

# Interoperabilidad para IOT

Integra y Mejora tu plataforma IOT

**Pierre-Yves Duquesnoy**  
Senior Sales Engineer



# AGENDA

<https://github.com/es-comunidad-intersystems/webinar-awsiot-telegram.git>



**IOT Devices**

1

**MQTT**

2



**AWS IOT**

3



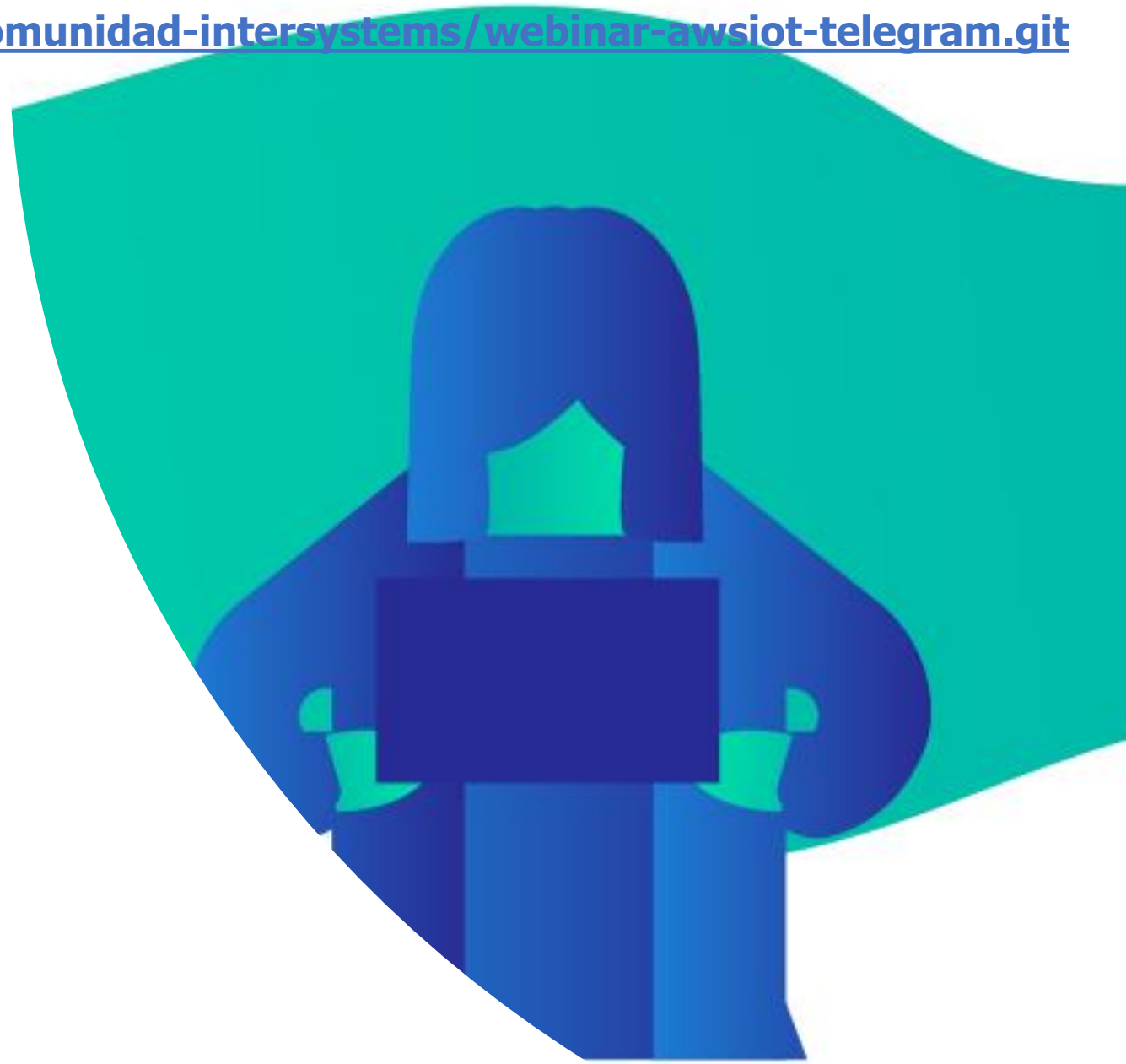
**Telegram**

4

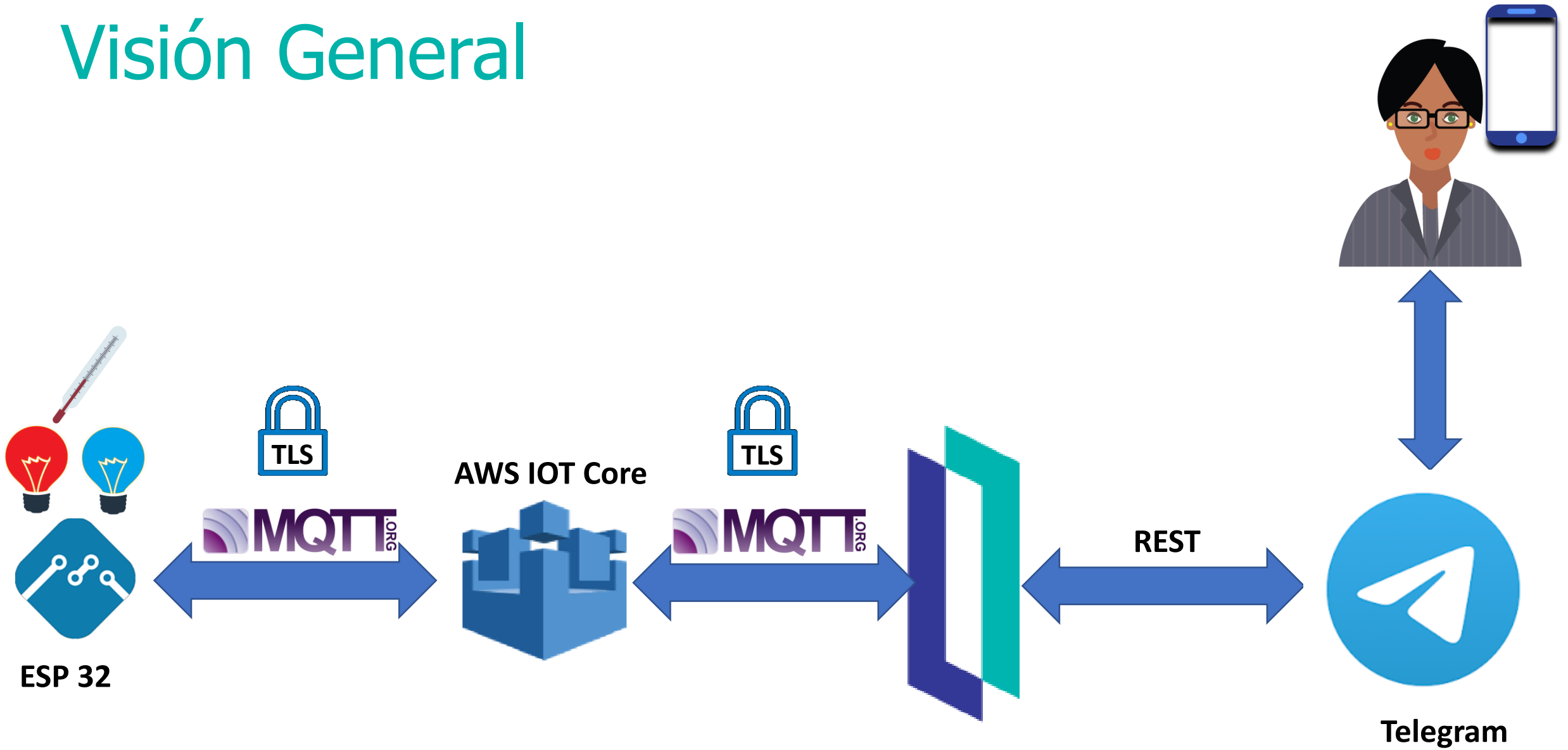


**InterSystems IRIS**

5



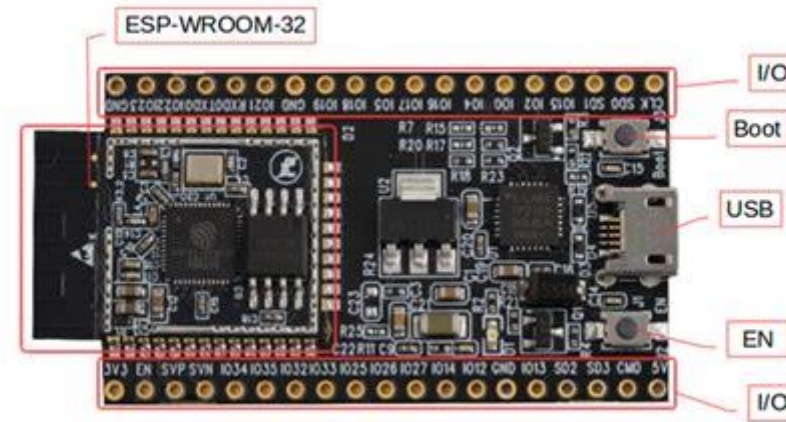
# Visión General



# Dispositivos IOT

## Categorías

- Micro-Controladores
  - Arduino
  - ESP8266
  - ESP32
- CPUs con Linux
  - Raspberry PI



## ESP-32

- Wifi
- BLE
- Dual Core
- 4 GB Ram
- TLS 1.2 Capable
- DevKitC < 10 Euros



# Plataformas Firmware para ESP32

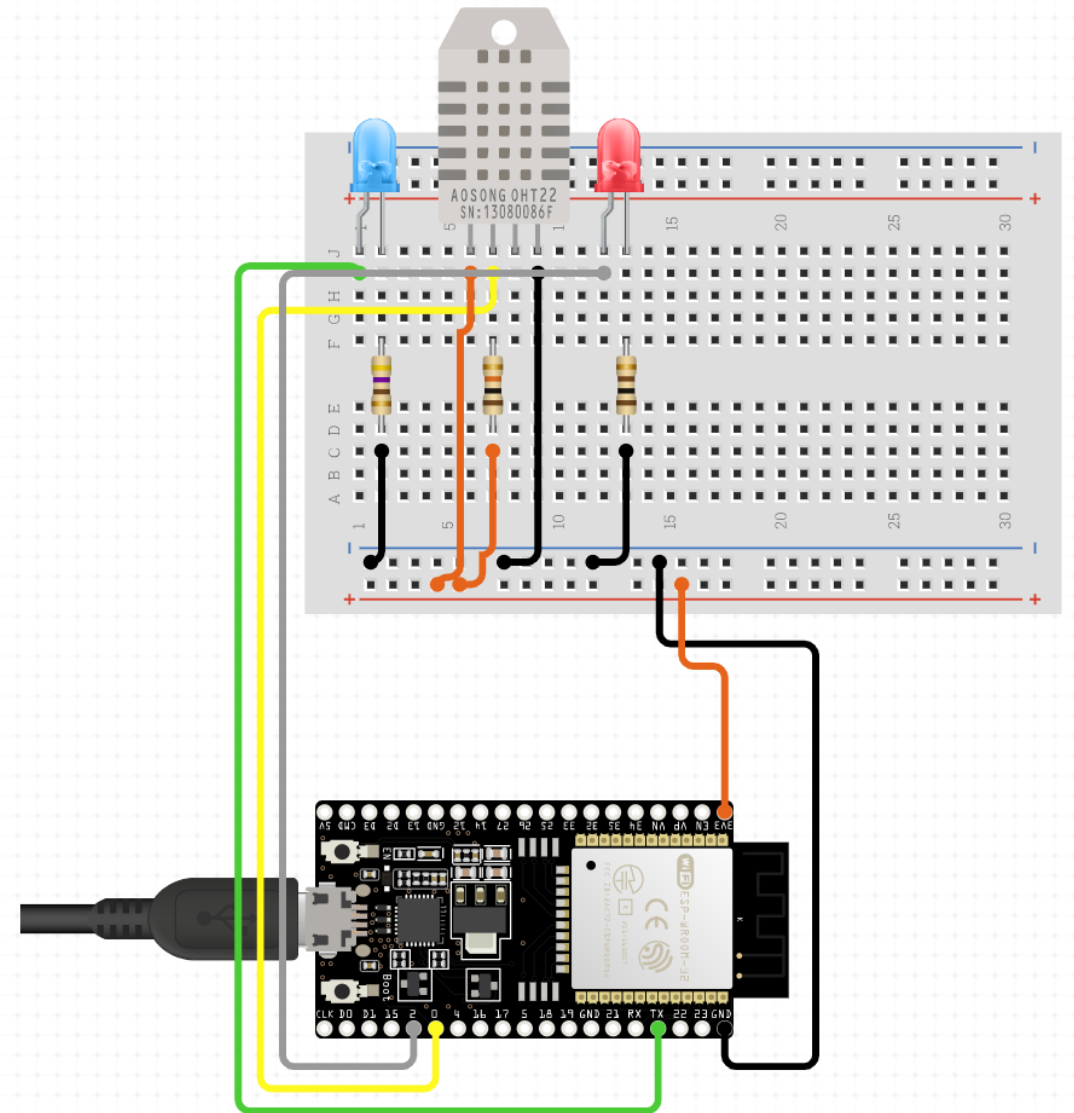
## Requisitos Iniciales

- Conectividad WIFI+MQTT
- Soporte TLS 1.2
- Posibilidad de OTA solido



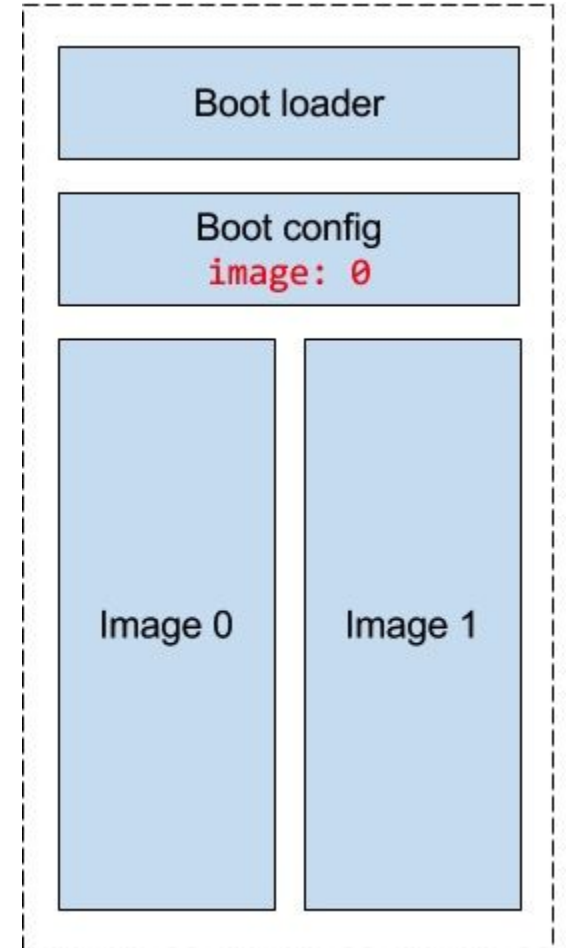
# Esquema del Dispositivo

- ESP-32
- 1 Sensor de Temperatura /Humedad
  - DTH 11(+/- 2 Celcius), conectado a GPIO 5
  - + 10K Ohm
- 1 Led Rojo
  - Conectado a GPIO 2 (+ 100 Ohm)
- 1 Led Azul
  - Conectado a GPIO 4 (+ 100 Ohm)
- Circuito mantenido simple a propósito
  - Sin Dallas DS18B20 sensor
  - Sin Gestión de Energía



# Over the Air Update

- Trigger provoca Descarga de una nueva imagen
- Se hace un reboot con la nueva imagen
  - El sistema pone una alarma
- Nueva Imagen arranca y conecta a la nube: Commit
- Si hay un crash o no conecta
  - Alarma
  - Rollback a la imagen precedente
- **Mongoose + AWS**
  - AWS Shadow sirve de "trigger" para desplegar
  - Imagen se copia desde AWS S3





# Construcción dispositivo IOT

<https://www.circuito.io/>



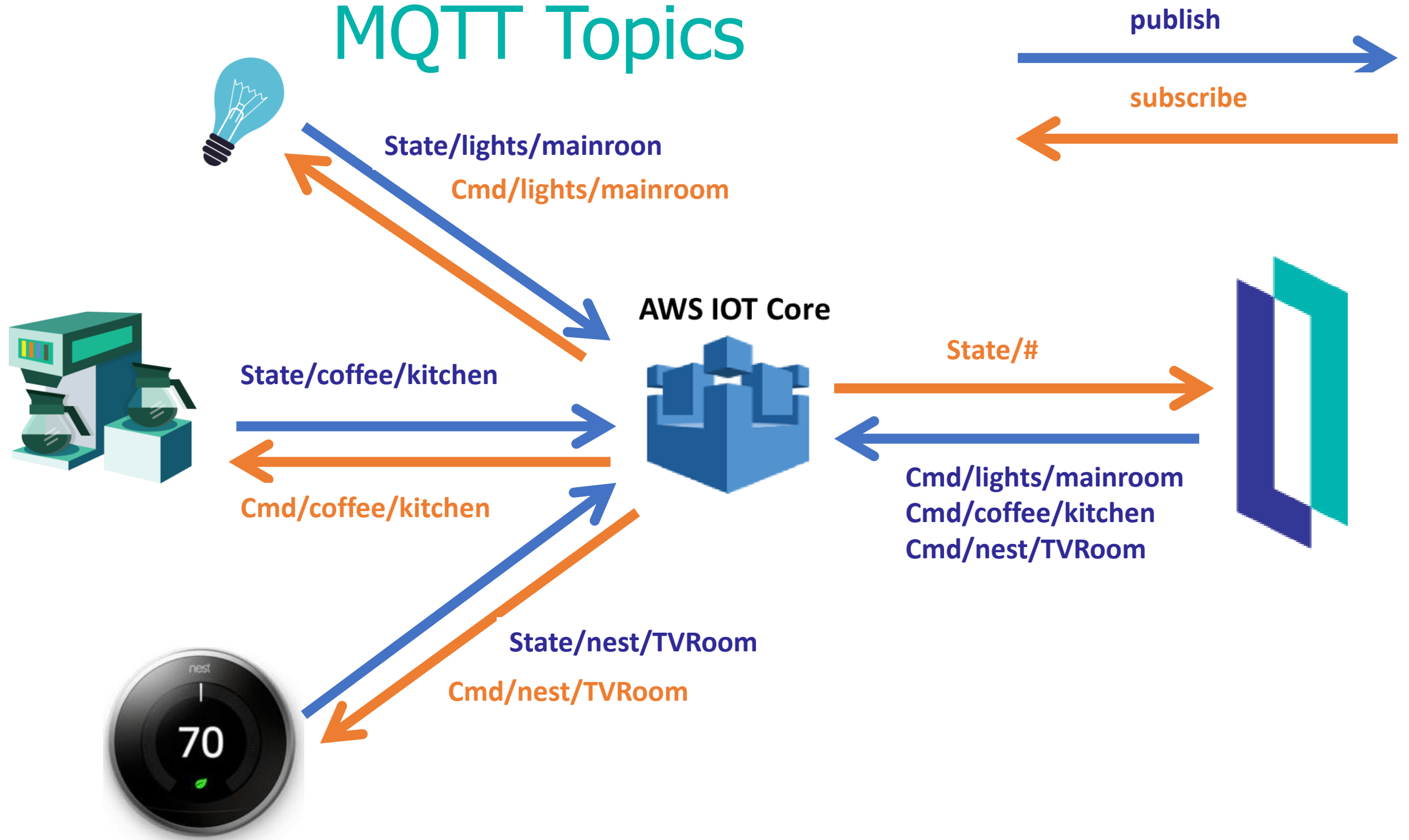


# MQTT

- Message Queuing Telemetry Transport
  - Protocolo ligero, sobre TCP/IP
  - Permite alto flujo de mensajería sobre ancho de banda bajo
  - Muy utilizado en IOT
- 
- Mecanismo Publish/Subscribe
  - Clientes conectan a un Broker
    - Publican a Topics
    - Suscriben a Topics



# MQTT Topics



# MQTT en InterSystems IRIS

- Soporte a MQTT 3.1
- QoS 0 y 1 (at most once, at least once)
- Adaptadores
  - EnsLib.MQTT.Adapter.Inbound (subscribe)
  - EnsLib.MQTT.Adapter.Outbound (publish)
- Componentes pre-construidos (BS, BO)
  - EnsLib.MQTT.Service.Passthrough
  - EnsLib.MQTT.Operation.Passthrough



# Parametrización MQTT en IRIS

- **ClientID** – identifica el cliente al broker
- **Url** – Dirección del Broker
- **Topic** – El topic a cual se publica/subscribe
- **QoS** – 0: Fire and Forget (at most once), 1: Wait for Delivery (at least once)
- **SSL** – Configuración SSL con certificados del bróker (AWS IOT)
- **KeepAlive** – intervalo de keepalive de con broker
- **LWTTopic, LWTMessage** – enviado por bróker en caso de desconexión inesperada

# Implementación MQTT: Device

- PUBLISH: devices/sensors/esp32\_F30B2C

```
{"deviceId": "esp32_F30B2C", "humidity": 29, "temperature": 27,  
  "led_state": {"Blue": "0", "Red": "0"}}
```

- SUBSCRIBE: devices/commands/esp32\_F30B2C

```
{"Blue": "0", "Red": "1"}
```

# Implementación MQTT: InterSystems IRIS

- PUBLISH: devices/sensors/esp32\_F30B2C

```
{"Blue": "0"}
```

- SUBSCRIBE: devices/sensors/#

```
{"deviceId": "esp32_F30B2C", "humidity": 29, "temperature": 27,  
  "led_state": {"Blue": "0", "Red": "0"}}
```

# Funciones MQTT desde Mongoose OS

```
MQTT.sub(TopicSub,  
    function(conn, topic, msg) {  
        print('Topic: ', topic, 'message:', msg);  
        let obj = JSON.parse(msg);
```

```
MQTT.pub(TopicPub,message,QoS,retain);
```



## Implementación en Dispositivo

- mos.Yaml
- Init.js





# AWS IOT

- Plataforma muy completa
  - FreeRTOS, AWS IOT Greengrass
  - Gestión de Flotas de dispositivos (Catalogo, Seguridad, OTA,...)
- Integrado con el Resto de Servicios AWS
  - Seguridad
  - Lambda, S3, DynamoDB, Aurora, SNS, SQS, ...
    - Beneficio pero también plétora de servicios para hacer una integración normal
- AWS IOT Core
  - Broker MQTT con TLS1.2 (en modo SaaS)
  - Solo acepta conexiones autenticadas con certificado
  - Certificado asociado con permisos otorgados

## AWS IOT Core



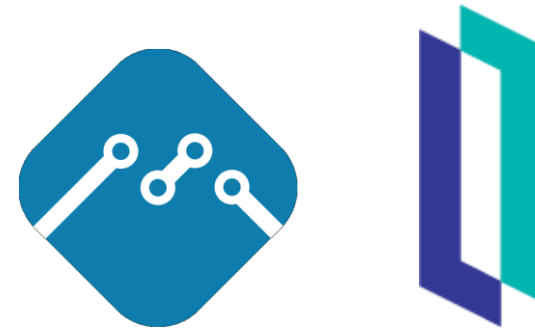
## AWS IOT Core



- Thing Type
  - Agrupación de dispositivos
- Thing
  - Un dispositivo identificado
- Policy
  - privilegios
- Certificados
  - Vinculan Thing <-> Policy



## AWS IOT Core



- CA.pem
- <Device>-certificate.pem
- <Device>-private.pem

# MQTT con AWS IOT Core



# Telegram

- Plataforma de mensajería (como WhatsApp y Signal)
- Bots
  - Agentes que permiten automatizar tareas
    - Envío de Notificaciones a usuarios y grupos
    - Recibir comandos y realizar acciones, responder
  - APIs REST para interacción
    - Muchos ejemplos de BOT disponibles
- Usado por un cliente nuestro para notificaciones internas y externas
  - Desde InterSystems IRIS
    - Notificaciones a grupos sobre eventos específicos
  - Respuestas a comandos



# Telegram Bots

- Creación con @BotFather
  - “One bot to rule them all”
- Comandos
  - /newbot
    - Devuele un token de autenticación
  - /setname
  - ....y mucho más...
- Notificaciones
  - Envío de mensajes a personas o grupos
- Callbacks / WebHooks
  - Para recibir comandos enviados por usuarios



# Telegram Bots



- Notificaciones

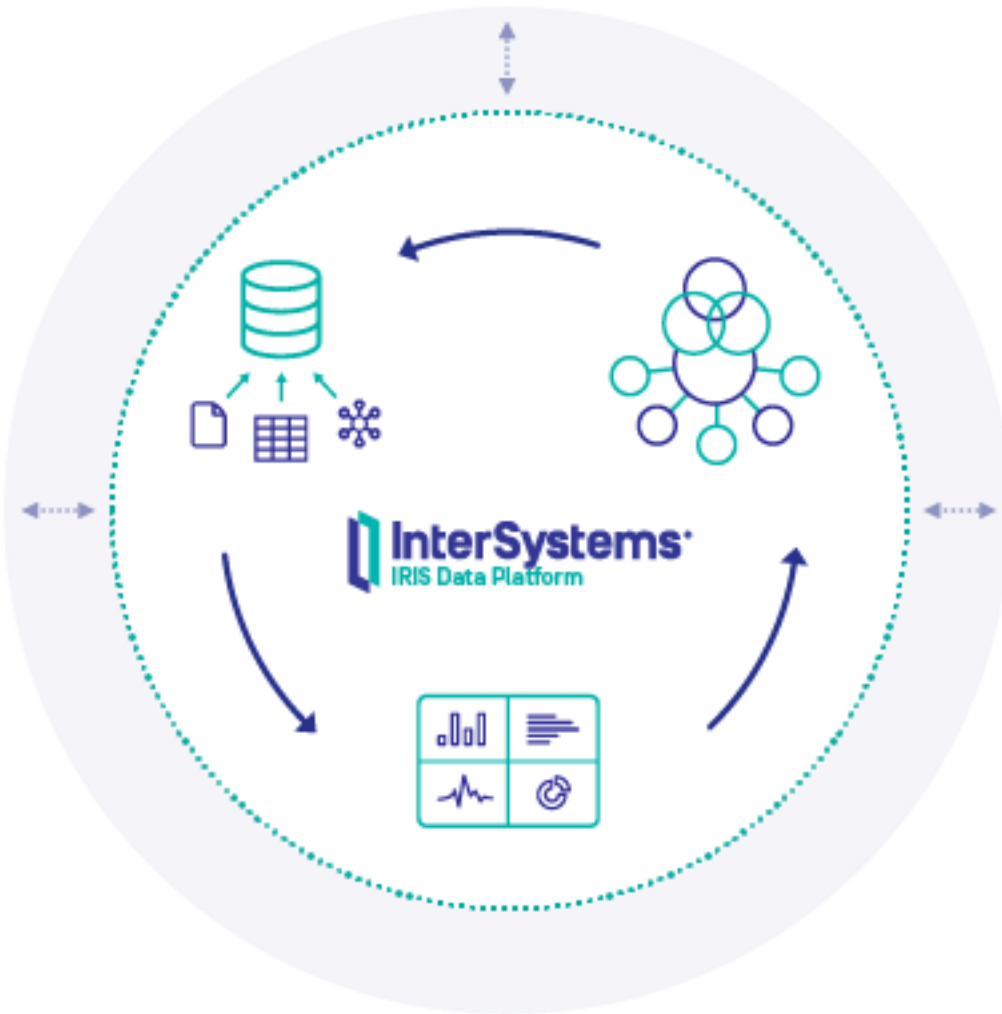
- API Rest+JSON:
- <https://api.telegram.org/<token>/sendMessage>
- Componente cliente IRIS: Business Operation REST

- Callbacks / WebHooks

- Para recibir comandos
- API Rest +JSON: Telegram llama a un Endpoint https que publicamos
- /setcommands + lista de comandos
- Activar Webhook:

```
https://api.telegram.org/bot<token>/setWebHook?url=<IRISHttpsEndPoint>
```

# InterSystems IRIS



- Plataforma de Datos
  - Ingestión masiva
  - Escalable VT y HZ
  - Para entornos críticos
    - Sanidad
    - Mercados financieros
- Interoperabilidad
  - MQTT
  - Rest+ JSON
  - ...Y Muchos más...



**202 secs**

Run Time

**22,701,000**

Records Inserted

**4,869,238**

Records Queried

**5,893.595**

MB Inserted

**1,352.177**

MB Queried

**0.04126**

Average Query Time in Ms

**201 secs**

Run Time

**2,427,000**

Records Inserted

**1,432,194**

Records Queried

**631.867**

MB Inserted

**352.292**

MB Queried

**0.24155**

Average Query Time in Ms

**831% more records ingested****485% faster average query rate**

<https://github.com/intersystems-community/irisdemo-demo-http>



[Learn More](#)

# **InterSystems®**

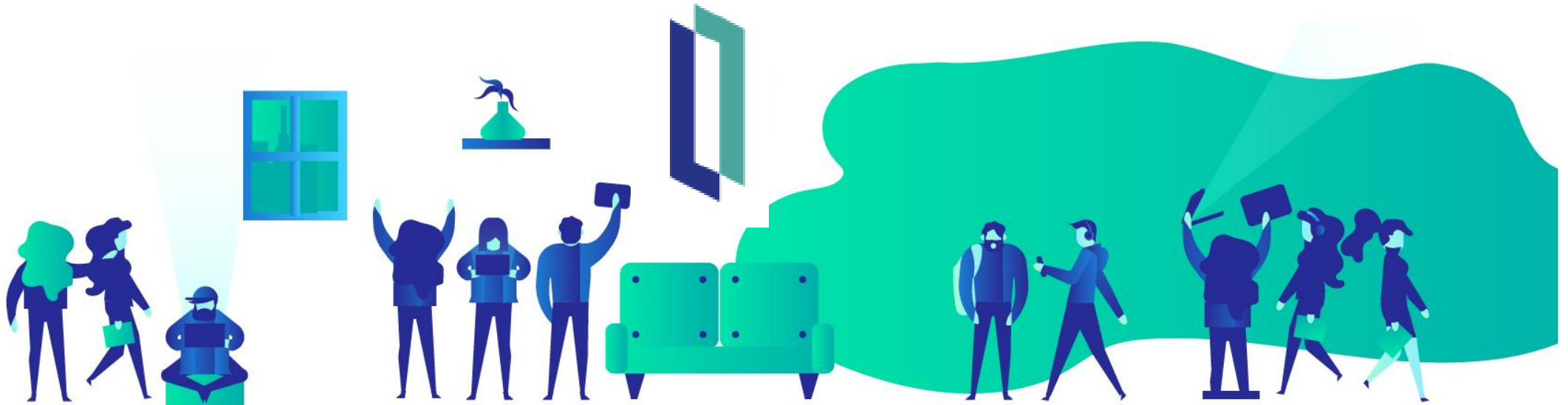
## **VIRTUAL SUMMIT 2020**

**Keynotes:** Oct 20 - 22

**Technical Sessions:** Oct 27 - 29

**Ask the Experts:** Oct 30, Nov 2

**Experience Labs:** Nov 3 - 5



<https://www.intersystems.com/virtual-summit-2020/>



# Circuito Completo

**@IRIS\_MQTT\_bot**

