

PROBLEM STATEMENT — MODEL 3

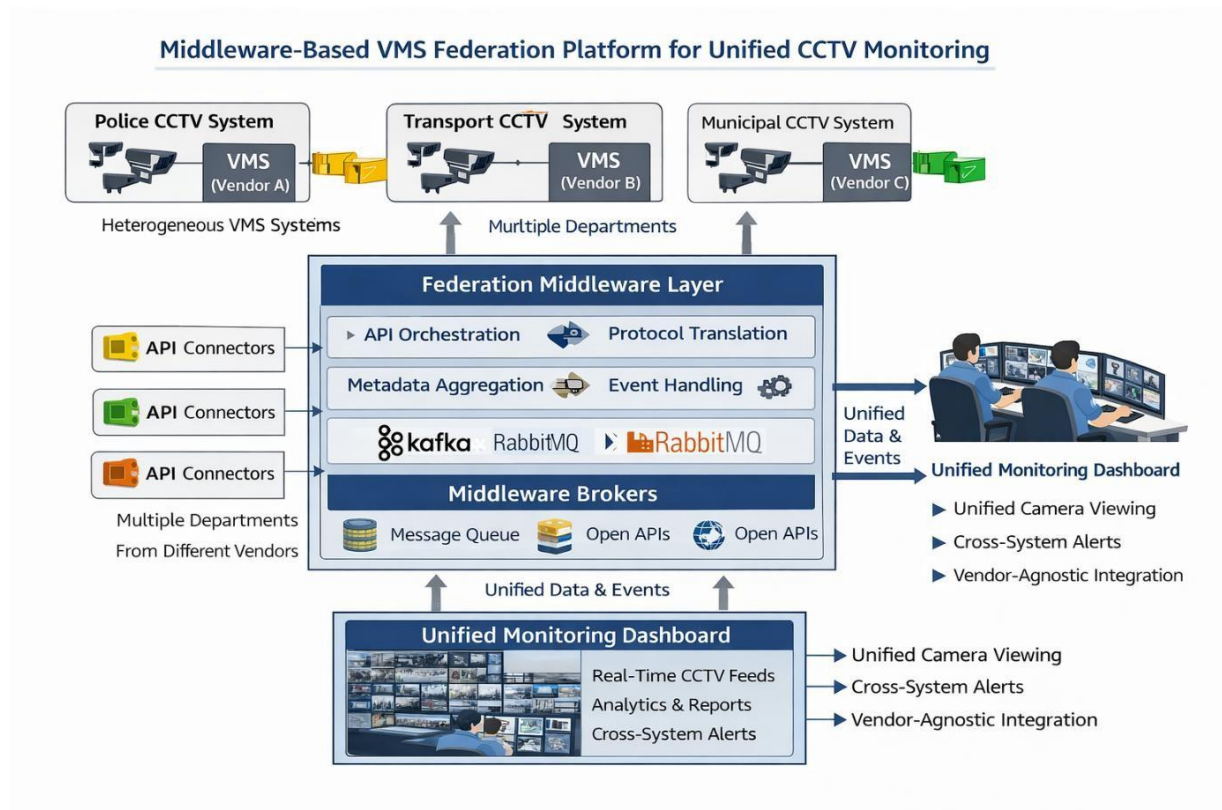


Fig 3

Tech Stack is illustrative , any open source may be used.

✦ Problem ID

- GOG-CCTV-03

✦ Problem Title

- Middleware-Based VMS Federation Platform for Unified CCTV Monitoring

✦ Theme

- Smart Governance / Interoperability

✦ Category

- Software

✦ Organization / Ministry

- Government of Gujarat – Home Department

✦ Industry Type

- Government Department

✦ Domain Tags

- VMS Federation, Middleware, API Integration, Interoperability, Multi-Vendor

✦ Problem Statement (Description)

Different departments across the state use heterogeneous CCTV systems supplied by multiple vendors, resulting in a fragmented monitoring ecosystem. This diversity in platforms and standards makes direct integration challenging and limits the ability to achieve a unified monitoring framework.

As outlined in Model 3 – VMS Federation, there is a need for a middleware layer that can aggregate metadata and facilitate unified viewing across systems without requiring replacement of existing infrastructure. Such an approach enables interoperability while preserving departmental autonomy and investments in current systems.

✦ Problem Statement (Short Abstract)

- Develop a middleware-based platform to integrate multiple VMS systems and provide a unified monitoring dashboard.

✦ Objectives

- Enable interoperability
- Provide unified monitoring
- Support vendor-agnostic integration

✦ Expected Solution / Deliverables

- API-based middleware layer
- Protocol translation system
- Unified dashboard
- Plug-and-play connectors

✦ Dataset / Inputs

- Multi-vendor simulated APIs
- CCTV metadata
- Integration with ≥ 3 different VMS APIs

✦ Constraints

- Must support heterogeneous systems
- Departments retain control

✦ Evaluation Parameters

- Integration capability
- Scalability
- Real-time synchronization

✦ Success Criteria

- Multi-VMS integration
- Unified dashboard functionality

✦ Expected Outcome

- Seamless cross-department integration
- Reduced vendor lock-in

✦ Suggested Technology Stack (Open Source Only)

- Kafka / RabbitMQ
- Node.js / Python
- PostgreSQL
- Any other open-source technologies may also be used

✦ Use Case Scenarios

- Multi-department coordination
- Legacy system integration

✦ Adoption / Deployment Readiness

- Deployable at state level
- Extendable architecture

✦ Difficulty Level

- High

✦ Impact

- Enables scalable integration
- Supports interoperability







| SCORING RUBRIC — MODEL 3 (Out of 100 Marks) | | |
|--|------------|--|
| EVALUATION CRITERIA | MARKS | DESCRIPTION / JUSTIFICATION |
|  Standardization & Protocol Conversion | 20 | Core of this model—accurate, lossless, and reliable conversion to standard protocols (RTSP/ONVIF) is essential for seamless integration. |
|  Hardware Design & Performance | 20 | The transponder/encoder must be stable, compatible, compact, and suitable for field deployment with diverse legacy devices. |
|  System Integration & Interoperability | 20 | Converted feeds must work seamlessly with downstream systems of all other Hackathon models (VMS, CCC platforms) across brands and formats. |
|  Latency & Stream Quality | 20 | Real-time performance with minimal added latency and good video quality is critical for operational use. |
|  Security | 10 | Secure boot, authenticated access, encrypted streaming, and device-level security are necessary for safe deployment. |
|  Innovation & Deployment Readiness | 10 | Value-added features (health monitoring, remote config, scalability, PoE support, rugged design) encouraged. |
| TOTAL | 100 | |

Fig 3.1