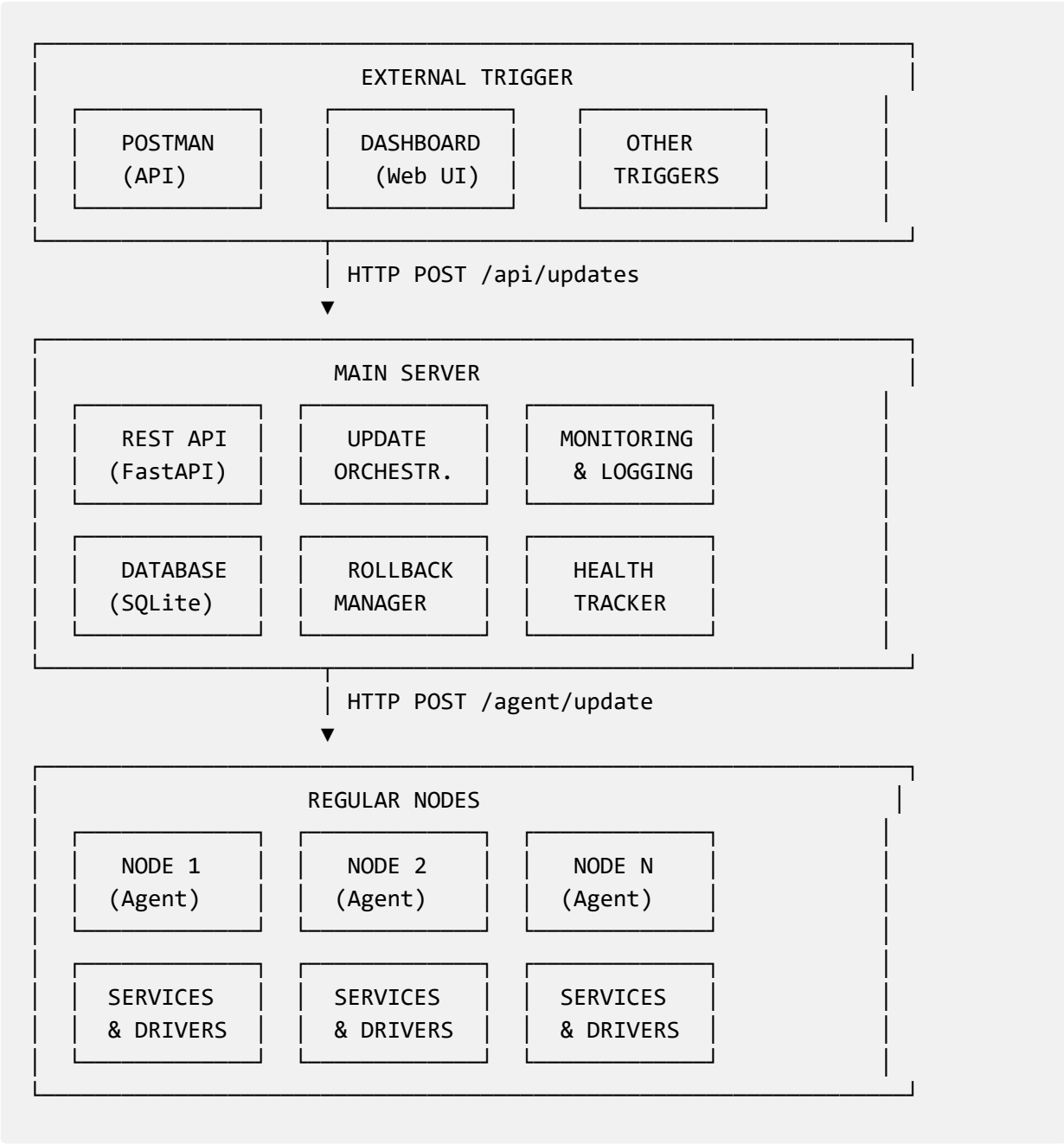


# Central Service & Driver Management System Architecture

## System Overview

This system manages service and driver updates across multiple Linux machines with a central Main Server coordinating updates to Regular Nodes.

## Architecture Diagram



## Technology Stack

### Main Server

- **Language:** Python 3.9+
- **Framework:** FastAPI (lightweight, async)
- **Database:** SQLite (embedded, no external dependencies)
- **Container:** Docker (for easy deployment)
- **Process Management:** systemd

## Regular Nodes

- **Language:** Python 3.9+ (agent)
- **Process Management:** systemd
- **Update Method:** Package manager (apt/yum) + Docker containers
- **Health Checks:** systemd + custom scripts

## Update Flow

---

1. **Trigger:** External system sends update request to Main Server
2. **Validation:** Main Server validates request and checks node availability
3. **Distribution:** Main Server sends update commands to all target Regular Nodes
4. **Execution:** Each Regular Node:
  - Downloads update package
  - Stops affected services
  - Installs update
  - Restarts services
  - Runs health check
5. **Reporting:** Regular Node reports status back to Main Server
6. **Rollback:** If any node fails, Main Server triggers rollback

## Key Features

---

- **Lightweight:** Minimal resource usage (suitable for 8GB RAM machines)
- **Scalable:** Can manage hundreds of Regular Nodes
- **Reliable:** Automatic rollback on failure
- **Monitorable:** Comprehensive logging and health tracking
- **Automated:** Full automation of update process

## File Structure

---

```
cvtask/
├── main_server/
│   ├── app/
│   │   ├── __init__.py
│   │   ├── main.py
│   │   ├── models.py
│   │   ├── database.py
│   │   └── update_manager.py
│   ├── Dockerfile
│   ├── requirements.txt
│   └── systemd/
│       └── main-server.service
├── regular_node/
│   ├── agent/
│   │   ├── __init__.py
│   │   ├── main.py
│   │   ├── update_handler.py
│   │   └── health_checker.py
│   ├── scripts/
│   │   ├── install_update.sh
│   │   ├── rollback.sh
│   │   └── health_check.sh
│   ├── Dockerfile
│   ├── requirements.txt
│   └── systemd/
│       └── node-agent.service
├── docker-compose.yml
├── setup.sh
└── README.md
```