

Web of Things

Kiev Gama

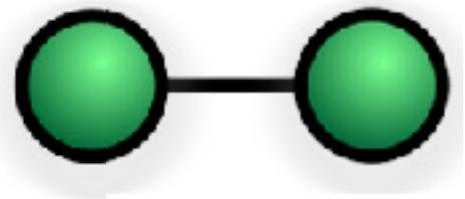
kiev.gama@gmail.com



@kievgama

slideshare.net/kievgama

Topologias de Rede na IoT

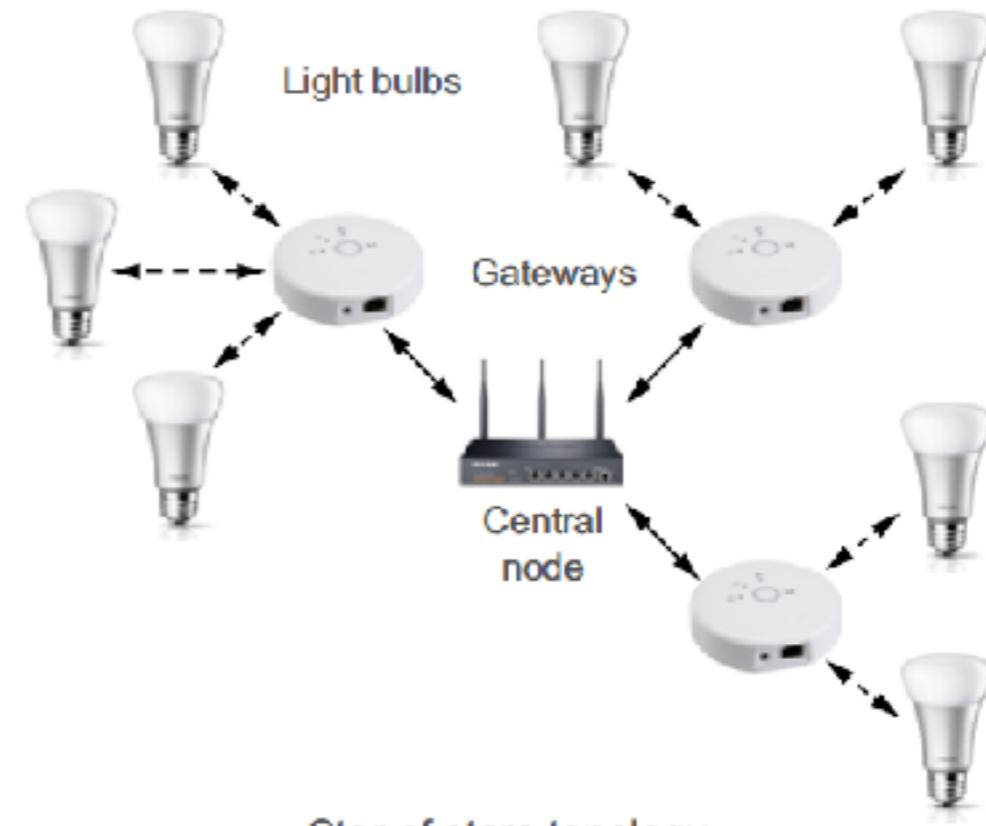


Ponto a ponto

conexão direta entre dispositivos

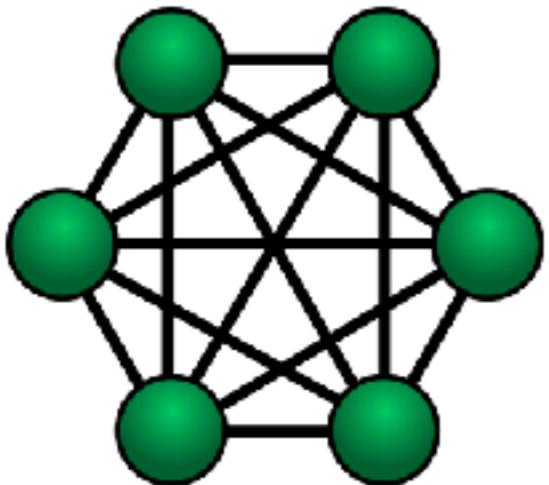
**cenário típico: pareamento com
wearable device (Bluetooth)**

Estrela

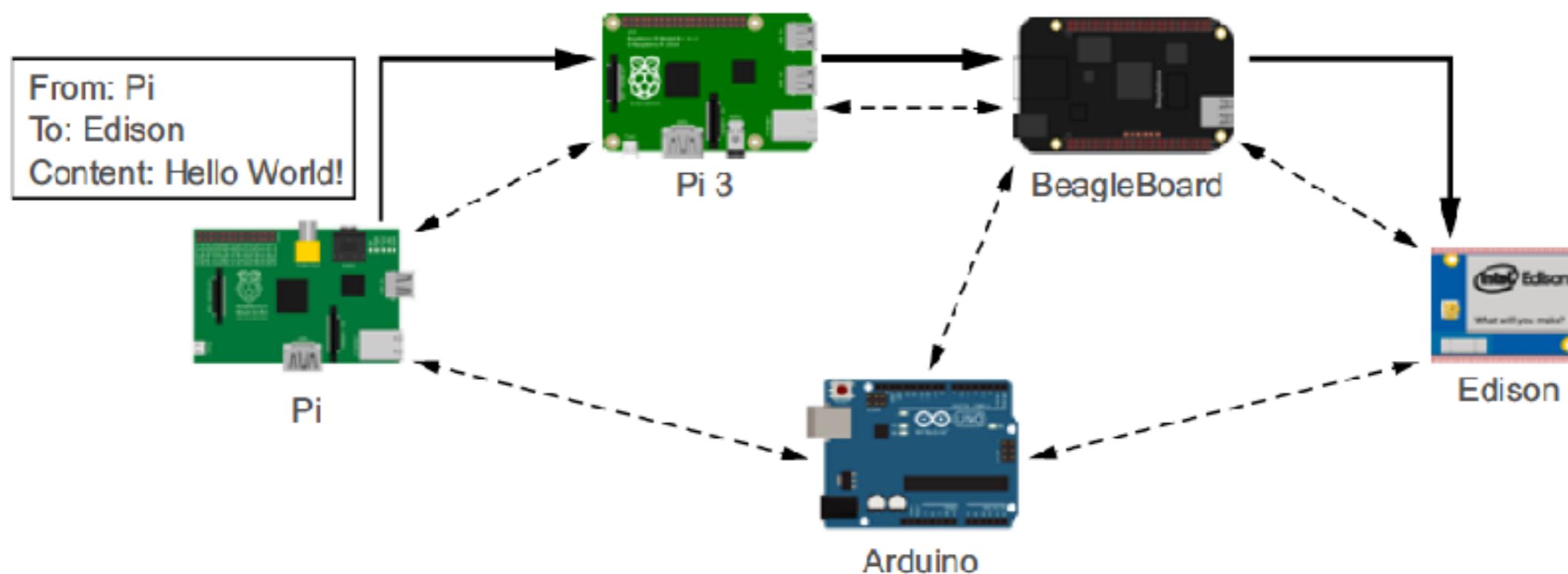


Os nós da rede se comunicam com um nó central

Single point of failure



Mesh



Nós intermediários (relays) encaminham as mensagens

A própria Internet funciona assim

Modelos de Comunicação

RFC 7452

Architectural Considerations in Smart Object Networking

Status: Informational

<https://www.rfc-editor.org/rfc/rfc7452.txt>



Internet Architecture Board (IAB)
Request for Comments: 7452
Category: Informational
ISSN: 2070-1721

H. Tschofenig
ARM Ltd.
J. Arkko
D. Thaler
D. McPherson
March 2015

Architectural Considerations in Smart Object Networking

Abstract

The term "Internet of Things" (IoT) denotes a trend where a large number of embedded devices employ communication services offered by Internet protocols. Many of these devices, often called "smart objects", are not directly operated by humans but exist as components in buildings or vehicles, or are spread out in the environment. Following the theme "Everything that can be connected will be connected", engineers and researchers designing smart object networks need to decide how to achieve this in practice.

This document offers guidance to engineers designing Internet-connected smart objects.

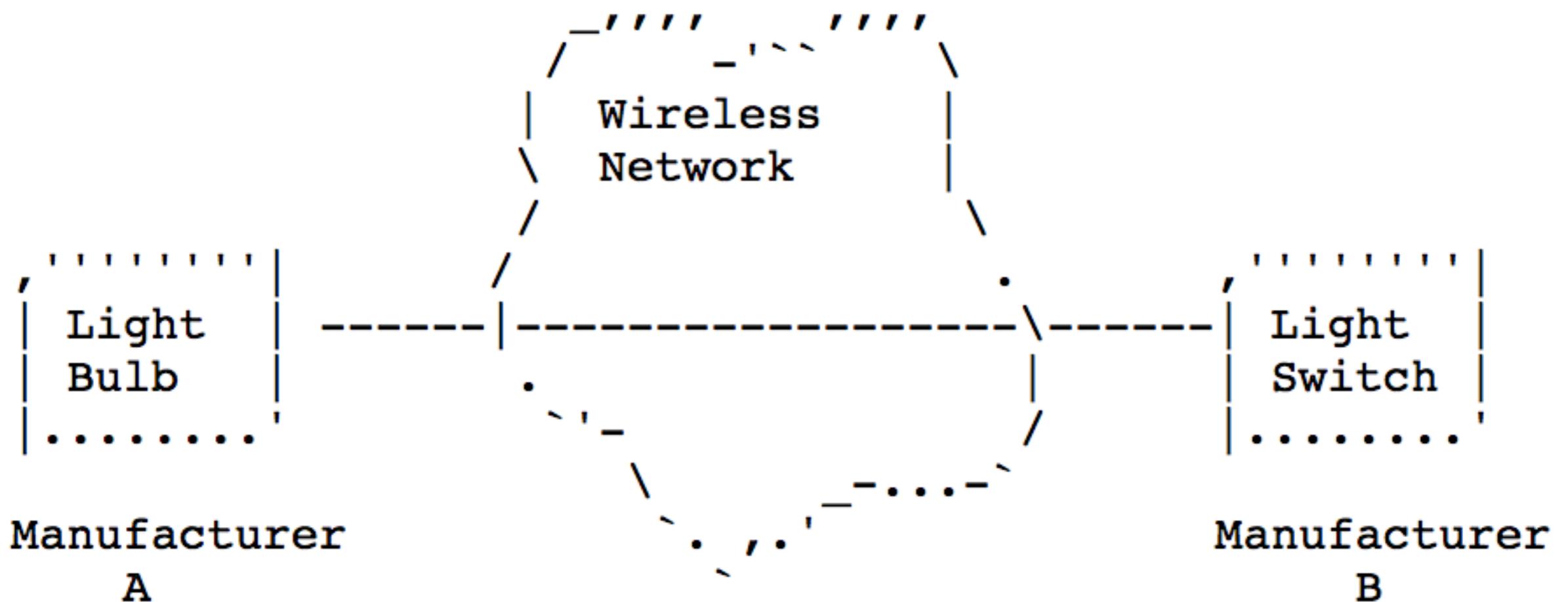
Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

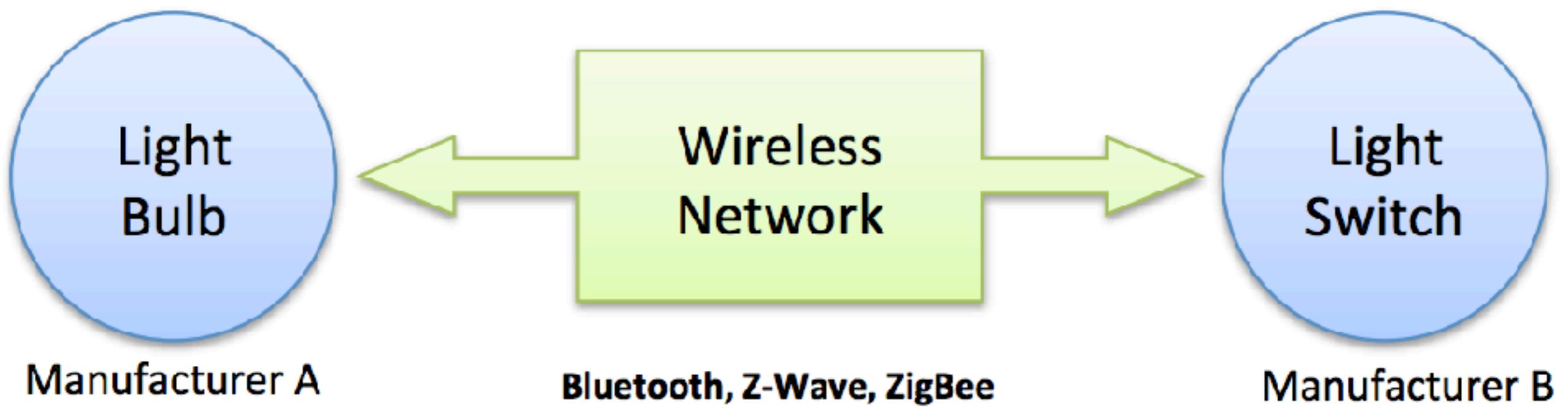
This document is a product of the Internet Architecture Board (IAB) and represents information that the IAB has deemed valuable to provide for permanent record. It represents the consensus of the Internet Architecture Board (IAB). Documents approved for publication by the IAB are not a candidate for any level of Internet Standard; see Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at
<http://www.rfc-editor.org/info/rfc7452>.

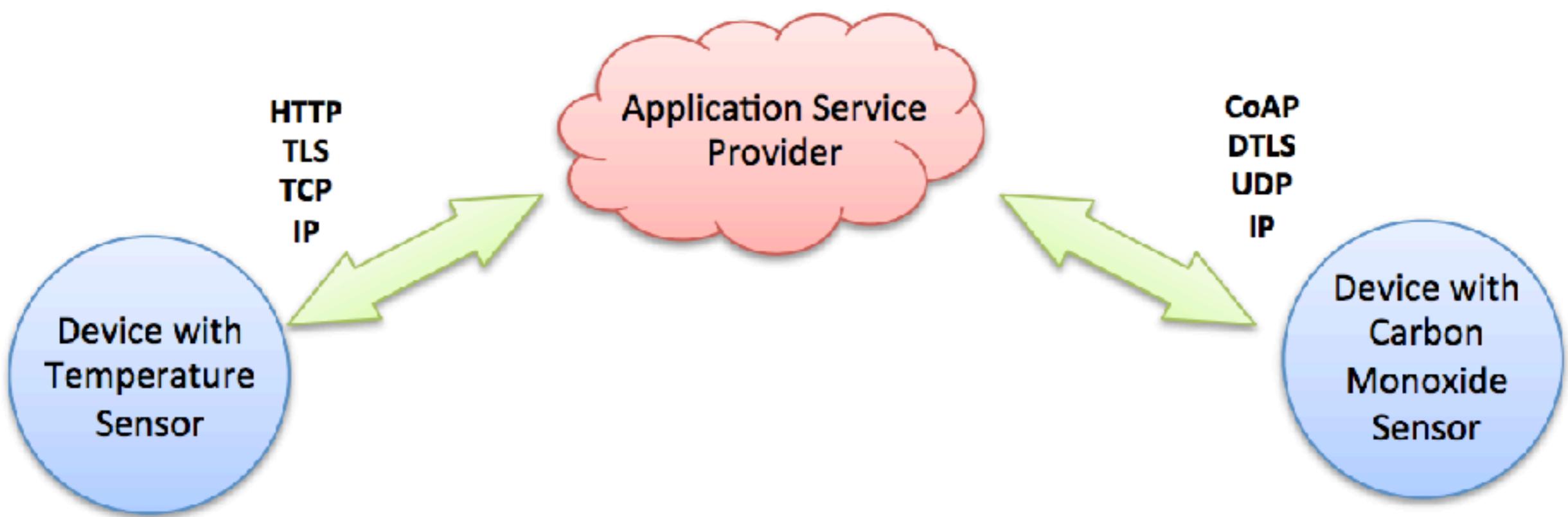
Device-to-device communication Pattern



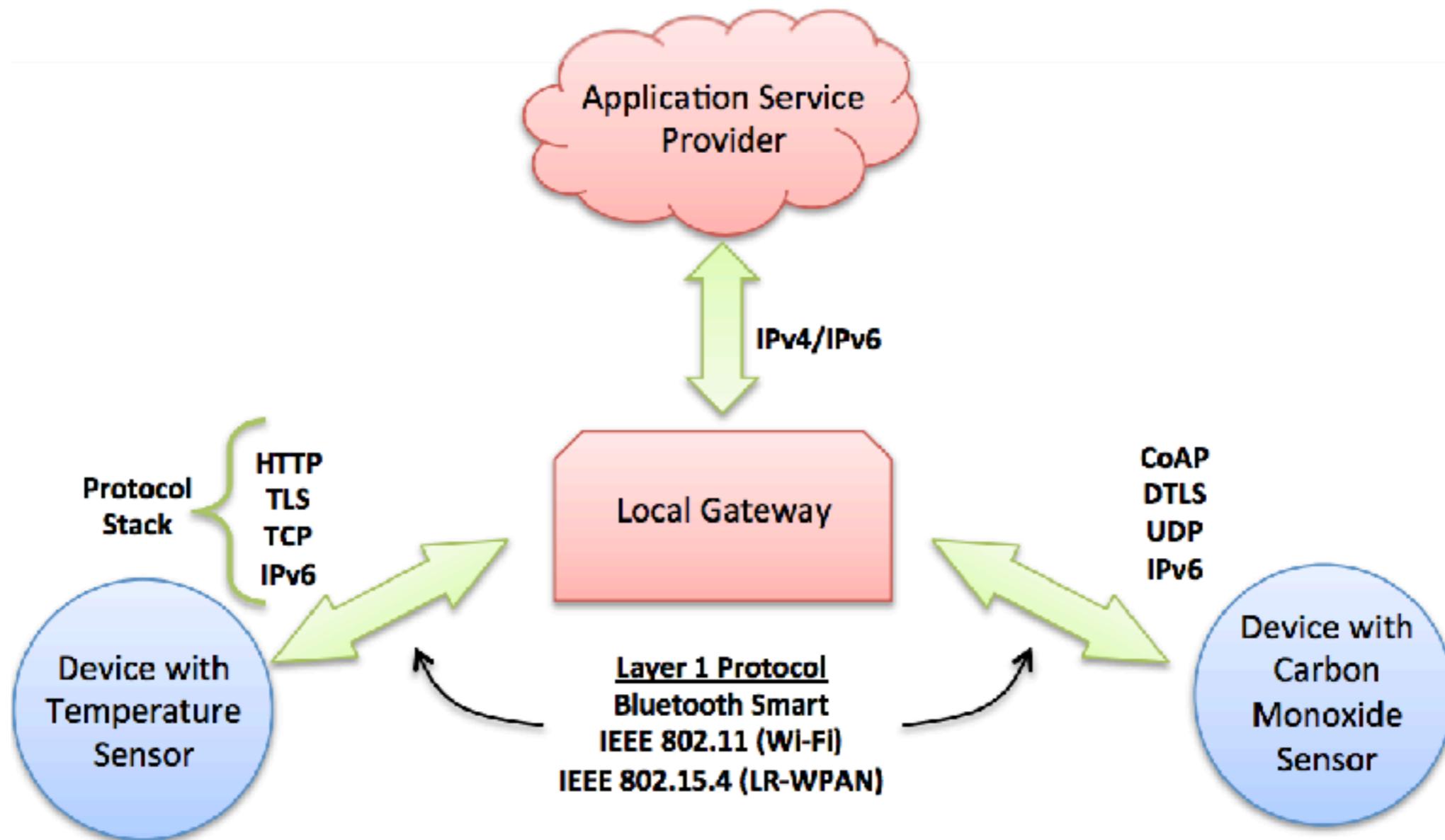
Device-to-Device



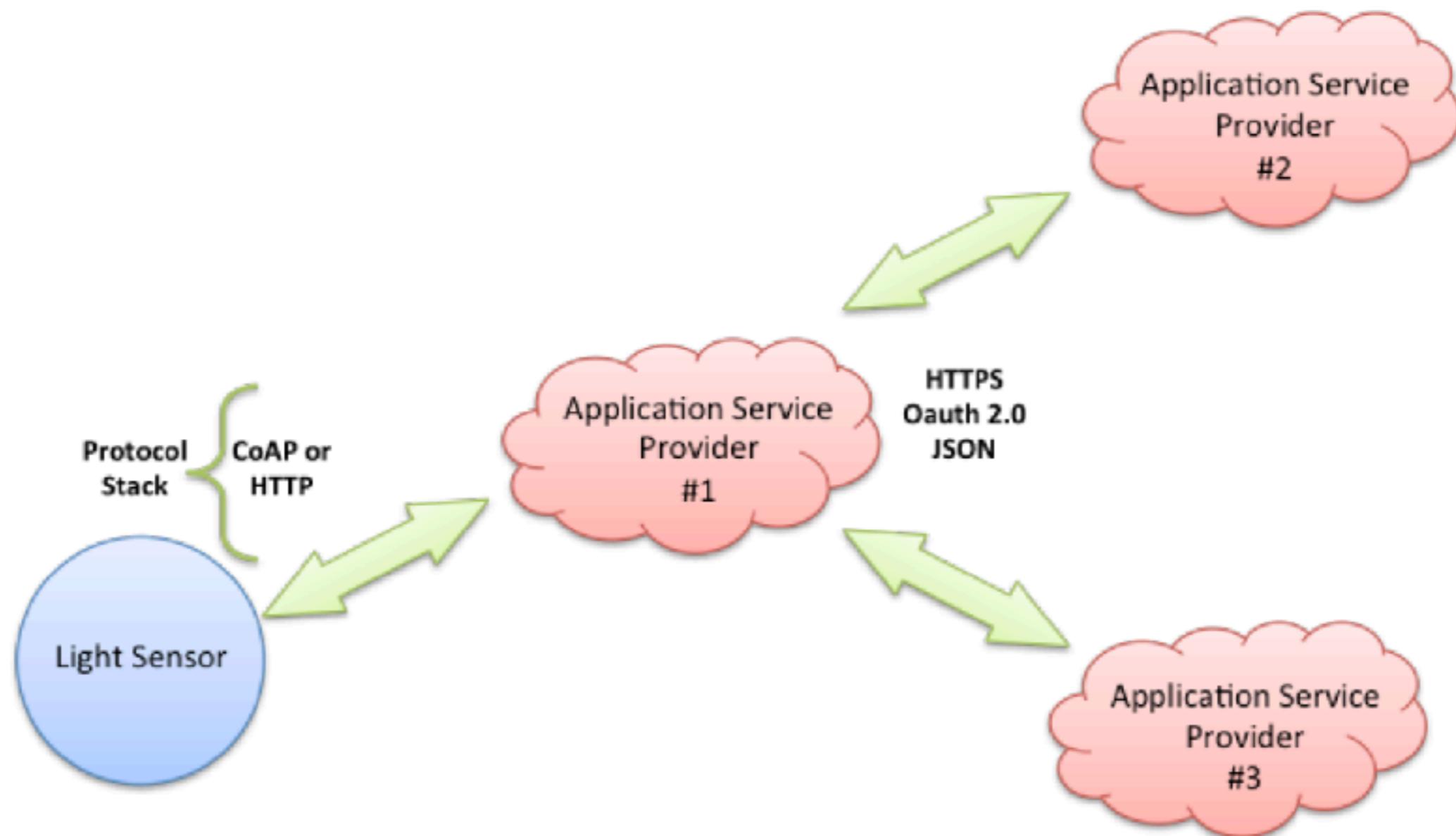
Device-to-Cloud



Device-to-Gateway



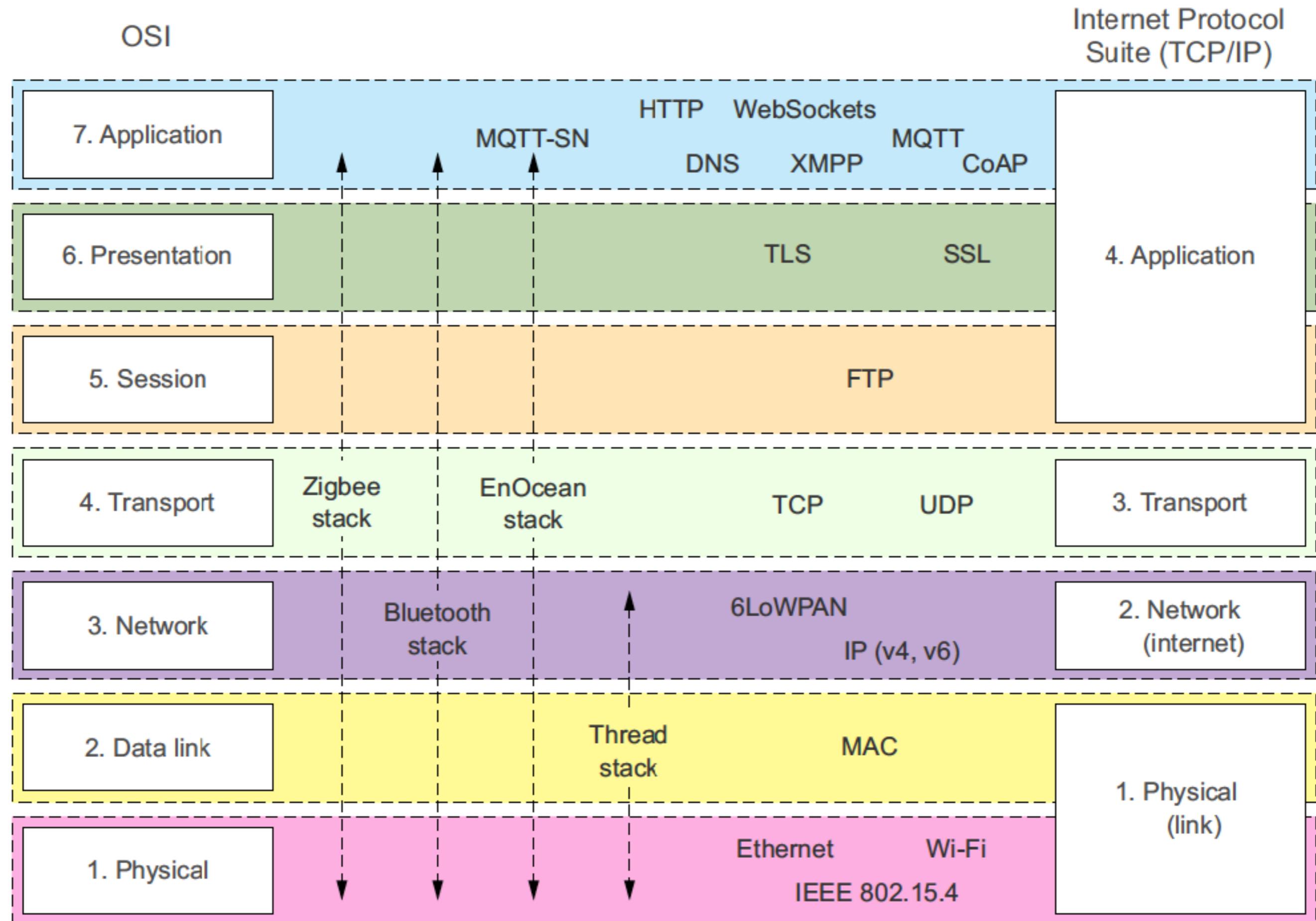
Back-end Data Sharing



IoT architectural styles

Architecture style	#studies
Layered	34
Cloud based	32
Service oriented	15
Microservices	6
Restful	5
Publish/subscribe	3
Information Centric Networking	2

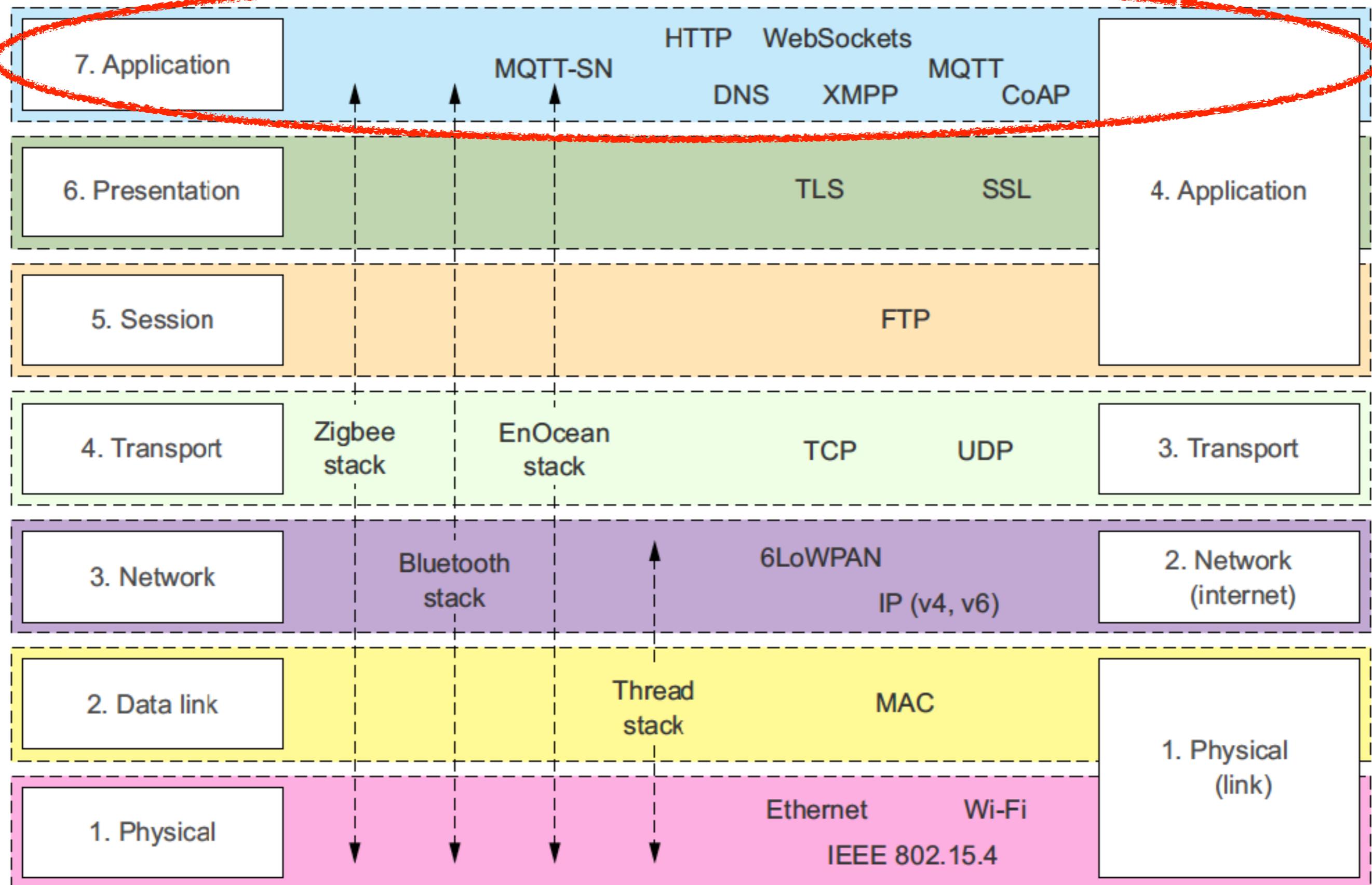
**E no modelo em camadas a
Internet das Coisas aparece
como?**



O foco desta disciplina é aqui

OSI

Internet Protocol
Suite (TCP/IP)



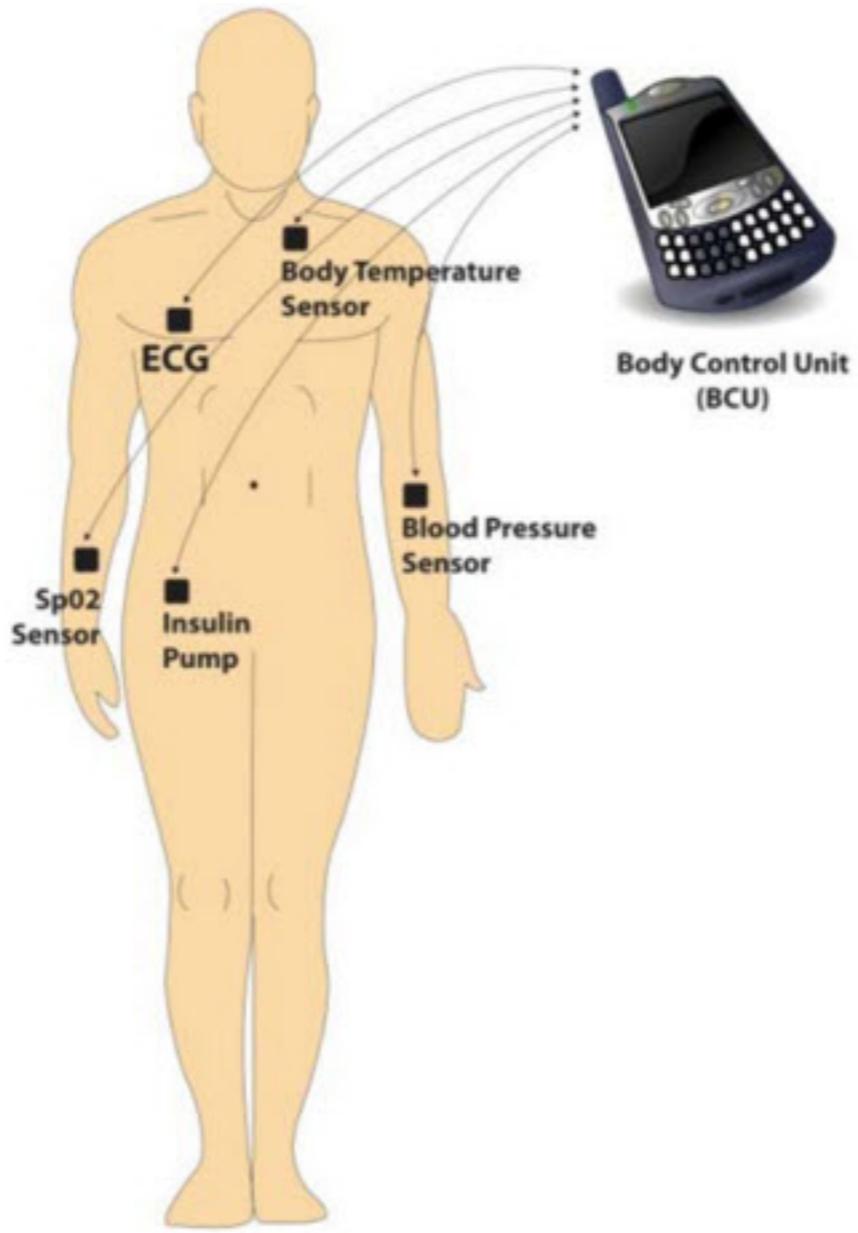
Os protocolos das coisas

Do ponto de vista de alcance

Spatial Scope	Typical Range	Examples
Near field (NFC)	< 10 cm	NFC Forum
Personal area network (PAN)	1 m–50 m	Bluetooth, ZigBee, Thread, IEEE 802.15
Local area network (LAN)	50 m–1 km	Wi-Fi, Ethernet
Wide area network (WAN)	1 km–50 km	SigFox, LoRa, 5G, 4G, GSM

Usos de NFC





Body Area Network (BAN)

Comparativo de protocolos PAN + Wifi

Name	Battery Usage	Observed Max Range (Indoors)	Mesh Networking	Openness	Ease of Use	Internet Integration
EnOcean	Very low	<30 m	No	Low	Medium	No
ZigBee	Low	<50 m	Yes	Low	Hard	No
Thread	Low	<50 m	Yes	Medium	Medium	Yes
Bluetooth	Low ^a	<50 m	Coming in 2016	Medium	Medium	No (upcoming)
Wi-Fi	High (low upcoming)	<30 m	No (internet)	High	Easy	Yes

Quanto maior a distância, maior o gasto de energia com o envio

**Isso vem sendo trabalhado fortemente em
redes LPWAN (Low-power WAN)**



**Soluções baseada em telefonia celular são vistas
como "provisórias" até que chegue o 5G (ETA
2020), que será low-power e terá suporte a mesh**

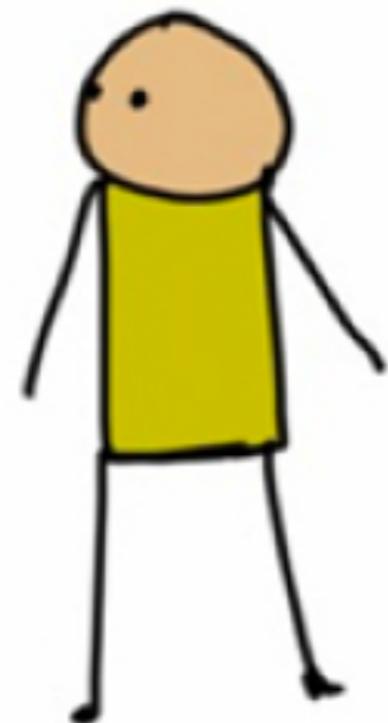
**Entretanto, LPWAN tem cobertura esparsa se
comparada com redes celulares**

Comparativo dos protocolos WAN mais comuns

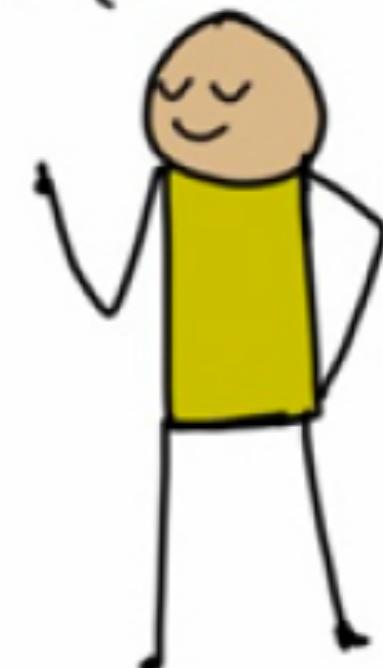
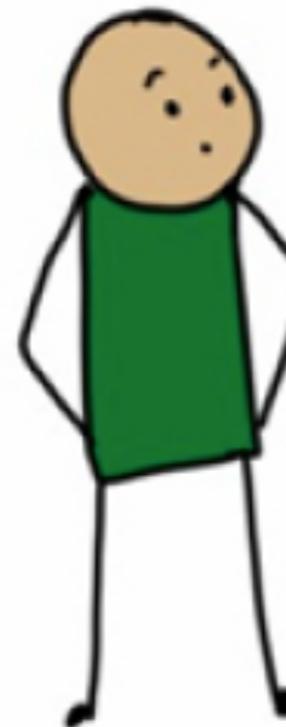
Name	Battery Usage	Max Range	Downlink	Openness	Coverage
Weightless	Very low	20+ km	Limited	Medium	Medium
SigFox	Very low	30+ km	Limited	Low	Medium-high
LoRa	Very low	30+ km	Limited	Medium	Medium
GPRS/3G/4G	High	50+ km	Yes	High	High
5G	Very low	?	? (probable)	High	Not deployed yet

TCP x UDP

I have a UDP joke, but you
might not get it!



Then, I would rather make a TCP joke because
it's a SYN to make a joke that others can't
ACKnowledge. \



**Dentre todas estas opções,
qual protocolo escolher?**

Depende...

Qual a distância necessária?

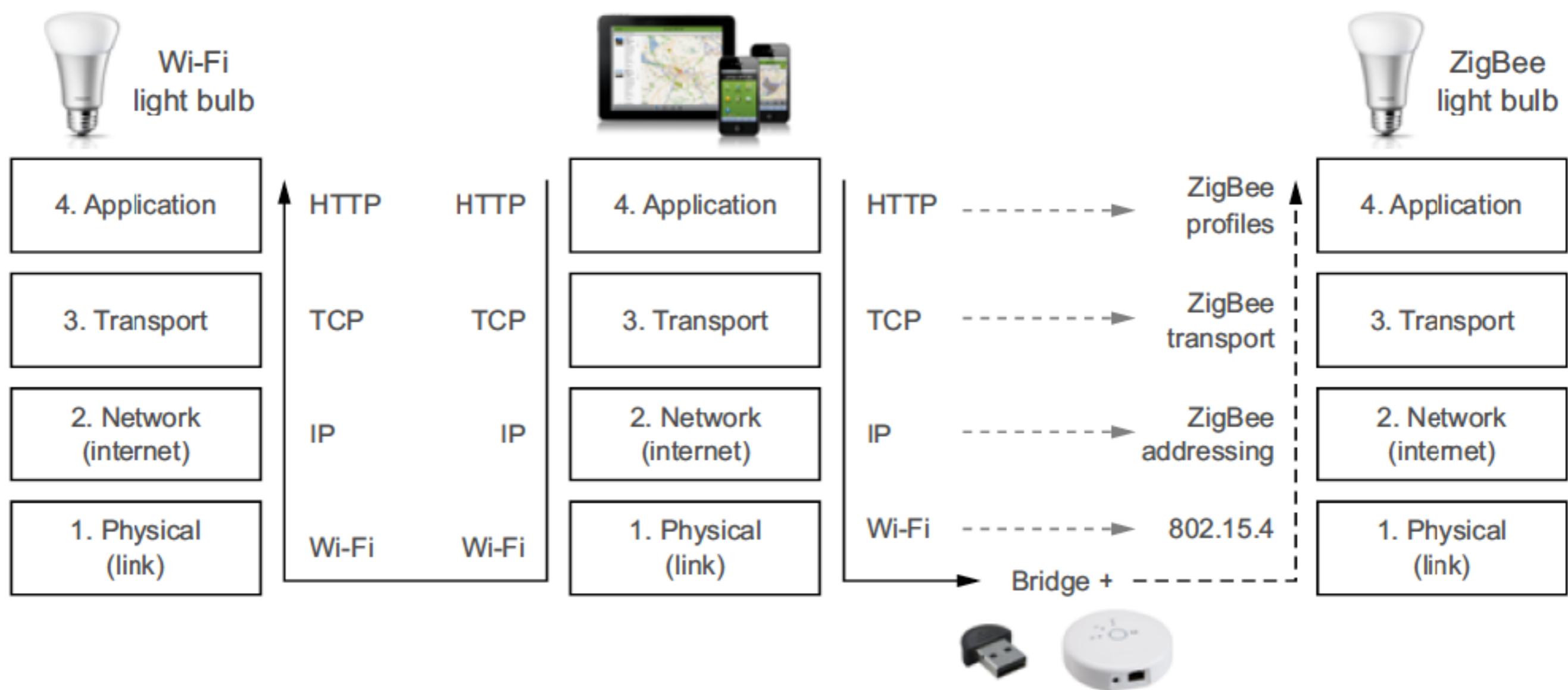
Qual será o custo do equipamento?

Envio precisa de garantia de entrega?

Requisitos de latência? Largura de banda? Envio de comandos (atuadores)?

As partes envolvidas falam o mesmo protocolo de aplicação?

Necessidade de tradução quando há pilhas de protocolo específicas

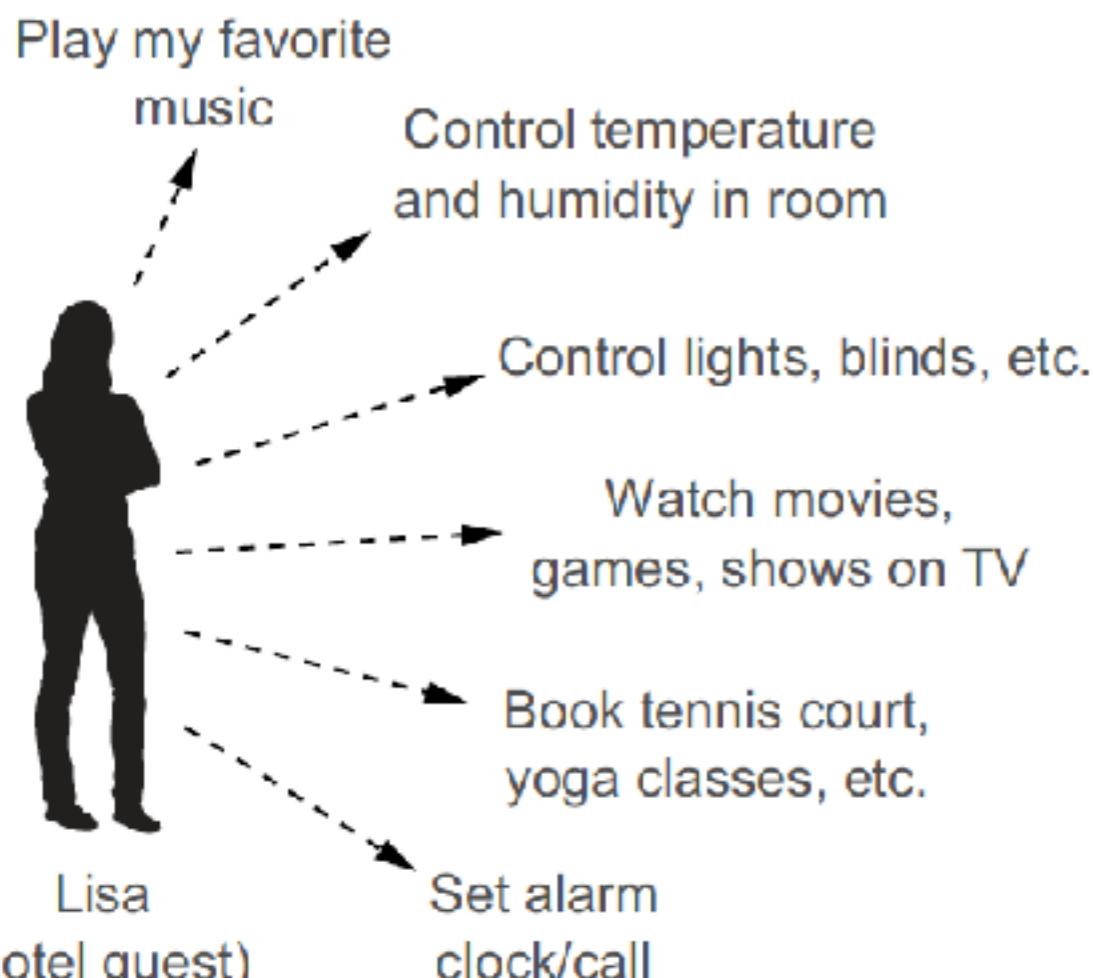


Algumas pilhas de protocolo específicas usadas na IoT

	Typical MQTT protocol stack	Typical MQTT-SN protocol stack	Typical CoAP protocol stack
4. Application	MQTT	MQTT-SN	CoAP CoRE
3. Transport	TCP	UDP	UDP
2. Network (internet)	IP	Not specified	6LoWPAN
1. Physical (link)	Not specified	Not specified	IEEE 802.15.4

"Intranet of things"

Room 202



Room 203



Room 204



Room 301



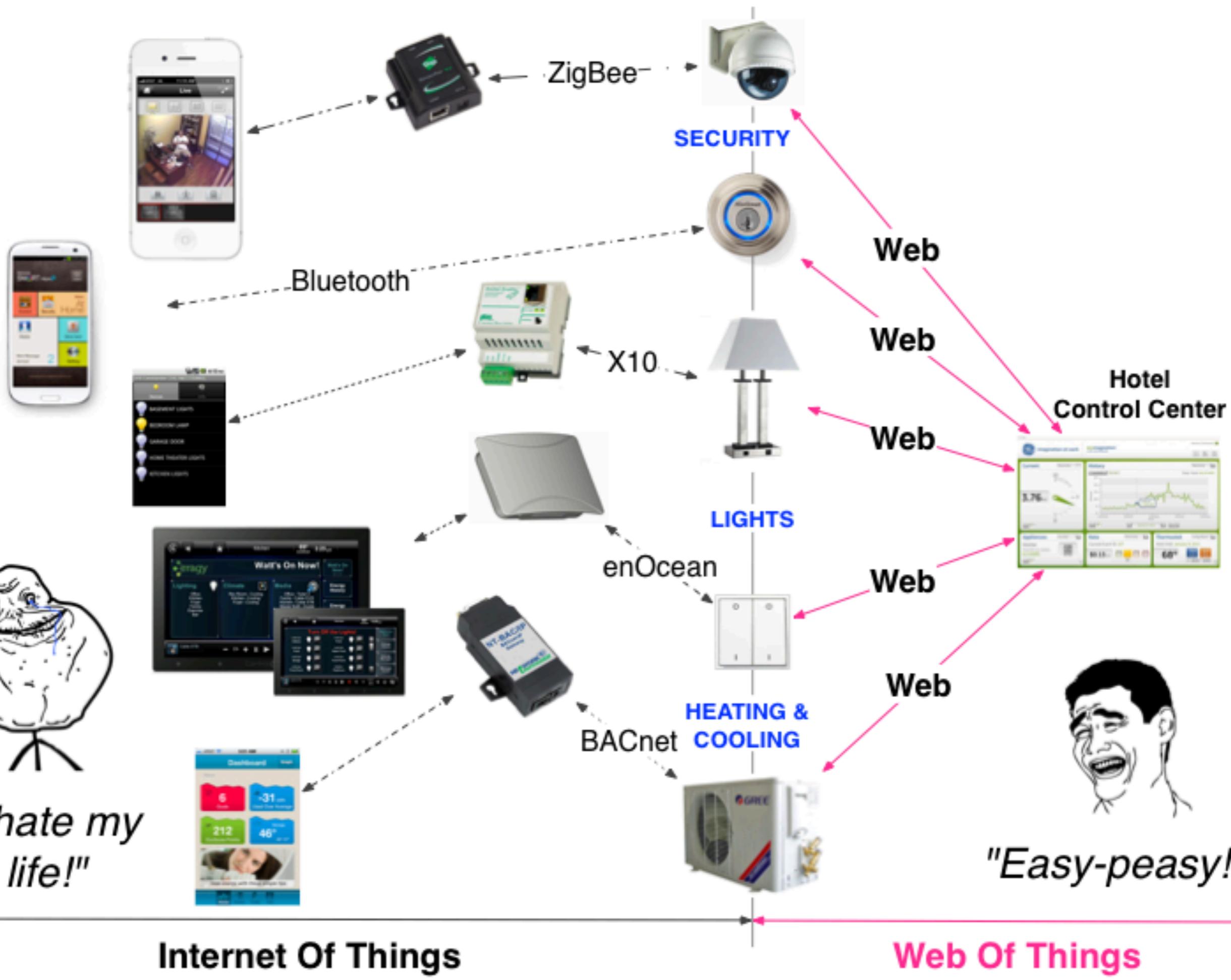
Hotel control center



- Turn off lights or air conditioning in all empty rooms
- Control security systems, cameras, smoke alarms, etc.
- Optimize room cleaning service
- Manage room services (Wi-Fi, TVs, etc.)
- Enable/disable room access for guests

Johnny (hotel owner)

**E se as coisas utilizassem
tecnologias da Web,
para se criar uma
Web das Coisas?**

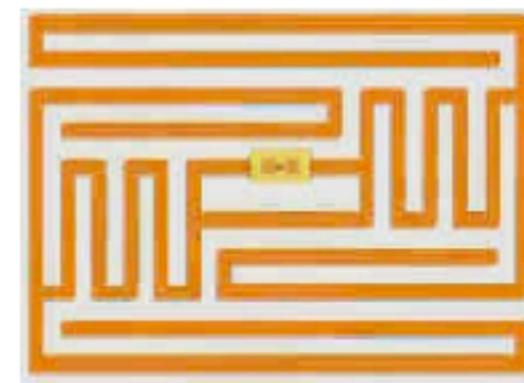


**A Web das Coisas pode
ser simplesmente vista
como um adorno dando
“Web Presence” para
objetos comuns...**

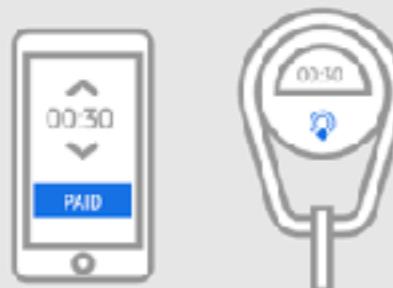
URLs para objetos físicos



URLs para objetos físicos

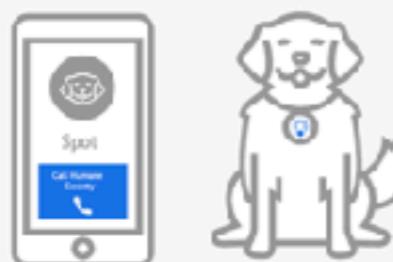


The Physical Web is an open approach to enable quick and seamless interactions with physical objects and locations.



Everything is a tap away

Walk up and interact with any object -- a parking meter, a toy, a poster -- or location -- a bus stop, a museum, a store -- without installing an app first. Interactions are only a tap away.



See what's useful around you

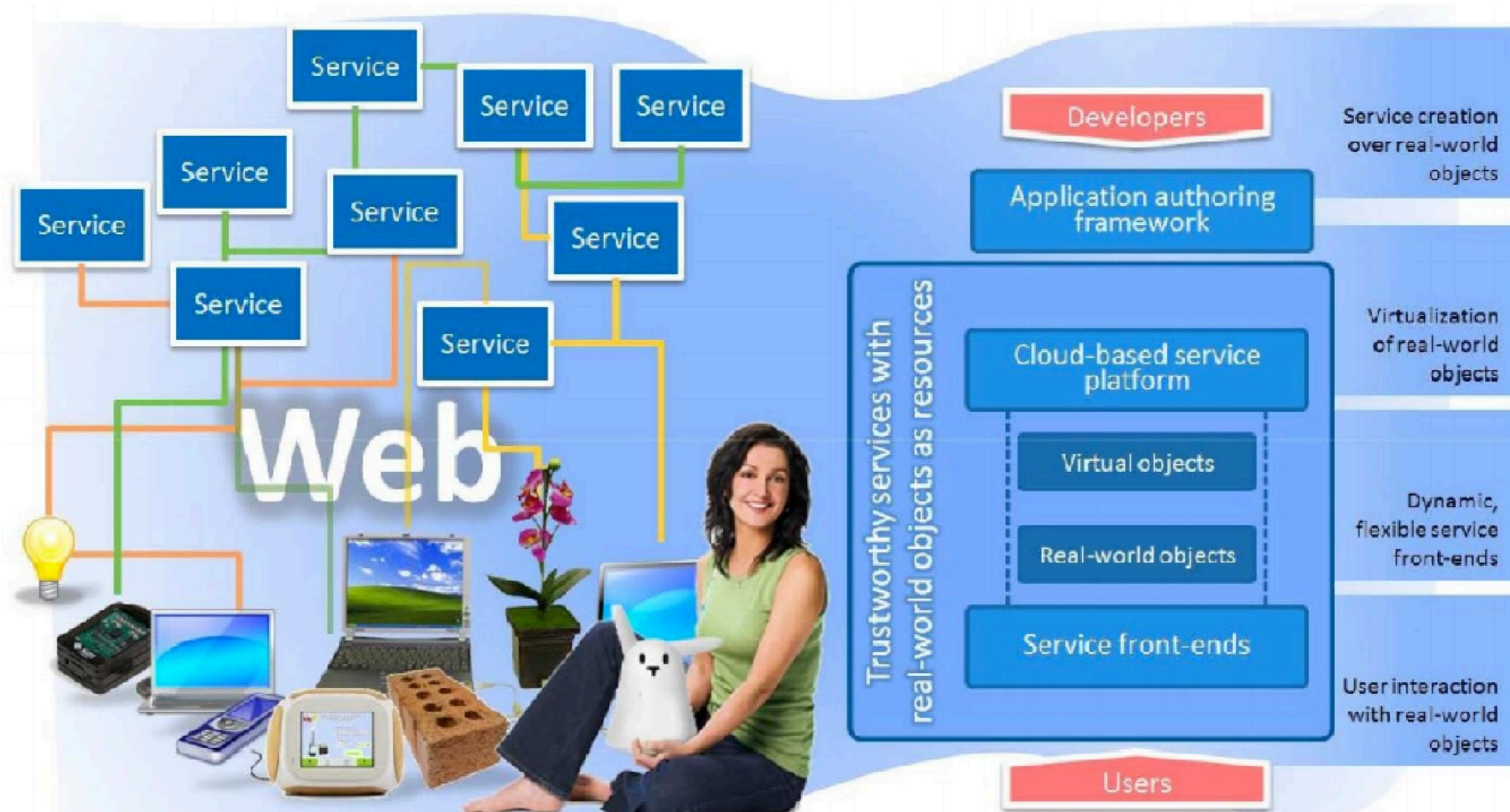
See web pages associated with the space around you. Choose the page most useful to you.



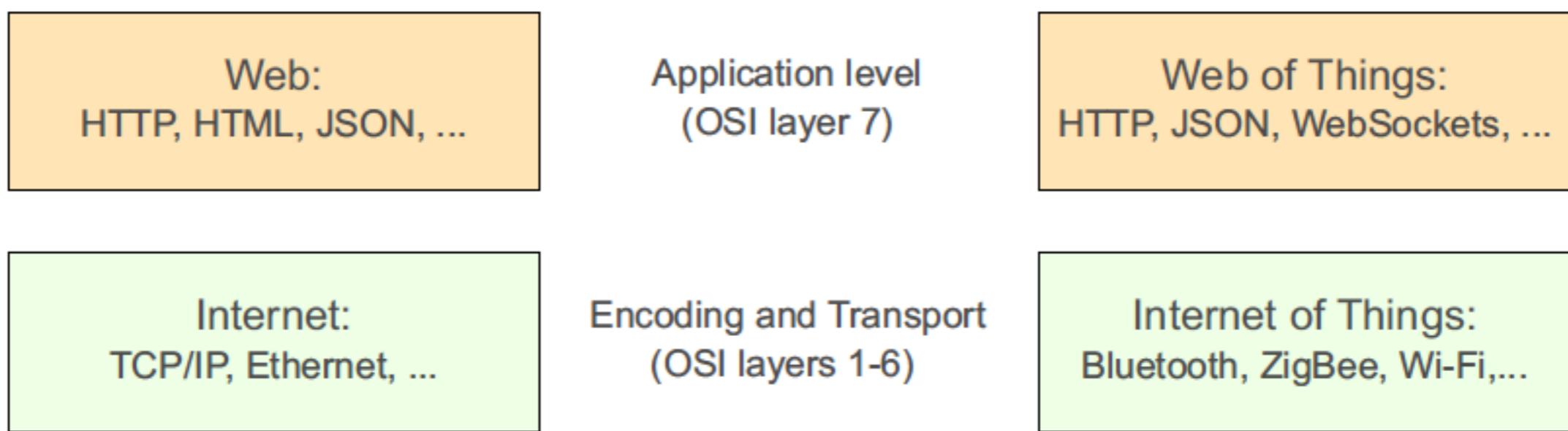
Any object or place can broadcast content

When *anything* can offer information and utility, the possibilities are endless.

**...ou pode ser uma
camada comum em cima
de objetos conectados**



Easier to program, faster to integrate data and services, simpler to prototype, deploy, and maintain large systems.



More lightweight and optimized for embedded devices (reduced battery, processing, memory and bandwidth usage), more bespoke and hard-wired solutions.

Layer 4: Compose	Systems integration WoT-a-Mashup	IFTTT Physical mashups Node-RED	Web applications Automated UI generation		
Layer 3: Share	Social networks PKI Encryption	API tokens OAuth Social WoT	TLS JWT Delegated authentication		
Layer 2: Find	REST Crawler HATEOAS Link header	RDFa Search engines mDNS	Web Thing Model Semantic Web Schema.org		
Layer 1: Access	HTML Webhooks URI/URL	JSON WebSockets Gateway	Proxy HTTP MQTT		
Networked things	NFC QR	6LoWPAN Beacons	Thread Bluetooth	Ethernet ZigBee	Wi-Fi 3/4/5 G
					

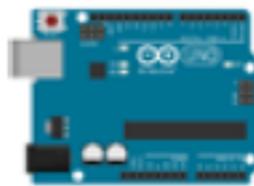
Social Web

Facebook
Twitter
OAuth Delegated authentication



Microdata JSON-LD
schema.org RDFa
Link headers

Semantic Web



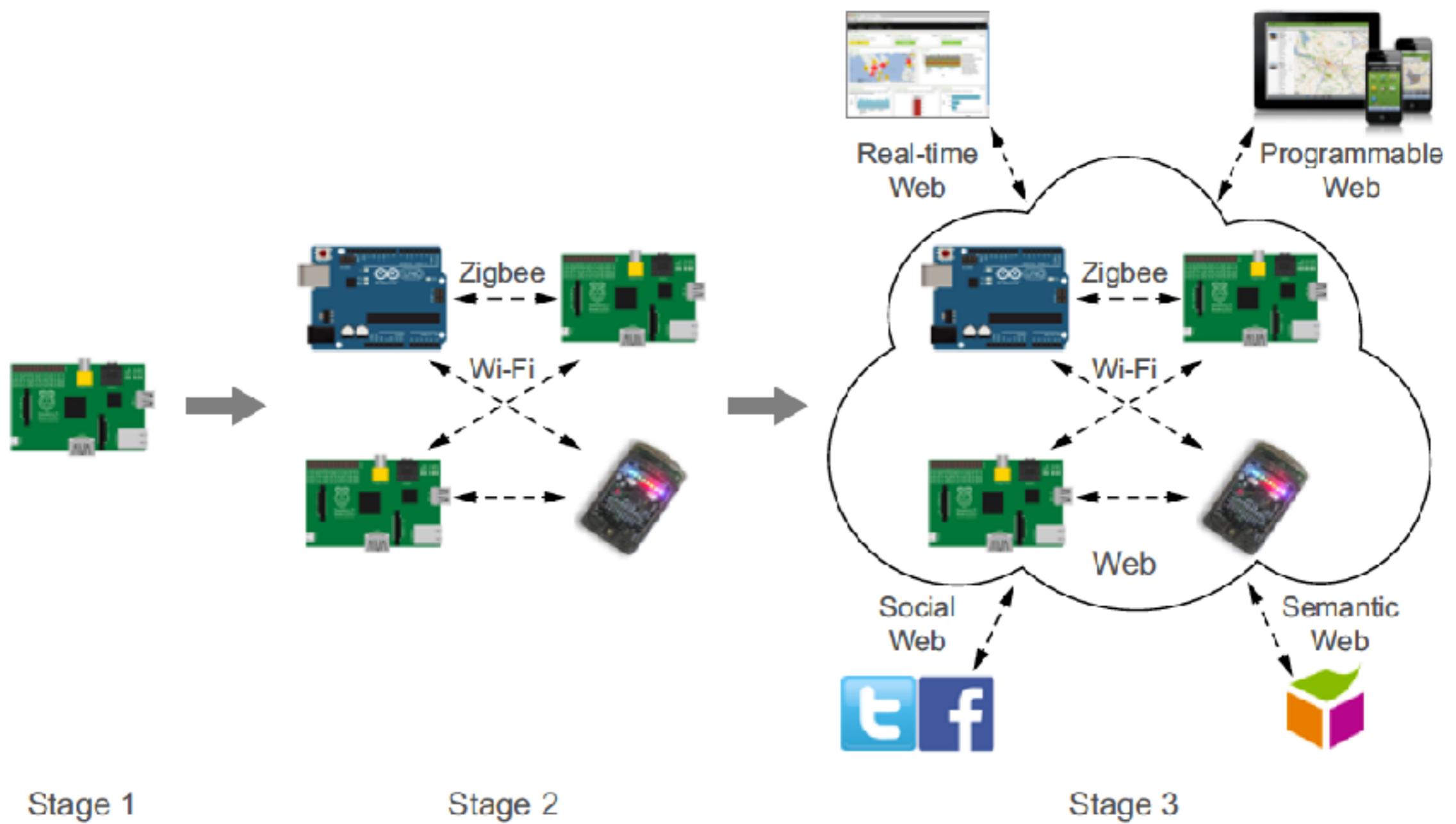
Real-time Web

RSS/ATOM
WebSockets
Web hooks
Push notifications



Mashups REST APIs
JSON
Web services

Programmable Web



Abordagem REST

REpresentational State Transfer

Princípios arquiteturais para a Web

Baseado em alguns *constraints*:

Client server

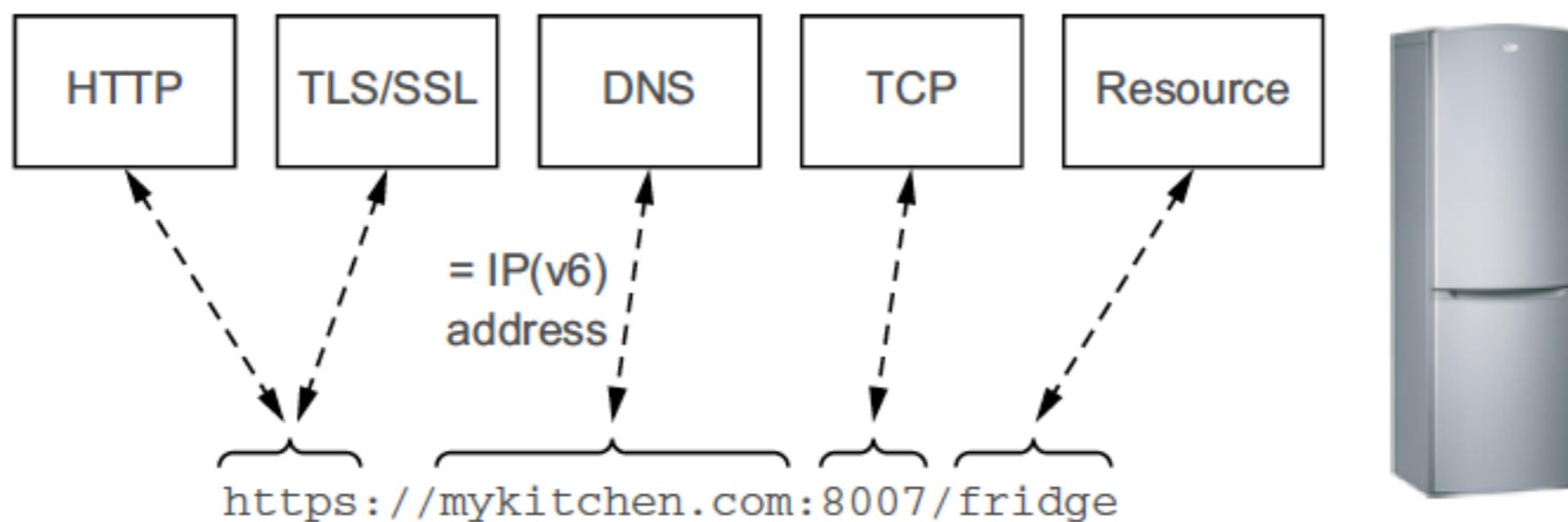
Uniform Interface

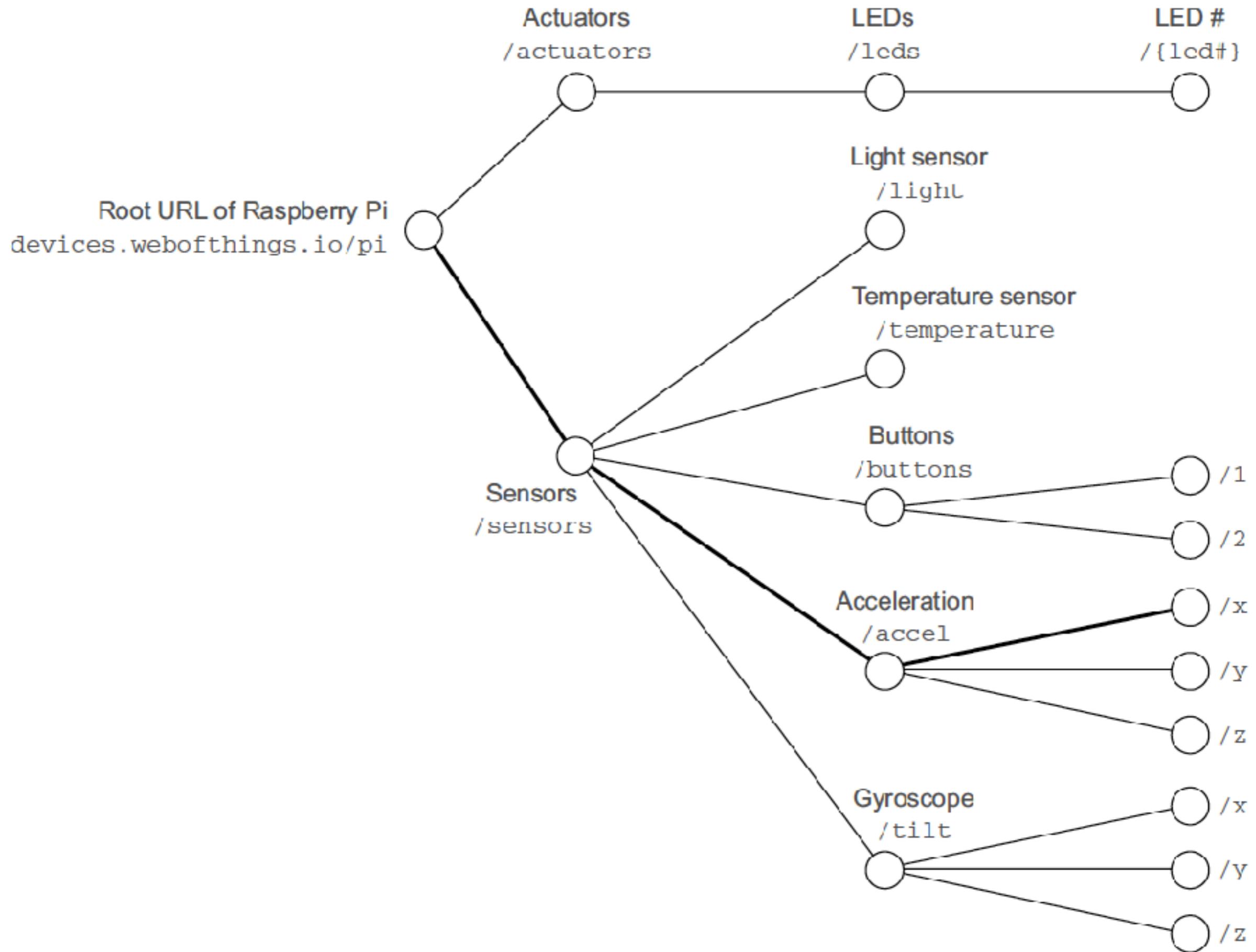
Stateless

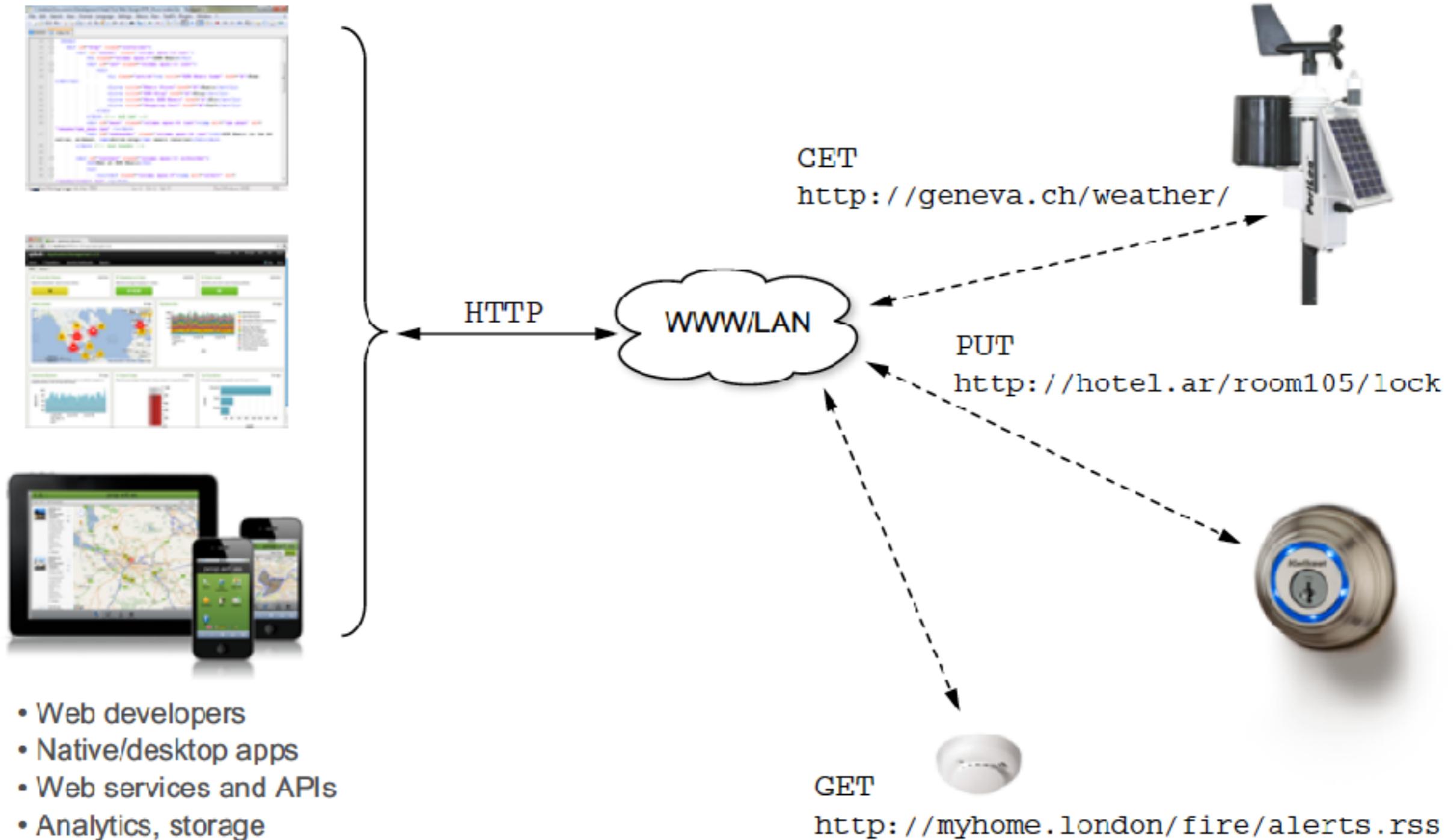
Cacheable

Layered System

Abordagem REST para coisas







HTTP

Hyper Text Transfer Protocol

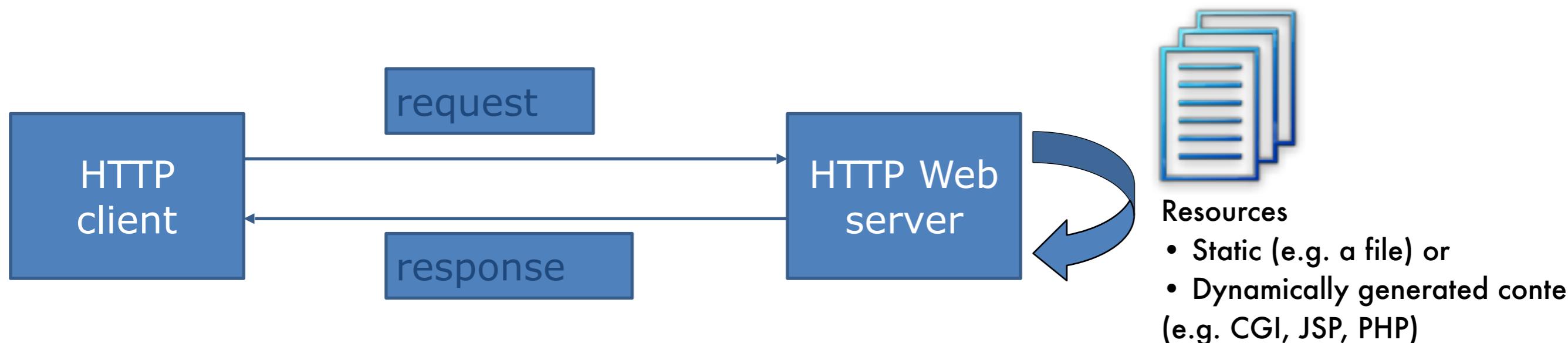
Baseado no paradigma request/response

Stateless

Fornece acesso a recursos referenciados por URLs

HTTP Clients

Browsers Web são tipicamente cliente HTTP



HTTP GET

/test/demo_form.asp?name1=value1&name2=value2

HTTP POST

POST /test/demo_form.asp HTTP/1.1

Host: w3schools.com

name1=value1&name2=value2

HTTP Status Codes

1xx Informational

100 Continue

101 Switching Protocols

102 Processing

2xx Success

200 OK

201 Created

202 Accepted

203 Non-authoritative Information

204 No Content

205 Reset Content

206 Partial Content

207 Multi-Status

208 Already Reported

226 IM Used

3xx Redirection

300 Multiple Choices

301 Moved Permanently

302 Found

303 See Other

304 Not Modified

305 Use Proxy

307 Temporary Redirect

308 Permanent Redirect

4xx Client Error

400 Bad Request

401 Unauthorized

402 Payment Required

403 Forbidden

404 Not Found

405 Method Not Allowed

406 Not Acceptable

407 Proxy Authentication Required

408 Request Timeout

409 Conflict

410 Gone

411 Length Required

412 Precondition Failed

413 Payload Too Large

414 Request-URI Too Long

415 Unsupported Media Type

416 Requested Range Not Satisfiable

417 Expectation Failed

418 I'm a teapot

421 Misdirected Request

422 Unprocessable Entity

423 Locked

424 Failed Dependency

426 Upgrade Required

428 Precondition Required

429 Too Many Requests

431 Request Header Fields Too Large

444 Connection Closed Without Response

451Unavailable For Legal Reasons

499 Client Closed Request

5xx Server Error

500 Internal Server Error

501 Not Implemented

502 Bad Gateway

503 Service Unavailable

504 Gateway Timeout

505 HTTP Version Not Supported

506 Variant Also Negotiates

507 Insufficient Storage

508 Loop Detected

510 Not Extended

511 Network Authentication Required

599 Network Connect Timeout Error

418 I'm a teapot

4xx CLIENT ERROR

418 I'M A TEAPOT

Any attempt to brew coffee with a teapot should result in the error code "418 I'm a teapot". The resulting entity body MAY be short and stout.

HTTP Request Headers Fields

Name	Description	Example	Status	Standard
A-IM	Acceptable instance-	A-IM: feed	Permanent	RFC 3229
Accept	Media type(s) that is/are	Accept: text/html	Permanent	RFC 2616, 7231
Accept-Charset	Character sets that are	Accept-Charset: utf-8	Permanent	RFC 2616
Accept-Datetime	Acceptable version in	Accept-Datetime: Thu, 31 May 2007 20:35:00 GMT	Provisional	RFC 7089
Accept-Encoding	List of acceptable	Accept-Encoding: gzip, deflate	Permanent	RFC 2616, 7231
Accept-Language	List of acceptable human	Accept-Language: en-US	Permanent	RFC 2616, 7231
Access-Control-Request-Method,	Initiates a request	Access-Control-Request-Method: GET	Permanent:	
Authorization	Authentication credentials	Authorization: Basic QWxhZGRpbjpvcGVuIHNlc2FtZQ==	Permanent	
Cache-Control	Used to specify directives	Cache-Control: no-cache	Permanent	
Connection	Control options for the	Connection: keep-alive	Permanent	
Content-Length	The length of the request	Content-Length: 348	Permanent	
Content-MD5	A Base64-encoded	Content-MD5: Q2h1Y2sgSW50ZWdyaxXR5IQ==	Obsolete ^[14]	
Content-Type	The Media type of the	Content-Type: application/x-www-form-urlencoded	Permanent	
Cookie	An HTTP	Cookie: \$Version=1; Skin=new;	Permanent:	
Date	The date and time at	Date: Tue, 15 Nov 1994 08:12:31 GMT	Permanent	
Expect	Indicates that particular	Expect: 100-continue	Permanent	
Forwarded	Disclose original	Forwarded:	Permanent	
From	The email address of the	From: user@example.com	Permanent	
Host	The domain name of the	Host: en.wikipedia.org:8080	Permanent	
HTTP2-Settings	A request that upgrades	HTTP2-Settings: token64	Permanent:	
If-Match	Only perform the action if	If-Match: "737060cd8c284d8af7ad3082f209582d"	Permanent	
If-Modified-Since	Allows a 304 Not	If-Modified-Since: Sat, 29 Oct 1994 19:43:31 GMT	Permanent	
If-None-Match	Allows a 304 Not	If-None-Match: "737060cd8c284d8af7ad3082f209582d"	Permanent	
If-Range	If the entity is unchanged,	If-Range: "737060cd8c284d8af7ad3082f209582d"	Permanent	
If-Unmodified-Since	Only send the response if	If-Unmodified-Since: Sat, 29 Oct 1994 19:43:31 GMT	Permanent	
Max-Forwards	Limit the number of times	Max-Forwards: 10	Permanent	
Origin ^[11]	Initiates a request	Origin: http://www.example-social-network.com	Permanent:	
Pragma	Implementation-specific	Pragma: no-cache	Permanent	
Proxy-Authorization	Authorization credentials	Proxy-Authorization: Basic	Permanent	
Range	Request only part of an	Range: bytes=500-999	Permanent	
Referer [sic]	This is the address of the	Referer: http://en.wikipedia.org/wiki/Main_Page	Permanent	
TE	The transfer encodings the	TE: trailers, deflate	Permanent	
User-Agent	The user agent string of	User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:12.0)	Permanent	
Upgrade	Ask the server to upgrade	Upgrade: h2c, HTTPS/1.3, IRC/6.9, RTA/x11, websocket	Permanent	
Via	Informs the server of	Via: 1.0 fred, 1.1 example.com (Apache/1.1)	Permanent	
Warning	A general warning about	Warning: 199 Miscellaneous warning	Permanent	

HTTP Response header fields

Field name	Description	Example	Status	Standards ref.
Access-Control-Allow-	Specifying which web sites can participate in cross-origin resource sharing	Access-Control-Allow-Origin: *	Permanent	
Accept-Patch ^[43]	Specifies which patch document formats this server supports	Accept-Patch: text/example; charset=utf-8	Permanent	
Accept-Ranges	What partial content range types this server supports via byte range requests	Accept-Ranges: bytes	Permanent	
Age	The age the object has been in a proxy cache in seconds	Age: 12	Permanent	
Allow	Valid methods for a specified resource. To be used for a 405 Method Not Allowed	Allow: GET, HEAD	Permanent	
Alt-Svc ^[44]	A server uses "Alt-Svc" header (meaning Alternative Services) to indicate multiple endpoint addresses	Alt-Svc: http/1.1="http2.example.com:443"	Permanent	
Cache-Control	Tells all caching mechanisms from server to client whether they can cache the response	Cache-Control: max-age=3600	Permanent	
Connection	Control options for the current connection and list of hop-by-hop headers	Connection: close	Permanent	
Content-Disposition ^[46]	An opportunity to raise a "File Download" dialogue box for a response	Content-Disposition: attachment; filename="file.txt"	Permanent	
Content-Encoding	The type of encoding used on the data. See HTTP compression .	Content-Encoding: gzip	Permanent	
Content-Language	The natural language or languages of the intended audience for the response	Content-Language: da	Permanent	
Content-Length	The length of the response body in octets (8-bit bytes)	Content-Length: 348	Permanent	
Content-Location	An alternate location for the returned data	Content-Location: /index.htm	Permanent	
Content-MD5	A Base64 -encoded binary MD5 sum of the content of the response	Content-MD5: Q2hLY2sgSW50ZWdyaxR5IQ==	Obsolete ^[14]	
Content-Range	Where in a full body message this partial message belongs	Content-Range: bytes 21010-47021/47022	Permanent	
Content-Type	The MIME type of this content	Content-Type: text/html; charset=utf-8	Permanent	
Date	The date and time that the message was sent (in "HTTP-date")	Date: Tue, 15 Nov 1994 08:12:31 GMT	Permanent	
Delta-Base	Specifies the delta-encoding entity tag of the response ^[10] .	Delta-Base: "abc"	Permanent	
ETag	An identifier for a specific version of a resource, often a message digest	ETag: "737060cd8c284d8af7ad3082f209582d"	Permanent	
Expires	Gives the date/time after which the response is considered stale (in "HTTP-date")	Expires: Thu, 01 Dec 1994 16:00:00 GMT	Permanent	
IM	Instance-manipulations applied to the response ^[10] .	IM: feed	Permanent	
Last-Modified	The last modified date for the requested object (in "HTTP-date")	Last-Modified: Tue, 15 Nov 1994 12:45:26	Permanent	
Link	Used to express a typed relationship with another resource, where applicable	Link: </feed>; rel="alternate" ^[49]	Permanent	
Location	Used in redirection , or when a new resource has been created.	• Example 1: Location: http://www.w3.org/pub/	Permanent	
P3P	This field is supposed to set P3P policy, in the form of a string	P3P: CP="This is not a P3P policy! See http://www.w3.org/standards/policy/forbidden"	Permanent	
Pragma	Implementation-specific fields that may have various effects	Pragma: no-cache	Permanent	
Proxy-Authenticate	Request authentication to access the proxy.	Proxy-Authenticate: Basic	Permanent	
Public-Key-Pins ^[51]	HTTP Public Key Pinning , announces hash of website's public key	Public-Key-Pins: max-age=2592000; pin-sha256=...; pin-list=...; includeSubDomains	Permanent	
Retry-After	If an entity is temporarily unavailable, this instructs the client to try again later	• Example 1: Retry-After: 120	Permanent	
Server	A name for the server	Server: Apache/2.4.1 (Unix)	Permanent	
Set-Cookie	An HTTP cookie	Set-Cookie: UserID=JohnDoe; Max-Age=3600; Path=/	Permanent	
Strict-Transport-Security	A HSTS Policy informing the HTTP client how long to cache the certificate	Strict-Transport-Security: max-age=31536000; includeSubDomains	Permanent	
Trailer	The Trailer general field value indicates that the given set of header fields continues	Trailer: Max-Forwards	Permanent	
Transfer-Encoding	The form of encoding used to safely transfer the entity to the client	Transfer-Encoding: chunked	Permanent	
Tk	Tracking Status header, value suggested to be sent in response to a tracking request	Tk: ?	Permanent	
Upgrade	Ask the client to upgrade to another protocol.	Upgrade: h2c, HTTPS/1.3, IRC/6.9, RTA/x11, sctp/1.0	Permanent	
Vary	Tells downstream proxies how to match future request headers to responses	• Example 1: Vary: *	Permanent	
Via	Informs the client of proxies through which the response was sent.	Via: 1.0 fred, 1.1 example.com (Apache/2.4.1)	Permanent	
Warning	A general warning about possible problems with the entity body.	Warning: 199 Miscellaneous warning	Permanent	
WWW-Authenticate	Indicates the authentication scheme that should be used to access the resource	WWW-Authenticate: Basic	Permanent	
X-Frame-Options ^[53]	Clickjacking protection: deny - no rendering within a frame	X-Frame-Options: deny	Obsolete ^[54]	

REpresentational State Transfer

Princípios arquiteturais para a Web

Baseado em alguns *constraints*:

Client server

Uniform Interface

Stateless

Cacheable

Layered System

Uniform Interface

- 1. Identification of resources**
- 2. Manipulation of resources through representations**
- 3. Self-descriptive messages**
- 4. Hypermedia as the engine of application state (HATEOAS)**

Uniform Interface

1. Identification of resources

Basicamente, URIs (Universal Resource Identifier)

2. Manipulation of resources through representations

Humano no browser quer a representação HTML, um programa utilizaria a representação JSON do mesmo recurso

3. Self-descriptive messages

Metadados (ex: Headers HTTP)

4. Hypermedia as the engine of application state (HATEOAS)

HATEOAS - Hypertext as the Engine of Application State

De forma resumida, refere-se à navegação entre links

A representação do estado de um recurso inclui links para outros recursos relacionados.

Uma API REST deve permitir essa naveabilidade entre recursos

HATEOAS

Ex: Suponha uma requisição HTTP GET para a URL
<http://api.domain.com/management/departments/10>

.. que retorna o JSON abaixo:

```
{  
  "departmentId": 10,  
  "departmentName": "Administration",  
  "locationId": 1700,  
  "managerId": 200,  
  "links": [  
    {  
      "href": "10/employees",  
      "rel": "employees",  
      "type" : "GET"  
    }  
  ]  
}
```

Há um link que permitiria a humanos e máquinas navearem até o recurso
<http://api.domain.com/management/departments/10/employees>

Exemplo da última aula

GET http://devices.webofthings.io/pi Send

Pretty Raw Preview JSON

```
1 {
2     "id": "1",
3     "name": "My WoT Raspberry Pi",
4     "description": "A simple WoT-connected Raspberry Pi for the WoT book.",
5     "url": "http://devices.webofthings.io/pi/",
6     "currentStatus": "Live",
7     "version": "v0.1",
8     "tags": [
9         "raspberry",
10        "pi",
11        "WoT"
12    ],
13    "resources": {
14        "sensors": {
15            "url": "sensors/",
16            "name": "The list of sensors"
17        },
18        "actuators": {
19            "url": "actuators/",
20            "name": "The list of actuators"
21        }
22    },
23    "links": {
24        "meta": {
25            "rel": "http://book.webofthings.io",
26            "title": "Metadata"
27        },
28        "doc": {
29            "rel": "https://www.raspberrypi.org/products/raspberry-pi-2-model-b/",
30            "title": "Documentation"
31        },
32        "ui": {
33            "rel": "*",
34            "title": "User Interface"
35        }
36    }
37 }
```

Tentando resumir a Web das Coisas

Uso de Padrões pré-existentes e já conhecidos

Tudo na camada de aplicação

Programação mais fácil

Melhor interoperabilidade