Data Movement & Cloud Strategy

The data movement and cloud layer is responsible for the **secure, reliable, and real-time transmission of waste bin data** from distributed field devices to a centralized monitoring dashboard. The design prioritizes **low bandwidth usage, scalability, and robustness**, making it suitable for dense urban deployments.

1. Communication Protocol

Protocol Used: MQTT (Message Queuing Telemetry Transport)

Technical Characteristics

- Lightweight publish–subscribe messaging protocol
- Operates efficiently over unreliable or low-bandwidth networks
- Minimal packet overhead compared to HTTP
- Supports Quality of Service (QoS) levels (0, 1, 2)
- Designed specifically for IoT and M2M communication

Justification

MQTT is selected because:

- Waste bins transmit **small, infrequent data packets**
- LoRaWAN provides limited bandwidth
- Publish–subscribe model decouples devices from servers
- Enables scalable communication for **100+ bin nodes**
- Reduces power consumption on ESP32 devices

2. End-to-End Data Flow

The system follows a structured, multi-layer data pipeline to ensure smooth data delivery from sensors to decision-makers.

Step-by-Step Data Flow

1. **Sensor Layer**
    - HC-SR04 ultrasonic sensor measures the distance between bin lid and waste surface.
    - ESP32 calculates fill percentage and battery voltage.

2. **Edge Processing (ESP32)**

    - Filters noise using multiple readings
    - Converts raw sensor values into meaningful metrics
    - Packages data into a compact JSON format

3. **LoRa Transmission**

    - JSON payload transmitted via LoRa to the nearest LoRaWAN gateway.
    - Data includes:
        - Bin ID
        - Fill percentage
        - Battery level
        - Timestamp

4. **LoRa Gateway**

    - Receives LoRa packets from multiple bin nodes.
    - Forwards decoded data to cloud servers using IP-based backhaul (Ethernet / LTE).

5. **AWS IoT Core**

    - Acts as the MQTT broker.
    - Authenticates devices using certificates.
    - Routes incoming messages to cloud services.

6. **Data Processing & Storage**

    - AWS Lambda processes incoming data.
    - Data stored in:
        - DynamoDB (real-time status)
        - S3 (historical analytics)

7. **GIS Dashboard**

    - Cloud API feeds processed data to a web-based dashboard.
    - Dashboard visualizes bin status on a city map.

3. Cloud Strategy (AWS-Based)

AWS IoT Core

- MQTT message broker
- Device authentication and authorization
- Scalable to thousands of devices

Data Processing

- **AWS Lambda**
  - Triggered by MQTT messages
  - Applies business logic (threshold checks, alerts)

Data Storage

- **DynamoDB**
  - Stores current bin status
  - Enables fast dashboard updates

- **Amazon S3**
  - Stores historical data for analytics and reporting

4. Dashboard & Visualization Concept

Dashboard Overview

- Web-based GIS dashboard accessible to municipal authorities
- Displays real-time bin status on an interactive city map

Color-Coded Visualization

- **Red Pin:** Bin fill level > 80% (Immediate collection required)
- **Yellow Pin:** Bin fill level between 50% – 80% (Monitor)
- **Green Pin:** Bin fill level < 50% (Normal)

Dashboard Features

- Zone-wise filtering
- Priority bin highlighting

- Route suggestion overlay for garbage trucks
- Historical data view for trend analysis

5. Benefits of the Proposed Data & Cloud Architecture

- **Low Latency:** Near real-time updates
- **Low Bandwidth Usage:** Optimized for LoRa networks
- **Scalable:** Easily supports city-wide deployments
- **Secure:** Certificate-based device authentication
- **Actionable Insights:** Visual decision-making for authorities