Networks:
The Technology of Connectivity

Ms. T.KAVITHA

Assistant Prof.(CSE)

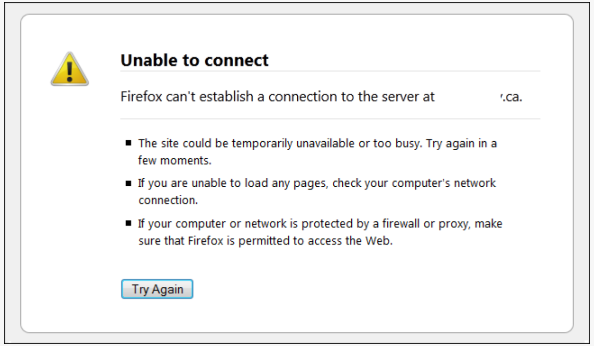
Agenda

- The impact of networks on the UX of an IoT system
- Some typical architectures of IoT systems
- The basics of different types of network used in IoT, both Internet and non-Internet local networks
- Network communication patterns, which govern how data flows between devices
- The role of the Internet service and application programming interfaces (APIs)

Normal Working Conditions (Expectations)

- A good network connection will be available (e.g., broadband or fast mobile data)
- The device will be constantly connected to the network
- The system will feel responsive to the user’s actions
- Where the system can’t fulfil the user’s request immediately, it can provide good feedback on progress
- If we’re accessing a service via more than one device, all those devices will always be in sync and be up to date with any service information

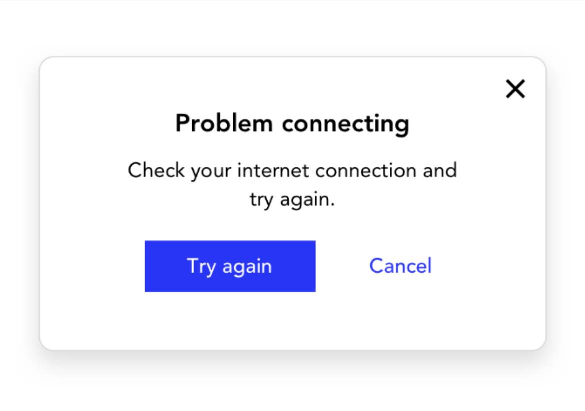
But the Reality is ………



Unable to connect

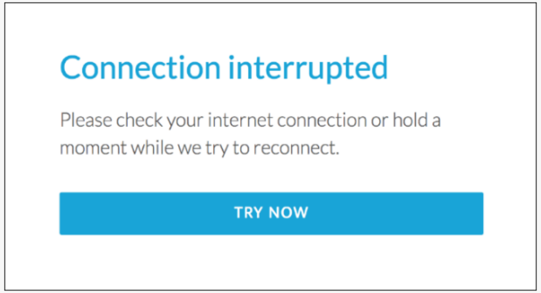
Firefox can't establish a connection to the server at '.ca.

- The site could be temporarily unavailable or too busy. Try again in a few moments.
- If you are unable to load any pages, check your computer's network connection.
- If your computer or network is protected by a firewall or proxy, make sure that Firefox is permitted to access the Web.



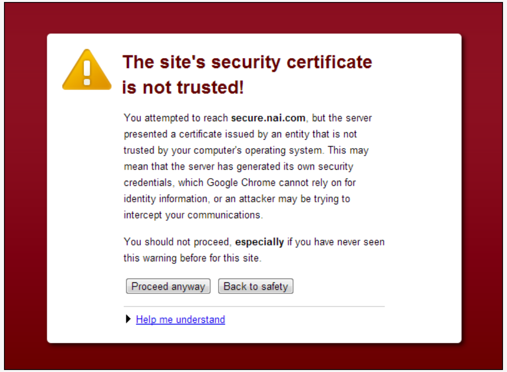
Problem connecting

Check your internet connection and try again.



Connection interrupted

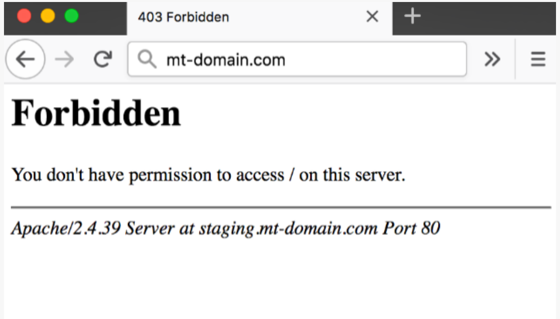
Please check your internet connection or hold a moment while we try to reconnect.



The site's security certificate is not trusted!

You attempted to reach **secure.nai.com**, but the server presented a certificate issued by an entity that is not trusted by your computer's operating system. This may mean that the server has generated its own security credentials, which Google Chrome cannot rely on for identity information, or an attacker may be trying to intercept your communications.

You should not proceed, **especially** if you have never seen this warning before for this site.



Forbidden

You don't have permission to access / on this server.

Apache/2.4.39 Server at staging.mt-domain.com Port 80

Networking Issues That Cause UX Challenges for IoT

- IOT DEVICES OFTEN CONNECT ONLY INTERMITTENTLY
- LATENCY AND RESPONSIVENESS MAY VARY
- NETWORKS ARE NOT 100% RELIABLE

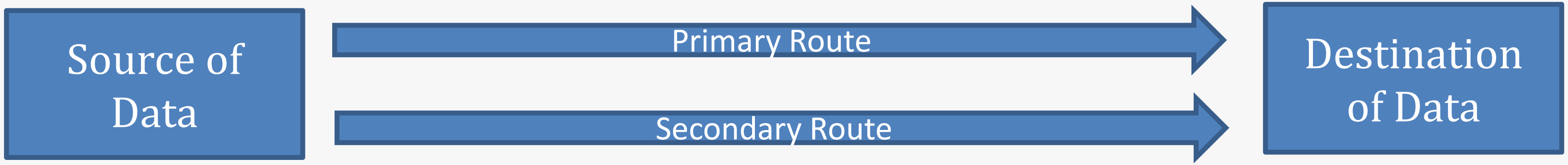
1. IOT DEVICES OFTEN CONNECT ONLY INTERMITTENTLY

- Many IoT devices only connect intermittently, often as a way to conserve power.
- This can result in parts of the service being out of sync with other parts.

2. LATENCY AND RESPONSIVENESS MAY VARY

- controlling latency is a big challenge for UX because it affects *responsiveness*. Instant responses cannot be guaranteed when interactions are routed via the Internet.

https://vimeo.com/87522764



3. NETWORKS ARE NOT 100% RELIABLE

- ***Reliability*: whether or not a message gets through.**
- The Internet is designed to maximize reliability and provide feedback on whether or not a message arrived at its destination.
- But connected devices may use local networks that don’t have the same built-in safeguards.
  - impossible to *guarantee* that a user command won’t get lost.
  - impossible to provide confirmation that the command has been executed.

3. NETWORKS ARE NOT 100% RELIABLE

- All networking is unreliable to some extent.
- This means that edge devices and gateways (where used) may need to *buffer* (temporarily store) data when the network goes down and resync once back online.

OTHER WAYS NETWORKING CAN IMPACT UX

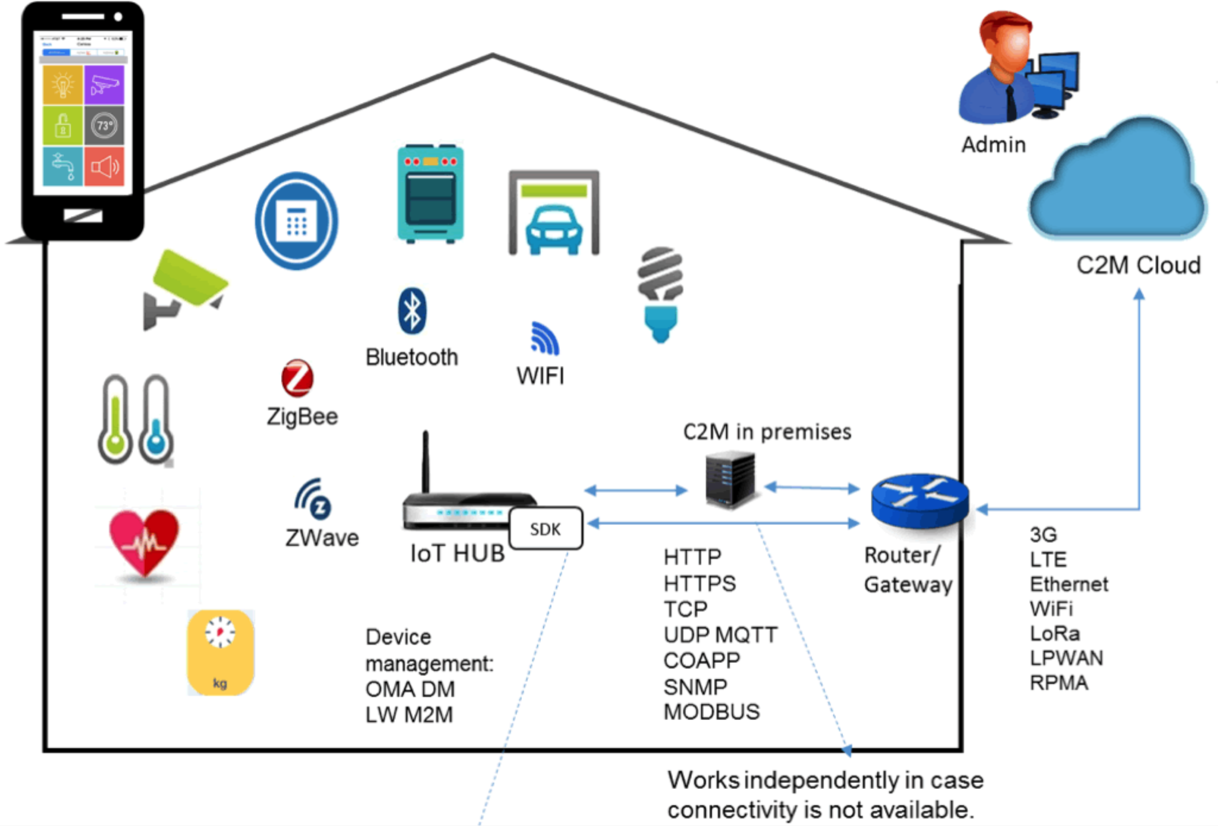
- Ease of installation
- *Interoperability*
- *Addressability on the Internet*

The Architecture of the Internet of Things

System Architecture

- The system architecture defines the path through which one device connects to another.
  - Dedicated Gateway
  - Smartphone as Gateway
  - Direct Internet Connection
  - Device to Device Connections
  - Service to Service Connections

Dedicated Gateway

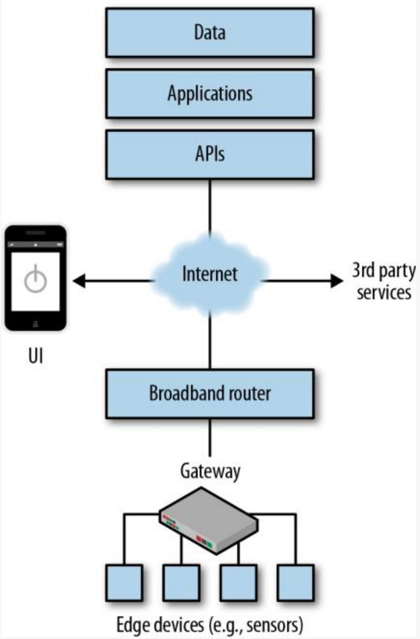


Works independently in case connectivity is not available.

Dedicated Gateway

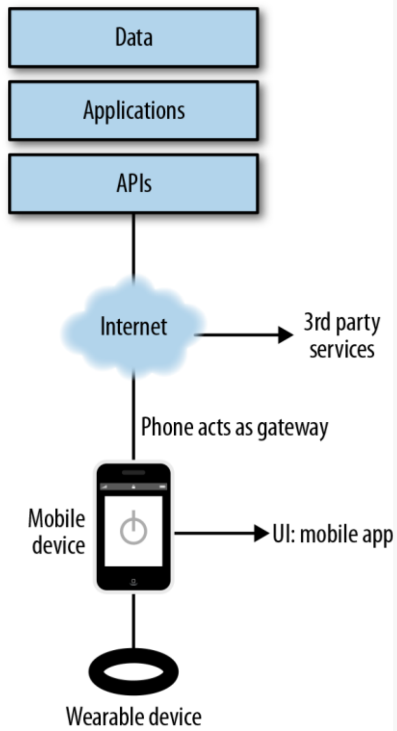
- A *gateway* is a combination of the hardware and software needed to link two different types of network.
- IoT gateways typically translate between Internet Protocol (IP)-based communications and local, non-IP networks used to connect the gateway to the edge devices.
- The gateway is the furthest point to which the proper Internet reaches.
- Consumer IoT gateways are often called *hubs or bridges*.

Dedicated Gateway

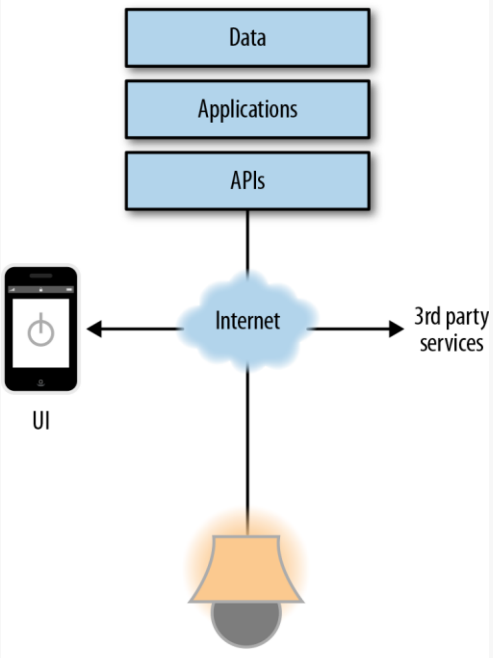


- *Edge devices* such as light switches and security sensors use a low-powered network protocol such as ZigBee or Bluetooth to talk to the gateway.
- The gateway translates between the ZigBee protocol and Internet Protocols
- It can also act as a local brain for the system—for example, running the security system or home automation rules.

Smartphone as Gateway



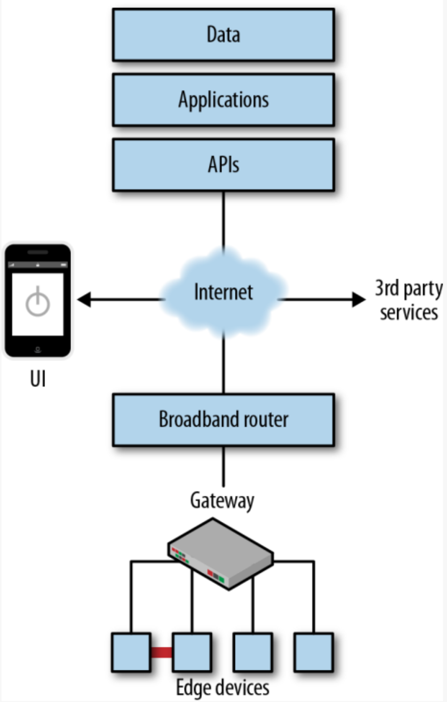
Direct Internet Connection



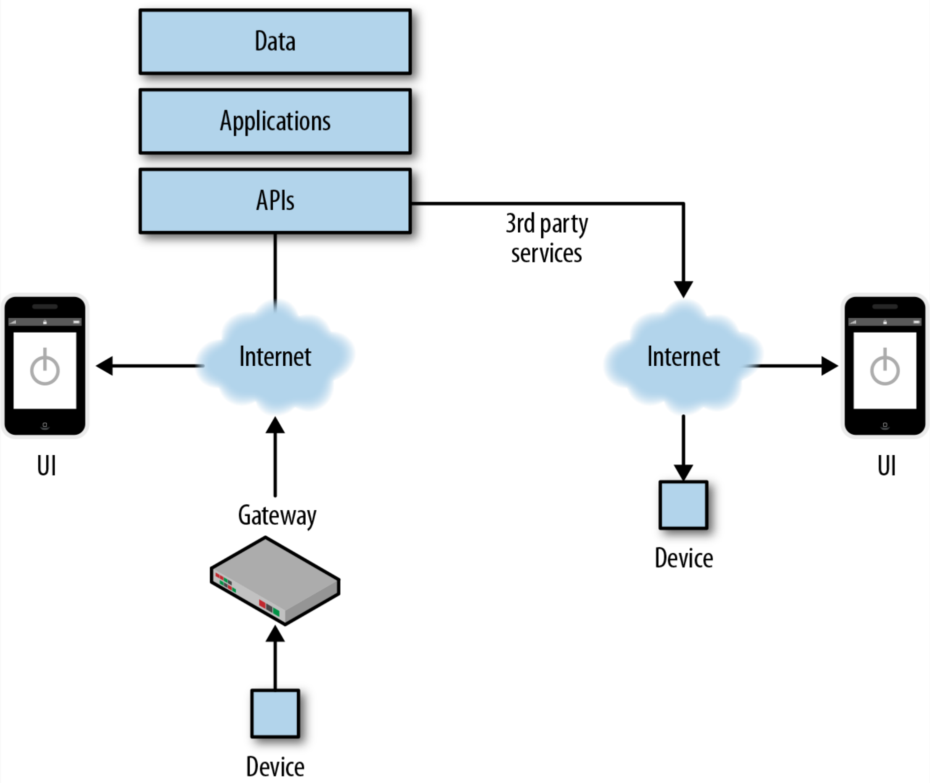
Edge device with direct connection

WeMo
INSIGHT SWITCH
belkin
Turn electronics on/off; manage cost and usage from anywhere.
Allumez/éteignez des appareils électroniques, gérez les coûts et leur utilisation où que vous soyez
Heater
ON / OFF
On now for:
3 h 24 min
Since:
6:21 pm
Estimated monthly: $13.37
Today: $0.44
Ave when on: 856 W
Now: 789 W
we mo
Free App for Android & iOS

DEVICE-TO-DEVICE CONNECTIONS



SERVICE-TO-SERVICE CONNECTIONS



Internet services can connect together over application programming interfaces (APIs). Devices from different manufacturer services can be integrated without the need for a shared gateway.

HOW SYSTEM ARCHITECTURE AFFECTS UX

- If a sensor is directly connected to an *actuator.*
  - *Quick and robust*
  - *Inflexible*
- If the path from the sensor to the actuator goes through a gateway.
  - Cause latency
  - More flexible

HOW SYSTEM ARCHITECTURE AFFECTS UX

- If the path goes through the Internet,
  - *Unreliable and latency.*
  - *More flexible*

Advantages of Gateways

- Edge devices can be kept simple.
- Easier setup and maintenance.
- The system can function when the internet is unavailable.
- Lower Latency
- Enabling interoperability

Disadvantages of Gateways

- Time consuming and costly to develop
  - Service provider creates their own proprietary gateway to meet their own needs.
- A potentially confusing point of failure

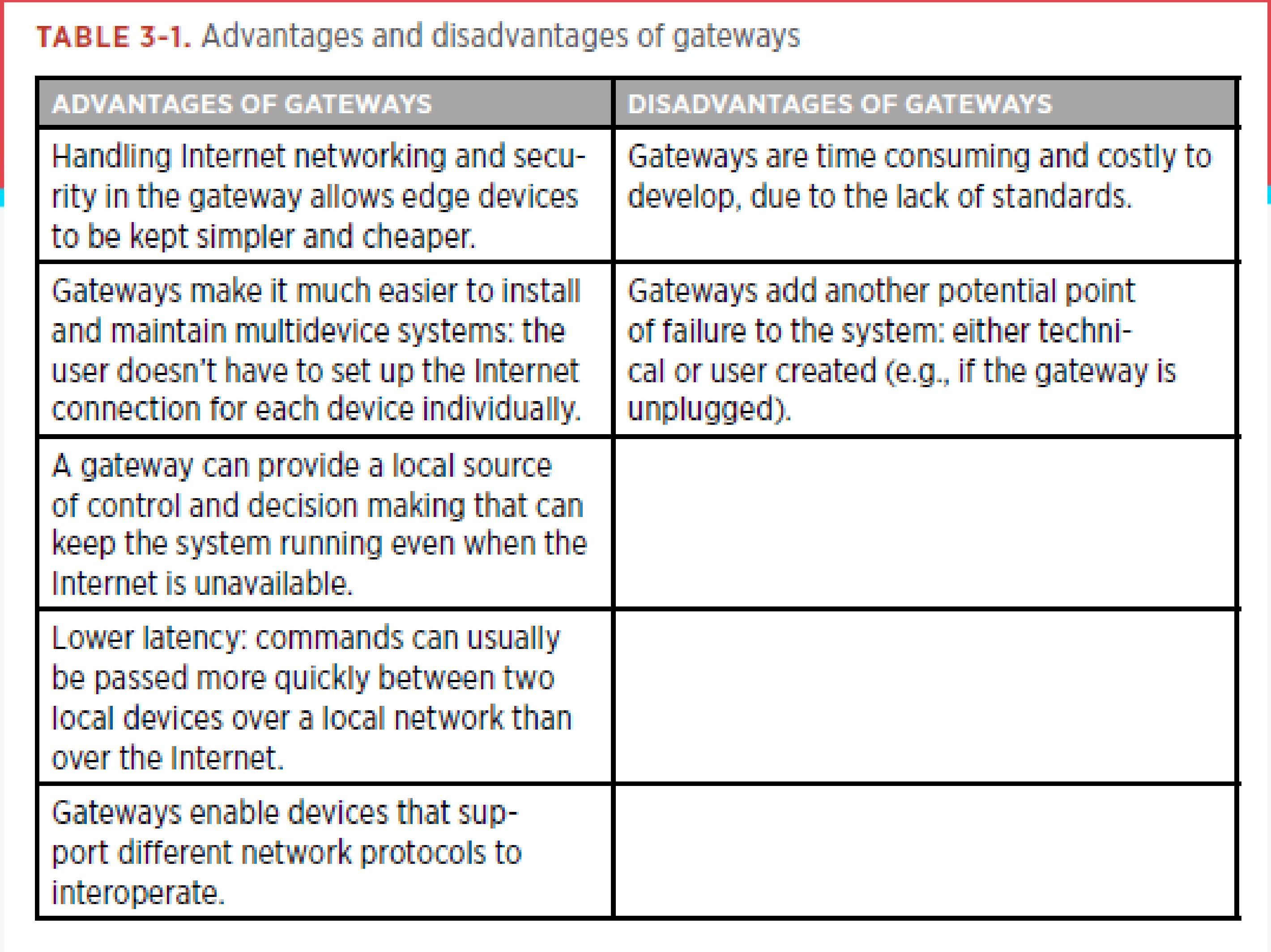


TABLE 3-1. Advantages and disadvantages of gateways

ADVANTAGES OF GATEWAYS	DISADVANTAGES OF GATEWAYS
Handling Internet networking and security in the gateway allows edge devices to be kept simpler and cheaper.	Gateways are time consuming and costly to develop, due to the lack of standards.
Gateways make it much easier to install and maintain multidevice systems: the user doesn't have to set up the Internet connection for each device individually.	Gateways add another potential point of failure to the system: either technical or user created (e.g., if the gateway is unplugged).
A gateway can provide a local source of control and decision making that can keep the system running even when the Internet is unavailable.	
Lower latency: commands can usually be passed more quickly between two local devices over a local network than over the Internet.	
Gateways enable devices that support different network protocols to interoperate.	

HOW INTERNET NETWORKING WORKS

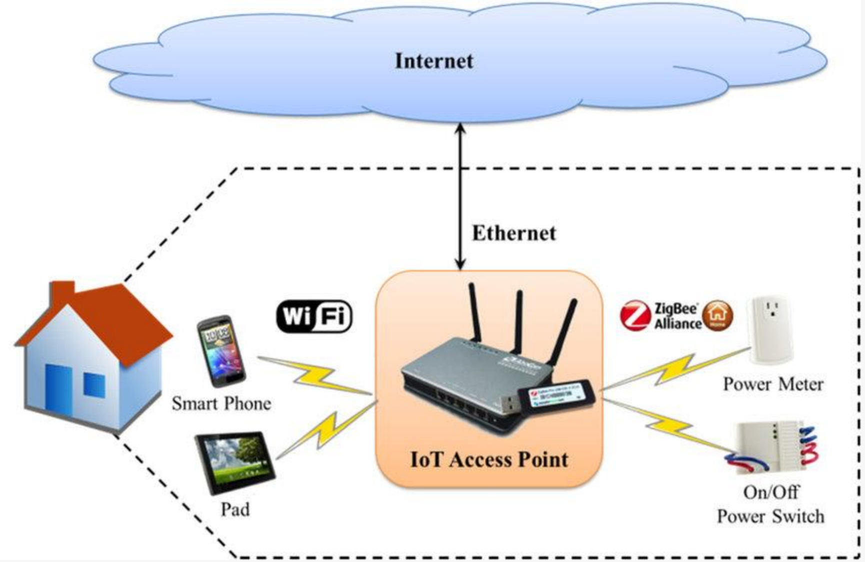
This clip is for non-commercial use only

TYPES
OF INTERNET
NETWORK

Ethernet

- The user won’t have a spare port
- Limits them to keeping the device near the router

Wi Fi



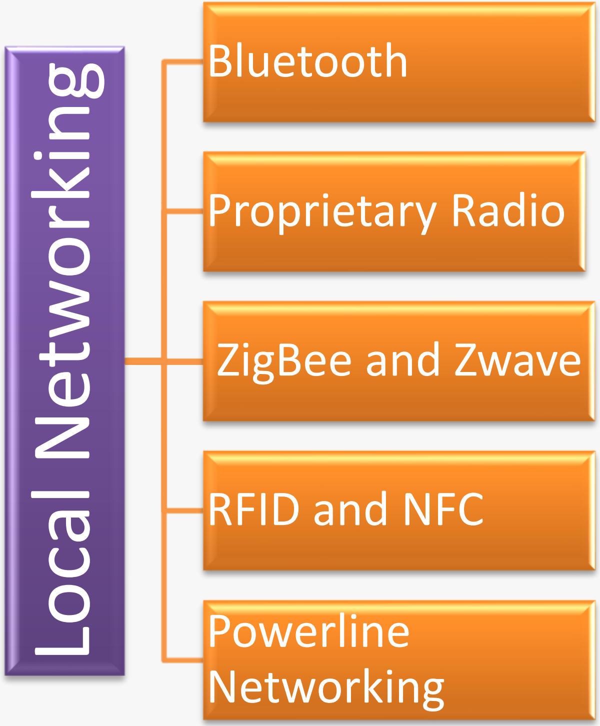
- If a device has reliable mains power and tends to stay within the home, it's generally easiest to connect it via Wi Fi.
- User must be in the range.

Cellular Data

- Cellular data uses the same data networks to which your mobile phone connects: usually GPRS or 3G/4G.
- uses a lot of power and can be expensive, requiring an ongoing subscription to a network provider.

Everything you need to know about 5G

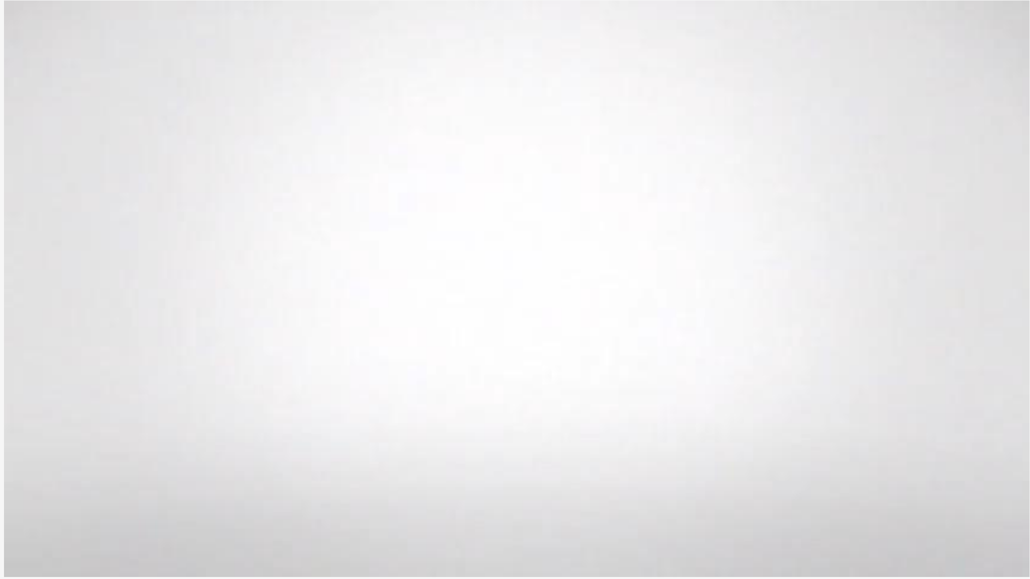
TYPES OF LOCAL NETWORKING



Local Networking

- Bluetooth
- Proprietary Radio
- ZigBee and Zwave
- RFID and NFC
- Powerline Networking

Bluetooth



- Bluetooth is useful when the device is primarily used in range of a phone and in the presence of the user.
- This works well for wearables and detecting the presence of the owner
- It does not yet support IP networking in the real world but the standard allows for it.

Bluetooth Low Energy

- Bluetooth LE (low energy; also known as Bluetooth 4 or Smart) is especially suited to connected devices with power constraints.
- Bluetooth LE allows for devices to connect and transmit data at regular intervals, or when they have something to share. This uses less power and preserves battery life.

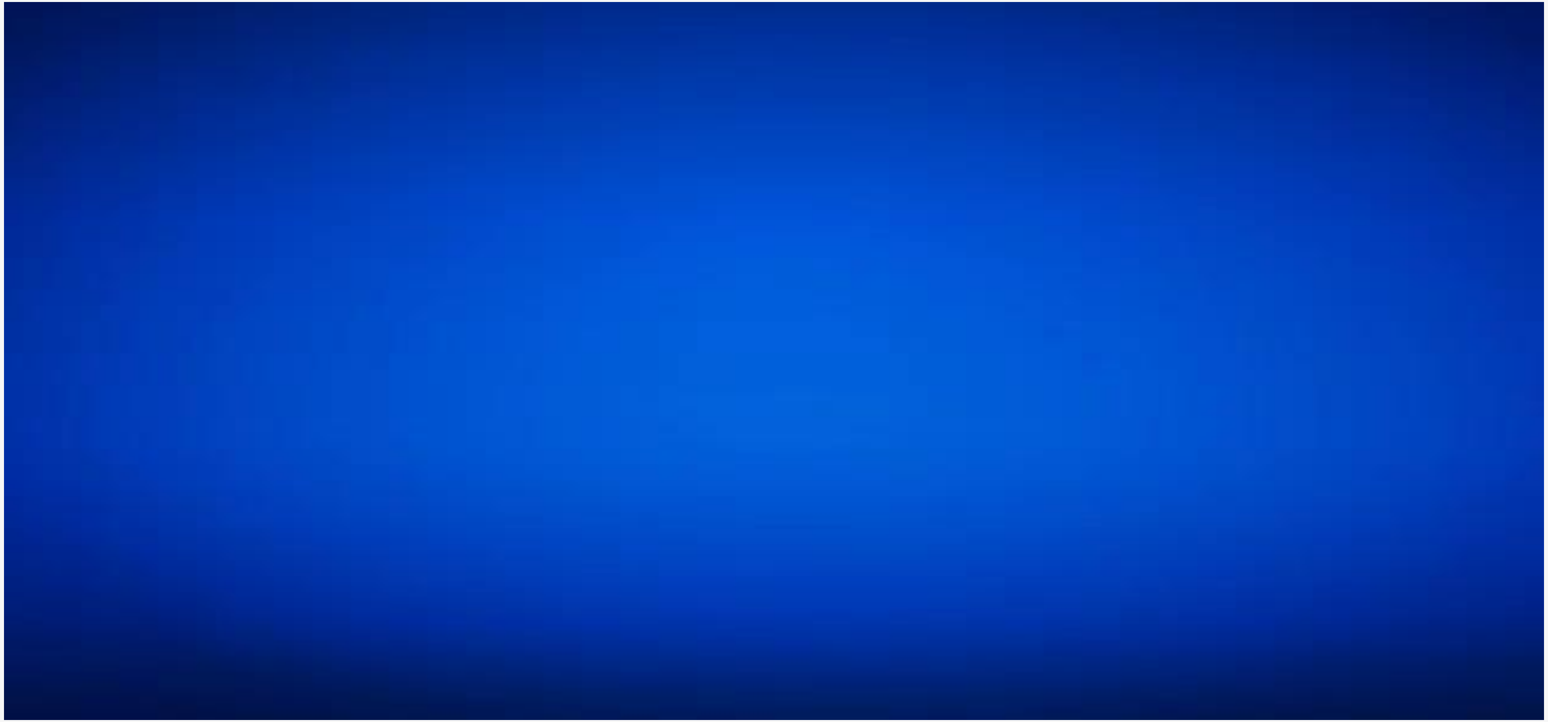
Proprietary radio

- custom connections entirely controlled by the manufacturer.
- They may be reliable and cheap

ZigBee

- ZigBee and ZWave are low-powered radio networks, suitable for battery-powered devices and designed for use in home automation systems.
- Unlike Wi-Fi, the range of a ZigBee or ZWave network can be extended through *mesh networking*.
- This allows any of the devices on the network that run on mains power to act as *repeaters*, extending the range of the network.

Difference Between WiFi and ZigBee



RFID and NFC

Powerline networking

- Where Ethernet is unavailable and wireless networking too unreliable, it is possible to run data over pre-existing electrical power lines using the Home Plug standard.
- Data is transmitted over the electrical wiring.

IP to the EDGE

- Delivering Internet communications (over IP) all the way to the edge device, across even low-powered networks, is a growing area of interest
- To provide many more edge devices with a unique identity on the Internet so they are able to contact (and be contacted by) any service and do anything they want.

IP to the EDGE

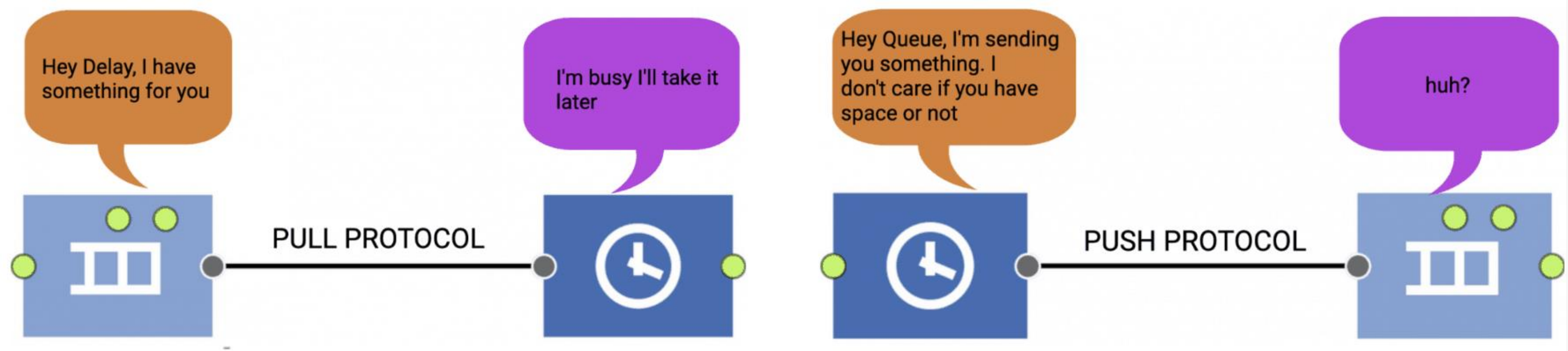
- Far greater degree of flexibility of functionality.
- A device can identify itself uniquely.
- The incoming standard IPv6 provides for the equivalent of 10 IP addresses for every atom on earth.
- 6LOWPAN is an IETF working group proposing solutions for IPv6 for low-power wireless personal area networks.

Network Communication Patterns

Communication Pattern

- Communication
  - how devices send and receive data to the network
  - can be handled in different ways.
- This is largely governed by protocols in the *application layer*.
  - Understanding the way data flows around the devices
  - allows you to understand how responsive

PUSH, PULL and Polling


Hey Delay, I have something for you
I'm busy I'll take it later
PULL PROTOCOL
Hey Queue, I'm sending you something. I don't care if you have space or not
huh?
PUSH PROTOCOL

Pull Protocol

- The user connects to the server and requests for the information what he needs.
- When the request of the user is fulfilled, he shuts down the connection.
- This kind of transformation of information over the internet is based on pull protocol, and information delivered via this protocol is called pull notification or pull messages.

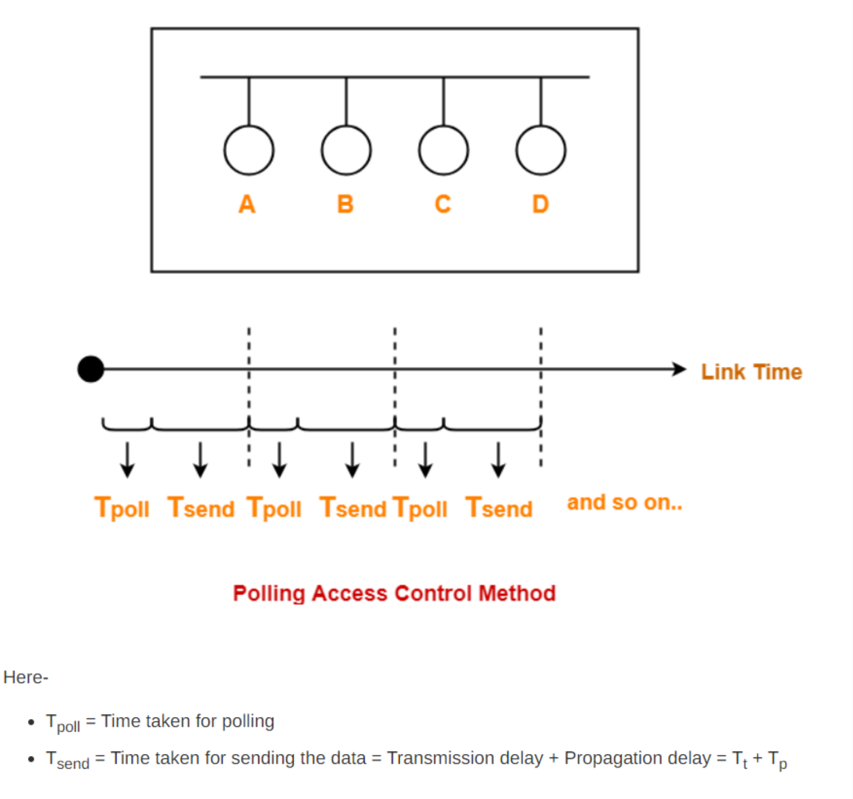
Push Protocol

- If we have subscribed a channel so, next time whenever they will post something on their channel we get notified by the message, that message is called push notification.

Polling Protocol

In this access control method,

- A polling is conducted in which all the stations willing to send data participates.
- The polling algorithm chooses one of the stations to send the data.
- The chosen station sends the data to the destination.
- After the chosen station has sent the data, the cycle repeats.



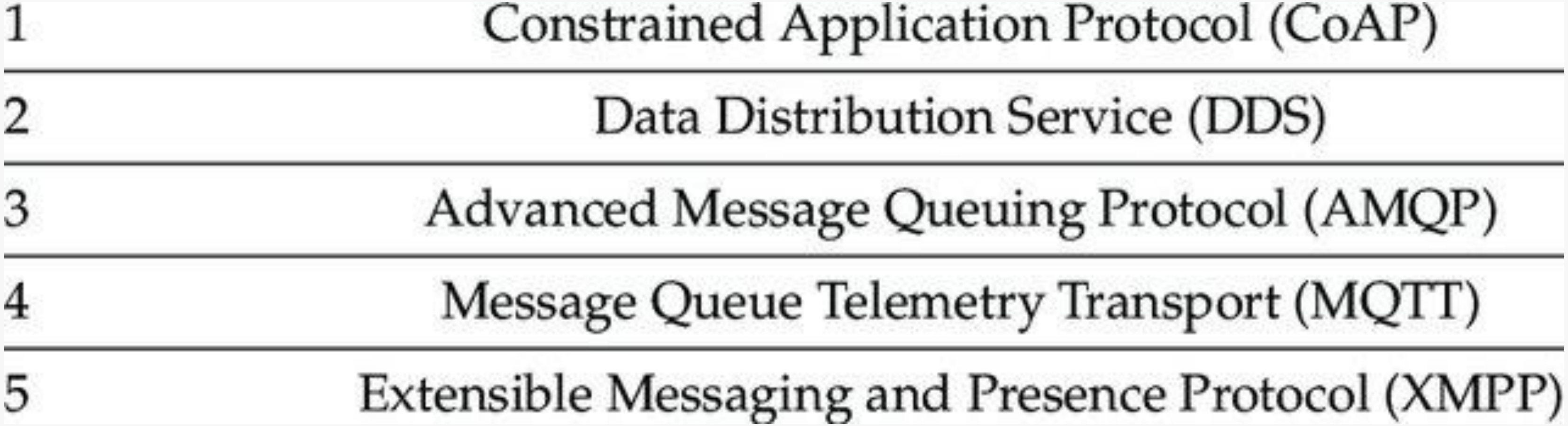
Polling Access Control Method

Here-

- T_{poll} = Time taken for polling
- T_{send} = Time taken for sending the data = Transmission delay + Propagation delay = T_t + T_p

IOT Application Protocols

IoT application Layer Protocol

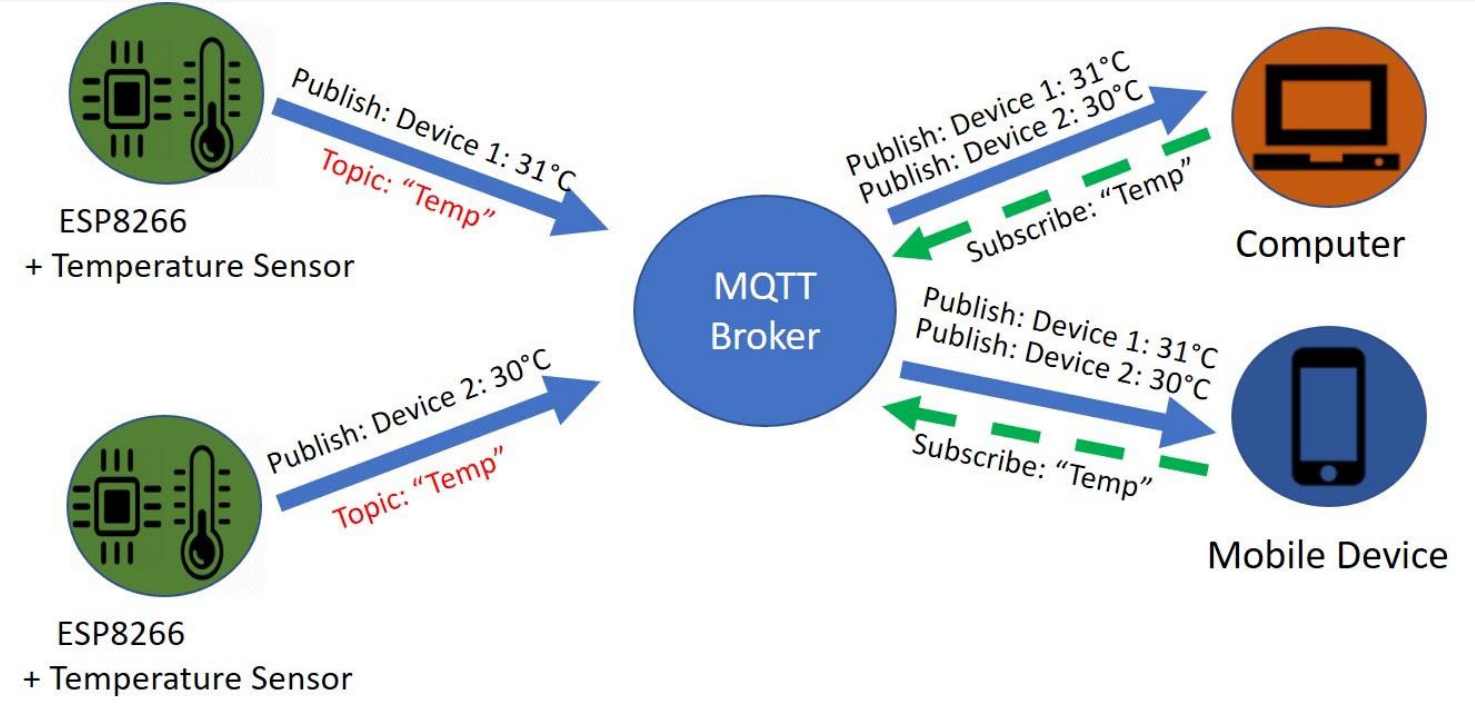


1	Constrained Application Protocol (CoAP)
2	Data Distribution Service (DDS)
3	Advanced Message Queuing Protocol (AMQP)
4	Message Queue Telemetry Transport (MQTT)
5	Extensible Messaging and Presence Protocol (XMPP)

HTTP

- HTTP is a reliable way to pass information around a small number of devices
- It is not designed for push or streaming and so usually requires polling.
- It’s also not the most streamlined method of passing data around lots of devices, especially ones that have power and computation constraints.
- It requires a direct connection to be established between any devices that need to share data
- Messages take up a lot of bytes.

MQTT



https://qr.ae/pNQKfw

Internet Service

THE ROLE OF THE INTERNET SERVICE

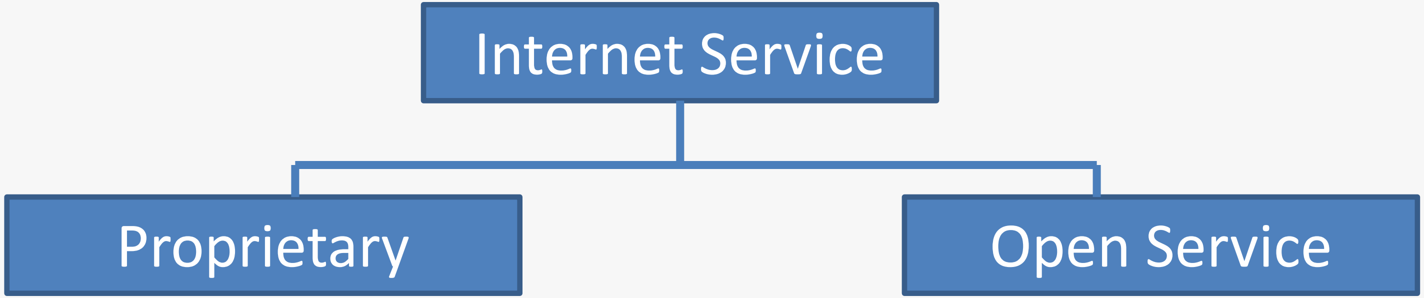
- Remote centralized Internet service - collect, process, and distribute data and instructions.
- The Internet service would generally include some way of storing and handling:
  - System data (e.g., sensor data or the current state of the devices)
  - Applications that run on top (e.g., health monitoring or lighting)

THE ROLE OF THE INTERNET SERVICE

- The service makes data and system commands available to mobile or web apps via application programming interfaces (APIs).
- Service providers make APIs available to third-party developers to enable other services to integrate with the system.

Processing in Cloud Good vs Bad?

- Maintain one centralized view of the system.
- But the system won’t work when devices lose connectivity and can’t access the cloud.
- Real time applications suffer when there is an outage.
- Emergency Baby Monitors – Local monitoring needed.



Internet Service

Proprietary

Open Service

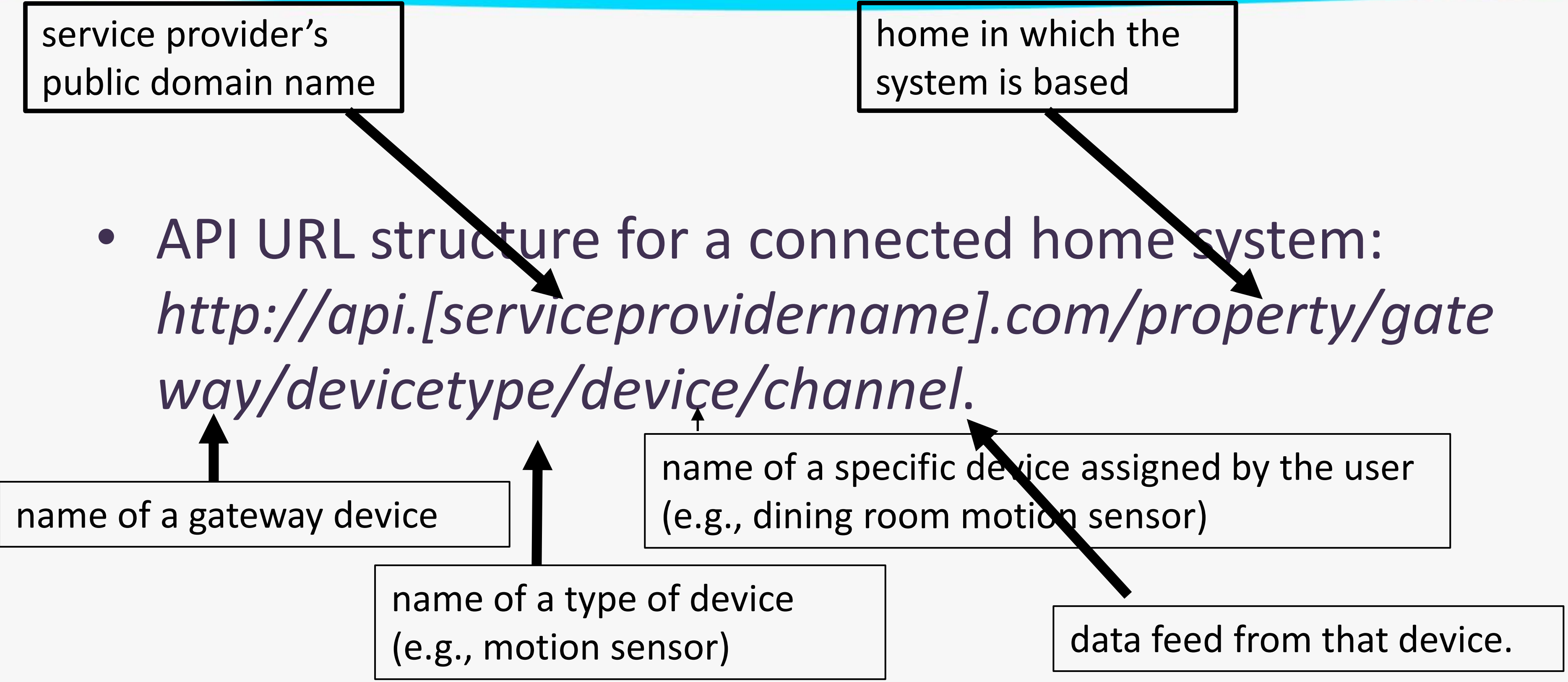
- A platform (in this context) is a type of software framework that can be used to build multiple remote service applications.
- Thingspeak, Kaa,[12] Xively,[13] and Thingworx[14] are examples of platforms that developers can use to create IoT services.

Application Programming Interfaces

APIs

- A set of functions and procedures allowing the creation of applications that access the features or data of an operating system, application, or other service.
- An *application programming interface* (*API*) is an interface for developers.
- It provides hooks for them to build things out of your system.
- This might be your own frontend web or mobile apps or third-party services that interact with your system.
- APIs are an essential part of creating ecosystems of interoperable devices and services.

How do web APIs work?



service provider’s public domain name

home in which the system is based

- API URL structure for a connected home system: *http://api.[serviceprovidername].com/property/gateway/devicetype/device/channel*.

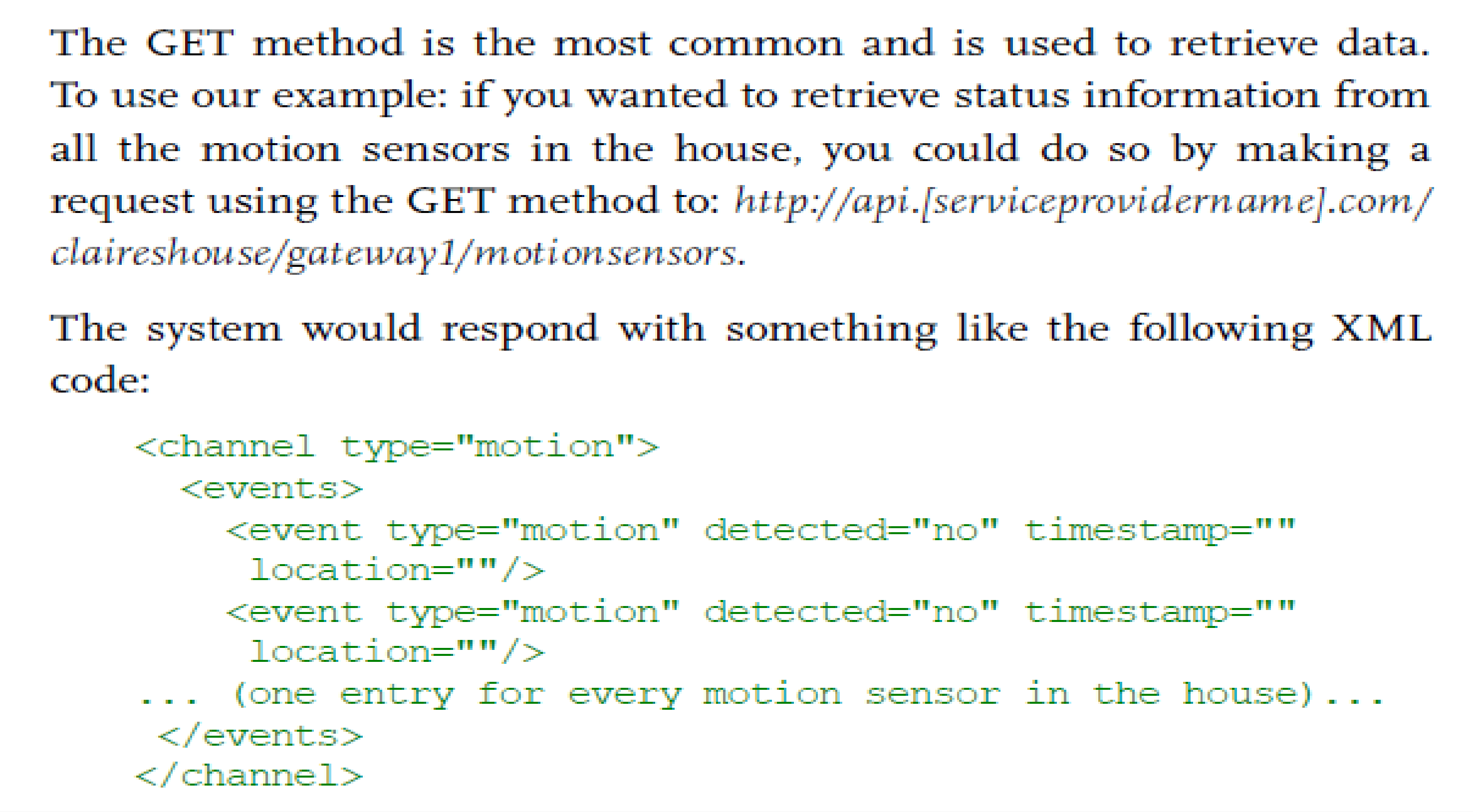
name of a gateway device

name of a type of device (e.g., motion sensor)

name of a specific device assigned by the user (e.g., dining room motion sensor)

data feed from that device.

Web API HTTP Method-GET

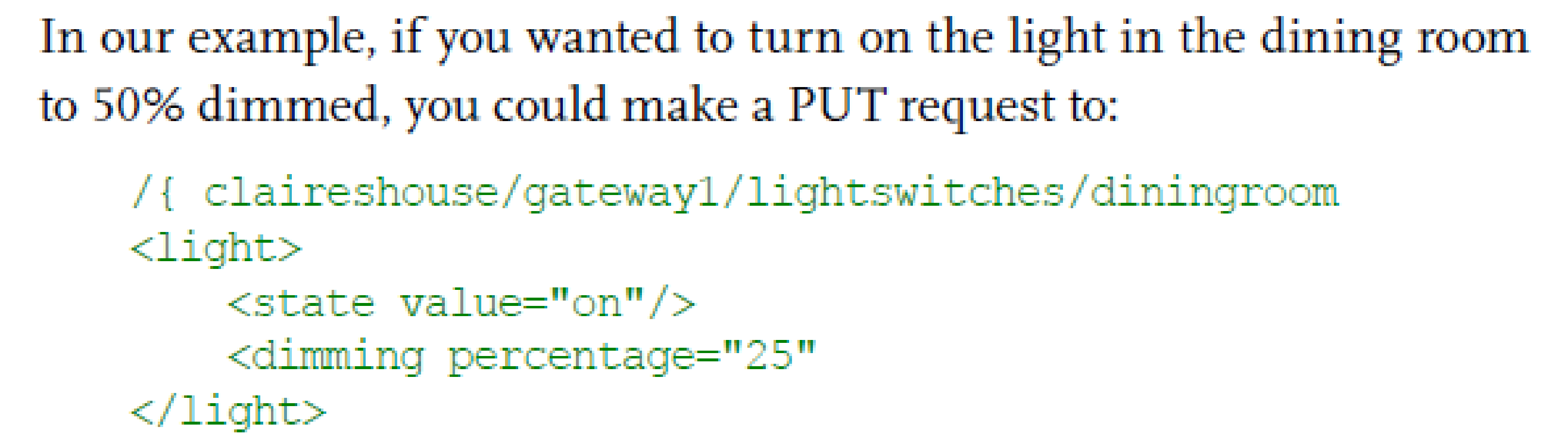


The GET method is the most common and is used to retrieve data. To use our example: if you wanted to retrieve status information from all the motion sensors in the house, you could do so by making a request using the GET method to: *http://api.[serviceprovidername].com/claireshouse/gateway1/motionsensors*.

The system would respond with something like the following XML code:

```
<channel type="motion">
  <events>
    <event type="motion" detected="no" timestamp=""
     location=""/>
    <event type="motion" detected="no" timestamp=""
     location=""/>
... (one entry for every motion sensor in the house)...
 </events>
</channel>
```

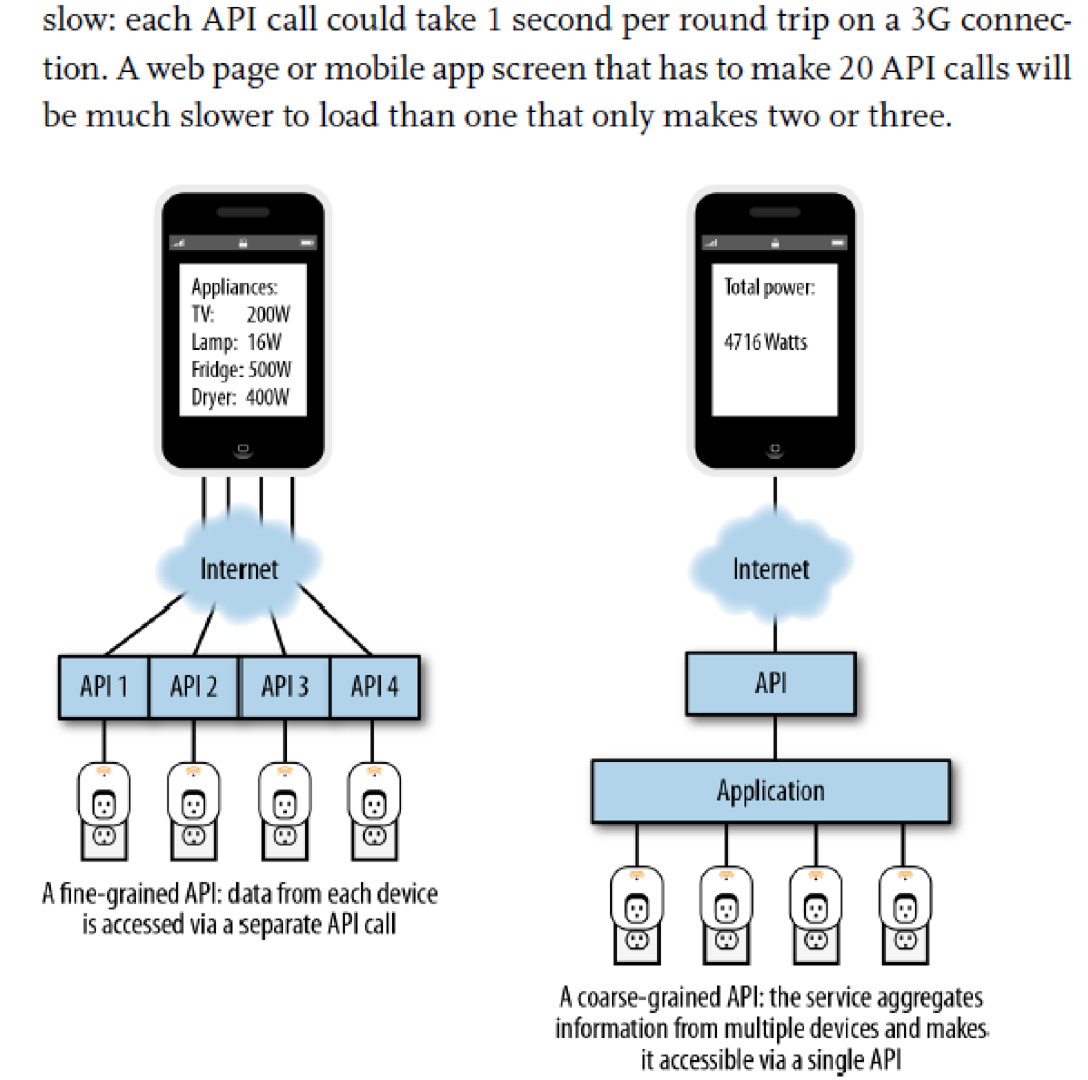
Web API HTTP Method - PUT



In our example, if you wanted to turn on the light in the dining room to 50% dimmed, you could make a PUT request to:

```
/{ claireshouse/gateway1/lightswitches/diningroom
<light>
    <state value="on"/>
    <dimming percentage="25"
</light>
```

Issues – SLOW for Multiple API Calls



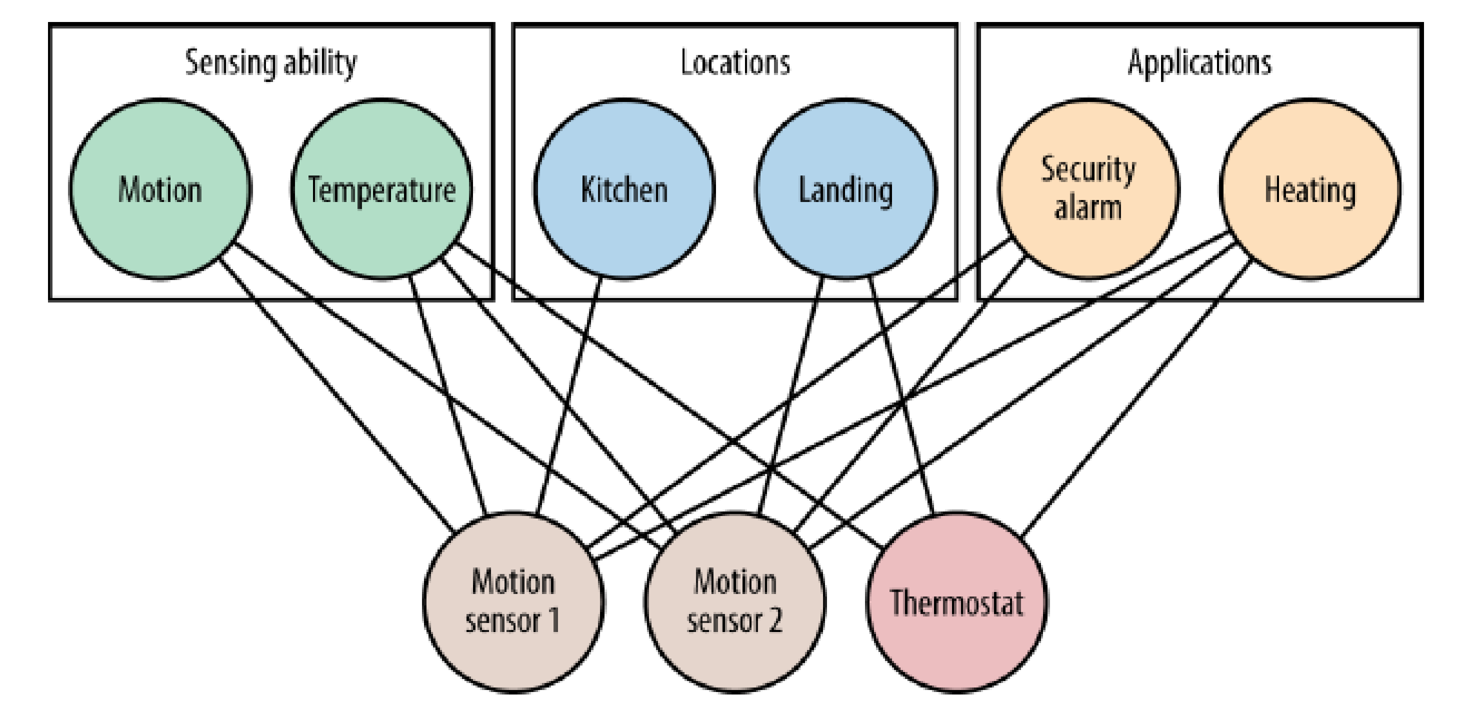
slow: each API call could take 1 second per round trip on a 3G connection. A web page or mobile app screen that has to make 20 API calls will be much slower to load than one that only makes two or three.

A fine-grained API: data from each device is accessed via a separate API call

A coarse-grained API: the service aggregates information from multiple devices and makes it accessible via a single API

Task-oriented APIs make it easier to support complex functions across multiple devices.

Have a well defined structure



Third-party APIs.

- You may be relying on a third-party API, such as Xively or Twitter.
- If so, you’re bound by their terms and you need to check how you can use them.
- Providers often impose limitations on the frequency of calls you can make, or the number of API calls per day to avoid overloading their own servers.
- They may also charge for API calls, as the weather service forecast.io does.

Thank You