

Unified Communication Models

(Life Analogies × Core Mechanisms × Technical Mapping)

This document summarizes the three fundamental communication models that appear in both real life and computer systems.

All digital communication patterns can be mapped back to these three life-originated models.

1. Request–Response Model (Active / Pull-Based)

Life Analogy:

- Sending a letter
- Sending an email
- Going to a bank counter to request a service
- Submitting an application and waiting for approval

Core Essence:

- The receiver **actively requests** information.
- The sender **responds only when asked**.
- One request → one response.
- Stateless, cause–effect chain.
- Pure **Pull** mechanism.

Core Mechanisms:

- Request()
- Response()
- Retry()
- Timeout()
- Polling (repeated requests)
- Long Polling (request held open until event occurs)

Technical Mapping:

- HTTP/1.1
- HTTP/2
- REST APIs
- RPC / gRPC
- GraphQL Query
- Traditional client–server interactions

2. Publish–Subscribe Model (Passive / Push-Based)

Life Analogy:

- Listening to radio broadcasts
- Subscribing to a newspaper or magazine
- Watching TV broadcast
- Receiving alerts from a subscription service

Core Essence:

- The publisher **pushes** events.
- Subscribers **declare interest** beforehand.
- Publisher does not know who receives the message.
- Decoupled, one-to-many.
- Pure **Push + Interest Declaration**.

Core Mechanisms:

- Subscribe(topic or filter)
- Publish(event)
- Unsubscribe()
- Event routing
- Asynchronous callbacks

Technical Mapping:

- Kafka
- MQTT
- RabbitMQ
- Redis Pub/Sub
- Server-Sent Events (SSE)
- Event buses and message brokers

3. Session / Duplex Model (Bidirectional Communication)

Life Analogy:

- A phone call
- Face-to-face conversation
- Video conferencing
- Walkie-talkie (half-duplex variant)

Core Essence:

- Both sides can **actively send messages**.
- A persistent connection exists.
- Requires **state** and **heartbeat** to stay alive.
- Real-time, continuous event flow.

- **Push + Pull + State + Heartbeat** combined.

Core Mechanisms:

- Connect()
- Send()
- Receive(callback)
- Heartbeat(PING/PONG)
- Close()

Technical Mapping:

- WebSocket
- TCP
- QUIC streams
- WebRTC (real-time media)

Summary Table

Communication Model	Life Analogy	Core Essence	Mechanisms	Technical Mapping
Request–Response	Sending letters, emails, bank counter	Pull-based, synchronous, stateless	Request/Response, Polling	HTTP, REST, RPC, gRPC
Publish–Subscribe	Radio, newspaper subscription	Push-based, asynchronous, decoupled	Subscribe/Publish	Kafka, MQTT, RabbitMQ, SSE
Session / Duplex	Phone call, conversation	Bidirectional, stateful, real-time	Connect/Send/Receive/Heartbeat	WebSocket, TCP, QUIC

Final Insight

All digital communication originates from real-life communication patterns:

- **Request–Response = sending a letter**
- **Publish–Subscribe = listening to a broadcast**
- **Session/Duplex = having a phone call**

These three models form the **complete foundation** of all communication systems.