



Artificial Intelligence in Network Operations

From Reactive Operations to Autonomous Networks

Akhilesh Thakur

Principal Architect– HPE Networking

#apnic62

MUMBAI, INDIA

4 – 10 September 2026



Agenda

01

The Case for AI in Networks

Typical use case



02

Agentic AI

Autonomous Network operations



03

MCP Server

Protocol making it possible



04

Practical Use Cases

MOP generation, MOP execution and Network Planner



05

AI End-to-End Architecture

A2A, MCP and AI Stack



The Problem: NOC Operations Today vs. AI-Powered Automation

● Today: Manual & Reactive



Thousands of alarms daily: Outages, flaps, degradation, and spikes overwhelm teams.



Siloed Systems: Engineers swivel between FM dashboards, PM KPIs, topology, and logs.



Manual Triage: Time wasted validating auto-clearing alarms before root-cause analysis.



Result: High fatigue, slow MTTR, and inconsistent remediation.

● AI-Powered: Zero-Touch Automation



Multi-source validation: Agent autonomously checks OSS/BSS via tool calls.



Automated Healing: Instant root-cause analysis and execution of remediation scripts.



Pattern Recognition: Filters self-recovering alarms using historical data.

Metric	Today	With AI
Response Time	Hours	Seconds
OpEx & MTTR	High / Slow	Reduced / Fast



End State: NOC shifts from firefighting to supervising automation.

AI Is Only As Good As Your Network Data

⚠ Most AI projects fail because of data quality — not model limitations. Fix the data foundation first.

Control Plane	Data Plane	Infrastructure	Business Layer	External Context
• BGP tables • IS-IS/OSPF state • MPLS labels • Route analytics • Prefix changes	• NetFlow/IPFIX • Queue metrics • Latency / jitter • Packet loss • Traffic matrix	• CPU / Memory • Interface util. • Optical power • Fan / PSU • Hardware errors	• SLA compliance • Subscriber KPIs • Service inventory • Customer tickets • Revenue impact	• Maintenance windows • Weather / outages • BGP route leaks • Vendor advisories • Seasonal events
💡 Tells you what the network THINKS is happening	💡 What is ACTUALLY flowing through the network	💡 Optical power degradation = early SFP failure warning	💡 Connects faults to customer & business impact	💡 Explains anomalies AI might falsely flag

DATA PIPELINE: Network Devices → Telemetry Collector → Data Lake → Feature Engineering → AI Models → Decision Engine → Automated Actions

💡 **APAC:** A mid-size ISP generates 2–8 TB operational data/day — humans cannot process this in real time. AI must.

💡 Analogy: You cannot bake a great cake with bad ingredients — no matter how skilled the baker. AI models trained on dirty data produce unreliable outputs.

Agentic AI: Use Cases for Service Providers

Optimize



Task automation & Documentation

MOP generation, Network audits, Security posture analysis, Training



Dynamic Network optimization

Failure impact, Automated re-routing, BGP hijack mitigation



Information distilling & Faster RCA

Multi-domain, multi-vendor event correlation; Needle in the haystack



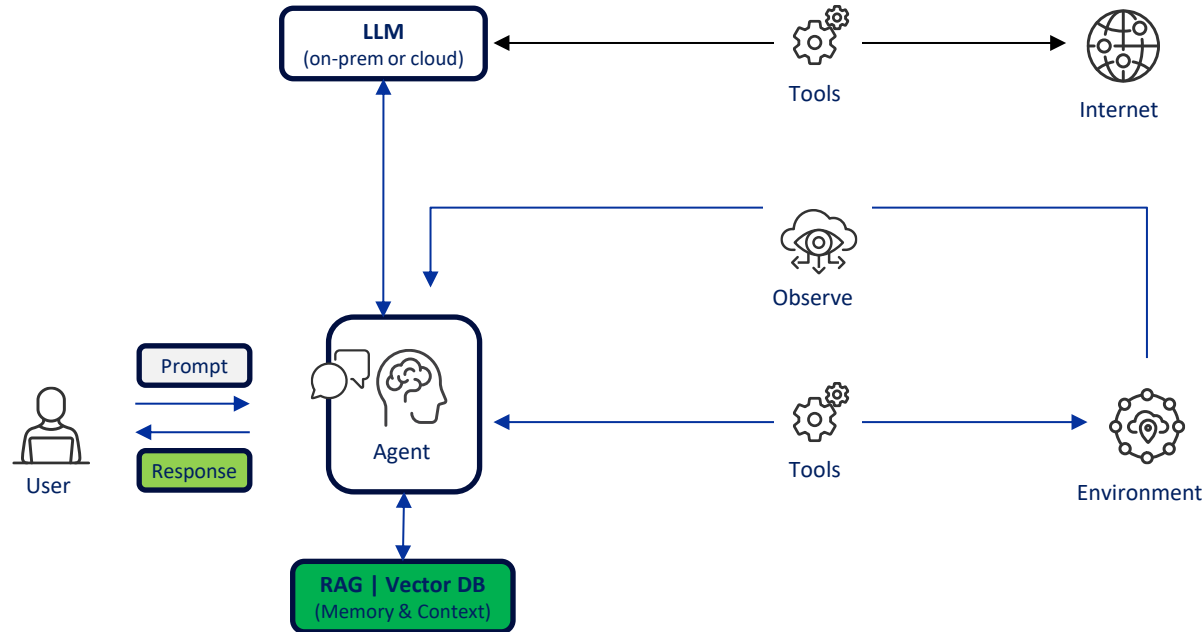
Digital Twins

Capacity planning, What-If analysis, MOP validation, Feature testing

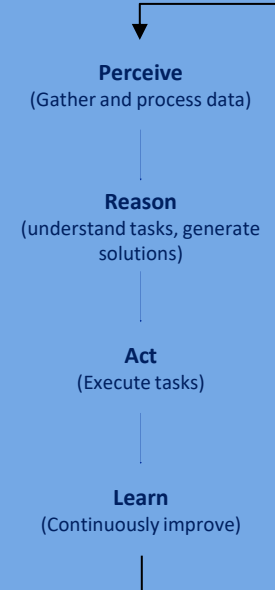
Agentic AI

Beyond traditional AI:

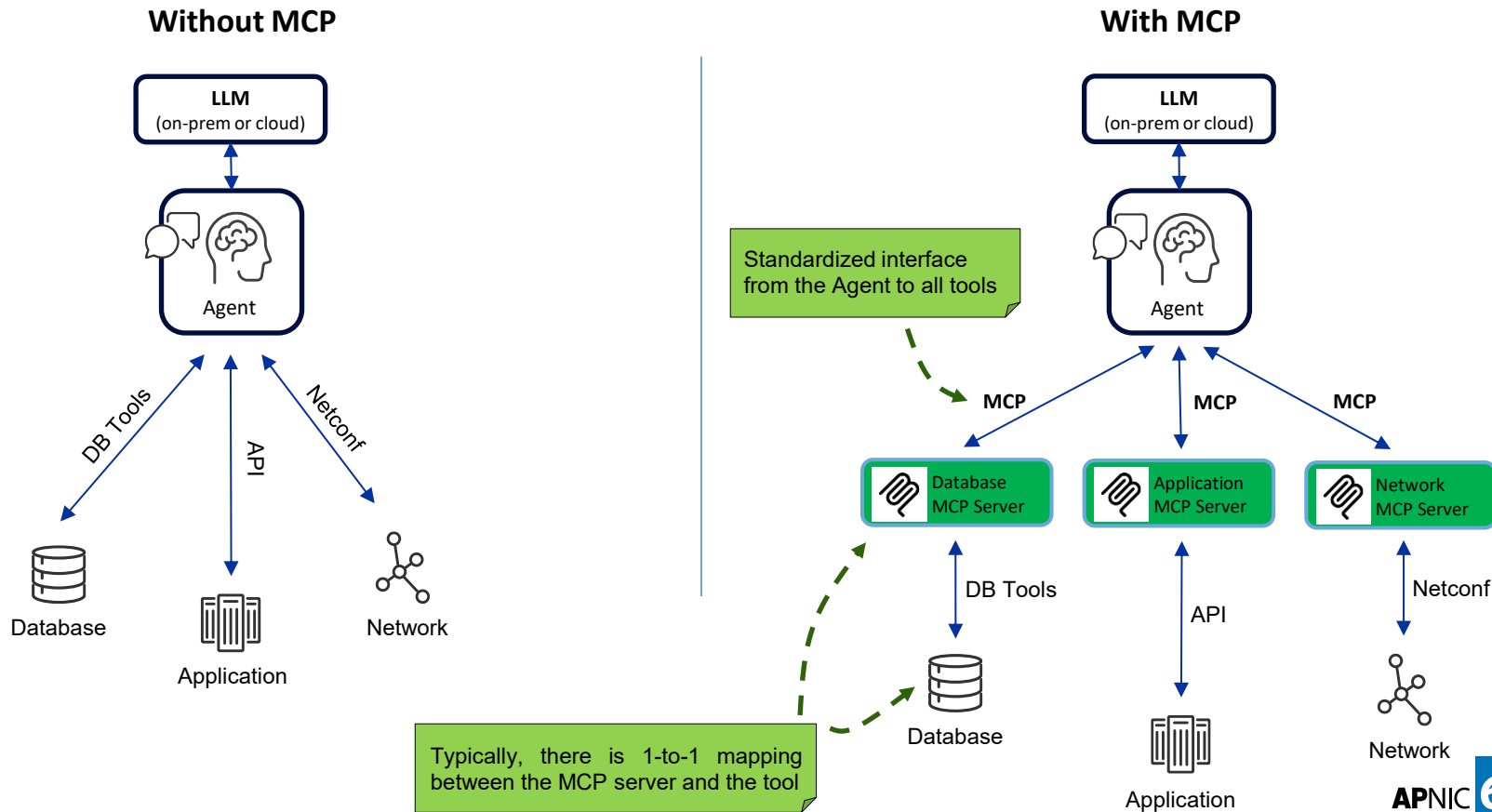
- Agentic AI doesn't just respond—it **plans, executes, and adapts** to achieve complex objectives.
- **Decompose complex goals** into actionable steps,
- **Maintain context** across multiple interactions, and iteratively refine its approach based on results.



Agentic AI process



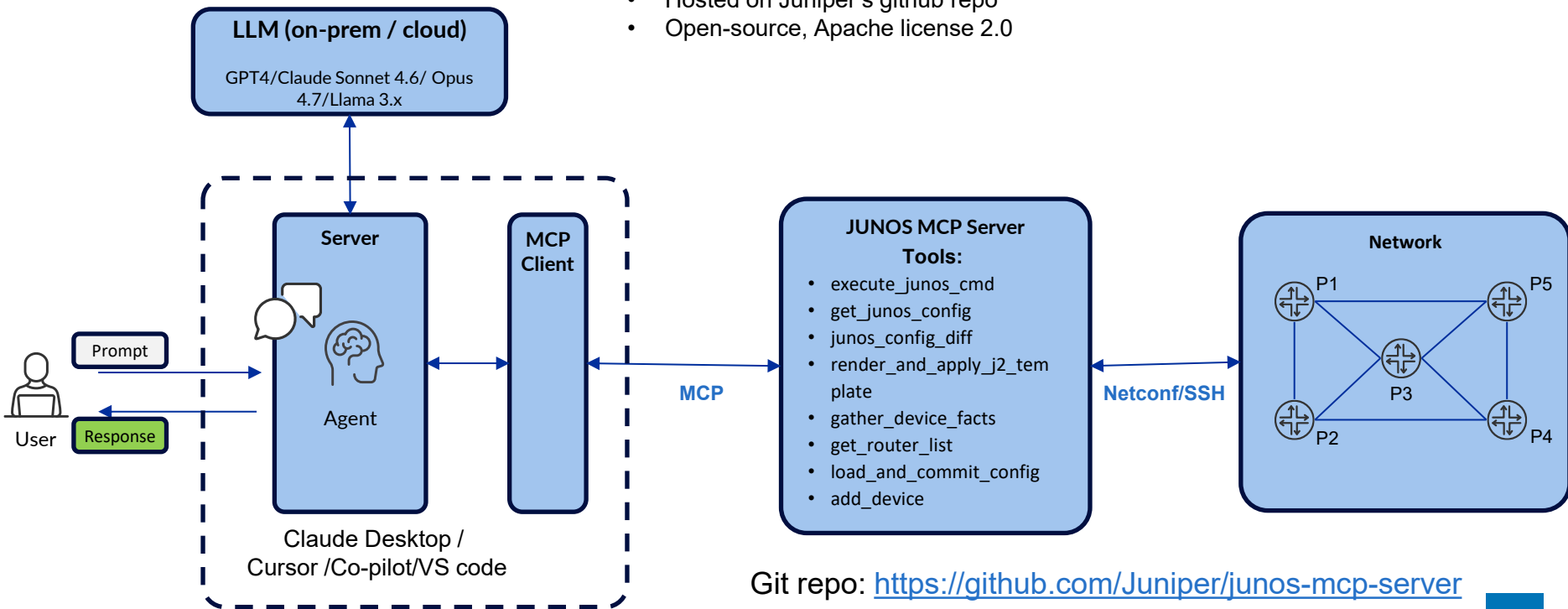
Model Context Protocol (MCP)



Example Junos MCP server

MCP server for JUNOS devices

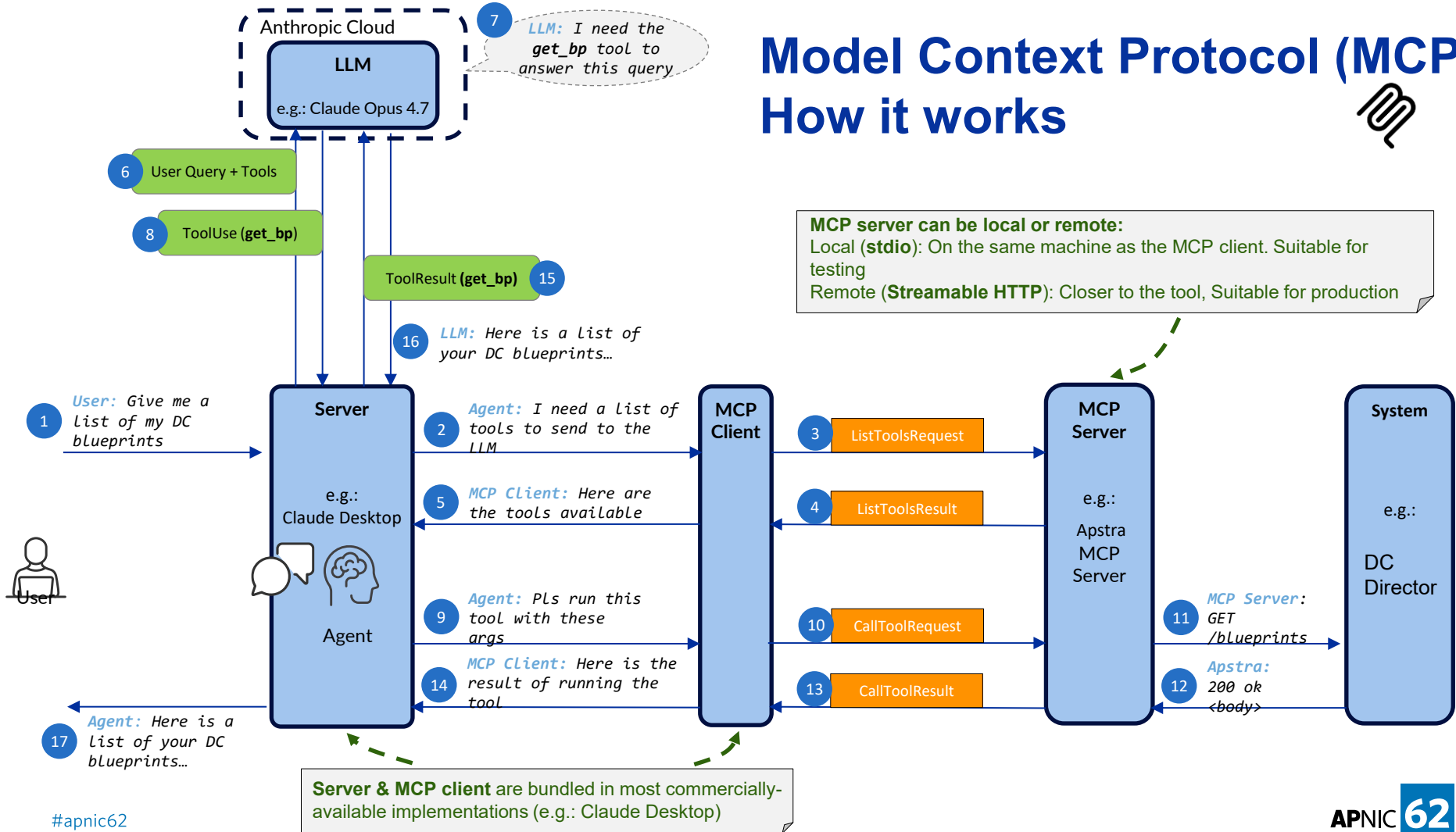
- Provides tools to List devices; Get device facts & config; Execute JUNOS commands & commit config; add_device, etc
- Hosted on Juniper's github repo
- Open-source, Apache license 2.0



Git repo: <https://github.com/Juniper/junos-mcp-server>

Model Context Protocol (MCP)

How it works



Use-cases

Use-case 1: Network infrastructure Assessment, Config Audit (BGP, SRv6)

- **Anatomy of an Effective Network Assessment Prompt**
- **Provide Structured Instructions:** Guide the AI with a clear, step-by-step process rather than a single, broad command.
- **Decompose the Objective into Granular Tasks:** Break down the high-level goal into specific areas of analysis. For example:
 - ISIS Topology & Adjacency Review
 - BGP Peering & Policy Audit
 - SRv6 Path Validation
 - VPN Service Configuration Check
 - Summary of Findings & Recommendations
- **Define Negative Constraints (The "Don'ts"):** Explicitly state what the agent must *not* do or include in its output. This is a critical guardrail to mitigate model hallucination and keep the analysis focused.

Analyse the entire network by checking the Juniper vMX logical systems in the POP1(192.168.1.221) and POP2(192.168.1.222) Create an extensive report with graphics, html but tabulation and illustrations , i want to download this report as html.

First chat/prompt should define the Part 1 and 2 subsequently other prompt should finish parts 3 & 4.

Requirements:

1. Detailed graphical topologies showing all nodes and how they're connected
2. OSPF Database and ISIS database details with ISO addressing with realtime examples.
3. Loopback addressing information
4. IBGP and eBGP peering diagrams (including Internet peering)
5. Route Reflector connection and BGP topology visualisation
6. PDF-ready format

Part 1 will include:

- * Executive summary and overall network health
- * Detailed ISIS topology with ISO addresses
- * ISIS database visualisation
- * SRv6 infrastructure deep dive with locator topology illustration

Part 2 will include:

PART 2: Routing Protocol Analysis (Next Chat)

- | — BGP peering topology & route reflector mesh
- | — iBGP/eBGP relationships & external connectivity
- | — Performance metrics & convergence analysis
- | — Traffic distribution & ECMP analysis

please add more details on the internet part, as-path visualisation, the analysis of various prefixes

Also show the entire BGP community structure with policies and mention the community and local-preference.

Part 3 will include:

- * VRF services detailed analysis
- * Flex-Algorithm 128 deployment details

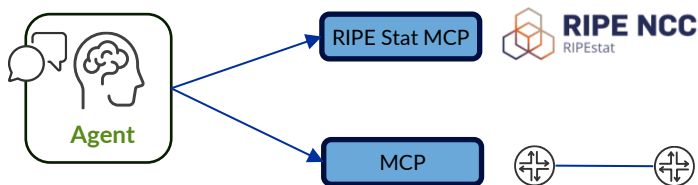
Part 4 will include:

- * Issues and remediation plans
- * Recommendations and conclusions

Use-case 2: BGP Route hijacking

Leveraging External Intelligence for BGP Analysis to mitigate issues like Route Hijacking

- **Integrates with Authoritative Sources:** The agent autonomously queries public data platforms like Cloudflare Radar and RIPE Stats.
- **Provides External Visibility:** Gains an essential "outside-in" view of BGP routing to validate internal observations.
- **Enables Event Correlation:** Correlates global routing data with internal network events for faster, more accurate root cause analysis.



Critical Finding: This is a Major Issue!

Who Owns 1.1.1.0/24:

Owner: Cloudflare (AS13335) / APNIC Labs

Purpose: Cloudflare's famous public DNS resolver service (1.1.1.1)

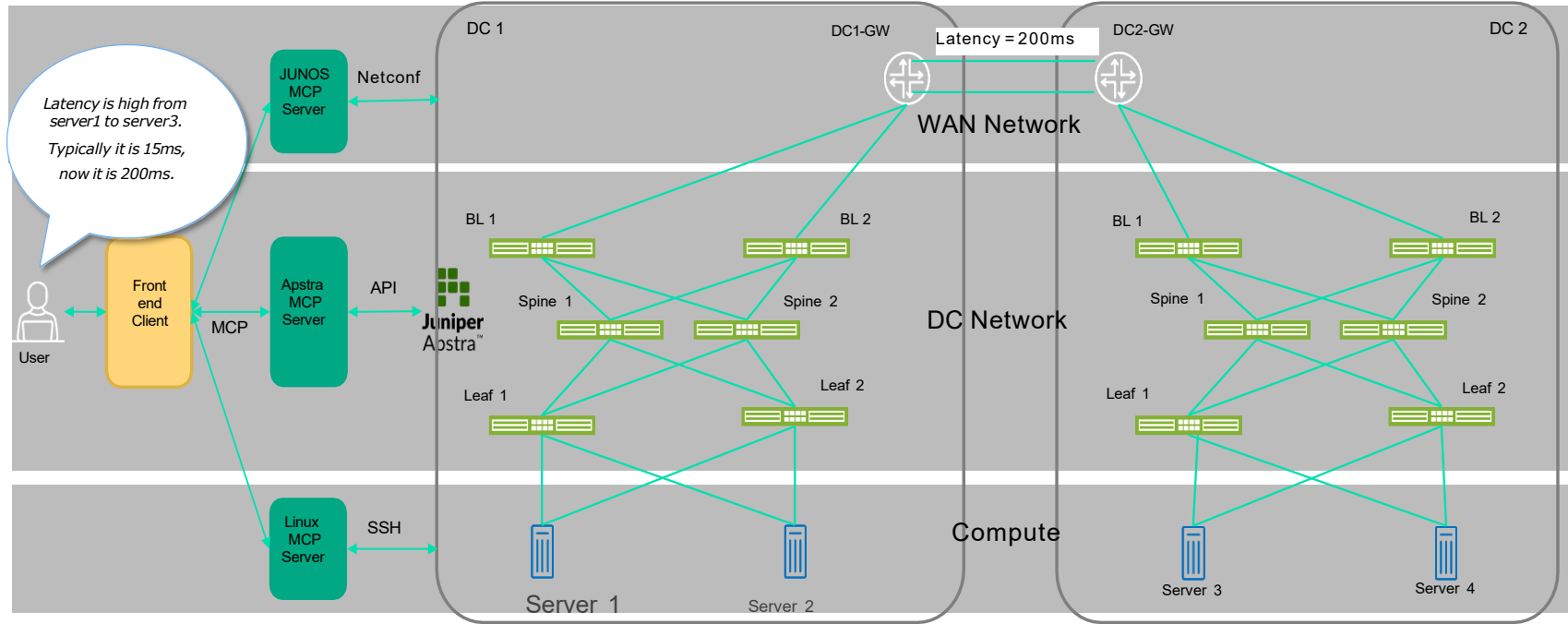
This is **one of the most well-known IP addresses on the internet** - Cloudflare's public DNS service!

What This Means for Your Network:

This is a **SERIOUS** Configuration Problem:

1. **IP Hijacking/Route Leak:** Your router B4 is incorrectly advertising Cloudflare's prefix (1.1.1.0/24) as if AS7473 owns it
2. **BGP Hijack Scenario:** When you advertise 1.1.1.0/24:
 - Some networks may route traffic destined for Cloudflare's DNS (1.1.1.1) to YOUR network instead
 - This creates a "blackhole" where DNS queries never reach Cloudflare
 - Users trying to use 1.1.1.1 as their DNS server would fail
3. **Why It's Not Globally Visible in RIPE:**
 - RIPE collectors may be filtering it as an invalid announcement
 - Most networks likely reject this route as invalid (AS13335 is the legitimate origin)
 - Your peer AS 2004 may be accepting it but others are not
4. **Reputation Risk:**
 - This looks like BGP hijacking (even if unintentional)
 - Could result in your network being filtered or blacklisted
 - Violates internet routing best practices 

Use-case 3: Multi-domain Root Cause Analysis

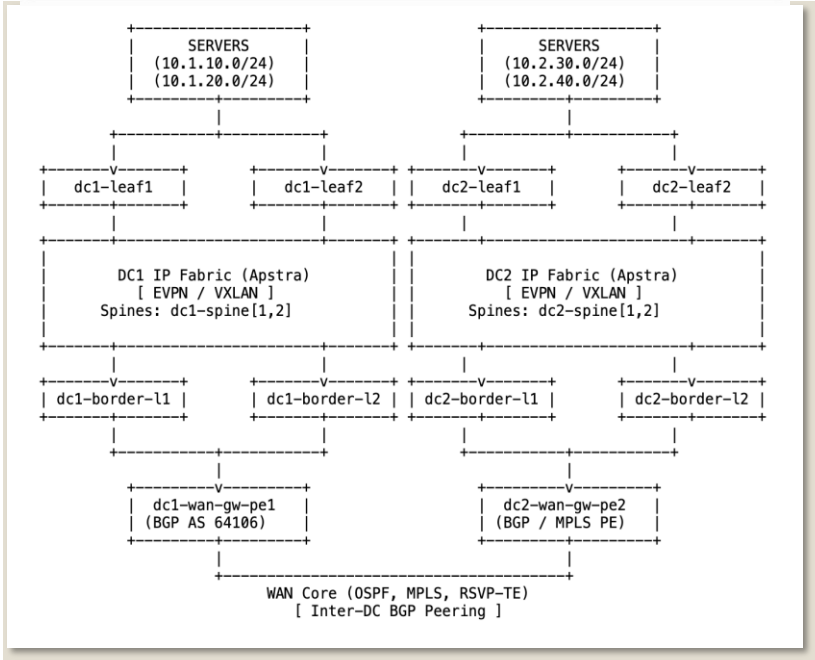


Demo scenario:

- User complains that latency is high from server1 to server3
- Typical latency = 5ms, Current latency = 200ms
- Agent troubleshoots the problem to find the root cause

Agent understands the underlay and overlay topology

give an overview of my datacenter. how are the underlay and overlay network built, how are the DCs connected over WAN, which are the servers in each DC and which networks do they belong to? give me a report including an ascii diagram of the network.



1. Data Center Architecture (Underlay & Overlay)

Both datacenters (JCL-DC1 and JCL-DC2) are managed as independent Juniper Apstra blueprints using a standard IP Fabric Spine-Leaf architecture.

- **Underlay Network:** Built using eBGP to provide robust, loop-free IPv4 connectivity between all leaf and spine switches.
- **Overlay Network:** Built using EVPN-VXLAN. This provides the Layer 2/Layer 3 multi-tenancy abstraction over the IP fabric, allowing virtual networks to span across the respective datacenters.

Hardware Footprint:

- **DC1 Nodes:** dc1-spine1, dc1-spine2, dc1-leaf1, dc1-leaf2 (compute leafs), dc1-border-leaf1, dc1-border-leaf2 (exit leafs).
- **DC2 Nodes:** dc2-spine1, dc2-spine2, dc2-leaf1, dc2-leaf2 (compute leafs), dc2-border-leaf1, dc2-border-leaf2 (exit leafs).

2. WAN Interconnect (Data Center Interconnect - DCI)

The two datacenters are connected over a WAN via Provider Edge (PE) Gateway Routers (dc1-wan-gw-pe1 and dc2-wan-gw-pe2).

- **Routing & Transport:** The WAN core operates on an OSPF / MPLS underlay.
- **Traffic Engineering:** The interconnect uses RSVP-TE (Resource Reservation Protocol with Traffic Engineering). Dedicated Label Switched Paths (LSPs) such as LSP-T0-DC2-PRIMARY and backup LSPs are established to strictly steer traffic over primary and backup links across the WAN.
- **DCI Overlay:** Inter-DC overlay connectivity relies on external BGP (WAN-INTERDC-L1) peered over the MPLS transport to exchange routes between the Border Leafs of DC1 and DC2.

3. Servers and Virtual Networks

The physical servers (Generic Systems) are connected to the compute leaf switches in their respective datacenters and are mapped into the EVPN-VXLAN overlay networks.

DC1 Networks (Mapped to dc1-leaf1/2):

- v10: Subnet 10.1.10.0/24 (VXLAN VNID: 10003)
- v20: Subnet 10.1.20.0/24 (VXLAN VNID: 10002)

DC2 Networks (Mapped to dc2-leaf1/2):

- v30: Subnet 10.2.30.0/24 (VXLAN VNID: 10005)
- v40: Subnet 10.2.40.0/24 (VXLAN VNID: 10004)

Agent runs server-to-server latency test

give me a table with the server mapping to the virtual networks and routing zones

Datacenter	Server Name	IP Address	Subnet	Virtual Network	Routing Zone
JCL-DC1	DC1-server1	10.1.10.100	10.1.10.0/24	v10 (VNI 10003)	RZ_10-20 (VNI 10001)
JCL-DC1	DC1-server2	10.1.20.100	10.1.20.0/24	v20 (VNI 10002)	RZ_10-20 (VNI 10001)
JCL-DC2	DC2-server1	10.2.30.100*	10.2.30.0/24	v30 (VNI 10005)	RZ_30-40 (VNI 10000)
JCL-DC2	DC2-server2	10.2.40.100	10.2.40.0/24	v40 (VNI 10004)	RZ_30-40 (VNI 10000)

check ping between servers within DC1 and from DC1 to DC2. ignore DC2-server1, its down. give a table with the latency

Source Server	Destination Server	Destination IP	Ping Status	Avg Latency (RTT)	Scope
DC1-server1	DC1-server2	10.1.20.100	✔ Success (0% loss)	2.74 ms	Intra-DC (DC1)
DC1-server1	DC2-server2	10.2.40.100	✔ Success (0% loss)	218.13 ms	Inter-DC (DC1 → DC2)
DC1-server2	DC2-server2	10.2.40.100	✔ Success (0% loss)	215.17 ms	Inter-DC (DC1 → DC2)

As expected, traffic within DC1 is very fast (~2.7ms routing between VNIs v10 and v20), while the WAN transit over to DC2 incurs a higher latency of ~215-218ms.

Agent diagnoses the cause of the high-latency

why is the latency so high between the DCs? Investigate

1. Two WAN Links with Drastically Different Profiles

The datacenters are connected via two distinct physical WAN links. Tests executed directly from the WAN gateway (`dc1-wan-gw-pe1`) show their raw performance:

- LINK1 (`172.16.252.1` via `ge-0/0/2.0`): Highly degraded/high latency link. Pings average >210 ms.
- LINK2 (`172.16.252.3` via `ge-0/0/4.0`): Low latency link. Pings average ~4 - 8 ms.

2. Forwarding Table Override

Despite having an active, low-latency backup path (LINK2), there is an explicit forwarding-table export policy (`RSVP-STEER-DCI-LINK1`) deployed on the routers that forces all DCI subnets onto the high-latency LINK1 path.

On `dc1-wan-gw-pe1`, the EVPN overlay transit subnets (`192.168.2.0/24`) are explicitly matched and steered over the `LSP-T0-DC2-PRIMARY` label-switched path:

Conclusion: All server-to-server traffic between DC1 and DC2 relies on EVPN/VXLAN encapsulated within these IPs. Because the WAN edge routers are strictly forcing this encapsulation traffic over the high-latency "LINK1", the overlay application traffic inherits the ~215ms latency.

To fix this, the forwarding-table export policy on both WAN routers needs to be updated to steer traffic to `LSP-T0-DC2-PRIMARY-LINK2` (and its DC1 equivalent) which maps to the fast `172.16.252.3` path. Would you like me to make these configuration changes to restore optimal latency?

Agent fixes the problem

yes do it

The configuration has been successfully applied to both WAN gateways (`dc1-wan-gw-pe1` and `dc2-wan-gw-pe2`). I swapped the forwarding-table export policy from `RSVP-STEER-DCI-LINK1` over to `RSVP-STEER-DCI-LINK2`, effectively moving the DCI overlay traffic onto the low-latency transport path.

I ran a quick verification ping from `DC1-server1` to `DC2-server2` (`10.2.40.100`), and the latency has successfully dropped from ~218 ms down to ~15 ms.

give me the same latency table again

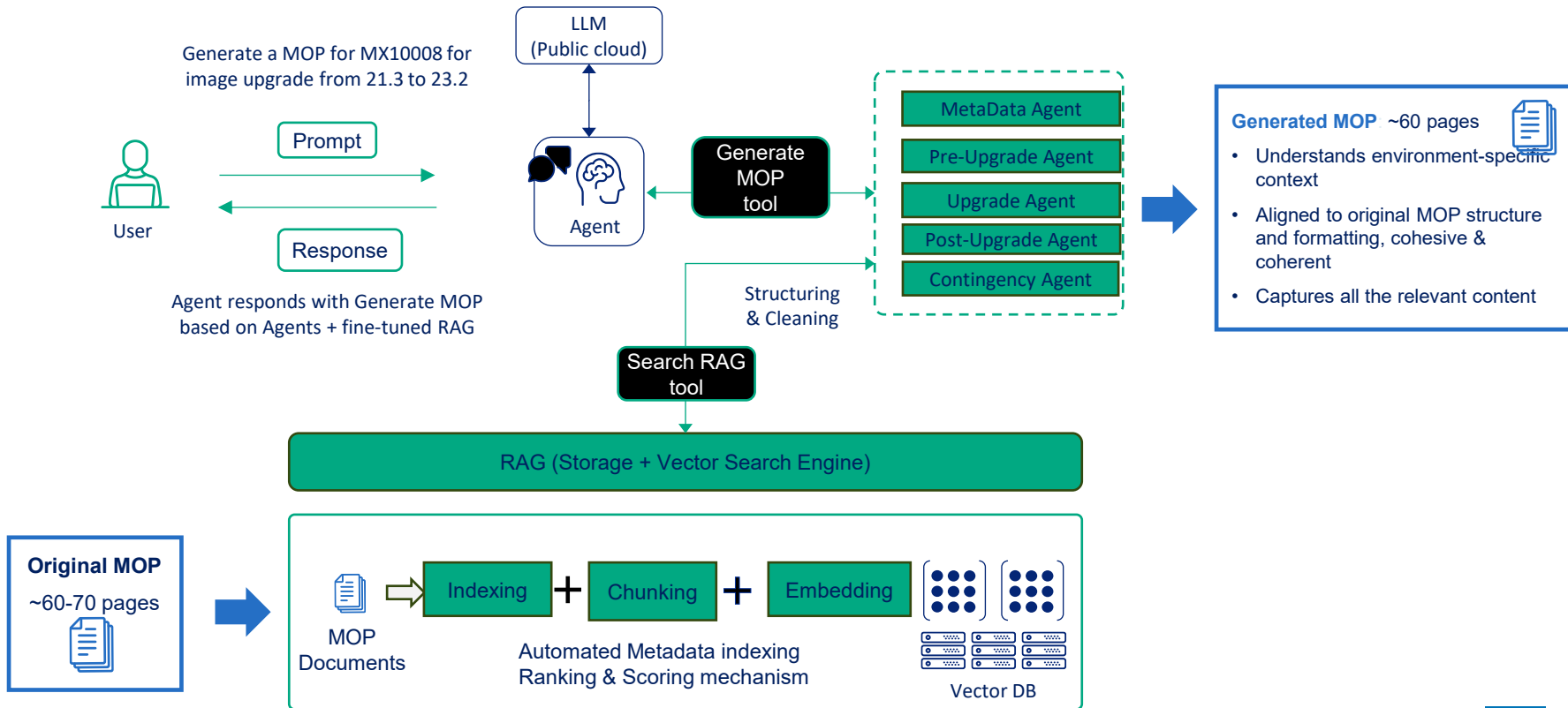
Here is the updated latency table reflecting the changes after routing traffic over the low-latency DCI link:

Source Server	Destination Server	Destination IP	Ping Status	Avg Latency (RTT)	Scope
DC1-server1	DC1-server2	10.1.20.100	✓ Success (0% loss)	2.62 ms	Intra-DC (DC1)
DC1-server1	DC2-server2	10.2.40.100	✓ Success (0% loss)	17.75 ms	Inter-DC (DC1 → DC2)
DC1-server2	DC2-server2	10.2.40.100	✓ Success (0% loss)	13.73 ms	Inter-DC (DC1 → DC2)

The Inter-DC latency has successfully dropped from ~215-218 ms down to ~13-17 ms.

Use-case 4: MOP generation

Inference + Multi agent + Fine-tuned RAG



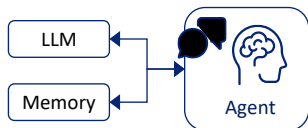
Why RAG & Multi-agent?

Original MOP: ~60-70 pages



Inference only

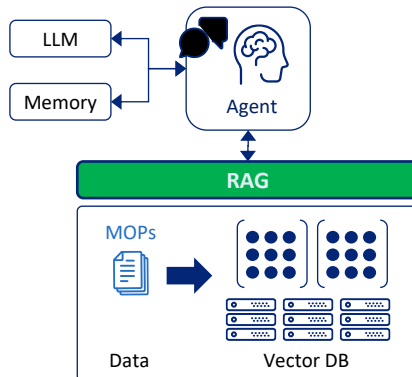
(no RAG, LLM fine-tuning or multi-agent)



Setup

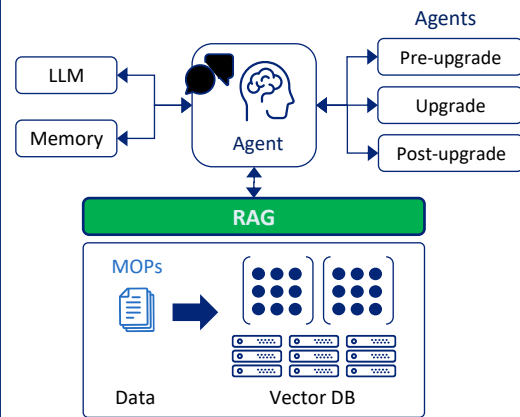
Inference + RAG

(no LLM fine-tuning or multi-agent)



Inference + RAG + Multi-agent

(LLM fine-tuning)



Generated MOP:



- 11 pages
- Generic MOP based on LLM's previously learnt knowledge
- No environment-specific context

Result

Generated MOP:



- 278 pages
- Understands environment-specific context
- Poorly structured, not coherent & cohesive

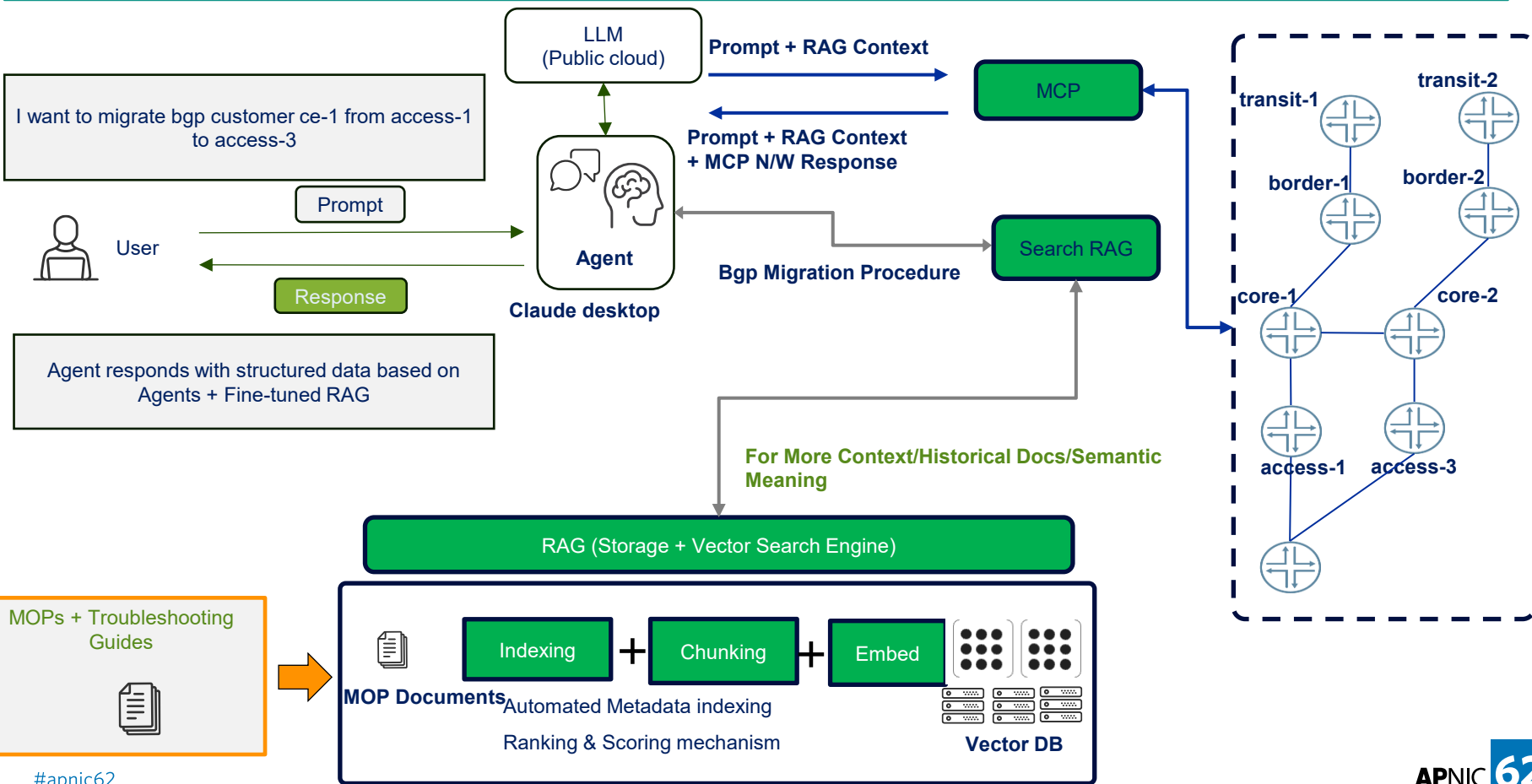
Generated MOP:



- ~60 pages
- Understands environment-specific context
- Aligned to original MOP structure and formatting, cohesive & coherent
- Captures all the relevant content

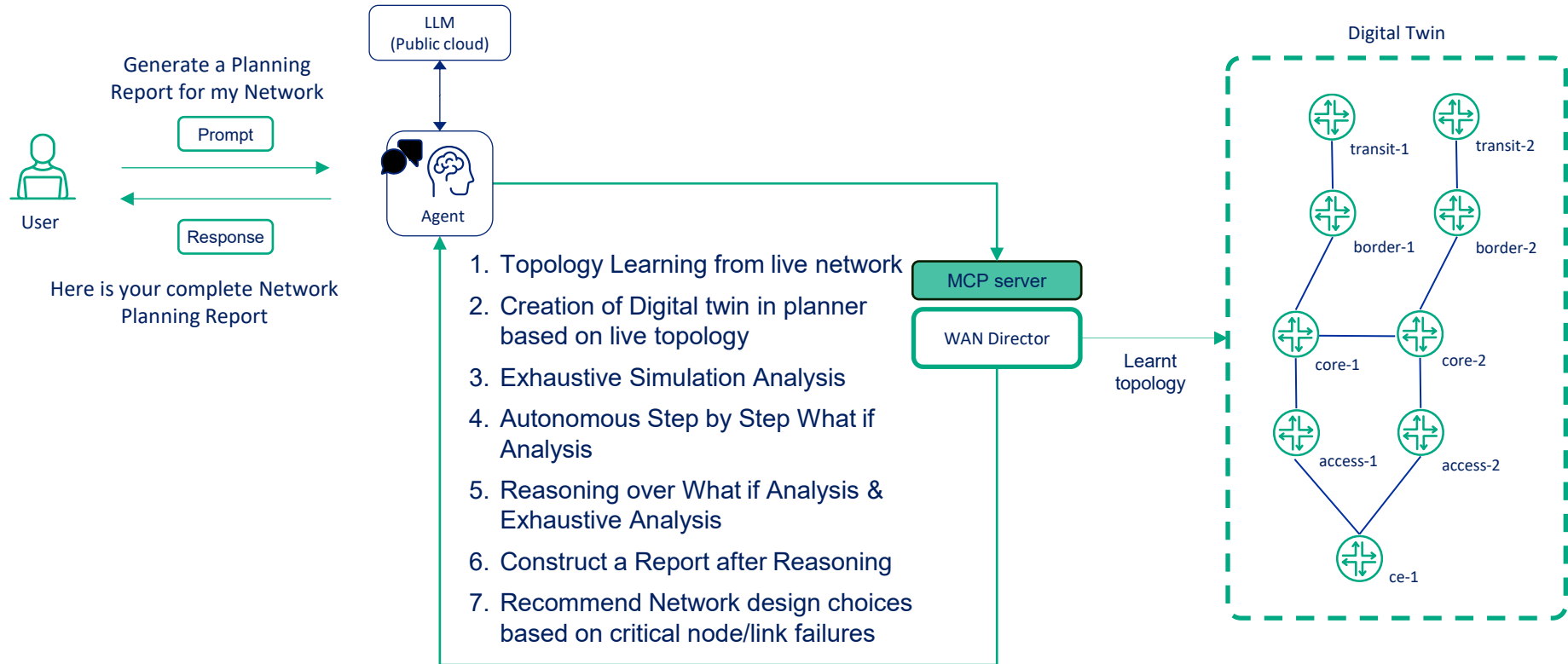
Use-Case 5: MOP Execution

Inference + Multi-Modal RAG + MCP



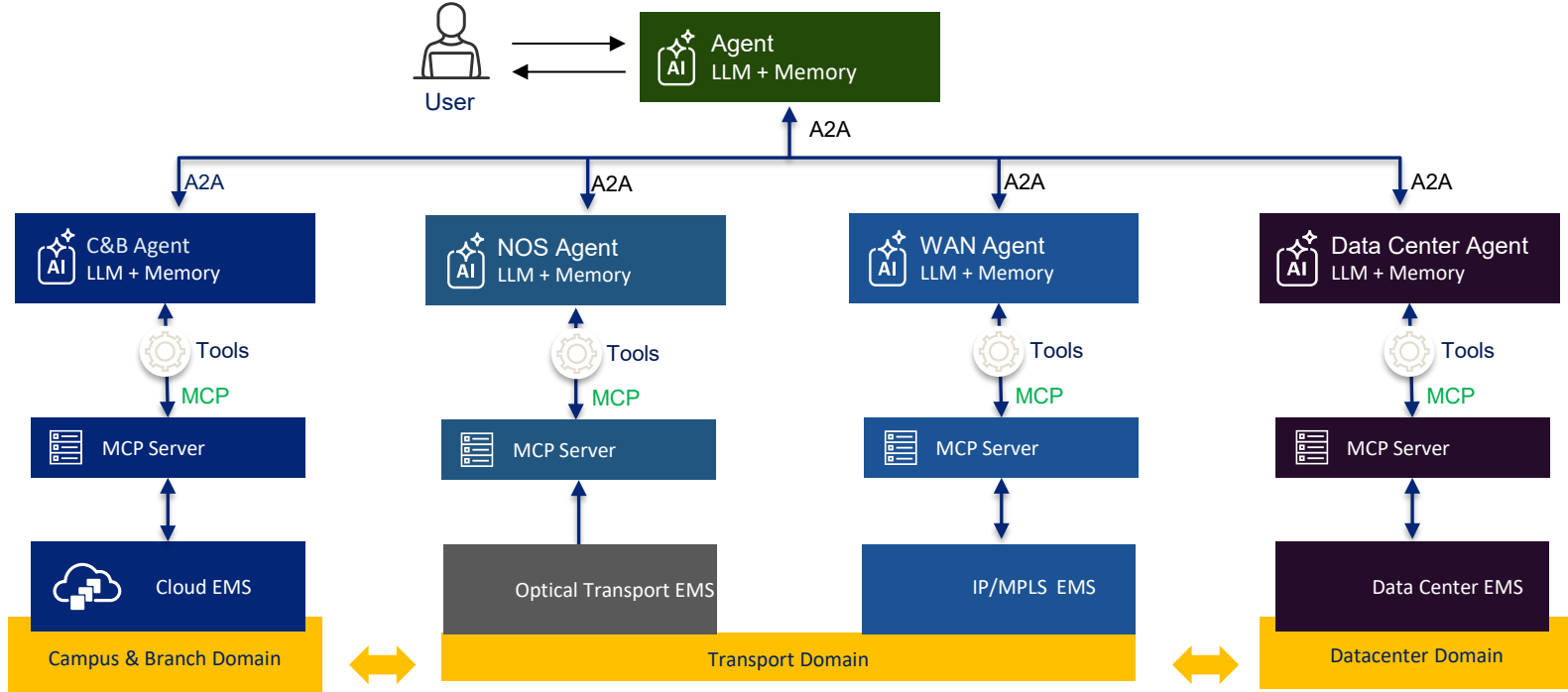
Use-case 6: Network Planning with WAN Director

Failure simulation and What-If analysis



End-to-end Architecture with A2A and MCP

Agent-to-Agent is an open standard that enables different AI agents to communicate and work together directly



AI Risk and Governance

R1

CRITICAL

Hallucination → Wrong Automated Action

LLM wrong RCA → agent executes config change on incorrect target → traffic black-hole. No factual grounding = no execution.

MITIGATION

commit-confirmed on every write · ≥85% confidence gate · tool-grounded facts required before execution

R2

CRITICAL

Blast Radius of Autonomous Config Changes

Mis-scoped playbook propagates across 40 PEs simultaneously. One bad SR-TE policy, no blast-radius guard = network-wide impact.

MITIGATION

Blast radius limit enforced at MCP layer · staged rollout (1→5→all) · Digital Twin pre-validation

R3

CRITICAL

Credential & Access Exposure via Tool Calls

SSH keys/API tokens in MCP config. Server compromise = full fleet access. LLM context logging can leak credentials.

MITIGATION

MCP as auth proxy — credentials never reach LLM · Vault for secrets · scoped read-only vs write accounts

R4

HIGH

Model Drift — AI Accuracy Degrades Over Time

Models trained on last year's topology miss new failure signatures. Anomaly detection degrades silently with no alert.

MITIGATION

Monthly model performance tracking vs human-reviewed incidents · quarterly retrain · RAG DB auto-refresh

R5

HIGH

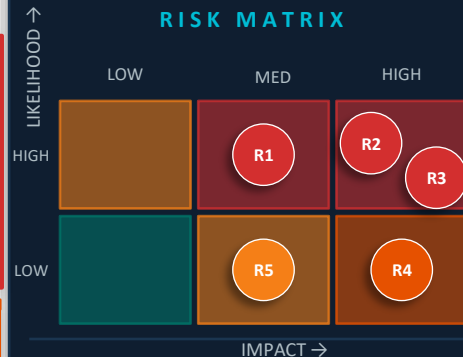
Incomplete Audit Trail for AI-Initiated Changes

AI changed it. Who approved? Without structured logs, AI-caused incidents are unrecoverable in post-mortem.

MITIGATION

Immutable append-only MCP audit log (timestamp, tool, args, reasoning) · auto-ITSM change ticket per commit

RISK MATRIX



LEGEND

- R1 Hallucination → Wrong Action
- R2 Blast Radius of Auto-Change
- R3 Credential / Access Exposure
- R4 Model Drift
- R5 Missing Audit Trail

Thank You

APNIC 62

Questions are welcome

MUMBAI, INDIA

4 – 10 September 2026

#apnic62



Akhilesh Thakur

Principal Architect · HPE Networking



Akhilesh.thakur@hpe.com



<https://www.linkedin.com/in/akhilesh-thakur-91801030/>

