

INTRODUCCIÓN AL INTERNET DE LAS COSAS

CONSTRUYENDO UN PROYECTO DE IOT

Alvaro Everlet

alvaro.everlet@carriots.com

@aeeverlet

Javier Pastor

javier.pastor@carriots.com

@javipstor

Nov 2013

CONSTRUYENDO UN PROYECTO DE IOT

1. INTRODUCCIÓN AL IOT
2. CONSTRUYENDO UN PROYECTO DE IOT
3. DISPOSITIVOS
4. PLATAFORMA
5. FRONTEND
6. INTEGRACIÓN
7. DEMOS

CONSTRUYENDO UN PROYECTO DE IOT

1. INTRODUCCIÓN AL IOT
2. CONSTRUYENDO UN PROYECTO DE IOT
3. DISPOSITIVOS
4. PLATAFORMA
5. FRONTEND
6. INTEGRACIÓN
7. DEMOS

INTRODUCCIÓN AL IOT

DE DONDE VIENE

1999 Kevin Ashton inventa “Internet of Things”



INTERNET OF THINGS

El IoT es el mundo en el que cada objeto tiene una identidad virtual propia y capacidad potencial para integrarse e interactuar de manera independiente en la Red con cualquier otro individuo, ya sea una máquina (M2M) o un humano.

INTRODUCCIÓN AL IOT

NUESTRO MUNDO ESTÁ CAMBIANDO

Estamos sufriendo una transformación donde “things” hablan con otros “things” y estas conversaciones están creando nuevos modelos de negocio, productos y compañías.

Hace 20 años, Internet se usaba principalmente como herramienta para buscar información. En los últimos 10 años hemos vivido una nueva forma de uso de Internet, donde todo se ha convertido en social, transaccional y móvil.

¿QUÉ SERÁ LO SIGUIENTE?

Internet of Things.

El número de cosas conectadas a internet sobrepasó en 2008 el número de habitantes del planeta. Se estima que habrá 50.000 millones de dispositivos conectados en 2020.

CIFRAS

Según un estudio de CISCO, solo este año el volumen de negocio de las cosas conectadas a internet ascenderá a 475.000 millones de euros.

El IoT representaría un negocio mundial de 10.900 billones de euros en 5 años por la mayor productividad, ahorro de costes y nuevos mercados para las empresas.

INTRODUCCIÓN AL IOT

POR QUÉ AHORA

Ahora convergen múltiples factores:

- Popularización de placas de HW libre
- Abaratamiento de sensores
- Mejora comunicaciones
- Plataformas IoT



RaspberryPi



ZigBee
Control your world



INTRODUCCIÓN AL IOT

EJEMPLOS

extraídos de postscapes.com

Baby check



Tracking deportivo



Termostatos inteligentes



Recordatorio pastillas



Domótica



Tracking de objetos (perdidos)



Pastillas inteligentes



Enchufes conectados



Vigilancia no intrusiva



INTRODUCCIÓN AL IOT

CASOS DE USO INDUSTRIALES



SMART CITY
SMART ENERGY
SMART OIL & GAS

SMART AGRICULTURE
SMART BUILDINGS
SMART RETAIL

SMART BANKING
SMART CONSUMER PRODUCTS
SMART LOGISTICS

INTRODUCCIÓN AL IOT

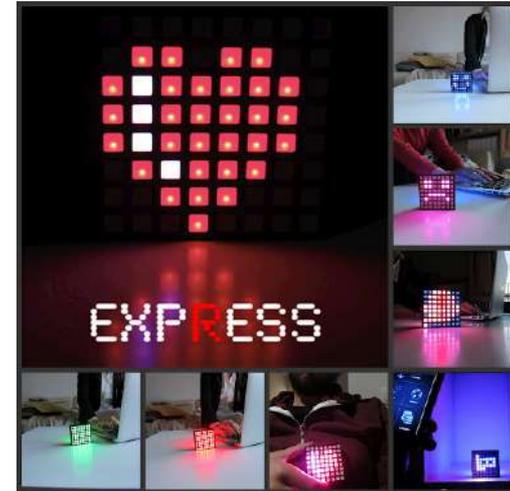
TENDENCIAS Y LOCURAS



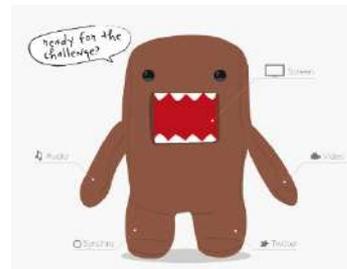
Good night lamp (kickstarter)



Smart egg container



L8 smartlight (kickstarter)



@domokunM2M

INTRODUCCIÓN AL IOT

PASOS A SEGUIR

Una cosa (thing)

Dotarlo de electrónica

Conectarlo



INTRODUCCIÓN AL IOT

HOMENAJE A LOS DEVELOPERS

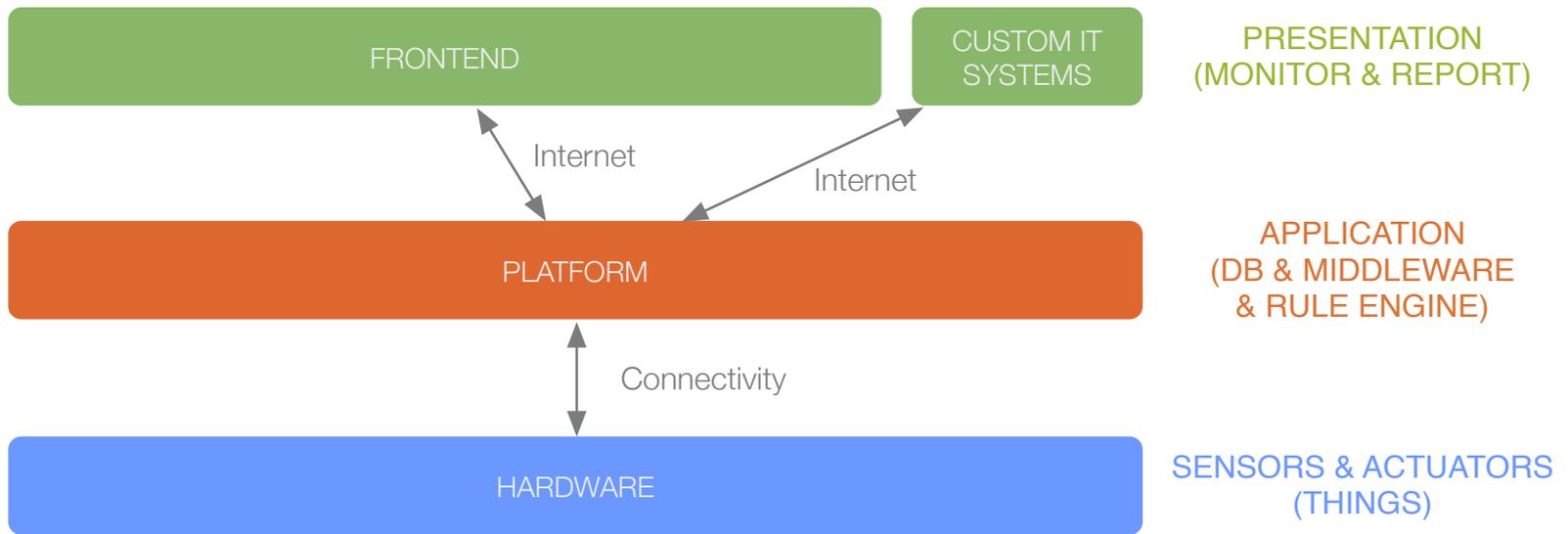


CONSTRUYENDO UN PROYECTO DE IOT

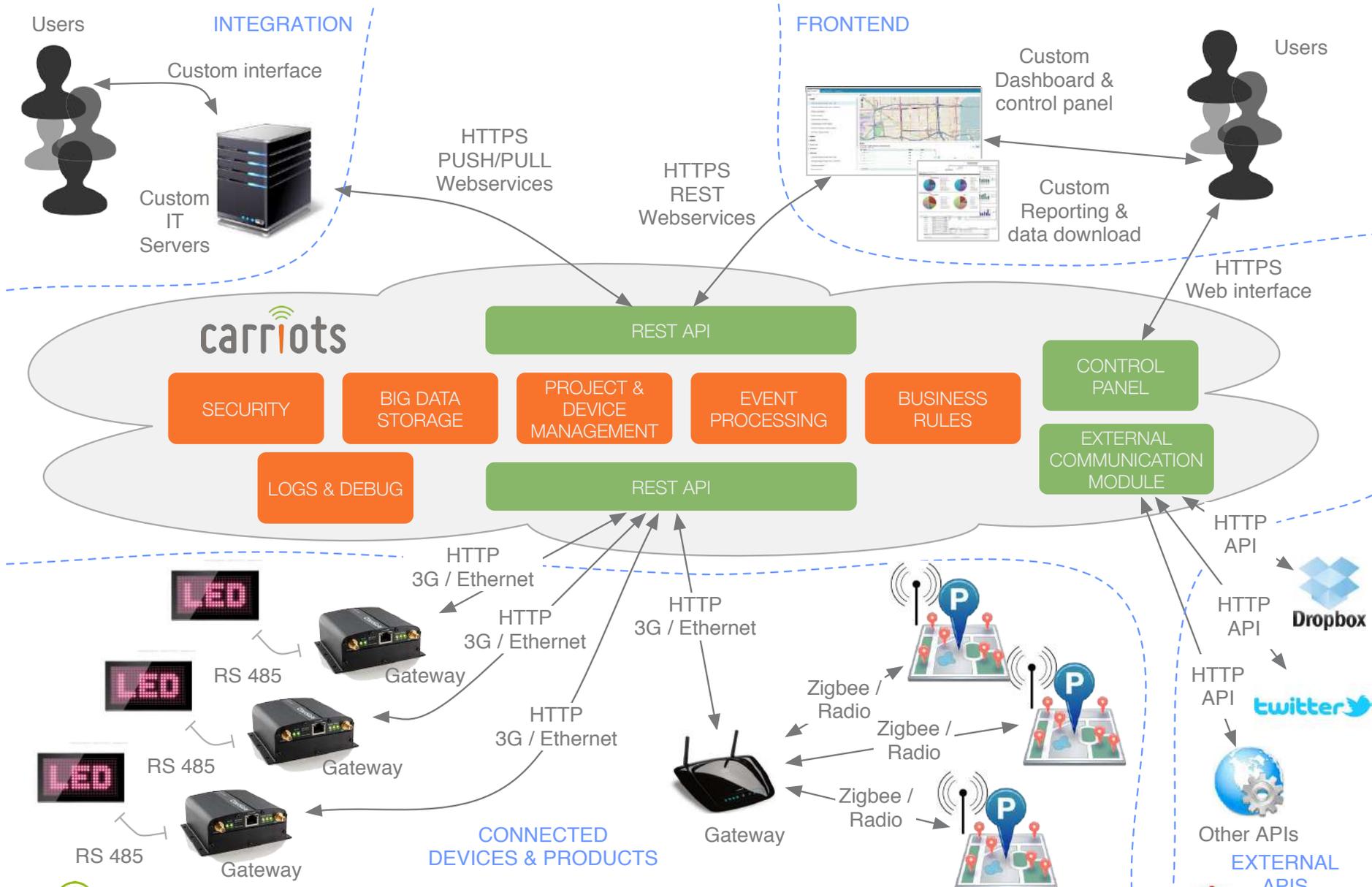
1. INTRODUCCIÓN AL IOT
2. CONSTRUYENDO UN PROYECTO DE IOT
3. DISPOSITIVOS
4. PLATAFORMA
5. FRONTEND
6. INTEGRACIÓN
7. DEMOS

CONSTRUYENDO UN PROYECTO DE IOT

PROJECT STACK



CONSTRUYENDO UN PROYECTO DE IOT



CONSTRUYENDO UN PROYECTO DE IOT

COMPLEJO O POTENTE?

PROPONGO UNA VUELTA RÁPIDA POR LOS ELEMENTOS DE UN PROYECTO DE INTERNET DE LAS COSAS SIGUIENDO UN EJEMPLO

SMART PARKING (CASO DE USO DE UNA SMART CITY)

CONSTRUYENDO UN PROYECTO DE IOT

DISPOSITIVOS

Tienen **sensores** captando información en bruto. Ej: Detector magnético de parking
Tienen actuadores para controlar cosas. Ej: Panel LED de señalización en las calles
Tienen conectividad a Internet. Ej: Wifi, ethernet o un modem 3g integrado

PLATAFORMA

Almacena la información en bruto: BD que puede ser consultada para análisis de BI
Contiene y ejecuta toda la **lógica de la aplicación**: reglas, alarmas, etc.
Proporciona la **gestión de dispositivos y su software**. Ej: provisión de dispositivos, habilitar o deshabilitar dispositivos, cambiar su firmware, etc.

FRONTEND

Para hacer una aplicación usable, el frontend proporciona la interfaz con el usuario final. Ej: un panel de control personalizado, dashboard, informes, etc.

INTEGRACIÓN

Para proporcionar el verdadero potencial de la conectividad a Internet la plataforma se integra con otros sistemas de información haciendo **PUSH o PULL** hacia o desde CRMs, ERPs o cualquier API HTTP disponible. Ej: IOC de IBM, Dropbox, Zoho, Twitter, etc.

CONSTRUYENDO UN PROYECTO DE IOT

1. INTRODUCCIÓN AL IOT
2. CONSTRUYENDO UN PROYECTO DE IOT
3. DISPOSITIVOS
4. PLATAFORMA
5. FRONTEND
6. INTEGRACIÓN
7. DEMOS

DISPOSITIVOS

SENSORES & ACTUADORES

Los **sensores** toman información en bruto que lee un dispositivo. Pueden medir temperatura, presencia, viento, campos magnéticos, flujos de líquidos, calidad del aire, propiedades de los líquidos, vibraciones, posiciones geográficas, altitud, presión y un largo etc.

Los **actuadores** permiten la interacción con el mundo físico. Pueden abrir puertas, mostrar mensajes, hacer encender luces, enviar señales, controlar un flujo de líquidos, controlar el aire acondicionado, encender y apagar motores, etc.

CONECTIVIDAD

En los proyectos de IoT los dispositivos se conectan a Internet para interactuar.

Conectividad autónoma (Standalone)

Modems 3G / GPRS embebidos en el dispositivo

Dispositivo + Pasarela

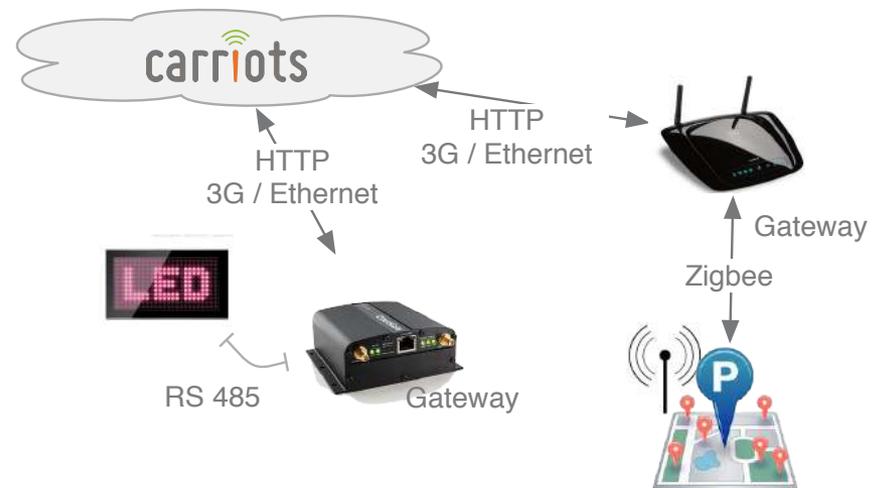
Comunicación dispositivo-pasarela serie (ej. RS232 or RS485) o radio (ej. 868MHz)

Red de sensores + Pasarela

Comunicación local por WIFI o ethernet entre dispositivos y pasarela.

COMMUNICATION PROTOCOL

Los dispositivos interactúan con Carriots con un protocolo estándar HTTP / HTTPS en la capa de Internet y JSON o XML en la capa de aplicación (REST API)



DISPOSITIVOS

EJEMPLO DE ENVÍO DE TRAMA: información de plaza de parking

HTTP request

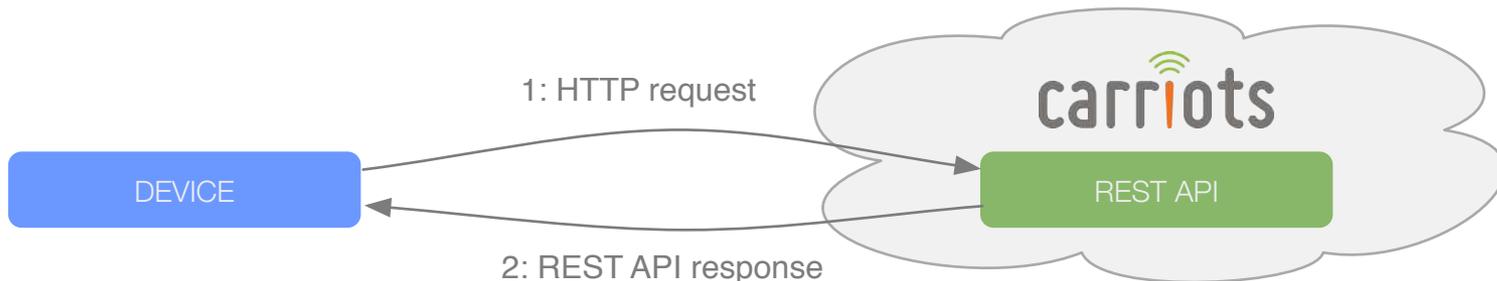
```
POST /streams HTTP/1.1  
Host: api.carriots.com  
Accept: application/json  
User-Agent: place_524@smartparking  
Content-Type: application/json  
carriots.apikey:98346673a637...5a0d83045425407ab4  
Content-Length: 182  
Connection: close
```

← REST API URL (POST request)
← JSON FORMAT
← SECURITY (APIKEY)

Data (PAYLOAD)

```
{  
  "protocol": "v2",  
  "at": "now",  
  "device": "place_524@smartparking",  
  "data": {"parking": "on"},  
  "checksum": "2c0766329b4d4b3beb08...97ae7b7de2160be"  
}
```

← Validación por checksum (HMAC)
← Carriots automatic data timestamping
← Dispositivo (Apikey debe tener permiso)
← La información a mandar



CONSTRUYENDO UN PROYECTO DE IOT

1. INTRODUCCIÓN AL IOT
2. CONSTRUYENDO UN PROYECTO DE IOT
3. DISPOSITIVOS
4. PLATAFORMA
5. FRONTEND
6. INTEGRACIÓN
7. DEMOS

PLATAFORMA

API REST

Siguiendo un estándar ampliamente adoptado en Internet, Carriots implementa una API REST sobre HTTPS para:

- **Recibir** cantidades masivas de datos de los dispositivos.
- **Interactuar con todas las entidades de Carriots** para construir paneles de control personalizados, dashboards y e informes

Ej: Sensores de parking envían su estado (libre u ocupado) a Carriots con una simple petición HTTP. Un panel de control personalizado usa la API REST para gestionar dispositivos.

ALMACENAMIENTO BIG DATA

Grandes cantidades de datos se almacenan en una arquitectura de big data con estructura

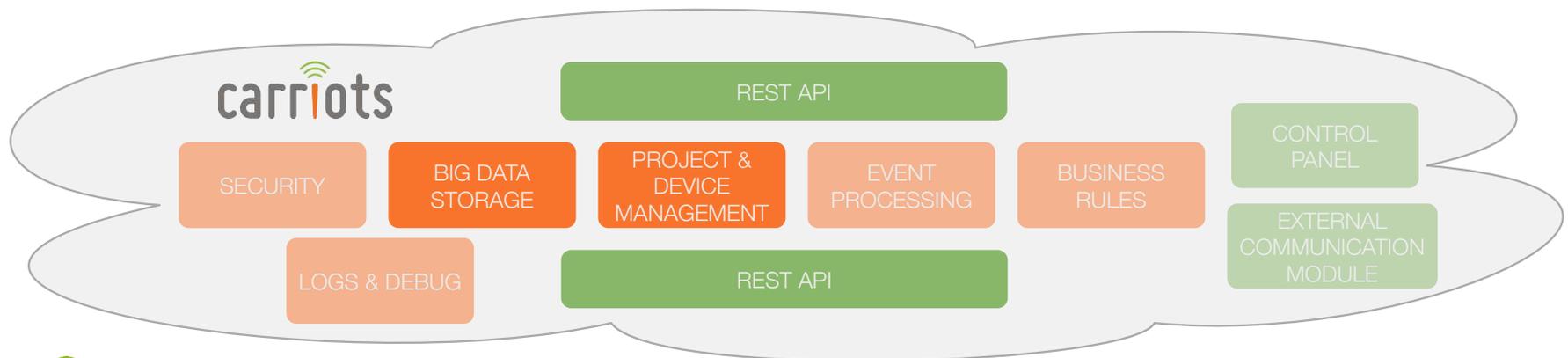
schemaless. Proporciona a los proyectos de IoT la flexibilidad de gestionar datos heterogéneos de diferentes dispositivos.

Ej: Temperatura, posición geográfica, estado de una plaza de parking, etc.

GESTIÓN DE DISPOSITIVOS Y PROYECTOS

Para cubrir cualquier requisito de un proyecto Carriots propone una jerarquía lista para usar y de uso flexible para la complejidad de cada caso Actualización de firmware y configuración remota de dispositivos también disponibles.

Ej: Ajustar la periodicidad del muestreo de los sensores o especificar un mensaje por defecto en un panel de señalización.



PLATAFORMA

LÓGICA DE NEGOCIO Y EVENTOS

La lógica del proyecto de IoT se aloja y se ejecuta en la plataforma. Scripts de Groovy se ejecutan (aislados y securizados) en el motor SDK y se lanzan en base a eventos tratados con una aproximación tipo **if-then-else**. Desde scripts sencillos a complejas reglas.

Ej: `If parking_sensor="ocupado" then parking_area.plazas -1.`
`If parking_area.plazas=0 then display.mensaje="Parking lleno"`

SEGURIDAD

En todos los proyectos IoT la seguridad debe formar parte de su diseño. La aproximación de Carriots a este tema:

- **Apikeys** definen privilegios y visibilidad.
- **HTTPS** para cifrar conversaciones con la API REST.
- **HMAC hash** y contraseñas pre-

compartidas para firmar el mensaje.

- **Cifrado** personalizado o medidas adicionales de seguridad en el código de los eventos.

LOGS & DEBUG

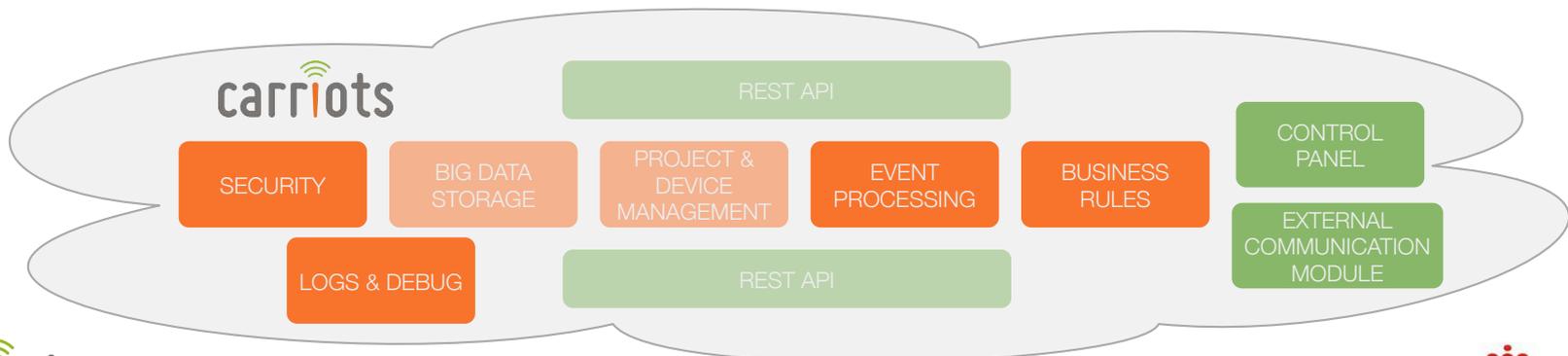
Mensajes de Log (accesibles desde el panel de control) y una consola para **debug console** necesarios para facilitar el desarrollo.

CONTROL PANEL

El panel de control es la interfaz web para desarrolladores de proyectos IoT con Carriots.

MÓDULO DE COMUNICACIÓN

Enviar emails o SMS, interactuar con otros sistemas o APIs se realiza con un módulo específico para garantizar la seguridad y el rendimiento.



PLATAFORMA

EJEMPLO: Lógica de Groovy ejecutada al recibir un dato de plaza de parking

```
// Libraries
import com.carriots.sdk.Device;
import com.carriots.sdk.utils.BasicHttp;

// Free places counter update
def device = Device.find('ParkingControl@smartparking');
def places = new Integer(device.device_properties.FreePlaces);
device.device_properties.FreePlaces=places-1;
device.update();

// Place update
def place = Device.find(context.device);
place.device_properties['free']='no';
place.update();

// Display location
def led = Device.find('LedDisplay-12@smartparking');
def address = led.device_properties.address;
def token = led.device_properties.token;

// Display update
def basicHttp = new BasicHttp();
basicHttp.url = "http://" + address + "/message";
basicHttp.params=["text":"Free places: " + device.device_properties.FreePlaces,
"token":token];
basicHttp.send();
```

← SDK LIBRARIES IMPORT

← DATA BASE ACCESS

← CUSTOM PROPERTIES MANAGEMENT

← CARRIOTS -> DEVICE COMMUNICATION
IP address stored as device property

PLATAFORMA

PANTALLAZO DEL PANEL DE CONTROL DE CARRIOTS EJEMPLO DE CREACIÓN DE LISTENER

The screenshot displays the 'Listener creation' interface in the Carriots Control Panel. The interface includes a navigation menu on the left with sections for Project Management, Device Management, Data management, and Rules management. The main content area contains the following fields and options:

- Name:** Text input field.
- Description:** Text input field.
- Entity type:** Dropdown menu with 'Project' selected.
- Id:** Text input field with 'defaultProject@smartcity'.
- Event:** Dropdown menu with 'Event Data Received' selected.
- If expression:** Text input field containing '1 context.data['parking']=='on'.
- Then expression:** Text input field containing '1 // Groovy expression'.
- Then rule:** Dropdown menu.
- Else expression:** Text input field containing '1 // Groovy expression'.
- Else rule:** Dropdown menu.
- Enabled:** Radio button (checked).

Annotations with arrows point to the following elements:

- ENTIDAD VIGILADA:** Points to the 'Entity type' dropdown.
- EVENTO QUE DESENCADENA LA LÓGICA:** Points to the 'Event' dropdown.
- APROXIMACIÓN IF-THEN-ELSE:** Points to the 'If expression', 'Then expression', and 'Else expression' fields.
- SCRIPTS DE GROOVY:** Points to the 'Then expression' and 'Else expression' fields.
- REGLAS PREDEFINIDAS:** Points to the 'Then rule' and 'Else rule' dropdowns.

At the bottom of the form, there are 'Create' and 'List' buttons.

CONSTRUYENDO UN PROYECTO DE IOT

1. INTRODUCCIÓN AL IOT
2. CONSTRUYENDO UN PROYECTO DE IOT
3. DISPOSITIVOS
4. PLATAFORMA
5. FRONTEND
6. INTEGRACIÓN
7. DEMOS

FRONTEND

PANEL DE CONTROL DE CARRIOTS

El panel de control proporcionado por Carriots es la primera herramienta necesaria para la creación de proyectos IoT. Desde gestionar los detalles del proyecto hasta la depuración y construcción de toda la lógica del proyecto.

Ej: Definir las zonas de parking, definir los mensajes de los paneles de señalización, etc.

FRONTENDS PERSONALIZADOS

Paneles de control personalizados, apps para móviles, widgets, etc. Lo que realmente quiere ver el usuario final de un proyecto IoT es la simplicidad de uso, abstrayéndose de la complejidad.

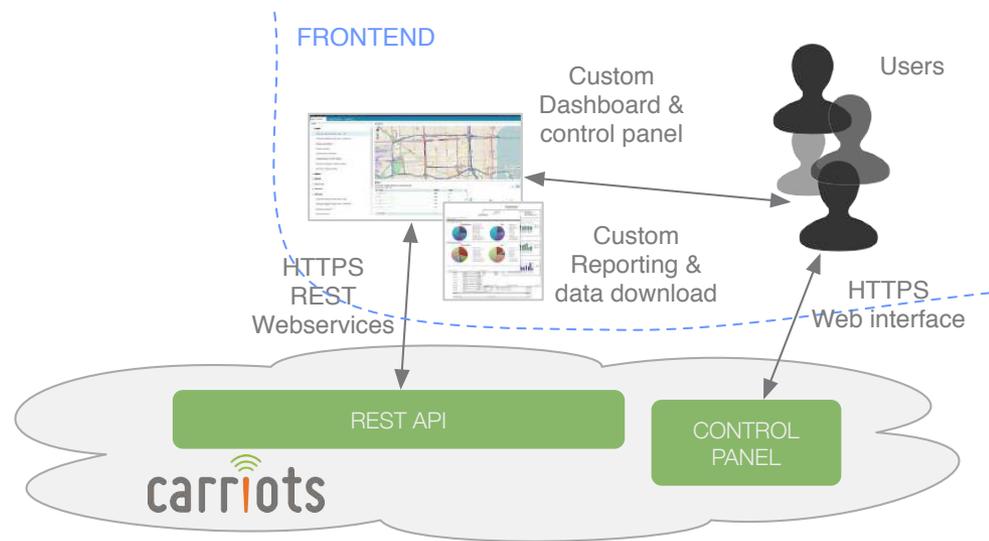
Ej: Mapas personalizados de las plazas de parking, gráficas de disponibilidad, gestión

remota del panel led, una REST API propia, etc.

MONITORIZACIÓN

Herramientas y técnicas disponibles para la monitorización:

- Pooling de los datos de la API REST.
- Descarga de datos en fichero
- Gráficas personalizadas de Carriots
- PUSH de datos basado en eventos



CONSTRUYENDO UN PROYECTO DE IOT

1. INTRODUCCIÓN AL IOT
2. CONSTRUYENDO UN PROYECTO DE IOT
3. DISPOSITIVOS
4. PLATAFORMA
5. FRONTEND
6. INTEGRACIÓN
7. DEMOS

INTEGRACIÓN

SISTEMAS EXTERNOS

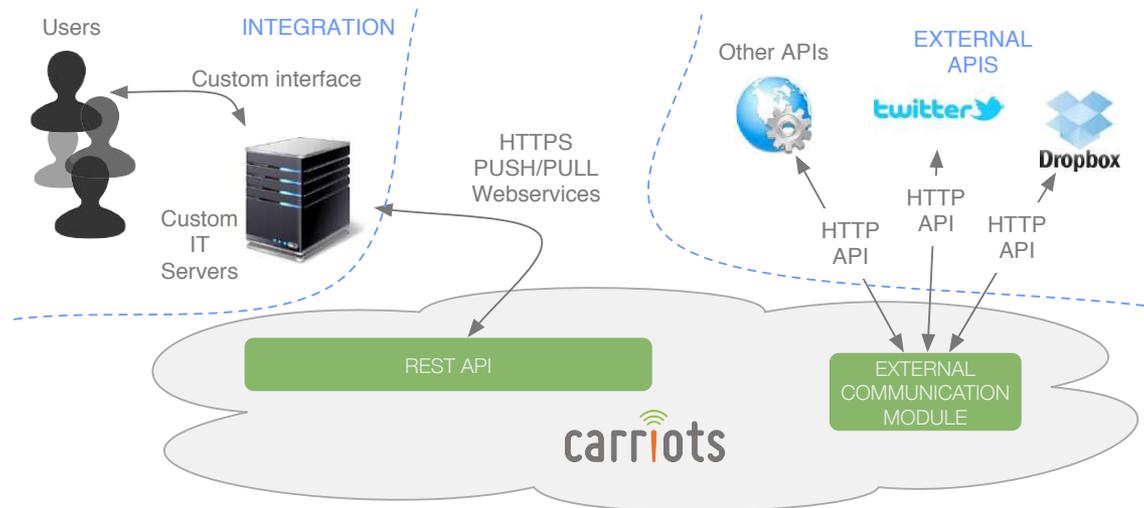
Para destapar el verdadero potencial de los proyectos IoT la integración es una pieza fundamental. Mediante la API REST de Carriots (data PULL), PUSH de datos personalizados o peticiones HTTP/s (RESTful) o sockets.

Ej: ERP corporativos, Zoho CRM, ArcGIS, data warehouses, IBM's Intelligent Operation Center, APIs de predicción meteorológica, bases de datos externas, etc.

MECANISMOS YA INTEGRADOS

Mecanismos accesibles desde el motor SDK:

- Dropbox
- Twitter
- Mailing
- SMS internacionales
- Sockets

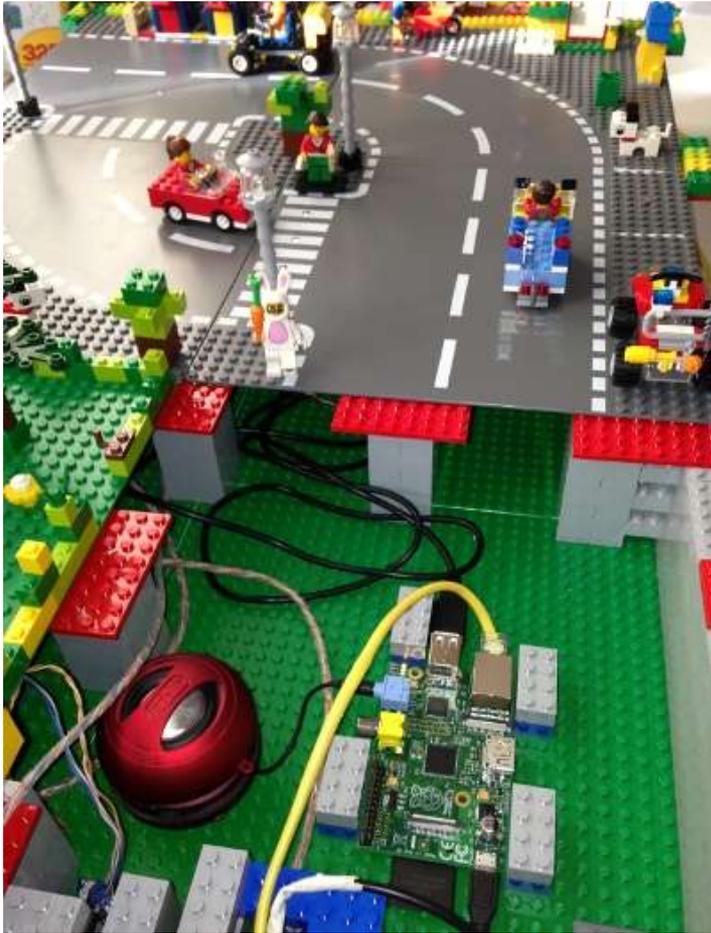


CONSTRUYENDO UN PROYECTO DE IOT

1. INTRODUCCIÓN AL IOT
2. CONSTRUYENDO UN PROYECTO DE IOT
3. DISPOSITIVOS
4. PLATAFORMA
5. FRONTEND
6. INTEGRACIÓN
7. DEMOS

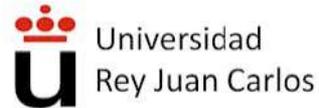
DEMOS

NUESTRA SMART CITY



BIG RED BUTTON





INTRODUCCIÓN AL INTERNET DE LAS COSAS

CONSTRUYENDO UN PROYECTO DE IOT

Alvaro Everlet

alvaro.everlet@carriots.com

@aeverlet

Javier Pastor

javier.pastor@carriots.com

@javipstor

Nov 2013