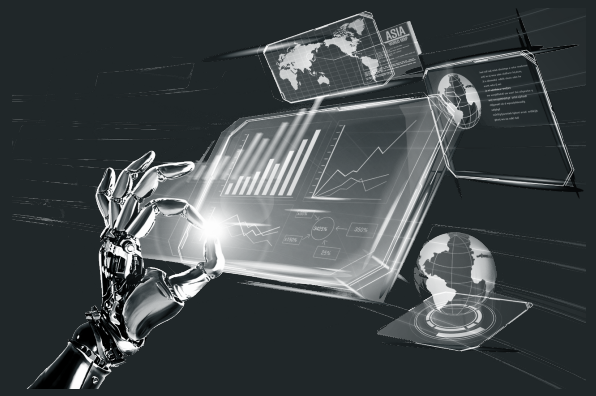


Gestión de Datos para Robótica

T4a - Sistemas de Almacenamiento y Consulta de Datos en Streaming

Álvaro Vázquez Álvarez
Departamento de Electrónica e Computación

alvaro.vazquez@usc.es

Pabellón III - Despacho 4

Curso 2023-2024

Tabla de contenidos

Introducción

Arquitecturas orientadas al dato: surgen de la necesidad de ingerir, procesar y generar valor de las enormes cantidades de datos que llegan a un sistema de información a gran velocidad.

Procesamiento batch (por lotes): orientado a procesar grandes cantidades de datos con un tamaño fijo y determinado y recursos limitados. El cliente tiene que esperar a que termine el procesamiento para disponer de todo el lote de datos.

Procesamiento en tiempo real: proporciona resultados a medida que entran eventos en el sistema, realizando las operaciones de forma inmediata, aunque el cliente no los demande.

Data Streaming: arquitectura orientada al dato, es un sistema en tiempo real (no estricto) que pone los datos disponibles en el momento que una aplicación cliente los necesita.

Casos de uso del procesamiento de datos en streaming:

- Monitorización de sistemas, de redes y de aplicaciones
- Dispositivos Internet of Things (IoT)
- Sistemas de recomendación y optimización de resultados
- Transacciones financieras, detección de fraude y trading
- Seguimiento de usuarios en páginas web y comercio electrónico
- Notificaciones en dispositivos y aplicaciones móviles en tiempo real

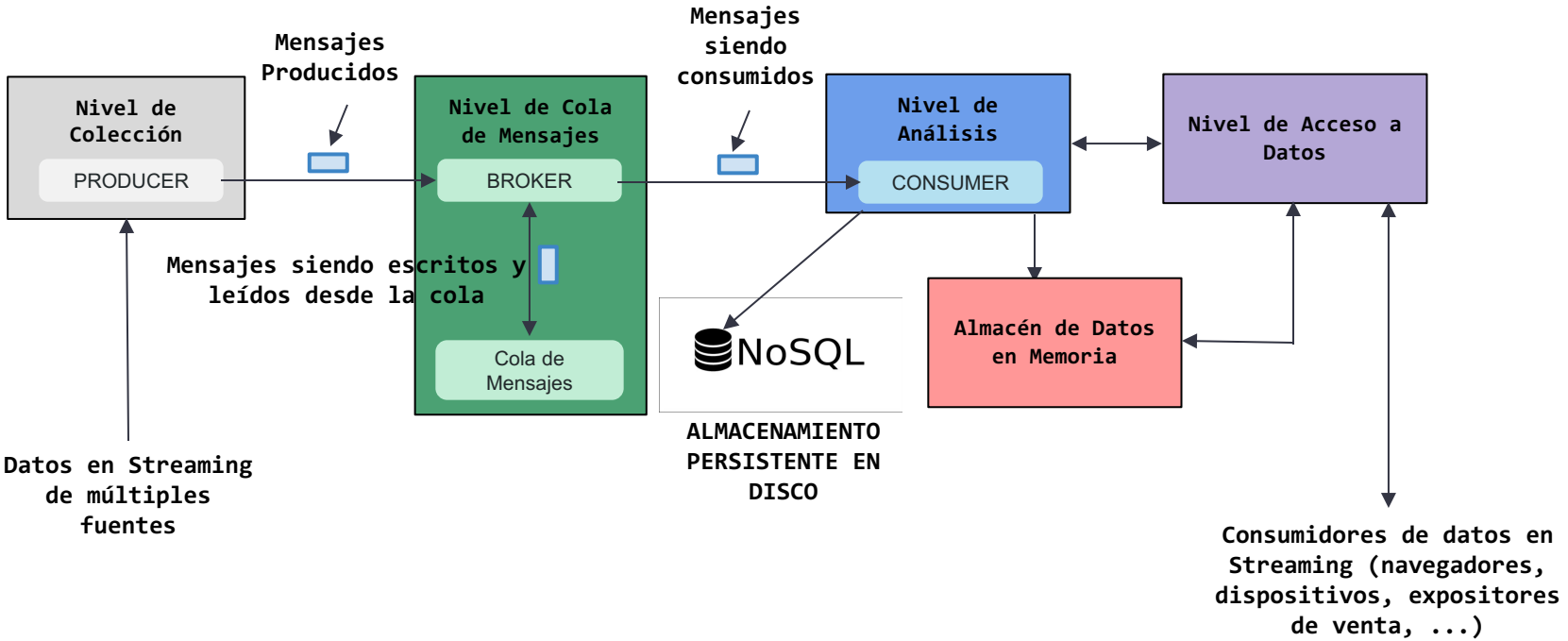
Sistemas Data Streaming

Suelen implementarse como sistemas distribuidos. Constan de distintos niveles o capas que implementan distintas funcionalidades y servicios:

- **Nivel de colección**: implementa los servicios encargados de recoger el gran volumen de datos de los streams que llegan desde multitud dispositivos y sensores (clientes productores). Un ejemplo sería un broker MQTT.
- **Nivel de cola de mensajes**: almacena temporalmente mensajes para su análisis ya que la localización del nivel de colección suele no ser la misma que la del nivel de análisis.
- **Nivel de análisis**: realiza el procesamiento continuo (en streaming) de los datos que llegan de las fuentes (productores) y las consultas de los clientes finales (consumidores).
- **Nivel de almacenamiento persistente**: suele implementarse como BD de tipo NoSQL en disco, optimizada para operaciones lectura/escritura de datos en streaming.
- **Nivel de almacén de datos en memoria**: puesto que las operaciones sobre los datos en streaming deben realizarse con baja latencia se realizan usando estructuras de datos en memoria.
- **Nivel de acceso a datos**: implementa una API con los servicios necesarios para hacer visibles los datos procesados a los usuarios (consumidores).

Además, los niveles de un sistema de procesamiento de datos en Stream pueden estar ubicados en distintas localizaciones.

Sistemas Data Streaming

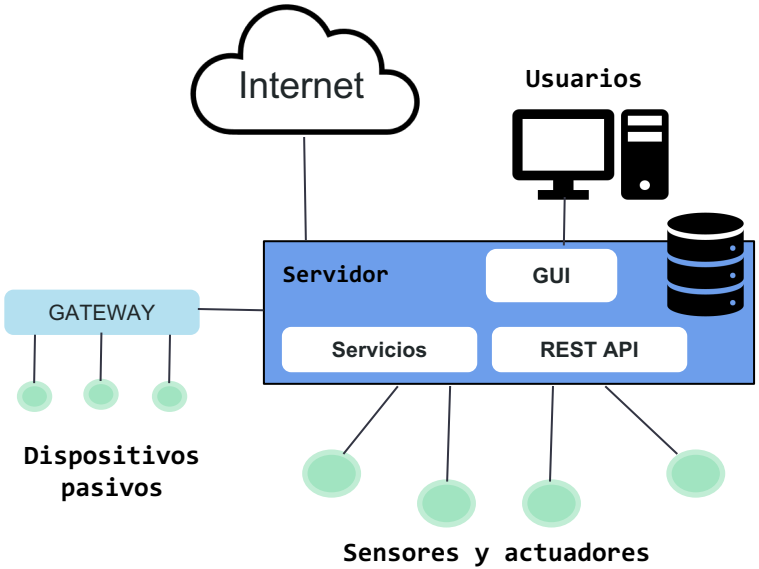


NIVEL DE ACCESO A DATOS

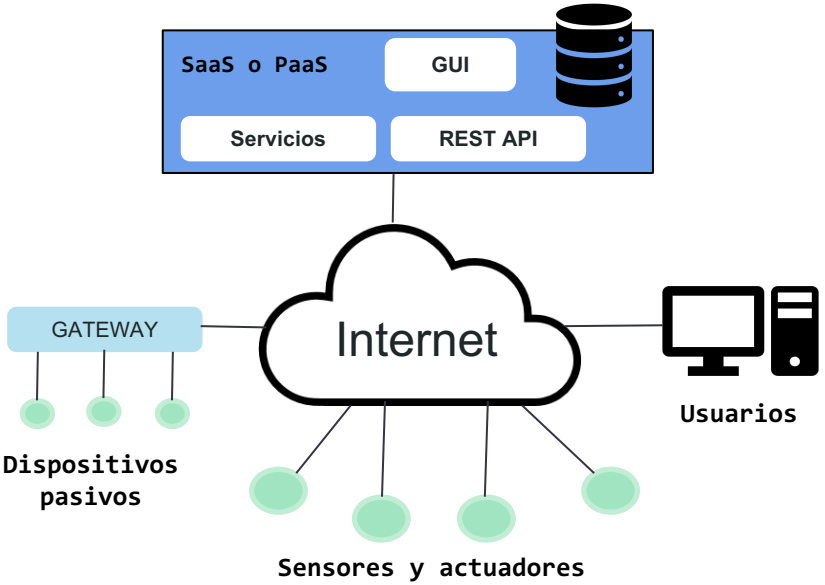
Existen 5 formas básicas (access patterns) de proporcionar a los clientes conexión con la API de acceso a datos (similares a los mecanismos del nivel de la cola de mensajes):

- Mediante método **push/pull** genérico.
- Usando **Data Sync**, mecanismos de sincronización de datos entre la BD con la API y el cliente.
- Con **RMI/RCP (Remote Method Invocation/Remote Procedure Call)**: la API de servidor invoca/llama a un método en un cliente conectado cuando llegan nuevos datos o se cumple una condición indicada por el cliente.
- **Mensajería simple**: el cliente inicia una petición a la streaming API preguntando por los datos más recientes, y la API responde devolviendo los últimos datos disponibles.
- Con un mecanismo de **publicación-suscripción**: el cliente se suscribe a un canal en particular, y la API envía mensajes a todos los clientes suscritos a dicho canal cuando cambian los datos.

Despliegue de Aplicaciones Data Streaming



Despliegue local



Despliegue en la Nube

SGBD en Streaming

Serán de tipo **NoSQL** ***(Not Only SQL)****, con niveles o extensiones adicionales para proporcionar mejor soporte a las operaciones con datos en Streaming.*

Vimos en el Tema 2 que las BD **NoSQL** ***(Not Only SQL)*** son las que mejor se adaptan a las caracteristicas del procesamiento Data Streaming:

- Fijan sus prioridades en la **escalabilidad** y la **disponibilidad**, permitiendo el procesamiento rápido y eficiente de conjuntos de datos dando la mayor importancia al **rendimiento**, la **fiabilidad** y la **agilidad**.

- Permiten comenzar con un modelo sencillo y con el paso del tiempo, añadir nuevos campos, a datos ya existentes, admitiendo la **inserción** de **datos** sin un **esquema** predefinido.

- Además proporcionan soporte para **grandes volúmenes** de **datos estructurados**, **semi-estructurados** y **no estructurados**.

Herramientas para Data Streaming

Se caracterizan por su **alta escalabilidad** y **capacidad** para **procesar grandes cantidades de datos en streaming**.

APACHE KAFKA
Plataforma de streaming de datos de código abierto de la ASF para la recopilación, almacenamiento y procesamiento de datos en tiempo real. Desarrollada originalmente por ingenieros de LinkedIn, usa el modelo **publicación-suscripción**: los datos se envían a un cluster central (productores de datos) y posteriormente se envían a los consumidores de datos (aplicaciones clientes).

CONFLUENT PLATFORM
Plataforma comercial desaroolada por Confluent basada en Apache Kafka, destaca por sus amplias capacidades de conectividad con otras herramientas.

AWS Kinesis
La herramienta de Amazon Web Services para la gestión de datos en tiempo real. Compuesta tres servicios principales: **Kinesis Data Streams** (transmisión y almacenamiento de datos), **Kinesis Data Firehose** (envío de datos en tiempo real), y **Kinesis Data Analytics** (procesamiento de datos en tiempo real usando SQL). Permite la integración con otras herramientas de AWS.

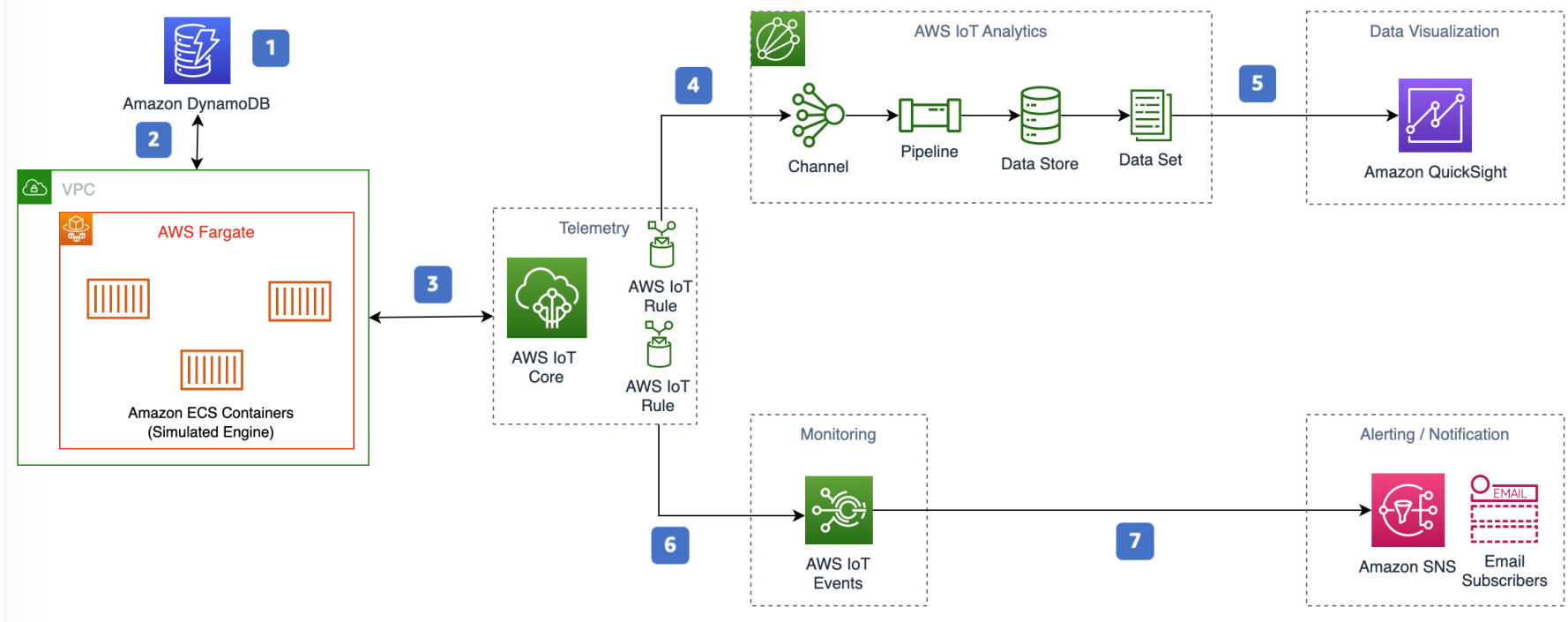
IBM Streams
Plataforma de IBM para procesamiento distribuído de datos en tiempo real. A diferencia de las tres primeras que se basan en sistemas de mensajería de flujo de eventos, IBM Streams implementa de forma nativa el procesamiento de flujo de datos. Permite la integración con otras herramientas de IBM como IBM Watson Studio, IBM Cloud o IBM BigSQL.

Microsoft Azure Stream Analytics
Comparte muchas de las características de AWS Kinesis e IBM Streams, destacando su facilidad de integración con otros productos de Microsoft.

EMQX-HStreamDB
Plataforma de la compañía china EMQ diseñada para aplicaciones de data streaming en el contexto de IoT. Herramienta de mensajeria MQTT (EMQX) acoplada a un SGBD en Streaming altamente escalable (HStreamsDB).

Aplicaciones Data Streaming

Monitorización remota de dispositivos IoT con AWS



Aplicaciones Data Streaming

Monitorización remota de dispositivos IoT con AWS

1. Amazon DynamoDB almacena la información de la simulación y del dispositivo.

2. Las tareas de Amazon ECS se ejecutan en AWS Fargate para simular dispositivos y enviar mensajes.

3. Los simuladores envían los datos de los sensores a AWS IoT Core.

4. Una regla de IoT preconfigurada se suscribe al tema de datos del sensor y envía los datos a AWS IoT Analytics.

5. Amazon QuickSight extrae datos de las vistas materializadas de AWS IoT Analytics para mostrar paneles y visualizaciones.

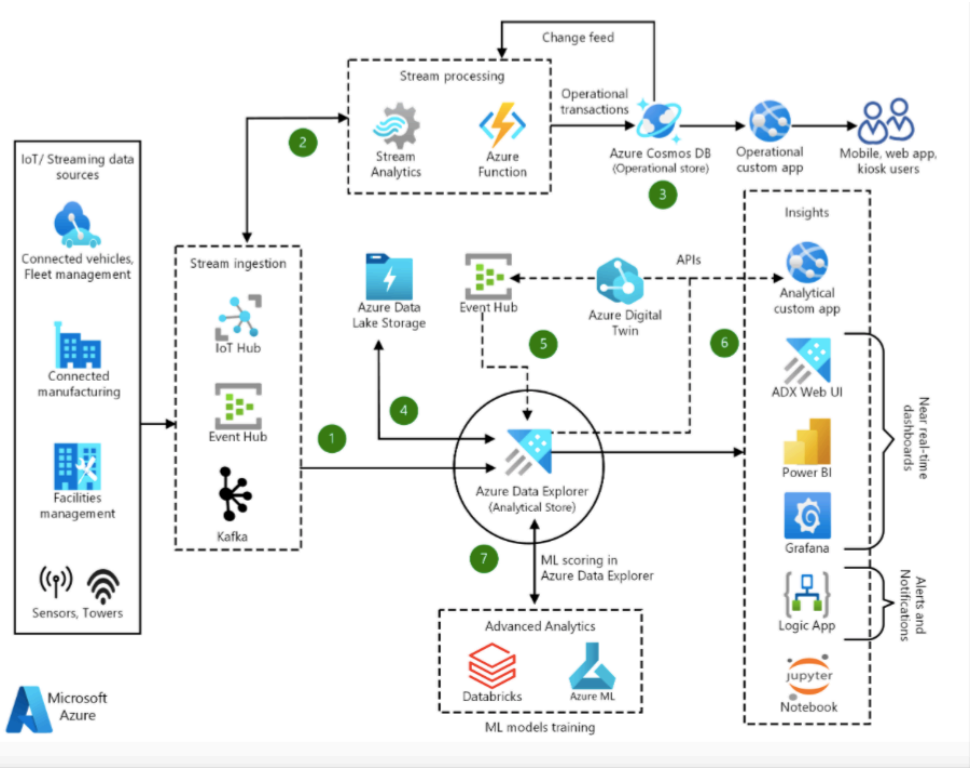
6. Una regla de IoT preconfigurada se suscribe al tema de datos del sensor y envía los datos a AWS IoT Events para detectar si el sensor se encuentra o no en estado de error.

7. Al detectar un estado de error, AWS IoT Events invoca una notificación por correo electrónico a los suscriptores a través de Amazon Simple Notification Service.

Aplicaciones Data Streaming

Análisis de Streaming Data proveniente de dispositivos y sensores IoT con Microsoft Azure

1. Captura de datos de Streaming con Azure Event Hubs, Azure IoT Hub o Apache Kafka.
2. Procesado de los datos casi en tiempo real con Azure Functions o Azure Stream Analytics.
3. Azure Cosmos DB almacena los mensajes transmitidos en formato JSON.
4. Azure Data Explorer ingiere datos de análisis, usando sus conectores de Azure Event Hubs, Azure IoT Hub o Kafka para reducir la latencia y aumentar el rendimiento.
5. Enrutamiento de datos entre aplicaciones.
6. Las interfaces extraen información de los datos almacenados en Azure Data Explorer.
7. Azure Data Explorer se integra con Azure Databricks y Azure Machine Learning para proporcionar servicios de aprendizaje automático (ML).



Aplicaciones Data Streaming

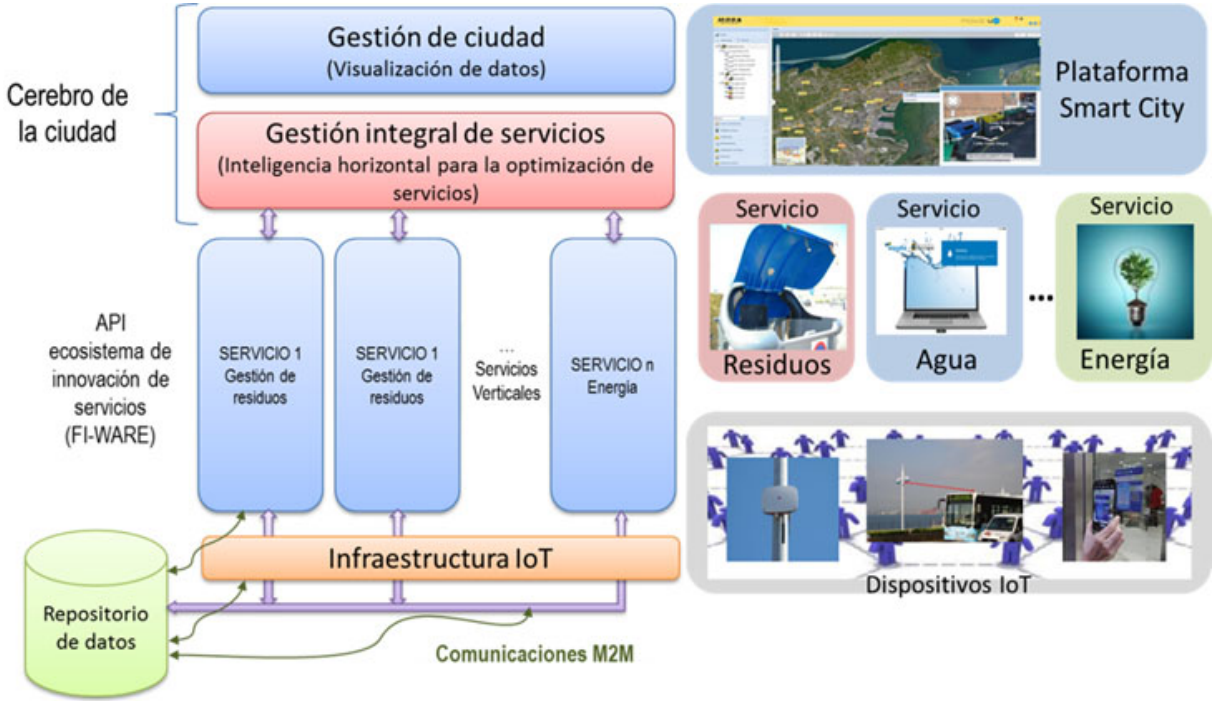
Plataforma Smart City Santander (STDRi)

OBJETIVO

Creación de una instalación experimental para el desarrollo y el estudio de arquitecturas, tecnologías facilitadoras, servicios y aplicaciones para el IoT dentro del entorno urbano de Santander.

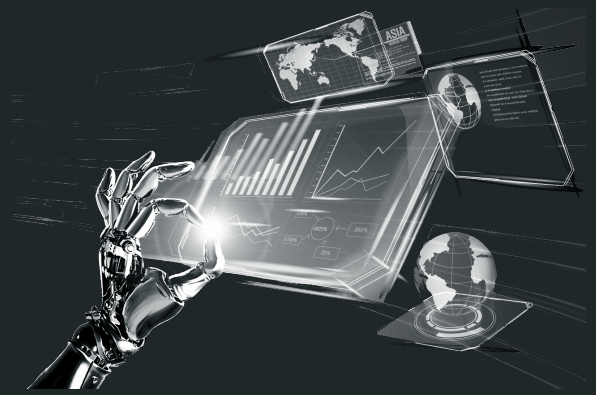
CARACTERÍSTICAS

- Despliegue de más de 12.000 dispositivos IoT.
- Sistema integrado de almacenaje y procesado de todos los datos procedentes de las distintas fuentes.
- Generación de servicios a partir del análisis de los datos.



Bibliografía

- *Streaming Data. Understanding the Real-Time Pipeline (2017)*. A. G. Psaltis.
- *Cloud Computing. Theory and Practice (2013)*. Dan C. Marinescu. Morgan Kaufmann **(cap 8. Storage Systems)**
- Big Data. Principles and Paradigms (2016). R Buyya, et al. Morgan Kaufmann **(cap 2. Real-Time Analitytics & cap 6. Database Techniques for Big Data)**



Gestión de Datos para Robótica

T4a - Sistemas de Almacenamiento y Consulta de Datos en Streaming

Álvaro Vázquez Álvarez
Departamento de Electrónica e Computación

alvaro.vazquez@usc.es

Pabellón III - Despacho 4

Curso 2023-2024