



**DATA
STREAMING**
WORLD TOUR

Real Time AI: Confluent Intelligence & watsonx Orchestrate

AI-powered support at streaming speed

IBM watsonx Orchestrate + Confluent

Data Streaming World Tour Paris · October 1, 2026 · 2:15–3:00 pm
François Estepa (IBM) · Daniel Petisme (Confluent)



What do we want to demonstrate today

01

AI at streaming speed is a real competitive advantage



Hand repetitive decisions to the stream, keep human intelligence for what really matters.

02

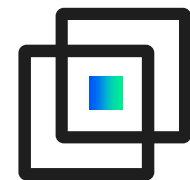
Confluent's AI capabilities are production-ready



Sentiment, forecasting, agents and MCP tools: all declared in Flink SQL, in the stream.

03

Confluent and watsonx Orchestrate integrate seamlessly



Events trigger Orchestrate agents, and live context flows back to them over MCP.

04

IBM Bob is the developer's best friend



The entire demo, from the app to the Terraform, was built with IBM's development assistant.



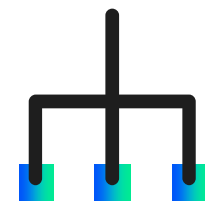
Would you cross the street based on yesterday's photo?

A support team receives hundreds of tickets. They are not all equal: some just need a standard answer, others flag an angry premium customer in the middle of an incident. Handling them first-come, first-served, on yesterday's data, means deciding blind.



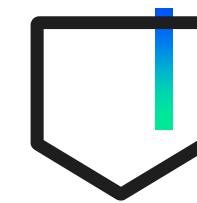
Triage everything, right away

Every ticket is enriched, qualified and decided the second it arrives, not in the next batch.



Escalate the right ticket

AI answers on its own when it's simple, and prepares the case when a human has to decide.

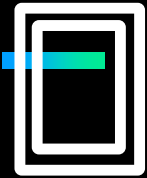


Act under control

Personal data masked, commercial policy applied, human approval before anything is sent.

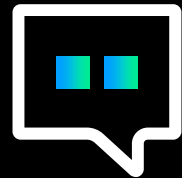


Your turn: create a ticket



01 Scan

The shop's support portal opens on your phone.



02 Write

A service, a subject, your problem. Be annoyed: it matters. Or hit "Generate a random ticket".



03 Watch

Your ticket shows up on screen within seconds, triaged by AI.

E-Shop Support User#31 Silver

New Support Ticket
Tell us what's going on and we'll take care of it.

[Generate a random ticket](#)

Service
Select a service

Subject
Payment declined but charged 0/100

Details (optional)
I was charged but the checkout page showed the payment as declined. 0/500

Urgency
Low Medium High

[Submit Ticket](#)

[New Ticket](#) [My Tickets](#)

The portal on mobile

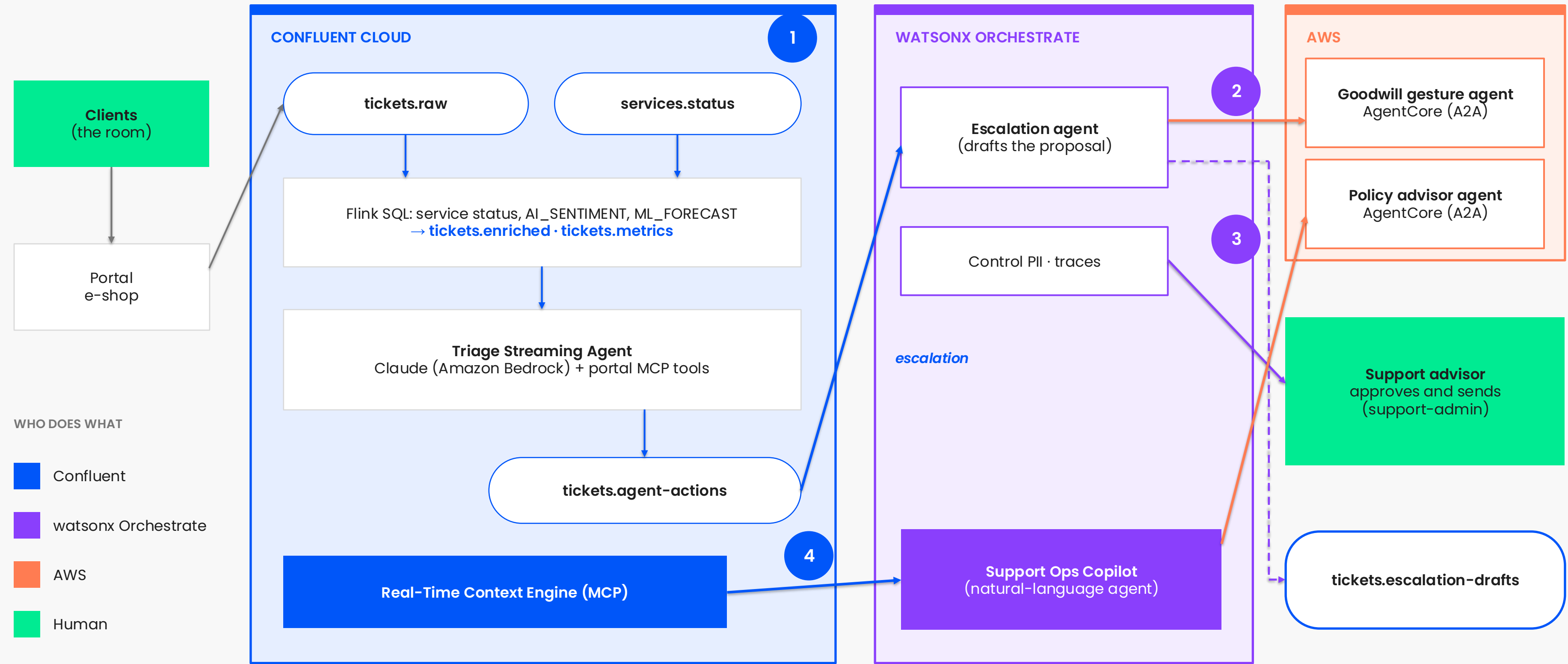


<https://ibm.biz/dswt2026-realtime-ai>

The shop's support portal



Four acts, one event stream



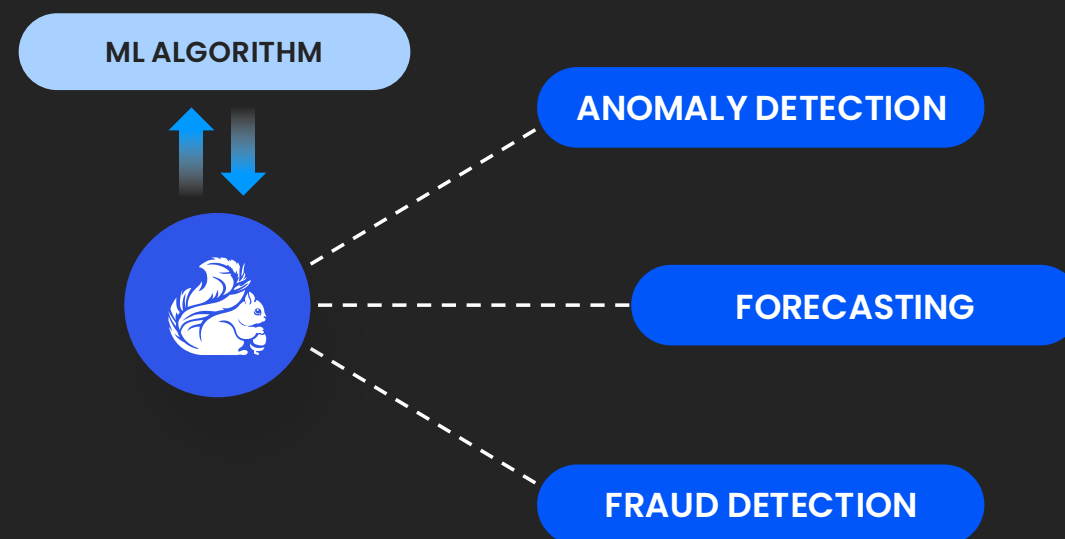


Confluent Intelligence

Fully managed service on Confluent Cloud for building real-time, context-rich, and replayable AI systems using Apache Kafka® and Apache Flink®, governed and observable by design.

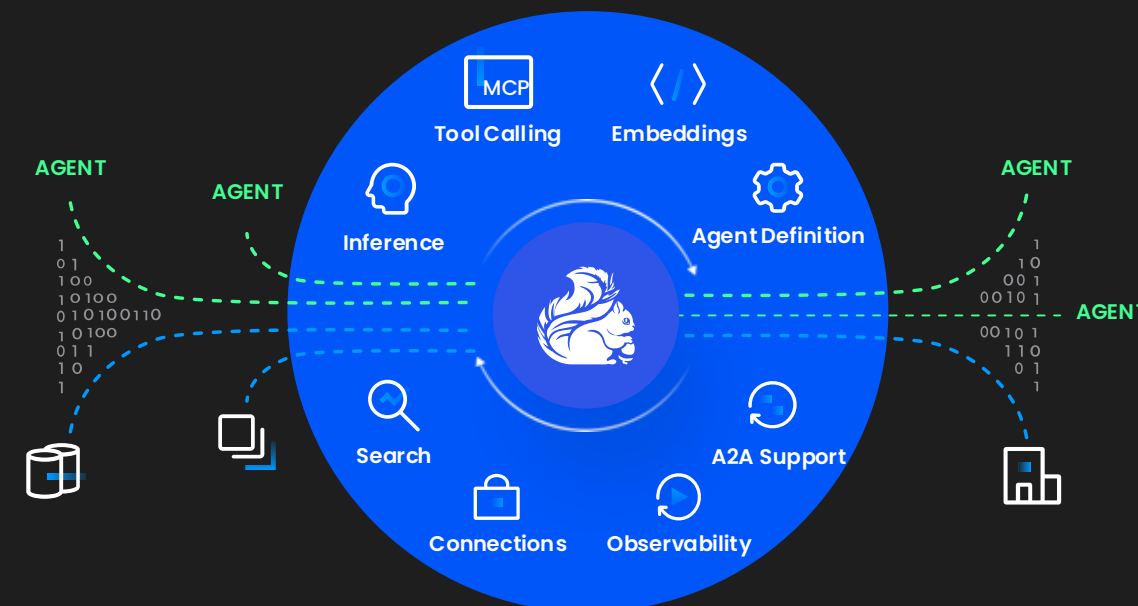
Built-in ML Functions

Perform anomaly and fraud detection,
forecasting, and more on streaming data



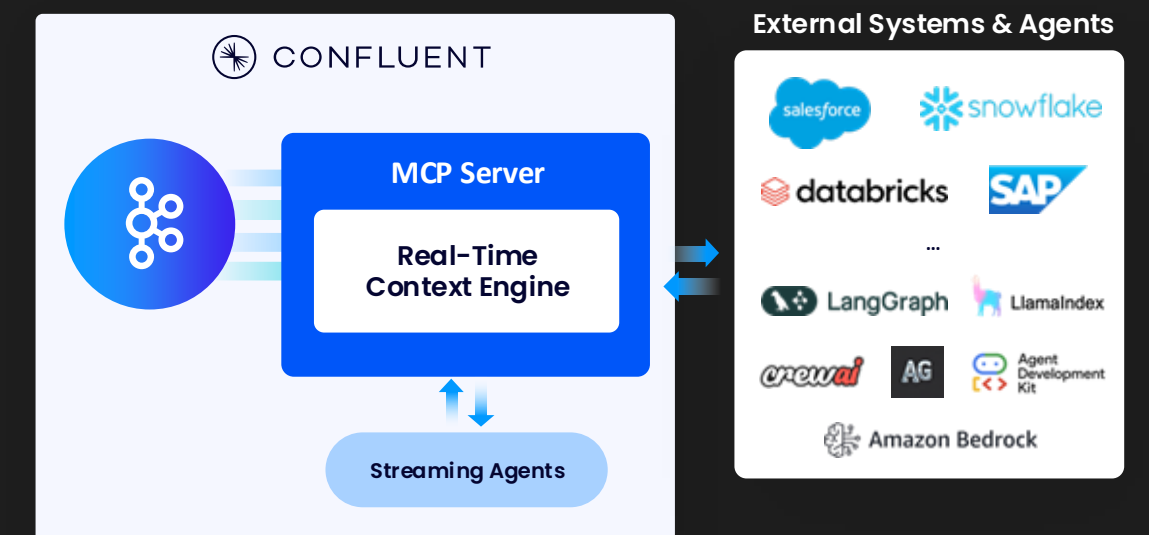
Streaming Agents

Build, deploy, and orchestrate event-driven agents on Flink



Real-Time Context Engine

Serve fresh context to
any AI agent or application





Confluent triages every ticket, in seconds

01 Enrich

Temporal join with the live status of services (login, payment, tracking...)

02 Understand

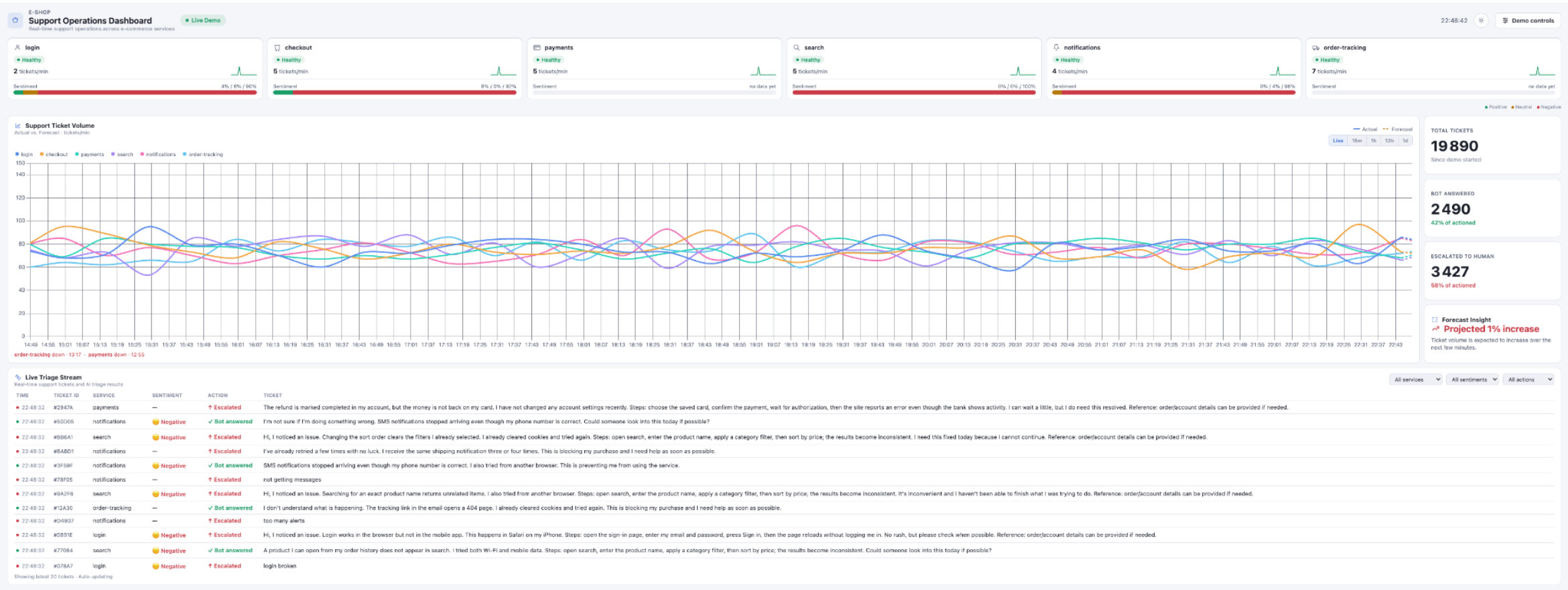
AI_SENTIMENT in Flink SQL: happy, neutral, annoyed

03 Anticipate

ML_FORECAST: 15-minute load forecast per service

04 Decide

Streaming Agent (Claude via Bedrock) + MCP tools: knowledge base, customer tier, history



Outcome

Automatic answer when it's simple, a reasoned escalation when a human needs to step in.

The Live Triage Stream shows every decision the moment it's made.



An AI agent, declared in a few lines of SQL

```
-- MCP tools exposed by the portal
CREATE TOOL get_account_context_tool
  USING CONNECTION portal_mcp_connection;

-- The agent: a Bedrock model, a prompt, tools
CREATE AGENT triage_agent
  USING MODEL triage_agent_model -- Claude Sonnet
  USING PROMPT 'You are a support triage agent...'
  USING TOOLS search_known_issues_tool,
              get_account_context_tool;

-- Every enriched ticket triggers the agent
CREATE TABLE `tickets.agent-actions` AS
SELECT * FROM `tickets.enriched`,
LATERAL TABLE(AI_RUN_AGENT('triage_agent', ...));
```

CHECKOUT

AUTO_RESOLVE

Promo code breaks checkout

« This is a known issue in our checkout flow... please sign out completely, then sign back in, rebuild your shopping cart... »

NOTIFICATIONS

ESCALATE

Duplicate shipping notifications

« ...doesn't match our standard troubleshooting steps, I'm escalating your ticket to our technical support team... »



The Streaming agent runs continuously on Confluent Cloud

triage_agent

More

Edit

Run

Sep. 28 2026 03:09 PM

Overview

Deployments

Configurations

Catalog
fe-dswt2026-env

Database
fe-dswt2026-cluster

Model
triage_agent_model

Recent deployments

View all deployments

Target	Status	Created	Account	
fe-dswt2026-run-triage-agent-4a019e42	Running	Sep. 28 2026 03:10 PM	fe-dswt2026-app	

Prompt

View full prompt

You are a support triage agent for an e-commerce platform. The input contains "User ID: <id>" then "Subject: <subject>" then "Service: <service>" then "Service Status: <status>" then optionally "Service Status Description: <description>" then "Ticket: <text>". Service is the name of...

Tools

View all tools

Tool	Type
get_account_context_tool	Model Context Protocol
search_known_issues_tool	Model Context Protocol



The AI agent reacts to any new event

fe-dswt2026-run-triage-agent-4a019e42

MoreStopResume

Running

CREATE TABLE AS

Service account: fe-dswt2026-app

Sep. 28 2026 03:10 PM

Activity

Logs

Query profiler

Settings

Catalog

Database

Compute pool

Provider

Region

Scaling status

Statement CFU

Messages behind (total)

Messages in (per min)

Messages out

fe-dswt2026-...

fe-dswt2026-...

fe-dswt2026-f...

aws

us-east-1

Fine

1

0

0

0

Table	# Messages be... (total)	Messages in (per min)	Messages out (per min)
tickets.enriched fe-dswt2026-cluster	0 ✓ Not behind	0 Watermark: --	
tickets.agent-act... fe-dswt2026-cluster	—	—	Water...

SQL

Edit SQL

```
1 CREATE TABLE `tickets.agent-actions` WITH (  
2   `value.format` = `json-registry`  
3 ) AS  
4 SELECT  
5   *,  
6   COALESCE(  
7     LOWER(REGEXP_EXTRACT(response, `(?)DECISION: *(AUTO_RESOLVE|ESCALATE)`,  
8     1)),  
9     `escalate`  
10  ) AS decision,  
11  COALESCE(  
12    NULLIF(TRIM(REGEXP_EXTRACT(response, `(?)MESSAGE: *(.*)`, 1)), ''),  
13    response  
14  ) AS message  
15 FROM `tickets.enriched` /*+ OPTIONS('scan.startup.mode' = 'latest-offset') */,  
16 LATERAL TABLE(AI_RUN_AGENT(  
17   `triage_agent`,  
18   CONCAT(  
19     `User ID: `, CAST(`tickets.enriched`.user_id AS STRING), CHR(10),
```

The triage Flink statement: always on, it calls the agent for every ticket in tickets.enriched



Orchestrate handles escalations

- 01

The escalated ticket triggers the agent

As soon as Flink decides, the Orchestrate escalation agent receives the ticket's context.
- 02

It summarizes, without personal data

Name, email, phone, order number: none of it leaves this summary.
- 03

It consults the AWS agent

The Customer Relations agent, hosted on Bedrock AgentCore, applies the goodwill gesture policy.
- 04

An advisor approves

Reply, gesture and context note land in the console. One click to reuse, adjust, send.

E-Shop Support

Support Admin

Manage escalated tickets and support customers.

[Back to queue](#)

Can't log in

Escalated

The page keeps reloading after I enter my password.

Login Medium 1m waiting #87325

Escalation reason

Thank you for reporting this login issue. We understand that the page keeps reloading after you enter your password, which is preventing you from accessing your account. Our technical team will investigate this immediately to identify the cause and get you back into your account as quickly as possible. You should hear from us shortly with a resolution.

Proposition watsonx Orchestrate

Agent d'escalade · geste commercial calculé par l'agent AWS · à valider avant envoi

Prête · 16 S

RÉPONSE SUGGÉRÉE

Nous avons bien pris connaissance de votre problème de connexion, la page se recharge sans vous permettre d'accéder à votre compte. Ce type d'incident peut parfois survenir même lorsque le service est opérationnel. Afin de vous aider au mieux, nous vous proposons un bon d'achat de 10 € ainsi que la livraison gratuite sur votre prochaine commande. Un conseiller vous contactera rapidement si vous avez besoin d'une assistance supplémentaire.

GESTE COMMERCIAL PROPOSÉ

voucher_eur : 10 €

extras : free_shipping

justification : Client gold, gravité faible (service opérationnel, problème de connexion). Geste de base : 5 € (section 3). Problème récurrent sur le même service : ajout de 5 € et livraison gratuite (section 4). Total : 10 €.



In Orchestrate: the agent and its reply

Agent d'escalade (Relatio...

Live

Build

Evaluate

Behavior

Knowledge

Tools

Agents

Notifications: 1 Dismissed | 1 total

Legal Notice

View all

Profile

Agent name

Agent d'escalade (Relation client)

Model

GPT-OSS 120B — OpenAI (via Groq)

Description

Prépare une proposition de réponse pour chaque ticket e-commerce escaladé par l'agent de triage. Consulte l'agent AWS de l'équipe Relation client pour le geste commercial, sans lui transmettre de données personnelles, et présente un brouillon que l'opérateur humain valide dans la console support-admin avant tout envoi

Instructions

Vous êtes l'assistant de support de niveau 2 d'une boutique en ligne. Chaque demande concerne UN ticket que l'agent de triage a jugé trop sensible pour une réponse automatique. Vous ne répondez jamais vous-même au client : vous préparez un brouillon qu'un conseiller humain valide et envoi

The escalation agent in the builder

You 10:25 PM

Ticket escaladé #[MASQUÉ]

Tier : gold

Service : login

État du service : operational

Problème récurrent sur ce service : true

Objet : Can't log in

Texte : The page keeps reloading after I enter my password.

Message de l'agent de triage : Thank you for reporting this login issue. We understand that the page keeps reloading after you enter your password, which is preventing you from accessing your account. Our technical team will investigate this immediately to identify the cause and get you back into your account as quickly as possible. You should hear from us shortly with a resolution.

Agent d'escalade (Relation client) 10:25 PM

Show Reasoning

RÉPONSE SUGGÉRÉE :

Nous avons bien pris connaissance de votre problème de connexion, la page se recharge sans vous permettre d'accéder à votre compte. Ce type d'incident peut parfois survenir même lorsque le service est opérationnel. Afin de vous aider au mieux, nous vous proposons un bon d'achat de 10 € ainsi que la livraison gratuite sur votre prochaine commande. Un conseiller vous contactera rapidement si vous avez besoin d'une assistance supplémentaire.

GESTE COMMERCIAL PROPOSÉ :

voucher_eur : 10 €

extras : free_shipping

justification : Client gold, gravité faible (service opérationnel, problème de connexion). Geste de base : 5 € (section 3). Problème récurrent sur le même service : ajout de 5 € et livraison gratuite (section 4). Total : 10 €.

NOTE POUR LE CONSEILLER :

- Escalade en raison d'un problème de connexion persistant malgré un service opérationnel.
- Aucun besoin d'approbation humaine supplémentaire (requires_human_approval = false).
- Vérifier que le bon de 10 € et la livraison gratuite sont bien appliqués avant d'envoyer la réponse.

Its reply to ticket #87325



Orchestrate governs agents, wherever they live



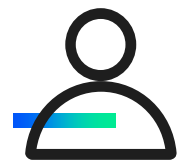
Control PII

Emails, phone numbers, cards, names: masked before the model reads the ticket. Visible in the traces.



Federated agents

The AWS agent is called over A2A, authenticated with OAuth. It only receives an anonymized summary.



Human in the loop

No agent has a tool to send a reply. Sending stays in the advisor's hands.



Everything is traced

Every proposal is published back to Confluent, with a schema in Schema Registry.

Message details

Timestamp	Offset	Partition
2026-09-28T20:25:20.946Z	0	0

Key	Value	Headers
-----	-------	---------

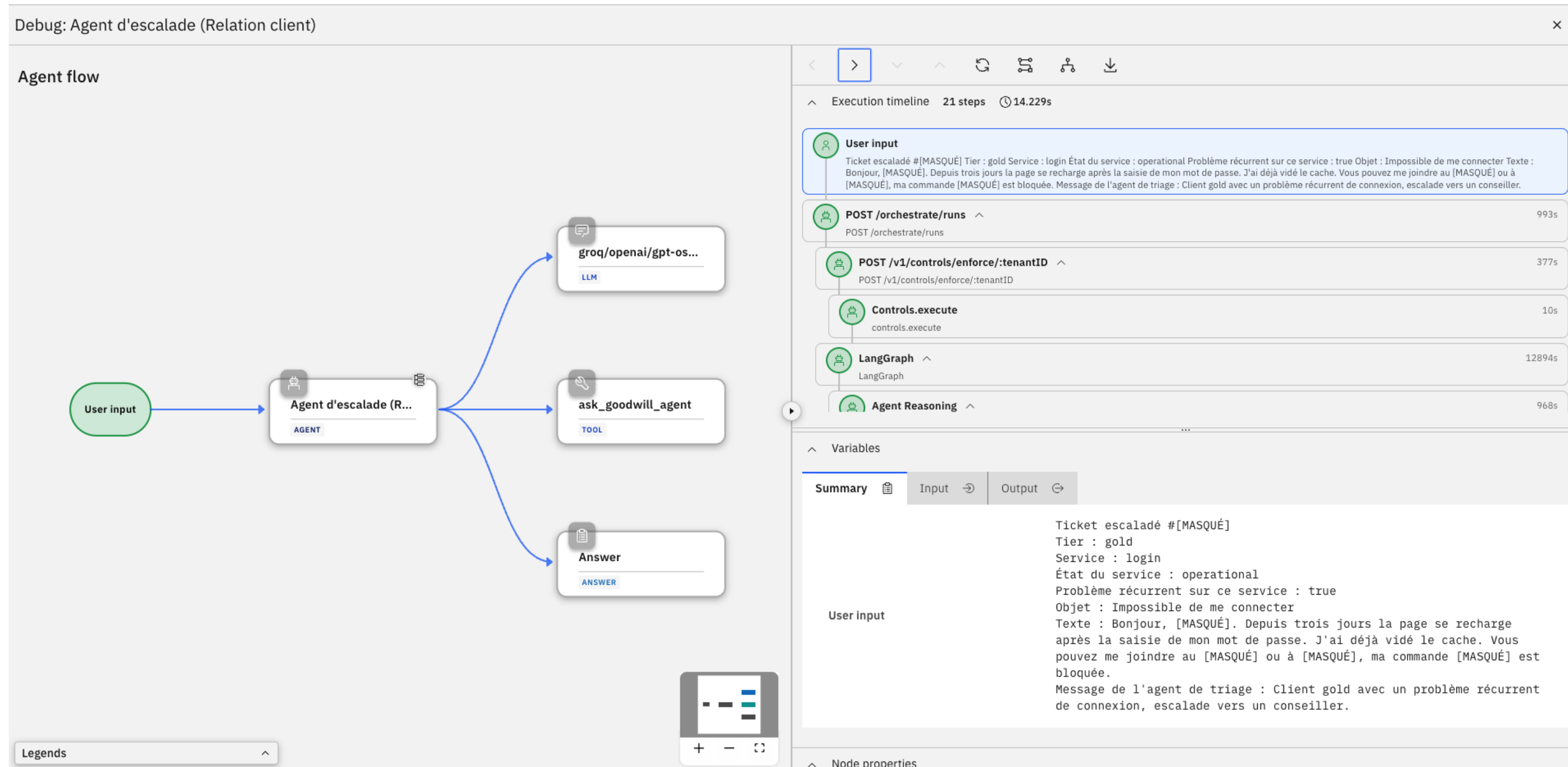
Schema ID: 100010

```
1 {
2   "ticket_id": "87325505-a7ae-4643-adab-c31f424e7a9f",
3   "ticket_number": "#87325",
4   "service": "login",
5   "tier": "gold",
6   "service_status": "operational",
7   "recurring_issue_same_service": true,
8   "reponse_suggeree": "Nous avons bien pris connaissance de votre
9     problème de connexion, la page se recharge sans vous permettre
10    d'accéder à votre compte. Ce type d'incident peut parfois survenir
11    même lorsque le service est opérationnel. Afin de vous aider au
12    mieux, nous vous proposons un bon d'achat de 10 € ainsi que la
13    livraison gratuite sur votre prochaine commande. Un conseiller
14    vous contactera rapidement si vous avez besoin d'une assistance
15    supplémentaire.",
16   "geste_commercial": "voucher_eur : 10 € \nextras :
17     free_shipping \njustification : Client gold, gravité faible
18     (service opérationnel, problème de connexion). Geste de base : 5 €
19     (section 3). Problème récurrent sur le même service : ajout de 5 €
20     et livraison gratuite (section 4). Total : 10 €.",
21   "note_conseiller": "- Escalade en raison d'un problème de
22     connexion persistant malgré un service opérationnel. \n- Aucun
23     besoin d'approbation humaine supplémentaire
24     (requires_human_approval = false). \n- Vérifier que le bon de
25     10 € et la livraison gratuite sont bien appliqués avant d'envoyer
26     la réponse.",
27   "voucher_eur": 10.0,
28   "requires_human_approval": false,
29   "duration_seconds": 16.0,
30   "agent": "watsonx Orchestrate escalation_agent",
31   "drafted_at": "2026-09-28T20:25:20.494459Z"
32 }
```

Message published to tickets.escalation-drafts



The proof: data masked before the model



Escalation agent trace: the Controls.execute step replaces name, phone, email and order number with [MASQUÉ] (masked)



Governing at scale: agent directory and controls

Agent directory is a preview feature. Functionality may change before general availability.

AI Gateway / Agent directory

Agent directory

Manage external AI agents available on the platform.

Add agent +

Agents - External

1

Search agents

Display name	Protocol	Imported	Agent Id
Geste commercial (AWS)	external_chat/A2A/0.3.0	28 sept. 2026, 18:00	7c9d185d-bd07-4474-99c8-afdbc4bed102

Items per page: 10

1-1 of 1 items

1 of 1 page

Agent directory: the AWS agent registered with A2A 0.3.0

Controls summary

Total controls

1

Agent controls

1

Tool controls

No tool controls

Model controls

No model controls

Recent controls

View all

Search controls

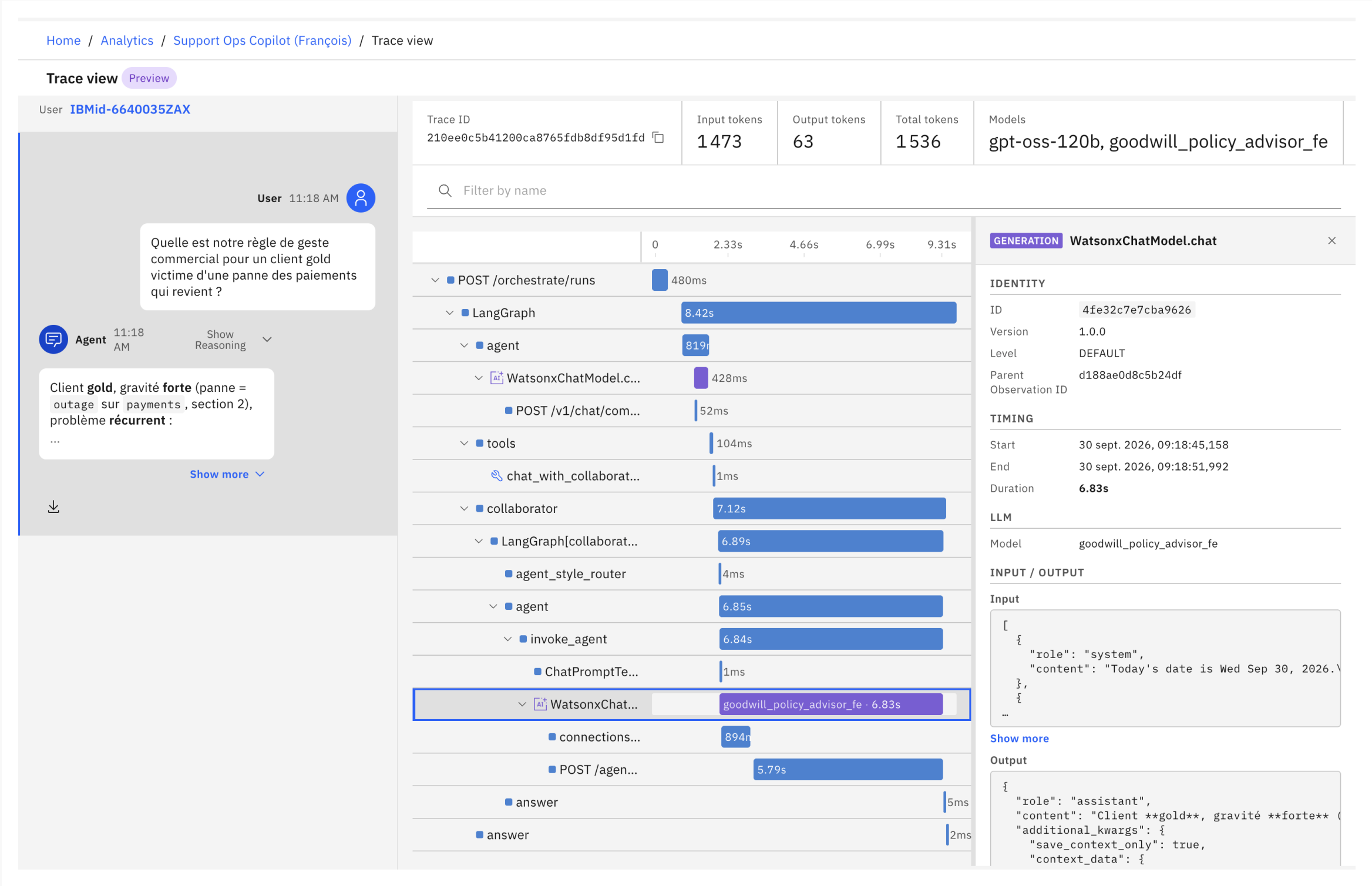
Control	Type	Asset type	Enforcement	Created on
PII - entrée de l'agent d'escalade	PII Filter	agent	INPUT	28 sept. 2026

Security and Risk: the PII Filter control, applied on input

A single point of control: external agents are registered and authenticated, controls apply per agent, tool or model, and everything is visible from the same dashboard.



An external agent, traced end to end



7.1 s
in the AWS collaborator, inside one copilot conversation

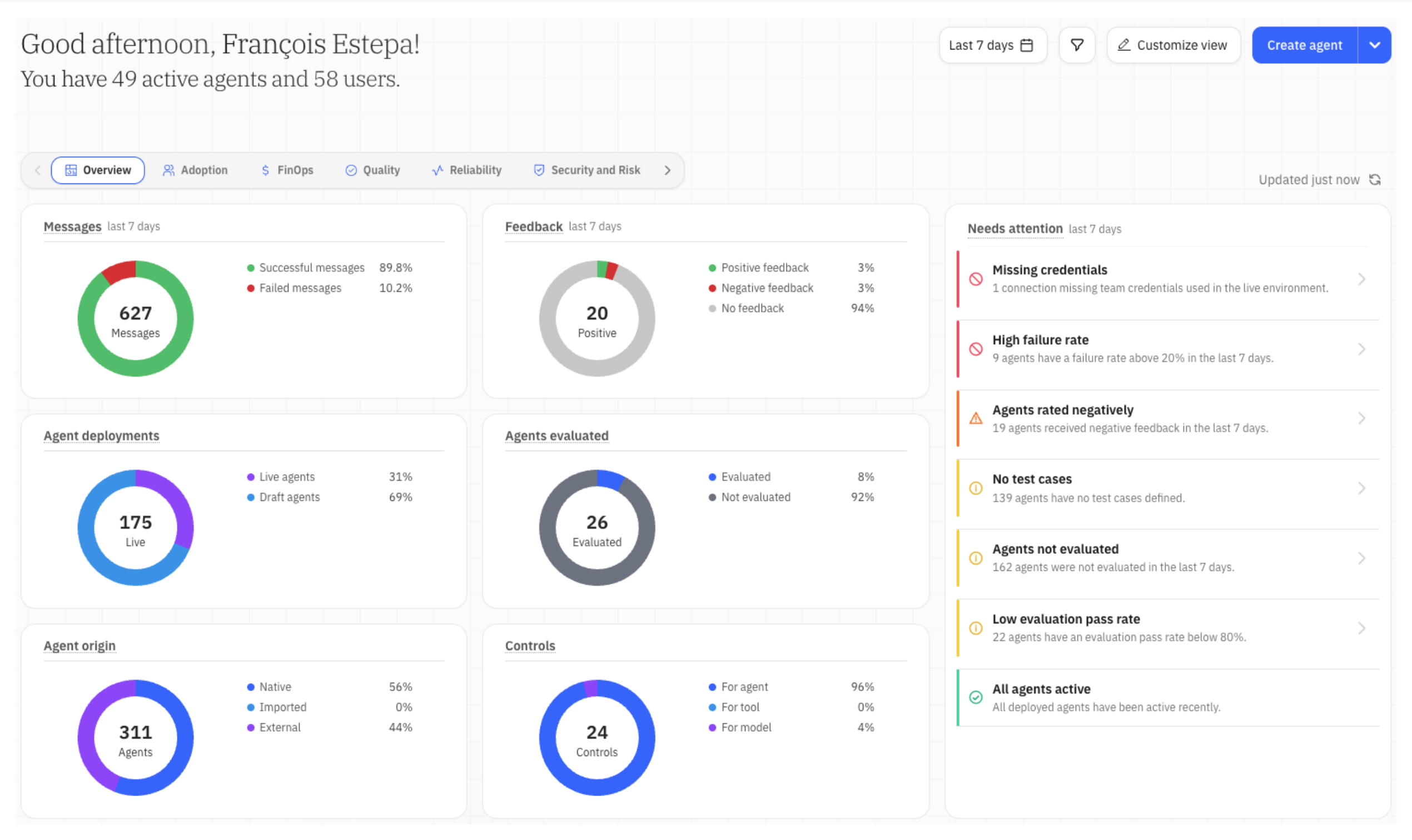
894 ms
OAuth token fetched by Orchestrate from the connection, never by the agent

5.8 s
A2A call to AgentCore, the policy advisor answers

Trace view of the Support Ops Copilot: the question is handed over to the AWS policy advisor over A2A



At scale: one control plane for hundreds of agents





A copilot plugged into live support

Confluent exposes its topics over MCP through the Real-Time Context Engine, enabled topic by topic. In Orchestrate, the IBM Confluent Real-Time Context Engine server is added from the catalog: the agent reads the current state of the business, not yesterday's export.

When the question is about policy rather than data, the copilot hands it over to the AWS policy advisor, a collaborator called over A2A.

Catalog / IBM Confluent Real-Time Context Engine

IBM Confluent Real-Time Context Engine

External

MCP Servers

Documentation available

Overview

Configuration details

Learn more about this MCP server

This is a remote MCP (Model Context Protocol) server. It exposes tools that agents can discover and invoke over a network transport. [Learn more](#)

Description

IBM Confluent Real-Time Context Engine (RTCE) MCP Server provides real-time, materialized data from Confluent Cloud Kafka topics. Use this server to discover which topics are enabled for context serving, inspect topic schemas and metadata, and query live topic data using key lookups, filters, ranges, and compound predicates. This server requires topics to have a registered schema and RTCE enabled, and serves continuously refreshed, low-latency context without direct interaction with Kafka or Flink. All tools are read-only and scoped to your API key permissions.

Details

Published

By: IBM

Categories

IT / Productivity

Version

1.0.0 (latest)

[View recent versions](#)

Last updated

14/09/2026 07:16:25

Languages supported

English

Resources

[Support](#)

[Documentation](#)

[Terms and Conditions](#)

IBM watsonx Orchestrate

EN Contact Sales Buy Now 20 trial days left DP

Support Ops Copilot

Build Evaluate Deploy

Behavior Knowledge Tools Agents

Give your agent access to your organization's tool to help it accomplish tasks.

Search for a tool

Add Tool +

rtce_dpetisme_dswt2026:getMe...

STEP 2: Call BEFORE queryData. Returns the schema for a topic -- column names, data types, and key information. Use this to learn what columns exist and their types. Alwa...

Type MCP

Edit details

rtce_dpetisme_dswt2026:lstTo...

STEP 1: Start here. Returns the list of available topics (tables) in the real-time data store. Call this first to discover what data is available, then call getMetadata() on each...

Type MCP

Edit details

rtce_dpetisme_dswt2026:query...

STEP 3: Run a SQL SELECT query against a topic. REQUIRES calling getMetadata() first to learn the schema -- never guess column names. Column references -- best practices...

Type MCP

Edit details

Draft Preview

Save as test

Running in debug mode

Today

You 11:39 PM

What topics you can read ?

Support Ops Copilot 11:39 PM | Show Reasoning

I can read the **tickets.agent-actions** topic (the automated triage-agent decision stream). No other topics are currently listed as available in the real-time data store.

You 11:39 PM

what have been the last 5 actions taken ?

Support Ops Copilot 11:39 PM | Show Reasoning

SER_ID	SERVICE	SUBJECT	URGENCY_SELF_RATED	STATUS	ACTION (from RESPONSE)
.071	notifications	Getting duplicate notifications	medium	SUCCESS	AUTO_RESOLVE -- instructions to re-enable notification permissions and restart the app.
.030	notifications	Order confirmation email missing	low	SUCCESS	AUTO_RESOLVE -- verify phone number, carrier blocking, and coverage.
.103	order-tracking	Delivery date keeps changing	low	SUCCESS	ESCALATE -- ticket handed to fulfillment team for package investigation.
.047	order-tracking	Tracking link is broken	medium	SUCCESS	ESCALATE -- sent to specialized team to probe carrier tracking issue.
.040	order-tracking	Tracking link is broken	low	SUCCESS	ESCALATE -- escalated to order-fulfillment

Save as test

Support Ops Copilot in watsonx Orchestrate: a real demo session



Each does its job, one system



Confluent

Process data in real time

- Kafka: every ticket becomes an event
- Flink SQL: enrichment, sentiment, forecasting
- Streaming Agents and MCP tools in the stream
- Schema Registry: governed data



watsonx Orchestrate

Govern the agents that act

- Federated agents: Orchestrate, AWS and others
- Controls: personal data masked
- Human in the loop, end-to-end traces
- Business copilot plugged into the live stream

Confluent turns events into decisions. Orchestrate turns decisions into controlled actions.



From ticket to approved reply in seconds

< 5 min

for a complete escalation proposal: reply, goodwill gesture, note to the advisor

measured in the demo, Sept. 28, 2026

0

personal data sent to the AWS agent: it only receives an anonymized summary

by design

100%

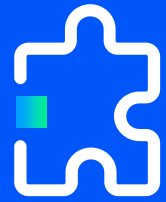
of escalated replies sent by a human, never by an agent

by design

For you: share of tickets resolved without a human, time to first response, time saved per advisor. These figures depend on your volumes: they are measured in a POC.

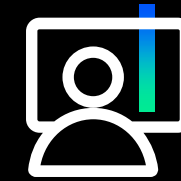


Your use case, on your data

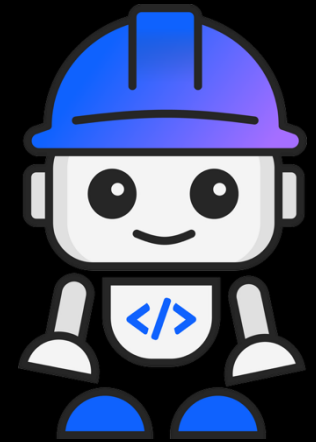


Workshop · less than 1 day

We identify the business event that is worth gold if known in under a minute, and we draw your blueprint.



Built with IBM Bob · 5 days



From the support portal to the Flink SQL and the Terraform, IBM Bob wrote and wired most of this demo: from idea to a live, deployed system in 5 days instead of weeks.

Support, fraud, maintenance, supply chain: the same pattern applies whenever an event needs to trigger a decision.



What did we demonstrate today

01

AI at streaming speed is a real competitive advantage



Hand repetitive decisions to the stream, keep human intelligence for what really matters.

02

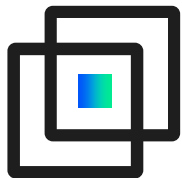
Confluent's AI capabilities are production-ready



Sentiment, forecasting, agents and MCP tools: all declared in Flink SQL, in the stream.

03

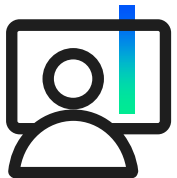
Confluent and watsonx Orchestrate integrate seamlessly



Events trigger Orchestrate agents, and live context flows back to them over MCP.

04

IBM Bob is the developer's best friend



The entire demo, from the app to the Terraform, was built with IBM's development assistant.

Thank you

Q&A