” service, and groups together individual calls to a particular endpoint.

Here we see all the instances of calls to GET/find/{sku}

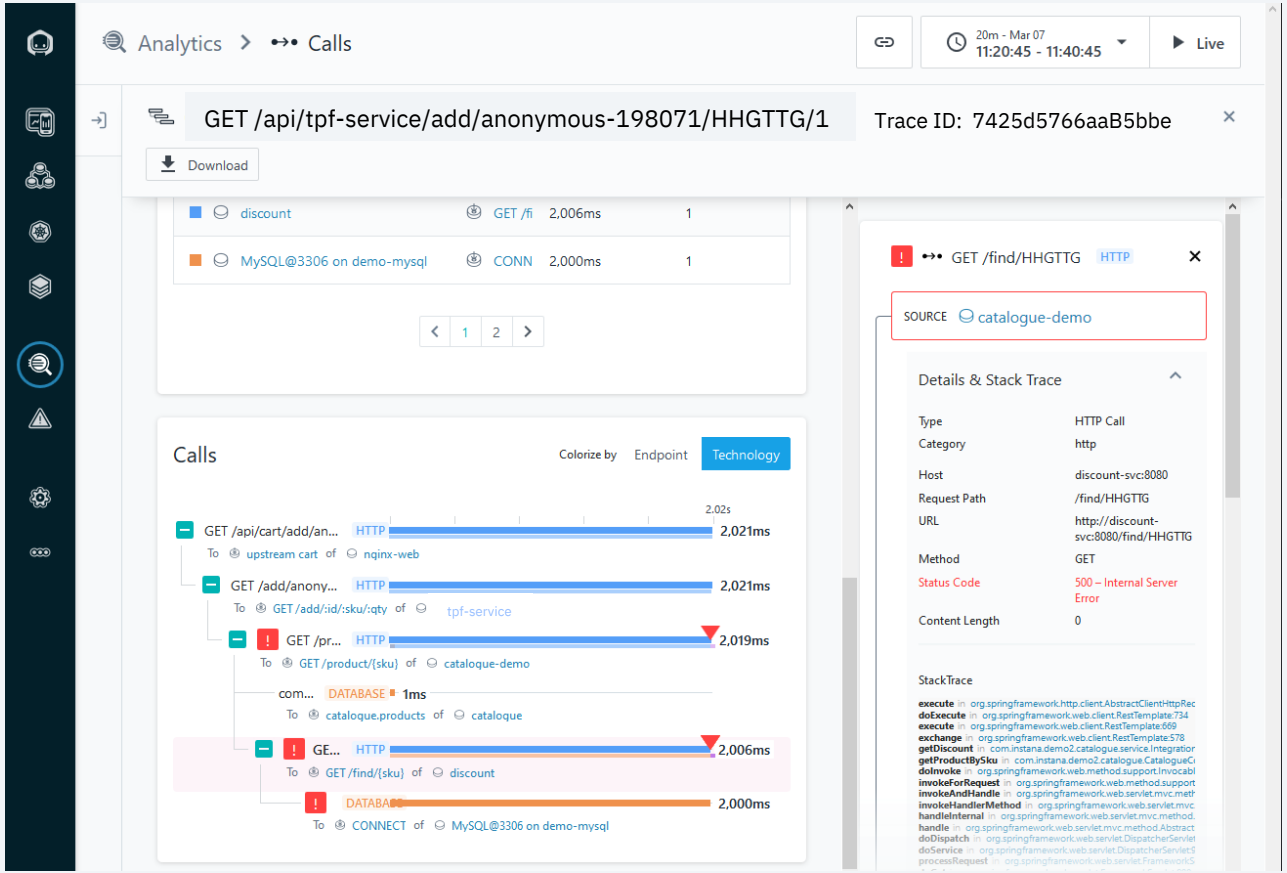


Drill Down on a Specific Request

Shows the end-to-end flow of the request showing the break-down of all the calls ending with the timeout of the call to the database.

The front end process nginx-web is calling the tpf-service, which is not the problem.

It's the backend service discount and it's call to the MYSQL database that is the source of the problem.



Value

The next level of tooling is about providing insights, not just collecting data.

- Built-in correlation between services and infrastructure to give context to issues
- Alerts based on rich set of rules including dynamic baselines and anomaly detection
- AI engine to group related events to quickly get to root cause

z/TPF is no longer a black box or black hole!

Why Instana?

APM Experts

Ranking the Observability Offerings

Capability	Splunk	Elastic	AppDynamics	Dynatrace	New Relic	Instana	Datadog	Honeycomb	Lightstep
Comprehensive Log Collection	●	●	○	◑	◑	◔	◑	◕	○
Comprehensive Metric Collection	◑	◑	◑	◑	◑	◕	◑	◕	◔
Comprehensive Tracing Collection	◕	◕	◑	●	●	●	●	◕	●
Comprehensive Dependency Collection	○	○	◑	◕	◑	●	◔	○	○
Comprehensive Relating of Logs, Metrics, and Dependencies	○	○	◔	◕	◑	●	◔	○	○
Automated and Instant Instrumentation	○	○	○	●	○	●	○	○	○
High Cardinality Analytics	◑	◑	○	◕	◕	◕	◕	●	●
Dependency Map and AI Based Root Cause	◑	◑	◔	◕	◑	●	◔	○	○
Rank	5th	5th	8th	2nd	3rd	1st	4th	5th	8th

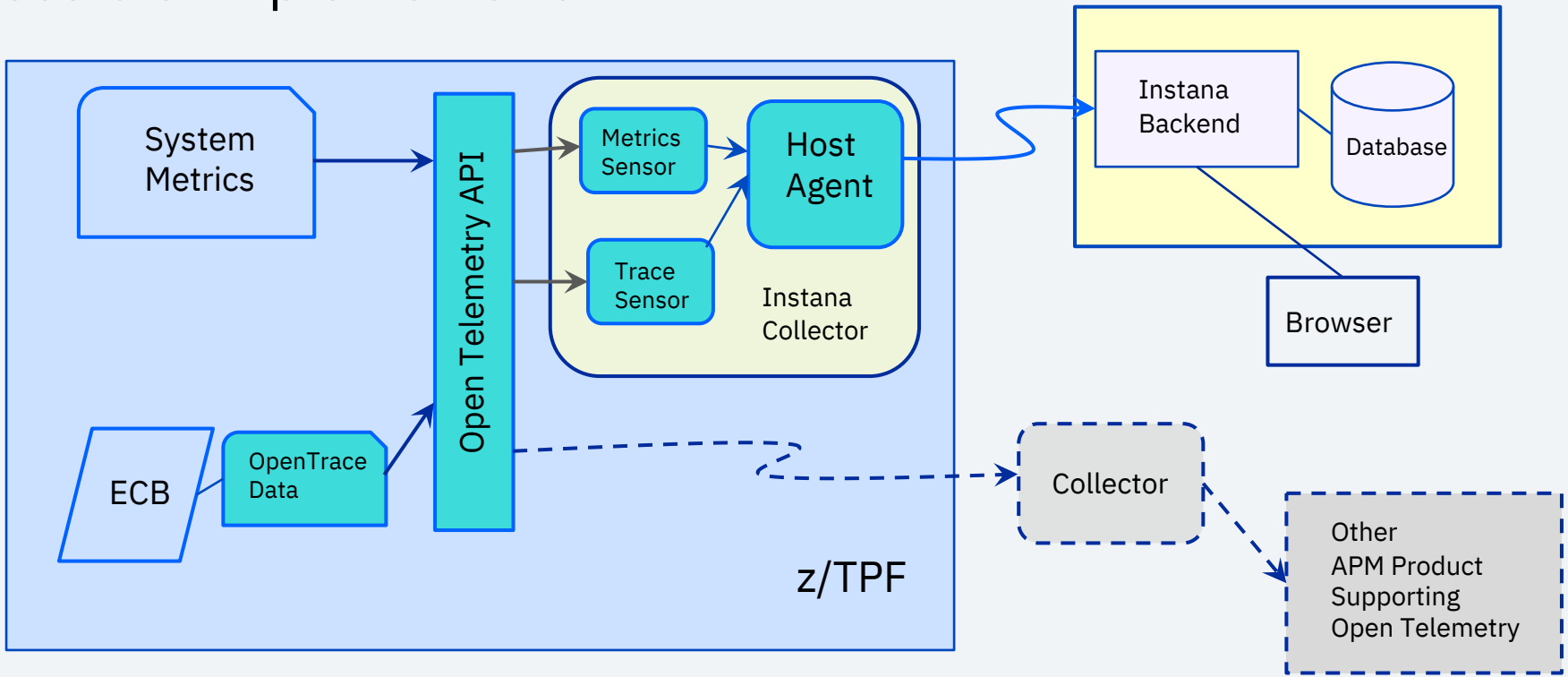
● – Strongly Differentiated Capability
◑ – Average Value
○ – Feature or Benefit not Present
◕ – Differentiated Capability
◔ – Less Value than Normal

Source: https://www.apmexperts.com/observability/ranking-the-observability-offerings/

Value Statement

A coverage programmer can view all the different components of an end-to-end transaction on a single dashboard, such that the component causing the problem can be identified in a matter of minutes.

Possible Implementation



Note: Overhead to z/TPF is expected to be minimal and will support sampling mode.
Collection process would be TE eligible

Call for Sponsor Users

Will be looking for Sponsor Users to assist in design and implementation, targeting the following personas:

- Application Architects
- Enterprise operators and coverage
- z/TPF developers

If you would like to be involved, contact:
Colette Manoni (cmast@us.ibm.com) or Dan Gritter (dgritter@us.ibm.com)

Thank you !