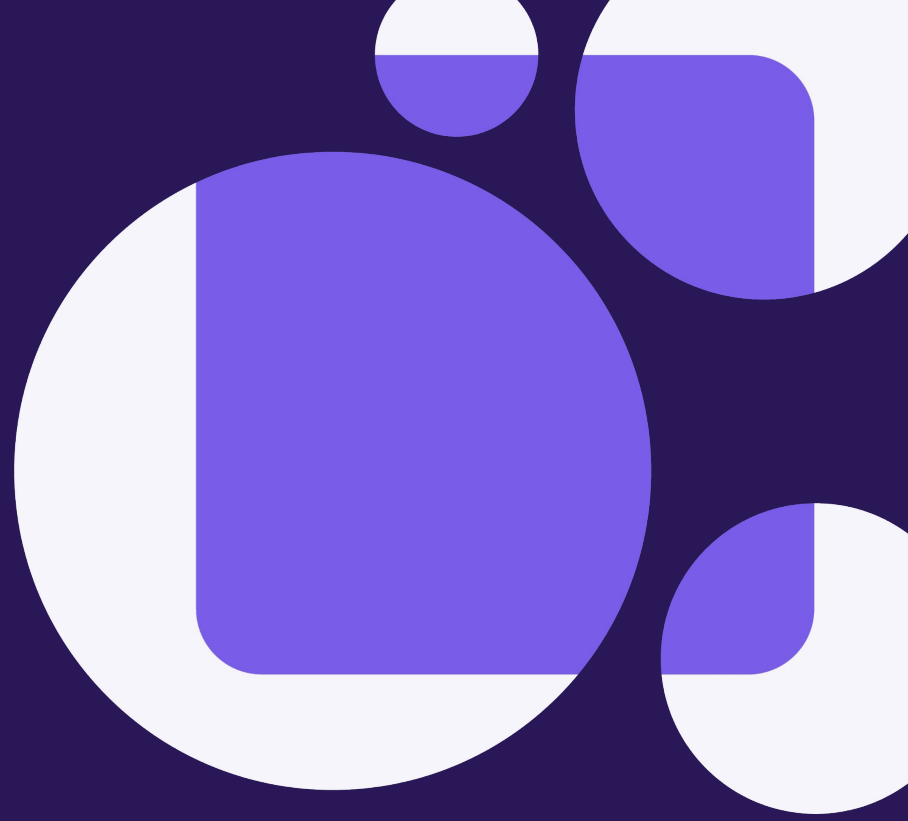
The AI-First Device Farm: Exposing Remote Hardware Infrastructure via MCP



Get Well

About Us

Kalyan Kolachala

Currently India MD at SAI Group, a global enterprise AI leader

Worked previously at Intuit and Hitachi Vantara as India site head for development platform, AI/ML, SaaS and Hybrid Cloud products

Over 3 decades of experience comprising of engineering leadership, product strategy, site leadership, SaaS, Cloud, design and architecture, AI/ML and GenAI

Active in open source, contributed to C++ standard and SOA standards with W3C/OASIS

Speaker in global events on technology and strategy

Graduate in Computer Science from Indian Institute of technology

SymphonyAI Group (saigroup.ai)

Building the leaders in Enterprise AI solutions for the largest vertical markets, including financial services, retail, manufacturing, life sciences and healthcare. SAIGroup India is the center of innovation and excellence.

Backed by $1 billion of flexible, committed equity capital from Dr. Romesh Wadhwani, a highly successful AI and software entrepreneur and philanthropist

Portfolio Companies:

SymphonyAI: Enterprise AI leader in financial services, retail, CPG, manufacturing

ConcertAI: Leader in diagnostics, clinical trials and AI solutions for life sciences and healthcare

GetWell RhythmX: Get Well, a leader in patient engagement software, and RhythmX AI, a leader in AI-powered precision care, combined to form GW RhythmX to usher in the next generation of precision care.

JazzX AI: JazzX is defining the future of enterprise work—by building AI-native digital workers that actually get the job done. The unerlying platform has advanced AI capabilities such as reasoner, knowledge fabric, self learning and these are extended into a verticalized solution using a low code/no code tool (builder studio).

About Us

Vaishali Shetty

Currently Principal Engineer, Get Well RhythmX, part of SymphonyAI Group..

Worked previously at Mycom and Hitachi Vantara as Architect focusing on Performance, QA and Observability

Over 16 years experience in modern testing frameworks observability, performance testing tools

Experience in observability, testing and evaluation of AI/ML and GenAI systems and AI first SDLC

Speaker in tech talks globally on AI, QA and observability

Get Well RhythmX

Get Well, a leader in patient engagement software, and RhythmX AI, a leader in AI-powered precision care, combined to form GW RhythmX to usher in the next generation of precision care.

The combined company, backed by multi-billion dollar SymphonyAI Group, has 150+ marquee health system clients and is led by RhythmX AI Founder and CEO, Deepthi Bathina; it will deliver the leading AI patient-centric precision care platform for industry-defining outcomes for patients, clinicians, nurses and health systems.

RhythmX AI is a generative AI-native health company driving a paradigm shift in hyper-personalized care. RhythmX AI’s precision care platform helps physicians pioneer a new era of whole-person care through generative and predictive AI-powered copilots.

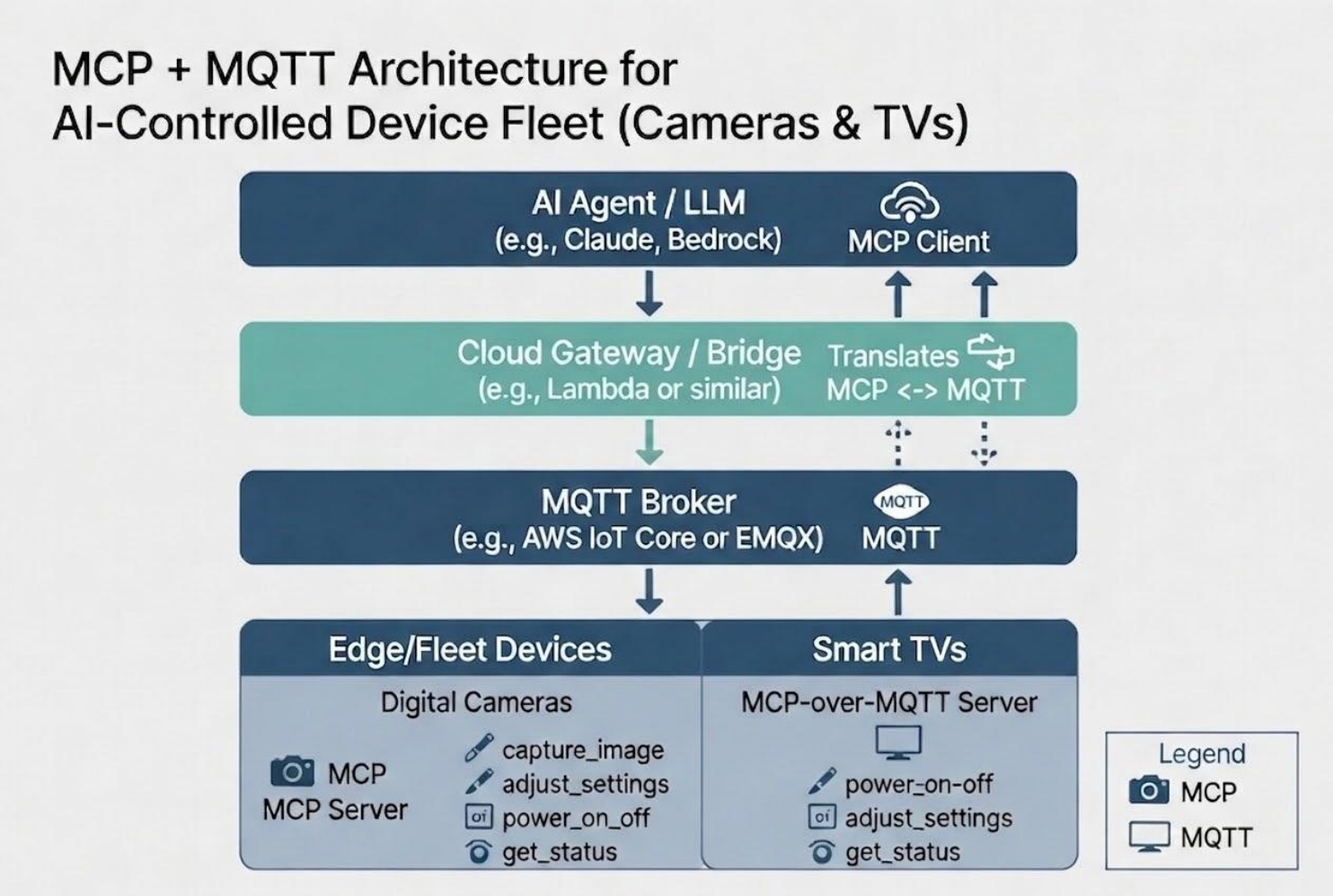
Agenda

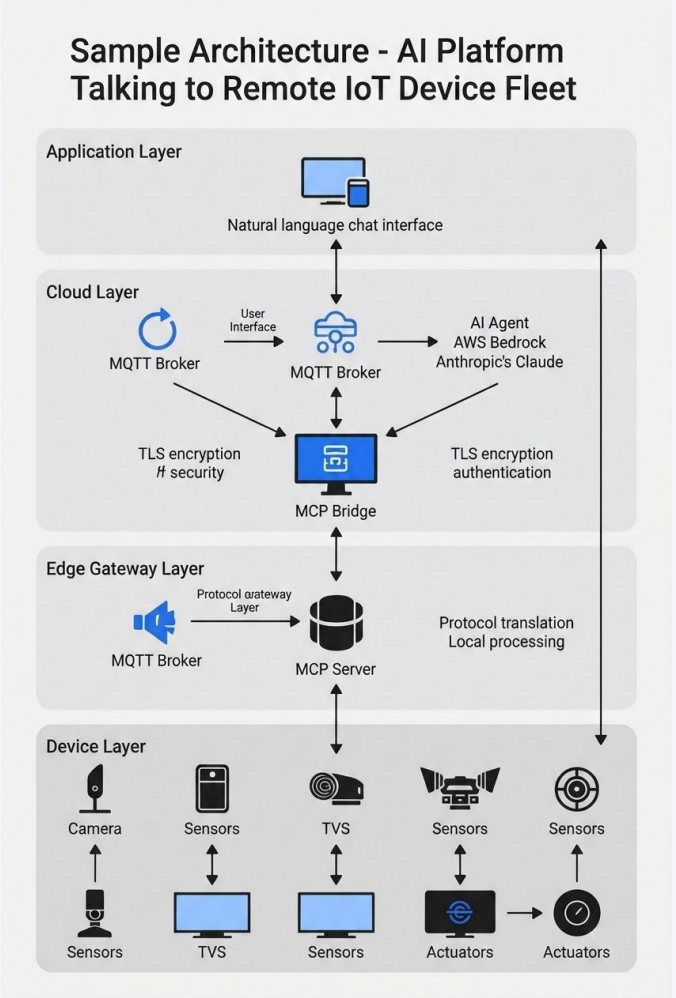
- Motivation and Drivers
- MCP for Edge/Remote devices
- MCP in Healthcare domain
- Security needs for edge devices
- Case Studies from GW RhythmX
  - AI device farm
  - Natural Language Testing & AI-Assisted Debugging
  - Virtual Care
- References and Acknowledgements
- Q & A

Motivation and Drivers

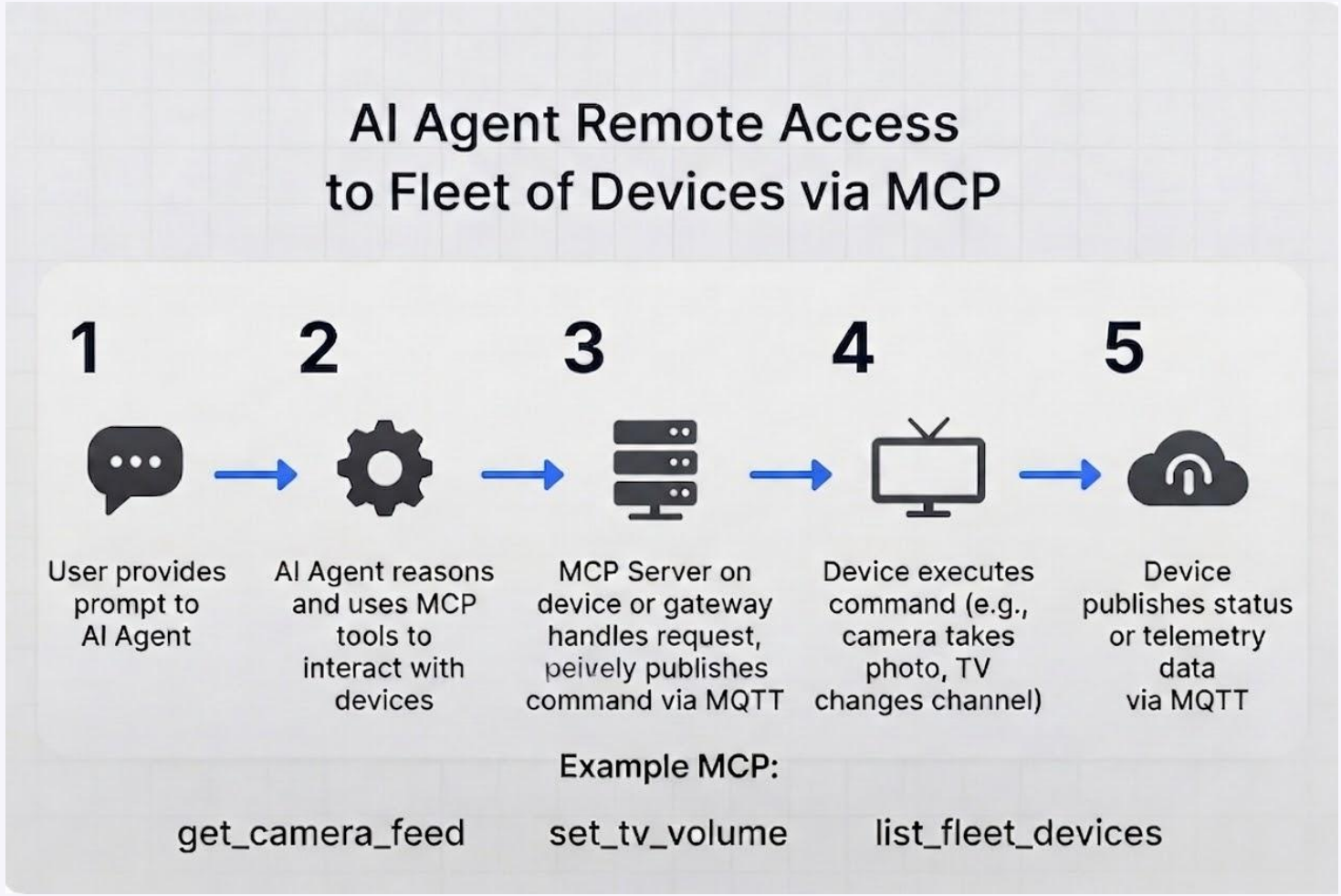
- Need for multiple integrations (EHRs, other healthcare systems, edge devices) and to avoid M x N integrations
- Support for natural language control/actions/reporting in both directions (query and response)
- Need to support multiple makes, models of devices and various protocols
- Security and Compliance
- Dynamic discovery and grouping

MCP for Edge/Remote devices





MCP for Edge/Remote devices



MCP-over-MQTT (EMQ and others) adapts it for IoT by running MCP semantics over MQTT topics.

Some alternatives for device interaction/communication (to be used with MCP):
Layered AIoT Architecture (common in industrial/smart factories):

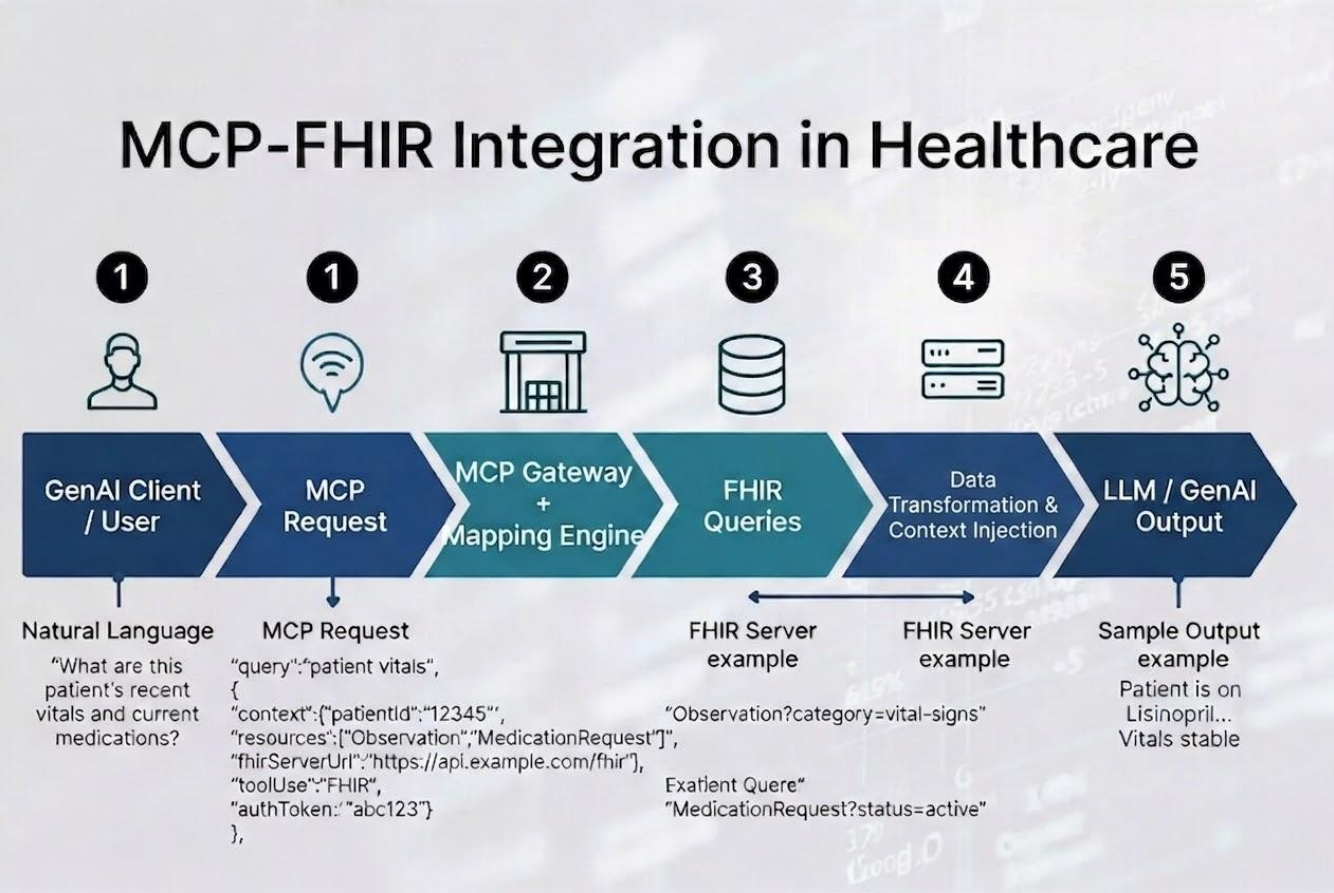
- **Perception/Devices**: Sensors + OPC UA.
- **Edge**: Gateways with local AI inference + MCP server.

EMQX File Transfer over MQTT extends MQTT with chunked, resumable transfers. Stores to local disk or S3-compatible storage. Ideal use cases include diagnostic logs, images, audio/video files, and offline IoT data.

HTTP(S) + pre-signed URLs (S3, Azure Blob, etc.) or dedicated file transfer

WAMP

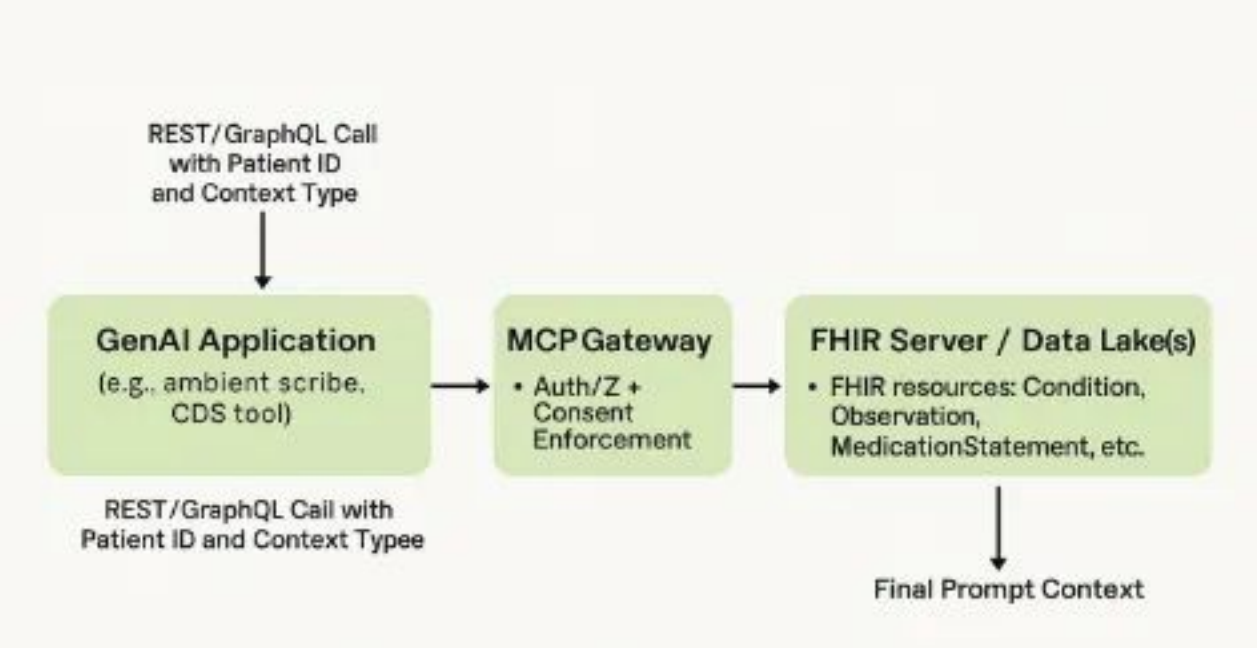
MCP in Healthcare domain

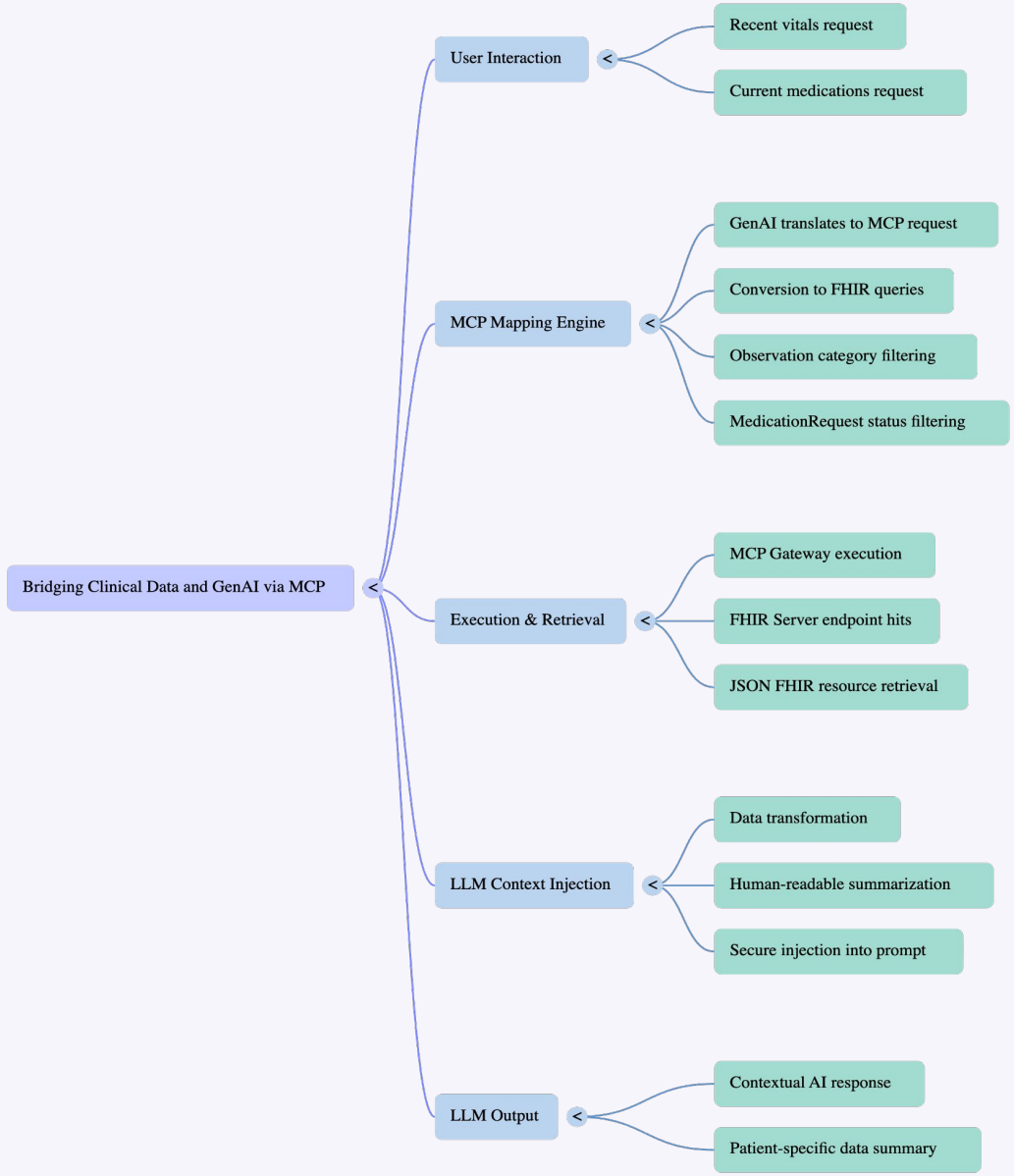


Typical End-to-End Workflow:

1. **User Prompt** — Clinician asks the GenAI assistant a natural-language question about a patient.
2. **GenAI → MCP** — Client creates an MCP request; the Mapping Engine converts it into one or more precise FHIR queries.
3. **MCP Executes** — Gateway queries the FHIR Server (Epic, Cerner, Medplum, etc.) and retrieves the JSON resources.
4. **Context Injection** — Retrieved data is summarized (e.g., "BP: 130/85 mmHg on 05/10/25") and securely injected into the LLM's prompt.
5. **LLM Output** — AI generates an accurate, contextual response based on real patient data.

MCP in Healthcare domain

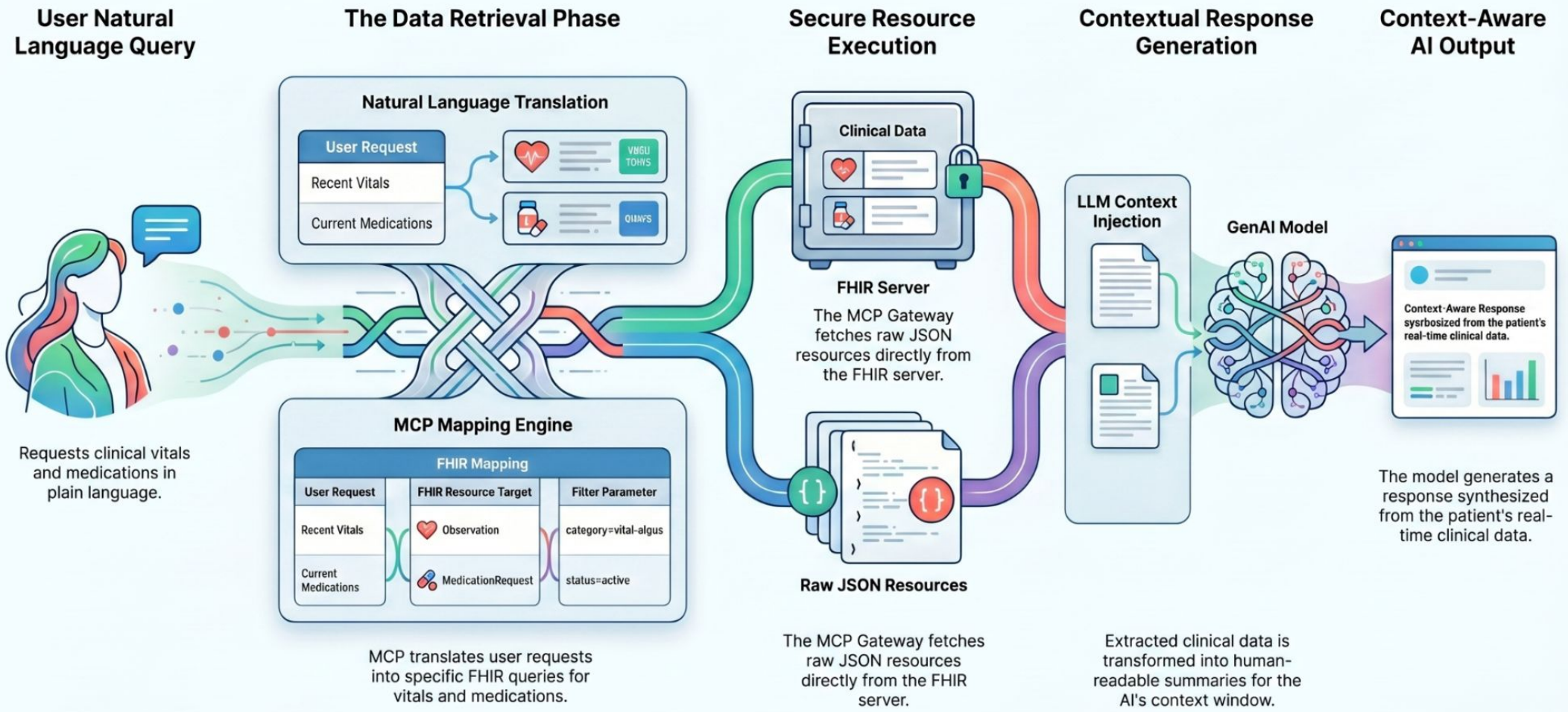




Open-source MCP servers (e.g., Flexpa mcp-fhir, AWS HealthLake MCP server, Momentum.ai FHIR-MCP) enable natural language CRUD/search over FHIR resources.

MCP gateways (e.g., from mintMCP, Keragon, Innovaccer HMCP) add enterprise controls: security, governance, audit trails, and pre-built connectors for major EHRs.

Bridging Clinical Data with GenAI: The MCP Workflow



MCP GATEWAY

Authentication (SSO/OAuth)

- Provides secure user login using enterprise identity providers (SSO, OAuth2, Entra ID, Okta, etc.).
- Eliminates shared lab credentials and ensures all device access is tied to an individual user.

Authorization (RBAC)

- Controls which users, teams, or AI agents can access specific devices and MCP tools.
- Enforces role-based permissions (Developer, QA, Admin, Automation Agent) to prevent unauthorized actions.

Device Reservation & Locking

- Prevents multiple users or AI agents from simultaneously controlling the same device.
- Supports exclusive device reservations with automatic lock acquisition and release.

Session Management

- Tracks active user and AI sessions across devices and MCP servers.
- Automatically expires inactive sessions and releases reserved devices to maximize utilization.

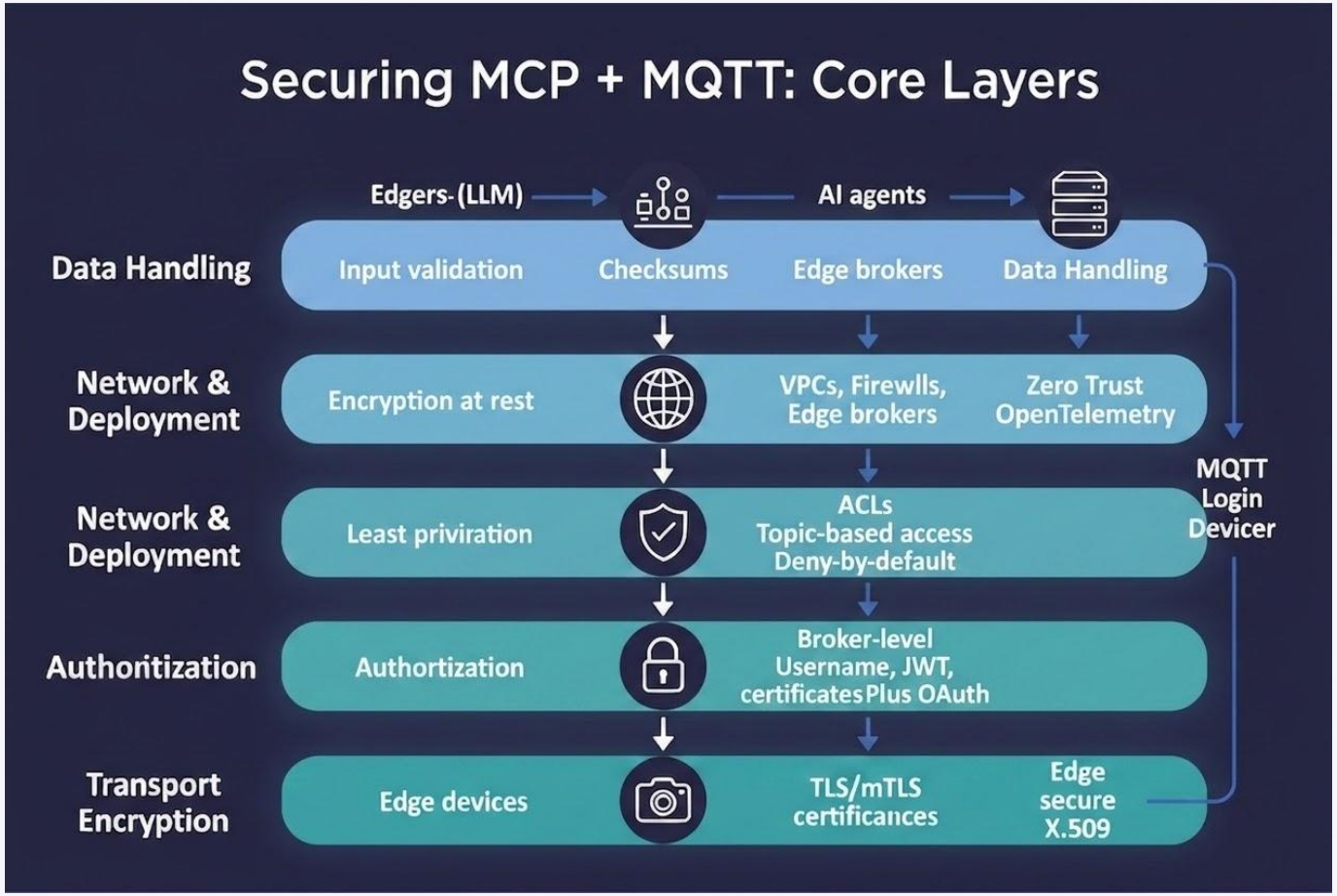
Audit & Activity Logging

- Records every action performed on a device, including user, timestamp, command, and outcome.
- Provides traceability for debugging, compliance, security investigations, and usage reporting.

Device Discovery / Inventory

- Maintains a centralized catalog of all available devices, configurations, firmware versions, and statuses.
- Enables users and AI assistants to discover and reserve devices based on attributes such as model, OS version, location, or availability.

MCP Security For Edge Devices



- **Upper layers**: Network controls + Data handling ensure safe deployment and processing.
- **Middle layers**: Authentication + Authorization (ACLs) control *who* can do *what*.
- **Bottom layer (Transport)**: Foundation — everything flows over encrypted mTLS.

Layer	Controls	Best Practices / Implementation
Transport Security	TLS 1.2/1.3, MQTTS (8883), mTLS	Disable SSLv3/TLS 1.0/1.1, enforce strong cipher suites, validate server certificates, use mutual TLS for device and AI client authentication, encrypt S3 transfers and storage (SSE-S3/KMS).
Authentication	JWT, OAuth2 Bearer Tokens, X.509 Certificates, PSK, LDAP, Username/Password	Use certificate-based authentication for device fleets, unique credentials per device, secure UUIDv4 session IDs, short-lived tokens, credential and certificate rotation, no anonymous access.
Authorization	Device-level ACLs, Topic ACLs, RBAC, Scoped Permissions	Deny-by-default policy, restrict publish/subscribe to device-specific topics, minimize wildcard usage, role-based access to MCP tools, service-level authorization through MCP Gateway.
MCP Tool Security	Tool Exposure Control, Scoped Tool Access	Expose only required tools (e.g., screenshot, capture image), restrict destructive operations (factory reset, firmware updates), assign permissions per user, AI agent, and role.
Network Security	VPC, Firewalls, VPN, Bastion Hosts, Reverse Proxies	Keep MQTT brokers and MCP servers private, avoid public internet exposure, use secure tunnels for remote access, segment networks by environment and device group.
Rate Limiting & DoS Protection	Connection Quotas, Throttling, Request Limits	Limit connections per device/user, throttle MCP requests and MQTT publishes/subscribes, detect abnormal traffic patterns.
Session Management	Secure Session IDs, Auto-Release, Timeout Handling	Use cryptographically secure UUIDv4 session IDs, automatically release device reservations after inactivity, enforce session expiration and cleanup.
Input Validation & Sandboxing	Validation, Sanitization, Execution Isolation	Treat all MCP requests as untrusted, validate inputs and outputs, sandbox tool execution, prevent command injection and unauthorized actions.

File Transfer Security	Secure MQTT File Transfer, S3 Policies, Checksums	Restrict access to $file/... topics, verify SHA-256 checksums, enforce least-privilege S3 IAM roles, encrypt files in transit and at rest.
Audit & Logging	Full Tool Invocation Logging, Audit Trails	Log every MCP action, authentication event, file transfer, and device reservation, maintain immutable audit records for compliance investigations.
Monitoring & Observability	OpenTelemetry, Metrics, Alerts, Anomaly Detection	Track device activity, MCP tool usage, MQTT traffic, failed authentication attempts, unusual file transfers, and suspicious AI actions.
Secrets Management	Vault-Based Secret Storage	Use HashiCorp Vault, AWS Secrets Manager, Azure Key Vault, rotate secrets regularly, never hardcode credentials or certificates.
Patch & Vulnerability Management	Software Updates, Security Fixes	Keep EMQX, MQTT clients, MCP Gateway, AI agents, and device software updated, perform vulnerability scanning and penetration testing.
Healthcare Compliance	HIPAA, FHIR Security Controls, PHI Protection	Encrypt PHI at rest and in transit, maintain audit logs, enforce consent validation, de-identify data where required, establish Business Associate Agreements (BAA).
Zero Trust Security	Continuous Verification	Verify every request, continuously authenticate devices and users, assume breach, apply least-privilege access throughout the architecture.
EMQX Enterprise Security	RBAC, SSO, Audit Logs, MCP Gateway Controls	Use enterprise RBAC, SSO integration, centralized audit logging, service-name-based ACL generation, and policy enforcement through EMQX MCP Gateway.

The Blueprint for Secure AI & IoT Architectures

A high-level security roadmap for organizations integrating AI agents (MCP) with IoT infrastructures (EMQX).

Access & Identity Control

Mandatory Encryption in Transit

Always use TLS 1.2/1.3 for MQTT and S3 transfers to ensure data privacy.

Multi-Layer Authentication

Combine MQTT broker authentication with strong MCP session IDs to verify all participants.

Least-Privilege Authorization

Implement topic-level ACLs and scoped tool permissions to restrict access to the absolute minimum.

Standard vs. Healthcare Security Requirements

Standard Requirement

Healthcare (HIPAA/FHIR)

Data Handling

Standard Requirement	Healthcare (HIPAA/FHIR)
Encryption in Transit	Encryption at Rest + De-identification

Access Control

Standard Requirement	Healthcare (HIPAA/FHIR)
Username/ Password	RBAC + Consent Checks

Audit Logs

Standard Requirement	Healthcare (HIPAA/FHIR)
Connection Events	Comprehensive PHI Access Auditing

Perimeter & Data Integrity

Network Isolation

Bind brokers to internal networks and use VPCs or firewalls to prevent public exposure.

Input Validation & Sandboxing

Treat all AI tool inputs as untrusted and execute them within secure sandboxes.

Continuous Monitoring & Audit

Use OpenTelemetry and anomaly detection to identify unusual tool calls or file uploads.

Case Studies from Get Well RhythmX

Get Well

GetWell Inpatient: The Operating System for the Patient Room

Patient-Facing Features

Entertainment
TV, Netflix, Swank movies

Education
Healthwise and Krames videos

Meal Ordering
CBORD, Sodexo, MyDining

Video Calling
Zoom/Caregility telehealth

Service Requests
Housekeeping, Facilities, Food, Advocate

Patient Pathways
Discharge, Fall Prevention, Pain, Hygiene

GetWell Inpatient PLS: The Room OS

Clinical & Nurse Features

Patient Census Board
All patients, statuses, teams

Staff Assignments
Rounding management

Digital Patient Careboard
Whiteboard per room

Real-time Alerts & Critical Notifications

BYOD Support
Patient phone

Integrations

48+ external integration providers across 12 categories

Clinical Systems
- HL7 HL7/ADT EHR
- FHIR R4 Epic CoExist
- mirth connect Mirth Connect

SSO
- LDAP/SAML Hospital SSO
- Active Directory

Device Communication
- MQTT Commands
- WAMP/Crossbar Remote Control

Get Well

AI Device Farm – Case Study

Hardware Access Bottleneck

- 4 device categories with different protocols
- Physical devices are limited shared resources
- Cross-timezone collaboration is difficult
- Scheduling conflicts and device contention
- Manual debugging and no unified automation

Current Solutions

- In-House Labs → Limited devices, location dependent
- Commercial Farms → Expensive at scale
- Custom TVs, STBs & Whiteboards often unsupported
- No single interface across all devices

The Device Fleet

LG WebOS TV / STB
Patient Bedside Experience
CDP WebSocket

Android / iPhone
GetWell Anywhere
Appium

iPad Tablet
Signal Nurse Station
WebKit

Whiteboard
Digital Patient Careboard
WAMP / MQTT

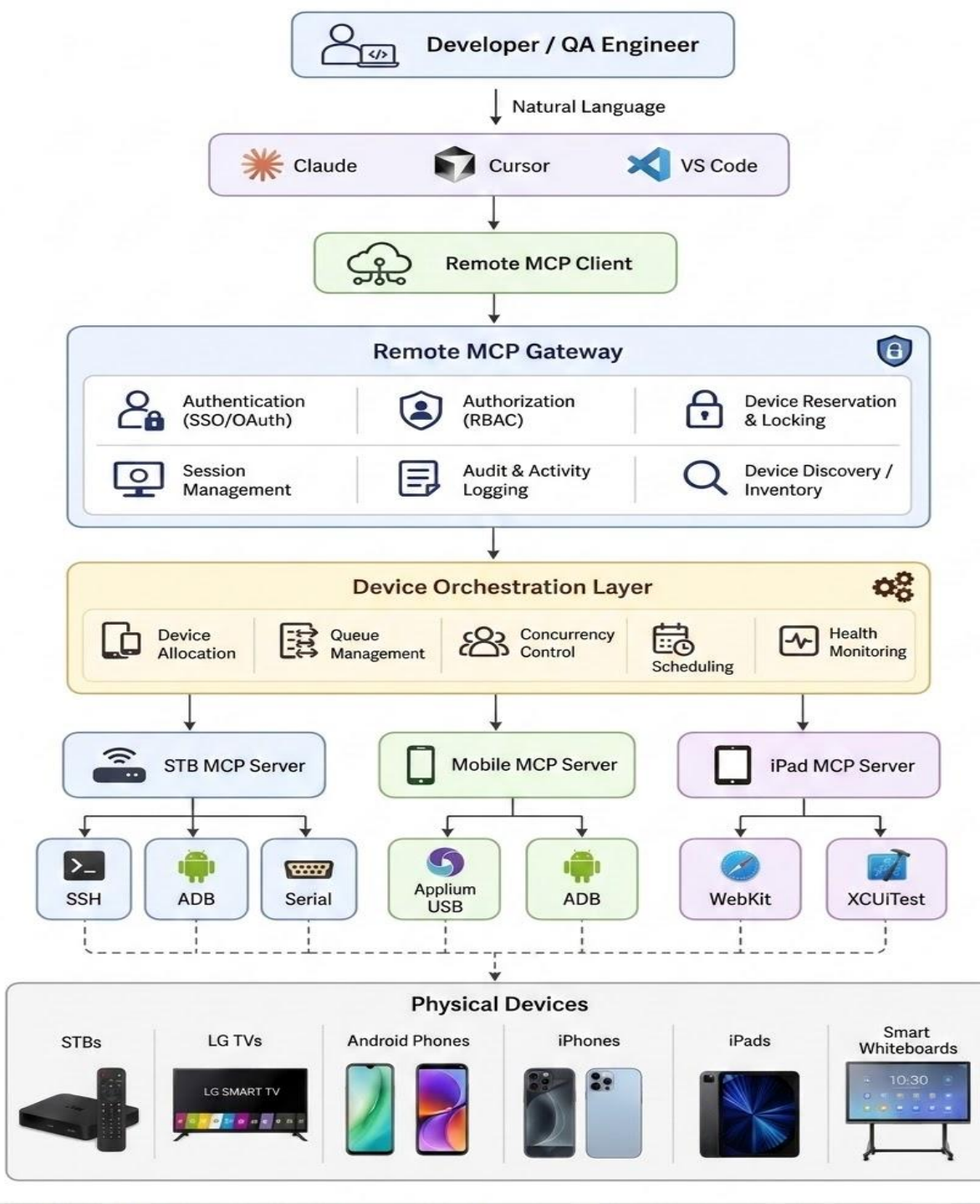
4 Device Types • 4 Protocols • Multiple Teams • One Future AI Interface

The Solution - MCP as the Unified Layer

Model Context Protocol (MCP) unifies all device interactions under one AI-accessible interface.

- AI assistant connects to any device through MCP tools
- Single natural language interface across all platforms
- Geographically distributed teams get 24/7 device access
- No protocol knowledge required from end users

Architecture Overview



MCP GATEWAY LAYER

Authentication (SSO/OAuth)

Provides secure user login using enterprise identity providers (SSO, OAuth2, Entra ID, Okta, etc.).
Eliminates shared lab credentials and ensures all device access is tied to an individual user.

Authorization (RBAC)

Controls which users, teams, or AI agents can access specific devices and MCP tools.
Enforces role-based permissions (Developer, QA, Admin, Automation Agent) to prevent unauthorized actions.

Device Reservation & Locking

Prevents multiple users or AI agents from simultaneously controlling the same device.
Supports exclusive device reservations with automatic lock acquisition and release.

Session Management

Tracks active user and AI sessions across devices and MCP servers.
Automatically expires inactive sessions and releases reserved devices to maximize utilization.

Audit & Activity Logging

Records every action performed on a device, including user, timestamp, command, and outcome.
Provides traceability for debugging, compliance, security investigations, and usage reporting.

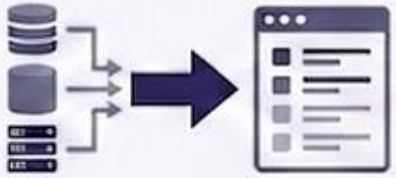
Device Discovery / Inventory

Maintains a centralized catalog of all available devices, configurations, firmware versions, and statuses.
Enables users and AI assistants to discover and reserve devices based such as model, OS version, location, or availability.

Get Well

Device Orchestration Layer

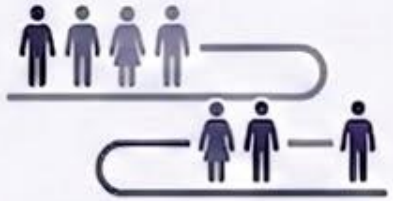
Device Allocation



Automatic assignment based on criteria

Optimizes device utilization

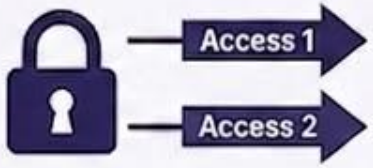
Queue Management



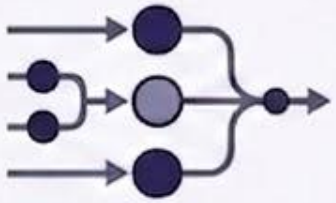
Places users/agents in waiting queue

Supports priority-based allocation

Concurrency Control



Prevents access conflicts



Maintains state consistency

Scheduling

Advance reservations

Supports recurring/ automated allocation

Health Monitoring

Continuously monitors metrics

Detects/removes unhealthy devices

Get Well

Natural Language Testing & AI-Assisted Debugging

User Request → AI Execution

User Says	AI Executes
"Verify patient details on LG TV in Lab1"	Connect → Navigate → Screenshot
"Check medication list on Android device"	Launch App → Capture Evidence
"Verify alerts on Signal iPad"	Check Alerts → Screenshot
"Run smoke test on Room 301 devices"	Parallel Validation (TV+iPad)

How It Works

- **Natural Language Request**
- AI Assistant Reasoning
- Device Reservation
- Automated Execution
- Evidence & Report Generation

- Autonomous Exploratory Testing
- Predictive & Visual Performance Optimization
- Smart Test Maintenance
- Network & Hardware Condition Simulation

Intelligence-Driven Debugging: From Error to Automated Resolution

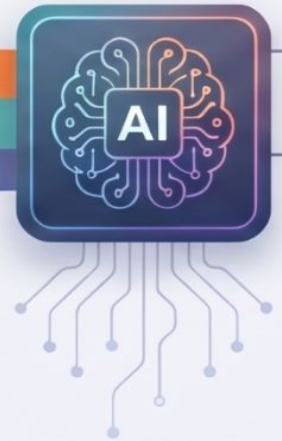
Intelligent Analysis & Reporting

Root Cause Identification via Log Clustering

AI groups similar errors to identify one root bug instead of multiple symptoms.

Automated Bug Reporting & Jira Integration

Generates reproducible reports with device logs and syncs them directly to management tools.



Precision Stack Trace Analysis

Flags the exact line of code causing failure by matching against historical data.

Performance Visibility & Classification

Anomaly Detection & Visual Timelines

Syncs video replays with live metrics to highlight performance spikes at the millisecond level.

Flakiness Classification

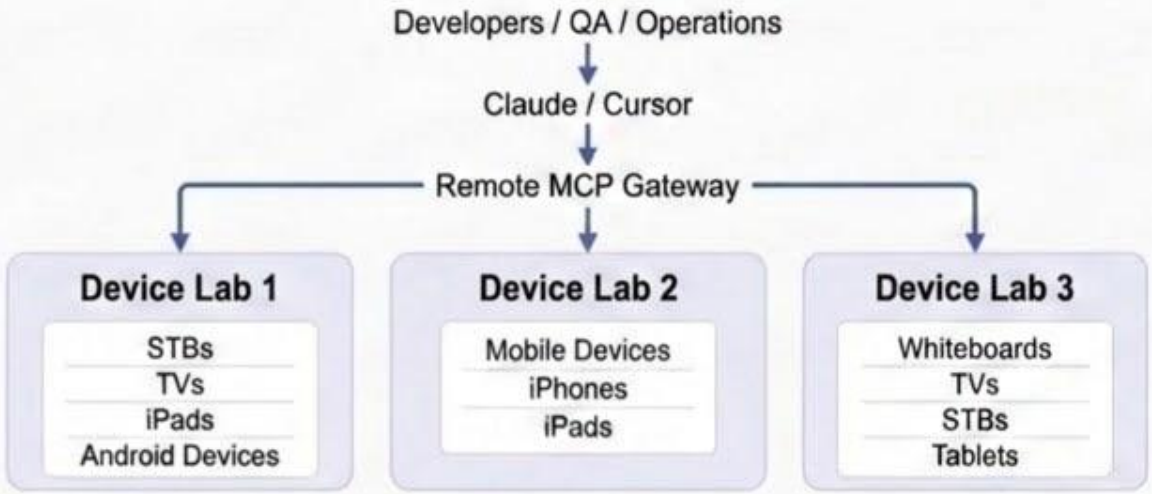
Distinguishes between actual code bugs and environmental issues like slow network timeouts.

Code vs. Environment Stability

Analyzes historic stability to prevent developers from wasting time on non-critical environment flakes.

Enterprise Deployment Architecture

Secure, Scalable, Multi-Site Device Lab Infrastructure



Key Capabilities

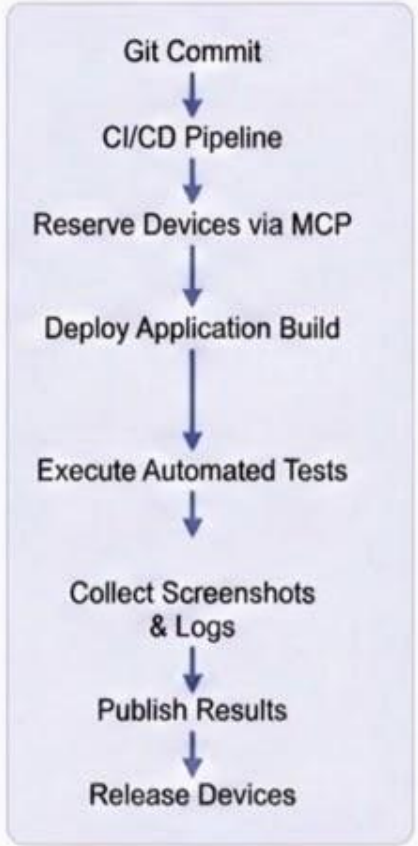
- Centralized authentication and authorization
- Global device discovery and reservation
- Site-aware device allocation
- Secure access without VPN or remote desktop
- remote desktop
- Unified interface for all physical devices
- Full audit trail for every action

Deployment Benefits

- ✅ 24/7 device access
- ✅ Shared infrastructure across teams
- ✅ Reduced operational overhead
- ✅ Better utilization of expensive hardware
- ✅ Automated post-deployment validation
- ✅ Faster release confidence
- ✅ Real-device testing in CI/CD
- ✅ Reduced manual QA effort

CI/CD Integration

Physical Device Validation as Part of Deployment Pipeline



CI/CD

Example Pipeline Actions

- Deploy Android APK
- Install Build
- Update Application
- Execute Smoke Tests
- Run Regression Suites
- Capture Failure Evidence
- Generate Test Reports

Example Natural Language Trigger

"Deploy build 5.2.1 and run smoke tests on Android, iPad, and TV devices."

CI/CD Benefits

- ✅ Automated post-deployment validation
- ✅ Faster release confidence
- ✅ Real-device testing in CI/CD
- ✅ Reduced manual QA effort

VIRTUAL CARE

Virtual Care Platform: System Context & Component Architecture

User Interfaces & Endpoints
(Nurse/Admin)

Nurse Console
(Clinical Interface)

Device Management UI
(Admin Portal)

Component Core
(Internal SaaS Backend)

Call Coordinator
(The Session Engine)

MQTT Broker
(The Messaging Backend)

Device Management Service (DMS)

Zoom Video SDK Integration

MQTT Topic Structure (MQTT MCP)

Component	Topic Level	Purpose
(agerCode)	Tanant Level	Isolates traffic by hospitalUsiant (e.g., H6SP A)
(facilityId)	Facility Level	Organizes devices by specific compuf of building)
(roomid)	Room Level	Ensores only the target room's device receives the call/command
(category)	Message Topic	Categoriess by 'call (sessiona) or "device' (hardcare)

System Context
(External Integrations & In-Room)

GWSL SaaS Ecosystem (GCR & EPP)

Inpatient PLS
(On-Premise Gateway)

Aver Android App
(In-Room Endpoint)

Virtual Care - Details

Nurse Console (GUAC Micro-Frontend)

- Web-based patient census board hosted as a micro-frontend in GUAC portal
- Multi-unit/ward selection with real-time patient status, diagnosis, and call history
- Initiate standard calls and breakthrough (urgent) calls to patient rooms
- In-call PTZ camera control — pan, tilt, zoom the bedside camera remotely
- Add participants mid-call: interpreter, family member, care team

Patient Room App (AVer)

- Native Android coordination app on bedside AVer camera device
- Digital Door Knock — incoming call notification on TV with accept/decline
- Two-way HD video and audio via WebRTC
- Camera PTZ via CGI on device loopback
- Two modes: headless (auto-answer) and TV-attached (patient controls)
- Custom MDM agent — heartbeat, telemetry, OTA updates, kiosk mode
- Security: mTLS certificates, Device Owner kiosk lock, factory reset on decommission

Data Flow

- Outbound: Nurse Console → VC Backend → PLS → Patient TV
- Inbound: Patient Response → WAMP WebSocket + HTTP Webhook → VC Backend → Nurse
- Fleet: Device → REST + mTLS → MDM API → Fleet Management
- Clinical: EMR → HL7 → Mirth → FHIR → Journey Manager → EPP → VC Backend

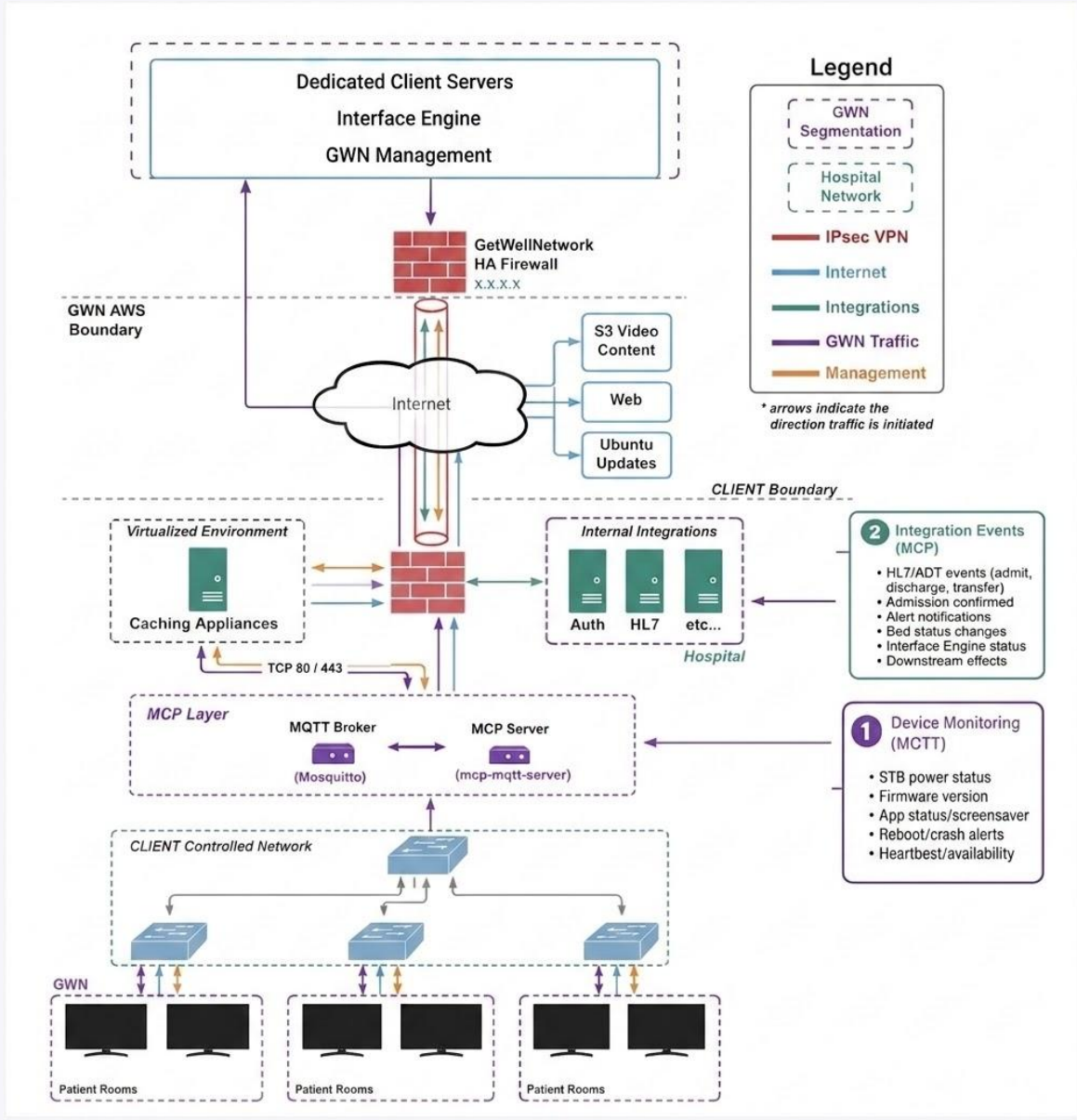
Virtual Care Backend

- Central hub for call signaling and video session management
- Custom MDM fleet API — register, approve, provision, OTA update, decommission devices
- PTZ command routing: nurse console → backend → in-room device → camera
- OAuth 2.0 integration with PLS for video call APIs
- Webhook receiver for patient accept/decline/ignore responses
- Built with: MQTT mCP

Integration Layer

- PLS (Inpatient): start_call, end_call, WAMP WebSocket for patient responses
- EPP (GUAC Services): Patient demographics, encounters, observations via FHIR
- Mirth Connect: HL7 v2 message processing (ADT admissions, ORU observations)
- Journey Manager: HL7 → FHIR transformation pipeline into EPP
- SMART on FHIR: EMR launch context for nurse console within hospital systems

Case Study -Patient Lifecycle System (PLS)



Get Well

References / Acknowledgement

https://medium.com/@harish.vadada/bridging-the-gap-how-fhir-and-the-model-context-protocol-mcp-power-generative-ai-in-healthcare-6e894ddae6b7 - Bridging the Gap: How FHIR and the Model Context Protocol (MCP) power Generative AI in Healthcare

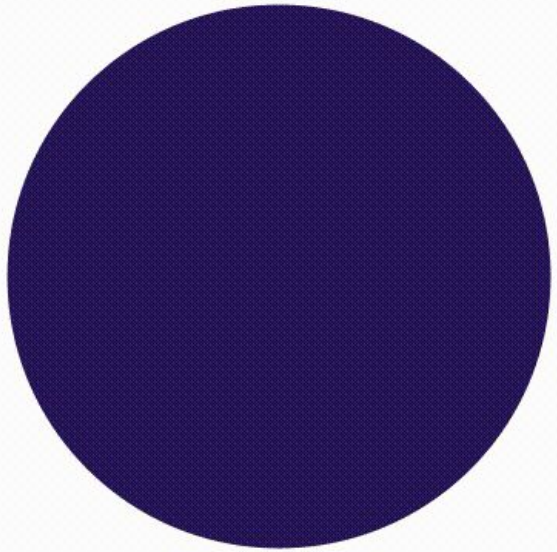
Use-cases.html - MQTT with MCP/AI

https://live.paloaltonetworks.com/t5/community-blogs/mcp-security-exposed-what-you-need-to-know-now/ba-p/1227143

https://arxiv.org/html/2510.01260v1 - **IoT-MCP**: Bridging LLMs and IoT Systems Through Model Context Protocol

https://www.getwellnetwork.com/patient-activation-growth/- GetWell

Thank you!



Get Well