Complete IoT Architecture Types & Detailed Explanation

Table of Contents

---

1. BASIC IoT ARCHITECTURE MODELS

Model 1: Three-Layer Architecture (Simplest)

Layer 1: Perception Layer (Physical Layer)

Purpose: Interface with the physical world

Components:

- **Sensors:** Devices that detect and measure physical phenomena
  - Temperature sensors (DHT11, DHT22, DS18B20, LM35)
  - Humidity sensors (DHT series, HIH-4000)
  - Pressure sensors (BMP180, BMP280, MPX5700)
  - Light sensors (LDR, BH1750, TSL2561)
  - Motion sensors (PIR HC-SR501, RCWL-0516)
  - Proximity sensors (Ultrasonic HC-SR04, IR sensors)
  - Gas sensors (MQ-2 smoke, MQ-135 air quality, MQ-7 CO)
  - Accelerometers (ADXL345, MPU6050)
  - Gyroscopes (L3G4200D, MPU6050)
  - GPS modules (NEO-6M, NEO-7M, NEO-8M)
  - Current sensors (ACS712, INA219)
  - Voltage sensors (Voltage dividers, DC sensors)
  - Sound sensors (Microphones, decibel meters)

    - Soil moisture sensors (Capacitive, Resistive)
    - pH sensors (Analog pH meters)
    - Flow sensors (Water flow, air flow)
    - Biometric sensors (Fingerprint, heart rate, pulse oximeter)
    - Camera modules (ESP32-CAM, Raspberry Pi Camera)
- **Actuators:** Devices that perform actions
    - DC motors (Brushed, brushless)
    - Servo motors (SG90, MG996R)
    - Stepper motors (28BYJ-48, NEMA 17)
    - Relays (5V relay modules, solid-state relays)
    - Solenoid valves (Water control, pneumatic)
    - LED indicators (Single color, RGB, LED strips)
    - Buzzers (Active, passive)
    - LCD/OLED displays (16x2 LCD, 128x64 OLED)
    - Speakers
    - Pumps (Water pumps, air pumps)
    - Heaters and coolers
    - Locks (Electromagnetic, solenoid)
- **Identification Technologies:**
    - RFID tags and readers (125kHz, 13.56MHz)
    - NFC modules (PN532, RC522)
    - QR codes and barcode scanners
    - Biometric identifiers

Characteristics:

- Direct interaction with physical environment
- Low power consumption requirements
- Often battery-operated
- Limited processing capability
- Analog and digital signal generation

Working Principle: Sensors convert physical parameters into electrical signals. These signals are conditioned (amplified, filtered) and converted from analog to digital format using ADC (Analog-to-Digital Converter). The digital data is then ready for transmission to the network layer.

Layer 2: Network Layer (Transport Layer)

Purpose: Transmit data from perception layer to processing systems

Components:

A. Communication Technologies:

Short-Range Communication (< 100m):

1. **Bluetooth Classic (802.15.1)**
   - Range: 10-100m depending on class
   - Data rate: 1-3 Mbps
   - Power: Medium to high
   - Use cases: Audio streaming, file transfer, peripheral devices
   - Topology: Point-to-point, piconet (1 master, 7 slaves)
2. **Bluetooth Low Energy (BLE 4.0/5.0)**
   - Range: 50-100m (BLE 5.0 up to 240m)
   - Data rate: 1-2 Mbps
   - Power: Very low (coin cell battery for months)
   - Use cases: Wearables, beacons, health monitors, proximity sensors
   - Topology: Star, mesh (BLE 5.0)
   - Features: Advertising, GATT profiles, connection-less
3. **WiFi (802.11 b/g/n/ac/ax)**
   - Range: 50-100m indoors
   - Data rate: 11 Mbps (b) to 9.6 Gbps (ax/WiFi 6)
   - Power: High consumption
   - Use cases: High bandwidth applications, video streaming, heavy data
   - Topology: Star (infrastructure mode), mesh (WiFi mesh)
   - Frequency: 2.4 GHz, 5 GHz, 6 GHz (WiFi 6E)
4. **Zigbee (802.15.4)**
   - Range: 10-100m per hop
   - Data rate: 250 kbps
   - Power: Very low

- Use cases: Home automation, smart lighting, industrial control
- Topology: Mesh, star, tree
- Features: Self-healing mesh, up to 65,000 nodes

5. **Z-Wave**
   - Range: 30m per hop, 100m with mesh
   - Data rate: 100 kbps
   - Power: Low
   - Use cases: Home automation, security systems
   - Topology: Mesh
   - Features: 232 devices per network, encrypted
6. **Thread**
   - Range: Similar to Zigbee
   - Data rate: 250 kbps
   - Power: Low
   - Use cases: Smart home, Matter protocol compatible
   - Topology: IPv6-based mesh
   - Features: Self-healing, secure, low latency
7. **NFC (Near Field Communication)**
   - Range: < 4cm (very short)
   - Data rate: 424 kbps
   - Power: Very low (passive tags need no battery)
   - Use cases: Contactless payment, access control, pairing
   - Modes: Read/Write, Peer-to-Peer, Card Emulation

Medium-Range Communication (100m - 1km):

1. **WiFi HaLow (802.11ah)**
   - Range: Up to 1km
   - Data rate: 150 kbps - 347 Mbps
   - Power: Lower than traditional WiFi
   - Frequency: Sub-1 GHz
   - Use cases: Smart agriculture, industrial IoT

Long-Range Communication (> 1km):

1. **LoRa (Long Range)**
    - Range: 2-5km (urban), 15km (rural)
    - Data rate: 0.3-50 kbps
    - Power: Very low (battery life 2-10 years)
    - Use cases: Smart cities, agriculture, asset tracking
    - Frequency: 433 MHz, 868 MHz (EU), 915 MHz (US)
2. **LoRaWAN (Protocol for LoRa)**
    - Architecture: Star-of-stars
    - Classes: A (lowest power), B (scheduled), C (always on)
    - Security: AES 128-bit encryption
    - Network capacity: Millions of devices
3. **Sigfox**
    - Range: 10-50km
    - Data rate: 100 bps (uplink), 600 bps (downlink)
    - Power: Ultra-low
    - Use cases: Asset tracking, environmental monitoring
    - Message limit: 140 messages/day (uplink)
4. **NB-IoT (Narrowband IoT)**
    - Range: Up to 10km
    - Data rate: 250 kbps (downlink), 20 kbps (uplink)
    - Power: Low
    - Use cases: Smart meters, parking, tracking
    - Network: Uses existing cellular infrastructure
    - Coverage: Deep indoor penetration
5. **LTE-M (LTE Cat-M1)**
    - Range: Similar to cellular
    - Data rate: 1 Mbps
    - Power: Low (better than regular LTE)
    - Use cases: Wearables, medical devices, fleet management
    - Features: Voice support, mobility, handover
6. **5G IoT**
    - Range: Variable based on frequency

- Data rate: Up to 10 Gbps
- Latency: < 1ms (Ultra-Reliable Low Latency)
- Use cases: Autonomous vehicles, industrial automation, AR/VR
- Features: Network slicing, massive IoT support

7. **Satellite IoT**
    - Range: Global coverage
    - Data rate: Low to medium
    - Use cases: Remote areas, maritime, aviation
    - Providers: Iridium, Globalstar, Inmarsat

B. Network Devices:

1. **Gateways:**
    - Protocol converters
    - Aggregate data from multiple sensors
    - Edge processing capabilities
    - Examples: Industrial IoT gateways, home automation hubs
2. **Routers:**
    - Direct data packets
    - Network address translation
    - Firewall capabilities
3. **Access Points:**
    - WiFi connectivity
    - Multiple device connections
    - Bridge between wired and wireless
4. **Switches:**
    - Connect multiple devices in LAN
    - Data link layer forwarding
    - Managed vs unmanaged

C. Communication Protocols:

1. **MQTT (Message Queuing Telemetry Transport)**
    - Type: Publish-Subscribe

- Port: 1883 (unencrypted), 8883 (encrypted)
- Overhead: Very low (2 bytes minimum)
- QoS Levels:
  - QoS 0: At most once (fire and forget)
  - QoS 1: At least once (acknowledged delivery)
  - QoS 2: Exactly once (guaranteed delivery)
- Components: Broker, Publisher, Subscriber
- Topics: Hierarchical structure (e.g., home/bedroom/temperature)
- Best for: Low bandwidth, unreliable networks, mobile

2. **CoAP (Constrained Application Protocol)**
   - Type: Request-Response (like HTTP)
   - Port: 5683 (UDP)
   - Overhead: Very low
   - Methods: GET, POST, PUT, DELETE
   - Features: Observe pattern, resource discovery
   - Best for: Constrained devices, low power networks
3. **HTTP/HTTPS**
   - Type: Request-Response
   - Port: 80 (HTTP), 443 (HTTPS)
   - Methods: GET, POST, PUT, DELETE, PATCH
   - Overhead: High compared to MQTT/CoAP
   - Best for: High bandwidth, RESTful APIs
4. **WebSocket**
   - Type: Full-duplex communication
   - Port: 80, 443
   - Features: Real-time bidirectional data
   - Best for: Live dashboards, real-time notifications
5. **AMQP (Advanced Message Queuing Protocol)**
   - Type: Message-oriented middleware
   - Port: 5672
   - Features: Reliable, queuing, routing, transactions
   - Best for: Enterprise applications, critical data

6. **DDS (Data Distribution Service)**
    - Type: Pub-Sub, real-time
    - Features: Quality of Service, discovery
    - Best for: Industrial control, military, aerospace
7. **Modbus**
    - Type: Industrial protocol
    - Variants: Modbus RTU (serial), Modbus TCP (Ethernet)
    - Best for: PLCs, industrial sensors
8. **OPC UA (OPC Unified Architecture)**
    - Type: Platform-independent industrial protocol
    - Features: Security, information modeling
    - Best for: Industry 4.0, interoperability

Layer 3: Application Layer (Processing & Services)

Purpose: Process, analyze, store data and provide services

Components:

A. Edge Computing:

- Processing at or near data source
- Reduces latency (< 10ms response time)
- Reduces bandwidth usage
- Local decision making
- Offline capability
- Examples: Industrial edge servers, smart gateways, edge AI processors

B. Fog Computing:

- Intermediate layer between edge and cloud
- Distributed computing resources
- Hierarchical architecture
- Better for geographically distributed systems

C. Cloud Computing:

Cloud Platforms:

1. **AWS IoT Core**
    - Device SDK support
    - Device shadows (digital twins)
    - Rules engine for data routing
    - Integration with AWS services (Lambda, S3, DynamoDB)
    - Security with X.509 certificates
2. **Microsoft Azure IoT Hub**
    - Device provisioning service
    - IoT Edge for edge computing
    - Device twins
    - Integration with Azure services
    - Azure Digital Twins for modeling
3. **Google Cloud IoT Core**
    - Device management
    - Protocol endpoints (MQTT, HTTP)
    - Integration with Cloud Pub/Sub
    - Dataflow for stream processing
4. **IBM Watson IoT Platform**
    - Device management
    - AI/ML integration
    - Blockchain integration
    - Industry-specific solutions
5. **Open Source Platforms:**
    - ThingsBoard: Device management, visualization
    - OpenRemote: Building automation
    - Kaa IoT Platform: Enterprise IoT
    - ThingSpeak: Data collection and visualization

D. Data Storage:

1. **Time-Series Databases:**
    - InfluxDB: High write throughput, time-based queries

- TimescaleDB: PostgreSQL extension for time-series
- OpenTSDB: Hadoop-based, scalable
- Prometheus: Monitoring and alerting

2. **NoSQL Databases:**
   - MongoDB: Document store, flexible schema
   - Cassandra: Wide-column store, highly scalable
   - Redis: In-memory, fast access, caching
   - CouchDB: Document store, offline-first
3. **SQL Databases:**
   - PostgreSQL: Feature-rich, reliable
   - MySQL: Popular, widely supported
   - SQLite: Lightweight, embedded
4. **Data Lakes:**
   - Hadoop HDFS: Distributed file system
   - AWS S3: Object storage
   - Azure Data Lake: Analytics service

E. Data Processing:

1. **Stream Processing:**
   - Apache Kafka: Distributed streaming, high throughput
   - Apache Flink: Real-time stream processing
   - Apache Storm: Distributed real-time computation
   - AWS Kinesis: Managed streaming service
2. **Batch Processing:**
   - Apache Spark: Fast, in-memory processing
   - Apache Hadoop MapReduce: Distributed processing
   - AWS EMR: Managed Hadoop framework
3. **Message Queues:**
   - RabbitMQ: AMQP broker
   - Apache ActiveMQ: Java-based messaging
   - AWS SQS: Managed queue service

F. Analytics & AI:

1. **Descriptive Analytics:**
    - What happened?
    - Historical data analysis
    - Dashboards and reports
2. **Diagnostic Analytics:**
    - Why did it happen?
    - Root cause analysis
    - Correlation analysis
3. **Predictive Analytics:**
    - What will happen?
    - Machine learning models
    - Forecasting
    - Anomaly detection
4. **Prescriptive Analytics:**
    - What should we do?
    - Optimization algorithms
    - Decision automation
    - Recommendation systems

G. Visualization:

- Grafana: Time-series visualization
- Kibana: Elasticsearch visualization
- Tableau: Business intelligence
- Power BI: Microsoft analytics
- Custom dashboards: Web-based (React, Vue, Angular)

H. Application Services:

- User management and authentication
- Device provisioning and management
- Firmware over-the-air (FOTA) updates
- Alert and notification services
- Business logic implementation

- API gateways
- Integration with third-party services

Model 2: Four-Layer Architecture (Most Common)

This adds a separate **Processing Layer** between Network and Application:

Layer 1: Perception Layer

(Same as three-layer model)

Layer 2: Network Layer

(Same as three-layer model)

Layer 3: Processing Layer (Middleware)

Purpose: Intelligent data processing and management

Functions:

- Data filtering and aggregation
- Data transformation and normalization
- Protocol conversion
- Device management
- Security and authentication
- Load balancing
- Service orchestration

Components:

- Event processing engines
- Complex event processing (CEP)
- Data enrichment services
- Business rule engines
- Workflow management
- Service bus

Layer 4: Application Layer

Purpose: User-facing applications and services

Types of Applications:

- Mobile applications (iOS, Android, cross-platform)
- Web applications and dashboards
- Desktop applications
- Voice assistants (Alexa, Google Assistant)
- Enterprise applications (ERP, CRM integration)
- Analytics and reporting tools
- Control and monitoring interfaces

Model 3: Five-Layer Architecture (Detailed)

Adds **Business Layer** on top:

Layers 1-4: Same as four-layer model

Layer 5: Business Layer

Purpose: Business logic, decision making, and value creation

Functions:

- Business process management
- Revenue models
- User management and billing
- Service level agreements (SLA)
- Compliance and governance
- Business intelligence
- ROI analysis and optimization

---

2. DETAILED LAYERED ARCHITECTURE MODELS

Architecture Type 1: Perception-Centric Architecture

Focus: Rich sensing and actuation

Structure:

```
Business Applications
       ↑
Application Processing & Analytics
       ↑
Edge/Fog Computing Layer
       ↑
Network & Communication
       ↑
Advanced Sensor Networks (Multiple sensor types, fusion)
```

Use Cases:

- Environmental monitoring
- Smart agriculture
- Industrial monitoring
- Healthcare monitoring

Characteristics:

- Large number of heterogeneous sensors
- Sensor data fusion
- Local preprocessing
- Redundancy for reliability

Architecture Type 2: Network-Centric Architecture

Focus: Communication and connectivity

Structure:

```
Applications & Services
       ↑
Application Support Layer
       ↑
Network Core (SDN, NFV)
       ↑
Access Networks (Multiple protocols)
       ↑
Gateways & Edge Devices
       ↑
Sensors & Actuators
```

Use Cases:

- Telecommunications IoT
- Mobile IoT applications
- Wide-area monitoring
- Connected vehicles

Characteristics:

- Software-defined networking (SDN)
- Network function virtualization (NFV)
- Multiple communication protocols
- Quality of Service (QoS) management
- Network slicing (5G)

Architecture Type 3: Cloud-Centric Architecture

Focus: Centralized processing and storage

Structure:

```
Cloud Applications & AI
        ↑
Cloud Storage & Analytics
        ↑
Cloud IoT Services
        ↑
Internet Gateway
        ↑
Local Network
        ↑
IoT Devices
```

Use Cases:

- Consumer IoT (smart home)
- Non-real-time applications
- Data-intensive analytics
- Machine learning applications

Characteristics:

- Centralized data processing
- Scalable computing resources
- Advanced analytics and AI
- Higher latency compared to edge
- Requires reliable internet connectivity

Architecture Type 4: Edge-Centric Architecture

Focus: Local processing and low latency

Structure:

```
Cloud (Backup & Long-term Storage)
        ↑
Edge Data Centers
        ↑
Edge Gateways/Servers (Primary Processing)
        ↑
Local Network
        ↑
IoT Devices
```

Use Cases:

- Industrial automation
- Autonomous vehicles
- Augmented reality
- Real-time video analytics
- Safety-critical applications

Characteristics:

- Ultra-low latency (< 5ms)
- Local data processing
- Offline capability
- Privacy and security
- Reduced bandwidth usage

Architecture Type 5: Fog Computing Architecture

Focus: Distributed, hierarchical processing

Structure:

```
Cloud Layer (Global services)
       ↑
Fog Layer 3 (Regional data centers)
       ↑
Fog Layer 2 (Local edge servers)
       ↑
Fog Layer 1 (Smart gateways)
       ↑
IoT Devices & Sensors
```

Use Cases:

- Smart cities
- Connected vehicles
- Large-scale industrial IoT
- Distributed monitoring systems

Characteristics:

- Hierarchical architecture
- Load distribution
- Geographically distributed
- Supports mobility
- Balances latency and scalability

---

3. IoT REFERENCE ARCHITECTURES

3.1 IoT World Forum (IoT-WF) Reference Model

7-Layer Model by Cisco:

Level 1: Physical Devices & Controllers

- Sensors, actuators, devices
- Controllers (PLCs, microcontrollers)

Level 2: Connectivity

- Communication and connectivity
- Protocol translation

- Switching and routing

Level 3: Edge (Fog) Computing

- Data filtering and reduction
- Evaluation and reformatting
- Local processing

Level 4: Data Accumulation

- Storage (databases)
- Event, data management

Level 5: Data Abstraction

- Aggregation and normalization
- Multiple data source access

Level 6: Application

- Reporting, analytics
- Control applications

Level 7: Collaboration & Processes

- People and business processes
- Integrations with existing systems

3.2 ITU-T IoT Reference Model

4-Layer Model:

Application Layer:

- IoT applications for different industries

Service Support and Application Support Layer:

- Generic support capabilities
- Specific support capabilities

Network Layer:

- Networking capabilities for control and transport

Device Layer:

- Device capabilities for sensing and actuating

3.3 Industrial Internet Reference Architecture (IIRA)

By Industrial Internet Consortium:

Viewpoints:

1. **Business Viewpoint:**
    - Business vision, values, objectives
    - Stakeholder concerns
2. **Usage Viewpoint:**
    - System activities and sequences
    - Use cases
3. **Functional Viewpoint:**
    - System components
    - Functional capabilities
4. **Implementation Viewpoint:**
    - Technologies and architectures
    - Deployment patterns
5. **System Characteristics:**
    - Safety, security, privacy
    - Resilience, scalability

3.4 Azure IoT Reference Architecture

Microsoft's Model:

IoT Devices:

- Hardware devices and sensors

Cloud Gateway:

- Protocol gateway
- Device connectivity

Stream Processing:

- Real-time data processing
- Rules engine

Storage:

- Warm storage (recent data)
- Cold storage (historical data)

User Interface:

- Dashboards
- Mobile apps

Business Applications:

- Integration with existing systems
- Business logic

---

4. DEPLOYMENT ARCHITECTURE TYPES

4.1 Standalone IoT System

Structure:

- Self-contained system
- No external dependencies
- Local processing and storage

Example:

```
Smart Thermostat
   ↓
Local Control Unit
   ↓
Heating/Cooling System
```

Use Cases:

- Embedded systems
- Offline applications
- Security-critical systems

4.2 Client-Server Architecture

Structure:

```
Multiple IoT Devices (Clients)
       ↓
   Server
       ↓
Database & Processing
```

Characteristics:

- Centralized control
- Clients send data to server
- Server makes decisions
- Traditional architecture

4.3 Peer-to-Peer (P2P) Architecture

Structure:

```
IoT Device 1 ←→ IoT Device 2
    ↕           ↕
IoT Device 3 ←→ IoT Device 4
```

Characteristics:

- No central server
- Direct device-to-device communication
- Distributed decision making
- Resilient to single point of failure

Use Cases:

- Mesh networks
- Home automation
- Disaster recovery systems

4.4 Hybrid Architecture

Structure:

```
        Cloud Services
             ↑
        Edge Servers
        ↗    ↑    ↖
Device 1  Device 2  Device 3
```

Characteristics:

- Combines multiple architectures
- Flexibility in processing location
- Optimized for specific requirements

---

5. COMMUNICATION ARCHITECTURE

5.1 Point-to-Point

Structure: Direct one-to-one connection

Example:

```
Sensor → Gateway
```

Protocols: Serial (UART), I2C, SPI

5.2 Star Topology

Structure: Central hub with multiple devices

Example:

```
   Gateway
  /  |  \
Device1 Device2 Device3
```

Protocols: WiFi, BLE, Zigbee

Advantages:

- Easy to set up
- Easy troubleshooting
- Centralized management

Disadvantages:

- Single point of failure (hub)
- Limited by hub capacity

5.3 Mesh Topology

Structure: Devices interconnected

Example:

```
Device1 ←→ Device2
 ↕   \ /   ↕
Device3 ←→ Device4
```

Protocols: Zigbee, Thread, BLE Mesh

Advantages:

- Self-healing
- Extended range
- No single point of failure
- Scalable

Disadvantages:

- Complex routing
- Higher power consumption

5.4 Tree/Hierarchical Topology

Structure: Parent-child relationships

Example:

```
    Gateway
   /     \
Router1  Router2
 | \    / |
D1  D2  D3  D4
```

Use Cases:

- Industrial control systems
- Building automation

5.5 Bus Topology

Structure: Single communication line

Example:

```
D1—D2—D3—D4—D5
```

Protocols: Modbus, CAN bus

Use Cases:

- Automotive
- Industrial automation

---

6. DATA FLOW ARCHITECTURE

6.1 Upstream Data Flow

Direction: Device → Cloud

Purpose:

- Telemetry data
- Sensor readings
- Status updates
- Alerts and alarms

Protocols:

- MQTT (publish)
- HTTP POST
- CoAP POST

Data Types:

- Temperature, humidity readings
- GPS coordinates
- Image/video data
- Device status

6.2 Downstream Data Flow

Direction: Cloud → Device

Purpose:

- Commands and control
- Configuration updates
- Firmware updates (OTA)
- Queries

Protocols:

- MQTT (subscribe)

- HTTP polling
- WebSocket
- CoAP PUT

Data Types:

- Control commands (turn on/off)
- Configuration parameters
- Firmware binaries
- Scheduling information

6.3 Bidirectional Data Flow

Most common in IoT:

Example Flow:

```
Device sends: Temperature = 25°C
Cloud processes: Temperature too high
Cloud sends: Turn on AC
Device confirms: AC turned on
Device sends: New temperature = 22°C
```

6.4 Device-to-Device Communication

No cloud involvement:

Example:

```
Motion Sensor → Smart Light
  (directly)
```

Use Cases:

- Home automation
- Industrial control
- Emergency systems

---

7. SECURITY ARCHITECTURE

7.1 Defense in Depth (Layered Security)

Layer 1: Device Security

- Secure boot
- Hardware root of trust
- Secure element chips
- Tamper detection

Layer 2: Communication Security

- TLS/SSL encryption
- VPN tunnels
- Secure protocols (MQTTS, HTTPS)
- Certificate-based authentication

Layer 3: Network Security

- Firewalls
- Intrusion detection systems (IDS)
- Network segmentation
- Access control lists (ACL)

Layer 4: Application Security

- Authentication (OAuth, JWT)
- Authorization (RBAC, ABAC)
- Input validation
- Secure APIs

Layer 5: Data Security

- Encryption at rest
- Encryption in transit
- Data anonymization
- Backup and recovery

Layer 6: Management Security

- Secure device provisioning
- Key management
- Certificate management
- Security monitoring and logging

7.2 Zero Trust Architecture

Principles:

- Never trust, always verify
- Least privilege access
- Micro-segmentation
- Continuous monitoring

Implementation:

```
Device → Authentication → Authorization → Access
          ↓                  ↓
   Identity Verification  Policy Check
```

---

8. SPECIALIZED ARCHITECTURE TYPES

8.1 Time-Sensitive Networking (TSN) Architecture

For real-time applications:

Characteristics:

- Deterministic latency
- Time synchronization (IEEE 1588)
- Traffic shaping
- Redundancy

Use Cases:

- Industrial automation

- Automotive
- Audio/video streaming

8.2 Event-Driven Architecture

Structure:

```
Event Sources (Devices)
        ↓
Event Bus/Broker
        ↓
Event Processors
        ↓
Actions/Responses
```

Characteristics:

- Asynchronous communication
- Loose coupling
- Scalability
- Real-time response

Example:

```
Motion detected → Event → Turn on lights
                       → Send notification
                       → Start recording
```

8.3 Microservices Architecture

For IoT applications:

Structure:

```
Device Management Service
Data Ingestion Service
Analytics Service
Notification Service
User Management Service
```

Characteristics:

- Independently deployable
- Scalable
- Technology diversity
- Fault isolation

8.4 Serverless Architecture

Structure:

```
IoT Device
   ↓
Cloud Function (Lambda)
   ↓
Database/Storage
```

Characteristics:

- Event-triggered
- Auto-scaling
- Pay-per-use
- No server management

Use Cases:

- Data processing
- Automation workflows
- API backends

---

9. ARCHITECTURE SELECTION CRITERIA

Choose Based On:

1. Application Requirements:

- Real-time vs batch processing
- Latency requirements

- Data volume
- Reliability needs

2. Scale:

- Number of devices (100s vs millions)
- Geographic distribution
- Growth projections

3. Connectivity:

- Network availability
- Bandwidth constraints
- Mobility requirements

4. Power:

- Battery-operated vs powered
- Energy harvesting
- Power budget

5. Cost:

- Infrastructure costs
- Operational costs
- Maintenance costs

6. Security:

- Threat model
- Compliance requirements
- Data sensitivity

---

10. PRACTICAL ARCHITECTURE EXAMPLES

Example 1: Smart Home

Architecture Type: Star + Cloud-Centric

Structure:

```
Cloud (Alexa/Google Home)
       ↑
   Internet
       ↑
  WiFi Router (Hub)
  ↙   ↓   ↘
Light  Thermostat  Camera
```

Components:

- WiFi-enabled devices
- Cloud services for AI/voice
- Mobile app interface
- Local automation rules

Example 2: Industrial Monitoring

Architecture Type: Edge + Fog

Structure:

```
Cloud (Analytics, ML)
       ↑
Edge Server (Local Processing)
       ↑
Industrial Gateway
  ↙   ↓   ↘
PLC   Sensors  Actuators
```

Components:

- Industrial protocols (Modbus, OPC UA)
- Real-time processing at edge
- Predictive maintenance
- SCADA integration

Example 3: Smart Agriculture

Architecture Type: Hybrid (Edge + Cloud)

Structure:

```
Cloud (Long-term analytics)
     ↑
  LoRaWAN Gateway
     ↑
LoRa Network
  ↙  ↓  ↘
Soil   Weather  Irrigation
Sensor Station  Controller
```

Components:

- Long-range communication
- Solar-powered sensors
- Local decision making
- Cloud-based insights

Example 4: Connected Vehicle

Architecture Type: Mobile + Edge + Cloud

Structure:

```
Cloud Services
```