TEMA

TRUSTED
EXTREMELY PRECISE
MAPPING AND PREDICTION
FOR EMERGENCY
MANAGEMENT

Data Storytelling and Big Data value chain in Natural Disaster Management

e-Course on big data analytics for natural disaster management
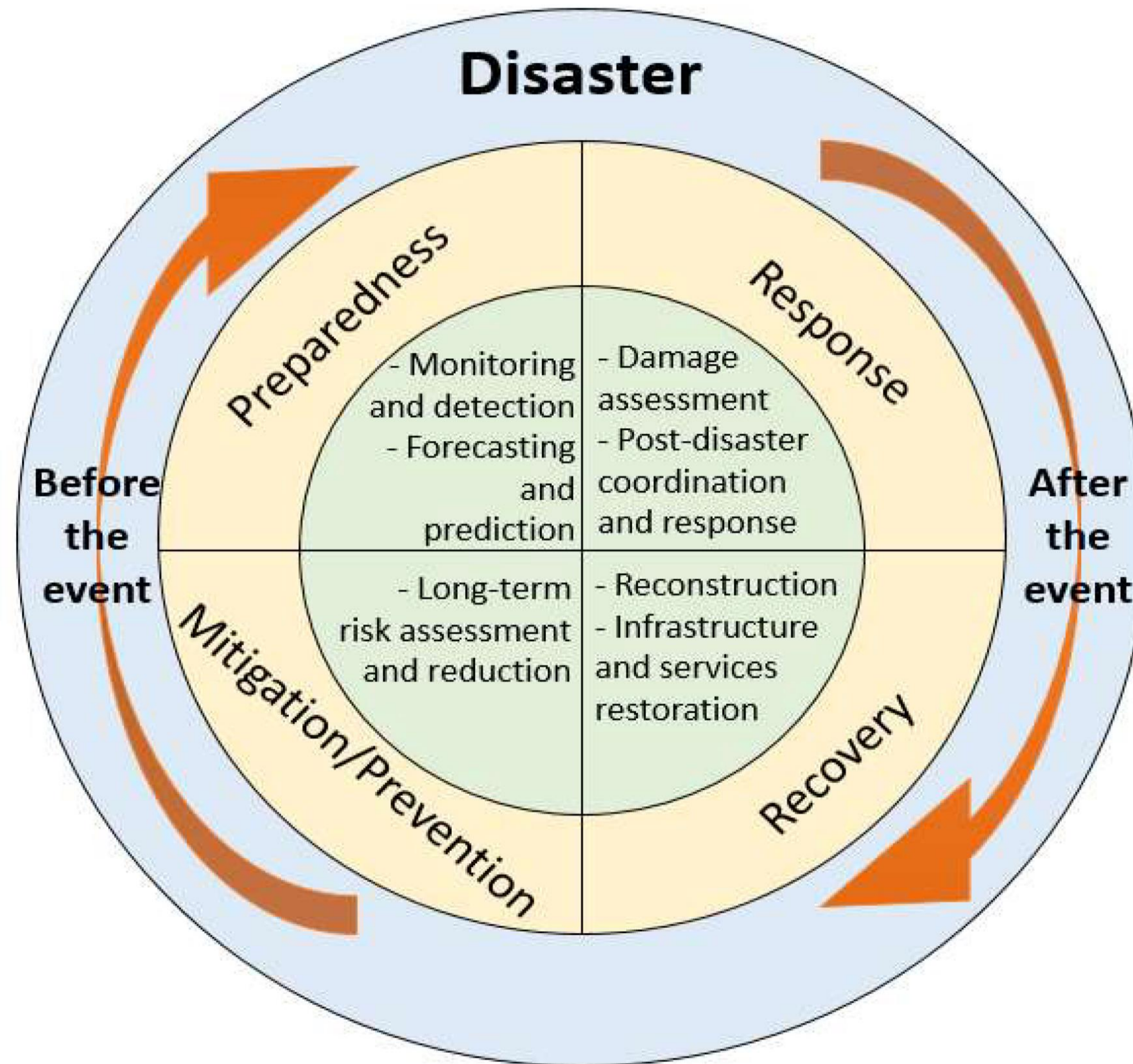
FRANCESCA D'AGRESTI
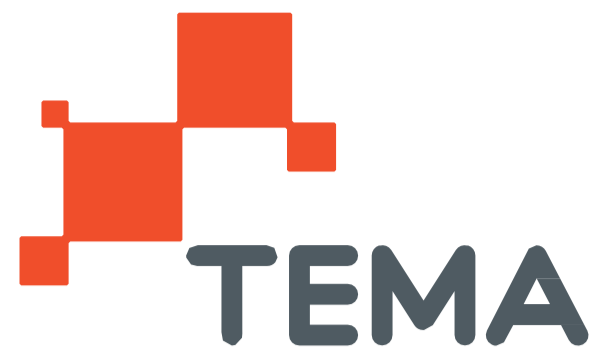
Engineering Ingegneria informatica S.p.A.

francesca.dagresti@eng.it

What Is Big Data?

# Big data in natural disaster management

Big data in natural disaster management

The number of weather, climate, and water extremes are increasing as a **result of climate change**

While the number of disasters has increased by a factor of five, **deaths** from disasters have **decreased** by a factor of three due to an **increase in early warnings** and an **improvement in disaster management**

How Big Data can help in emergency situations?

Crisis mapping
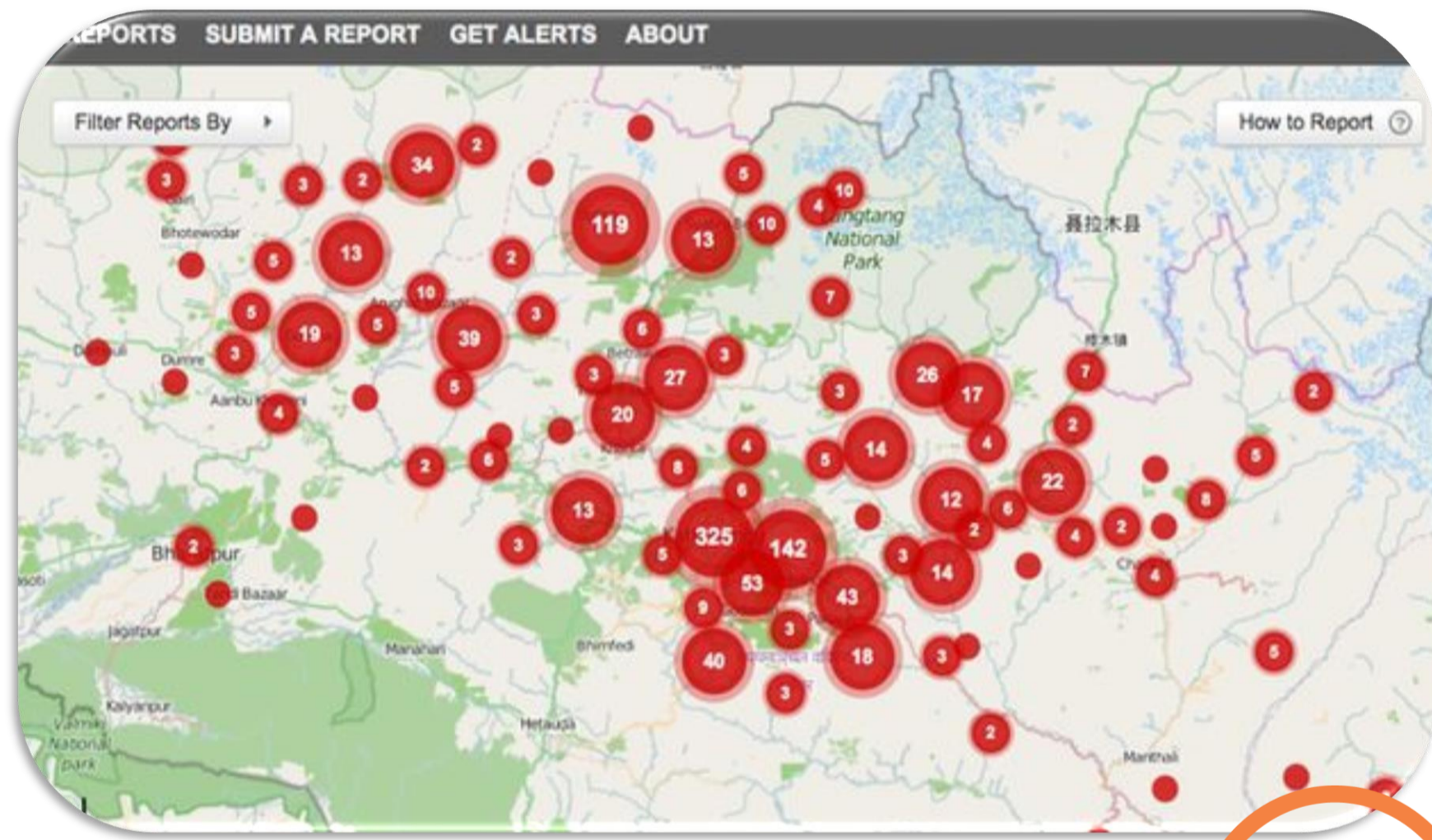
Be prepared

Stay connected

Big data & data storytelling approach

Collection of user needs
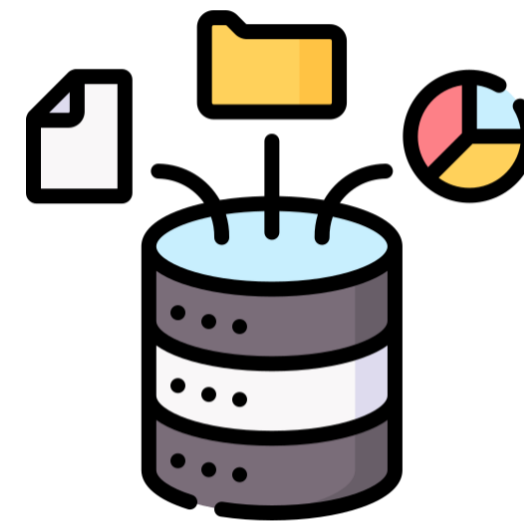
Creation of a Data Catalogue
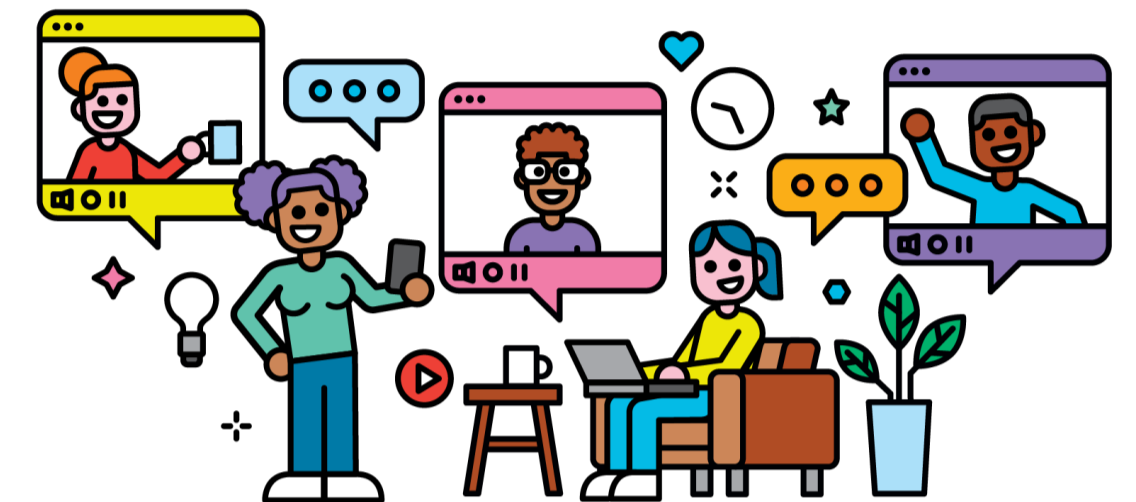
Data from co-creation activities
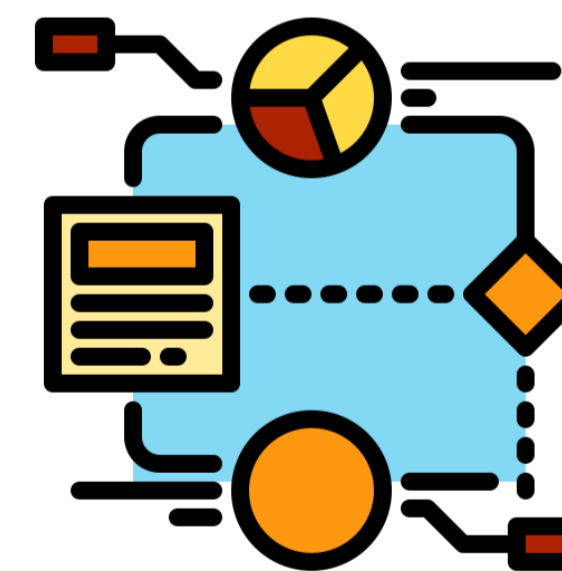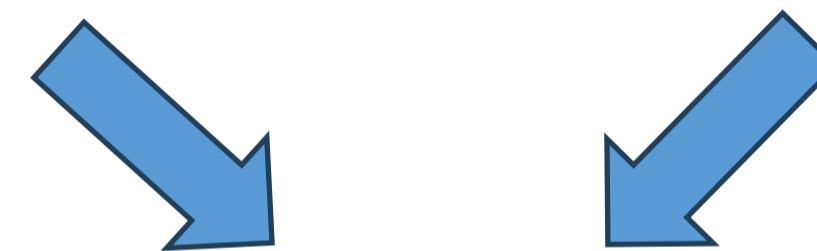
Implementation of algorithms to analyse those data

Visualisation of Dashboards to take decision

Technologies in Big data for Natural Disaster Management

TEMA
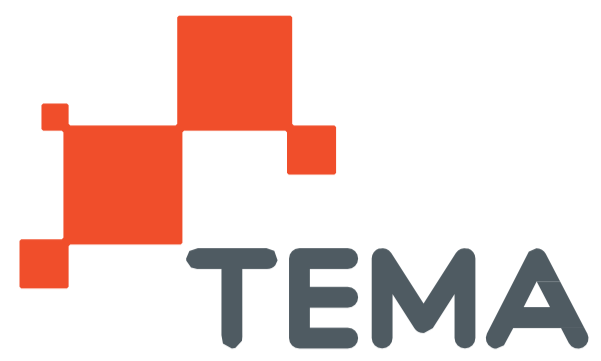
SATELLITE IMAGES
SAVE LIVES FROM SPACE

TEMA
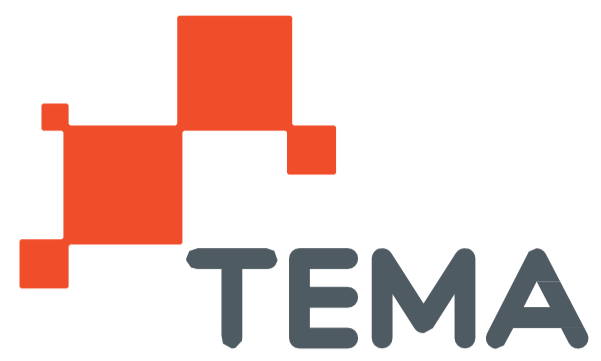
DRONES
SURVEY THE SCENE

Technologies in Big data for Natural Disaster Management

TEMA

SOCIAL MEDIA
TURN ON THE LIGHT

Technologies in Big data for Natural Disaster Management

SENSORS
SOUND EARLY ALARMS
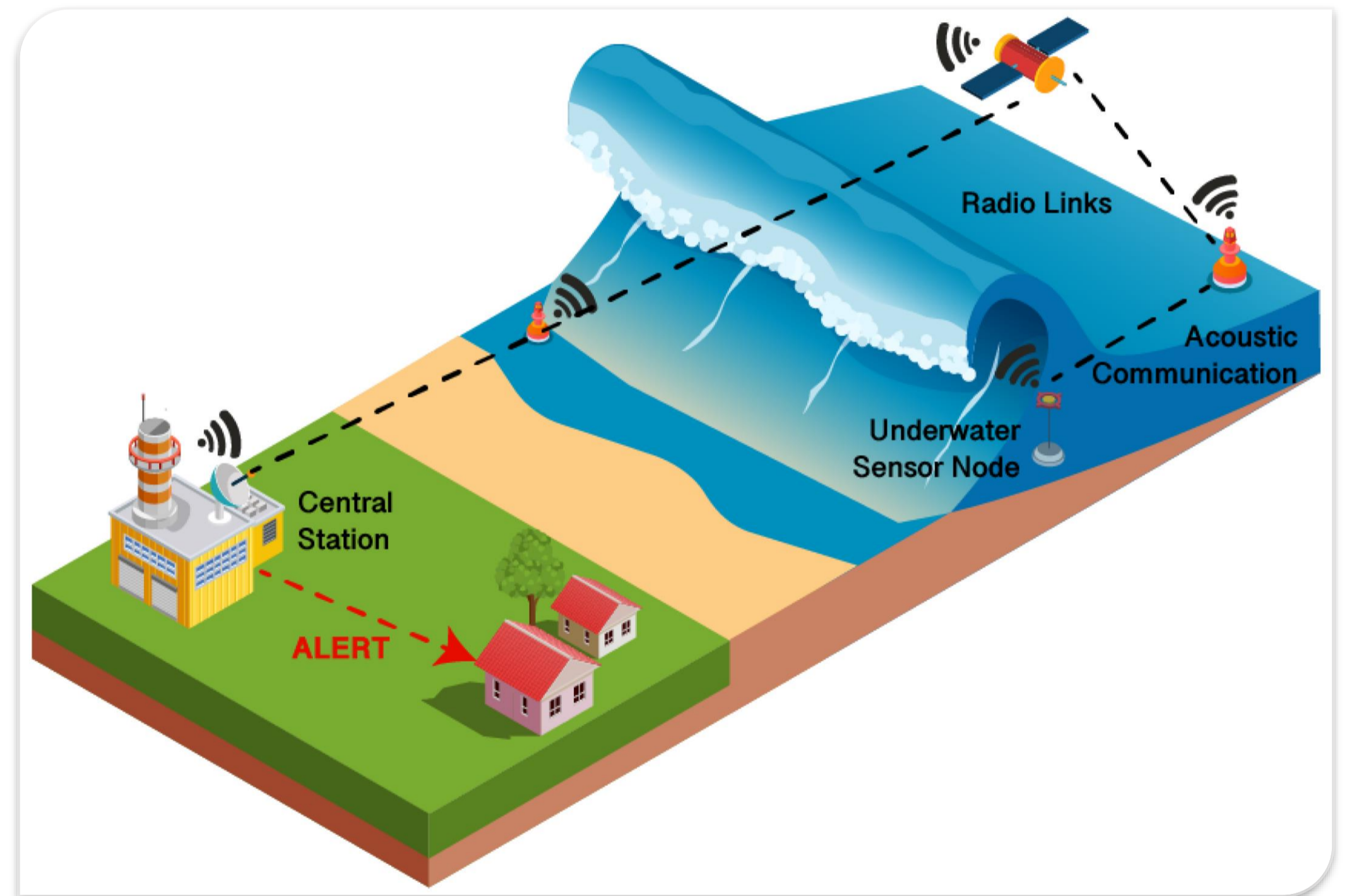
TEMA
A Research project on Big Data

PROJECT NAME:

TEMA
Trusted Extremely Precise Mapping and Prediction for Emergency Management

CALL:

HORIZON-CL4-2022-DATA-01-01
Methods for exploiting data and knowledge for extremely precise outcomes (analysis, prediction, decision support), reducing complexity and presenting insights in understandable way (RIA)

Project Consortium:

**19 partners
from 8 European countries**,
key players in the fields of data analysis, AI, modelling, drone technologies, simulation and visualization, analytics and cloud computing, as well as policy counselling

and emergency response authorities/public bodies.

Project Duration:

48 months.
Starting date: 1 December 2022

Budget:

€ 11.340.223,50

TEMA

The Vision

TEMA aims to develop beyond-state-of-the-art technologies for facilitating Natural Disaster Management, by dynamically exploiting **data sources and Artificial Intelligence technologies** in order to provide an accurate assessment of an evolving crisis situation.

The **goal** is to deliver a technical solution that is supportive in disaster response and management by **bringing situational data** to relevant end-users, enabling transferability to tackle different disaster types in various geographic regions, thus providing the **relevant information** that can help make the best possible **operative decisions**.

Digital Experience: Digital Enabler

Transforming Data into Knowledge through a data "ecosystem" platform

DE is a Data Management ecosystem platform that enables new data economy business models leveraging a complete suite of accelerators

Digital Enabler
Engineering's Ecosystem Platform

Main features

Cloud native **scalable** ecosystem platform that can be **easily composed (composable)**

Independent and interoperable tools allow to accelerate the development of **data oriented vertical applications**

Based on **opensource** software, it complies with **data interoperability standards** and with the GDPR

Main advantages

Industry, domain & technology independent, deployable everywhere, on-prem, on public/private cloud, on hybrid clouds, **no lock-ins**

Usability and flexibility for **reducing time to market** and improving responsiveness to business demands

Interoperability among third party systems and increasing value of existing assets by promoting **new business solutions**

OUR ACCELERATORS

Data discovery

Data collection from heterogeneous sources

Low/No code data integration

IoT and Edge management

Rule Engine, Adv. analytics, AI, Serverless

Digital interaction

DE leverages a suite of 'accelerators' to gather and gain data insight to drive decisions and monitor outcomes

Digital Enabler: functional Platform Landscape

Edge/Near Edge (IoT)

Distributed Endpoint

Edge

Gateway

End user Interfaces	Identity Manager
Analytics & AI	Device Security
Data Integration	
Device Management	Privacy mgmt
Data Source connection	Data Storage

Edge Computing

Cloud / On Prem Data Platform (core)

CDK | API Gateway | Dashboard Development

APPLICATION ENABLEMENT

AI/ML Studio | Real Time Analytics | KPIs Mngmt. | Early Warning & Anomaly Detection

ANALYTICS & AI

Data Mashup | Workflows | Rules and events processing | Serverless functions | Real Time Dispatcher

DATA INTEGRATION

Device Config. & Modeling | Device Data Mngmt. | IoT Edge gateway

DEVICE MANAGEMENT

Data Sources Federation | Data sources Discovery | Storage provisioning

DATA MANAGEMENT

Standard protocols connectors | IoT connectors | Legacy connectors | Third party data

DATA SOURCE CONNECTION

Digital Twin | AR/VR /Metaverse

DIGITAL INTERACTIONS

Access control | Identity mgmt. | Tenant mgmt. | Privacy mgmt. (GDPR)

Audit & Logging | Hybrid multicloud orchestration

OPERATIONAL & SECURITY SERVICES

Time Series DB | SQL DB | NoSQL DB | Big data ecosystem

Column DB | Graph DB | Object storage | Open Data

DATA STORAGE

kubernetes   **OPENSHIFT**

End users

Business Analysts

Third Party software

M2M & IoT

Application Developers

Data Scientists

Data Engineers

© Engineering

Components adopted in TEMA to support Natural Disasters management

Digital Enabler

Storage

Context Broker

FIWARE
Open APIs for Open Minds

All Tech Subscriber

All Tech Publisher

05: get Digital Object

04: push metadata

01: subscribe for updates

02: store Digital Object

03: update metadata

FIWARE Platform Architecture overview

Process / Analyze / Monitor

notify / query

Update

FIWARE Context Broker

Street
- location
- traffic
- pollution

Truck
- location
- route
- next stop
- time next stop

Waste Bin
- location
- capacity

Citizen claim
- location
- citizen id
- description

capture

actuate

Context (Real World)

FIWARE
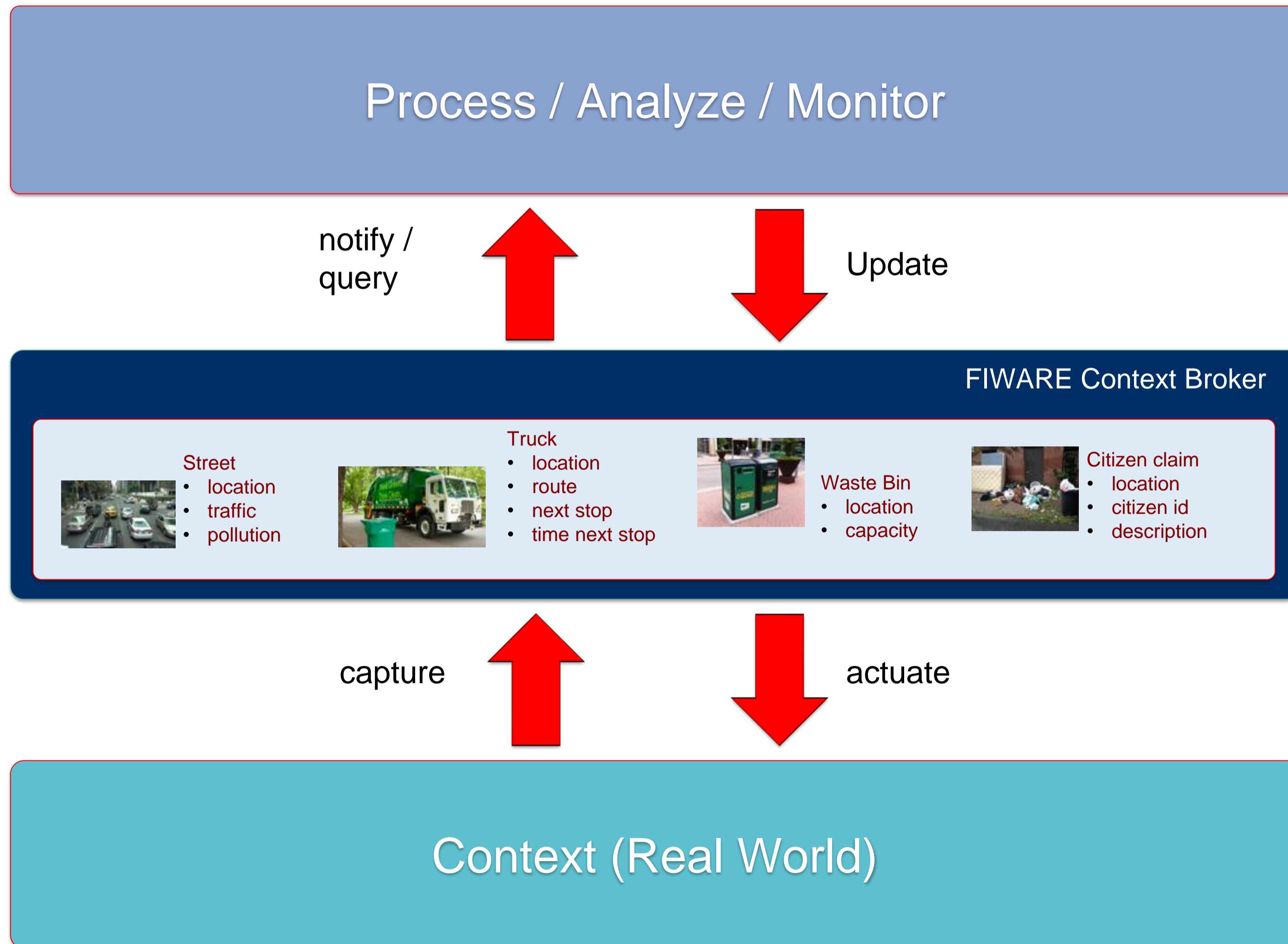
ETSI NGSI-LD: A standard API for Context Information Management

- The ETSI NGSI-LD API is a simple yet powerful public, royalty-free standard API for Context Information Management

- **Simple**: A RESTful API which any web programmer learns how to use in one day

- Yet **powerful**: It supports geo-queries, Linked Data (JSON-LD), subscription/notification, ...

Application/Service

Context Information

Pallet
- Id
- product
- Items quantity
- layers
- Size
- weight

Transport robot
- Id
- location
- speed
- transported items
- destination

Operator
- Id
- Location
- Assigned task
- Profile

FIWARE
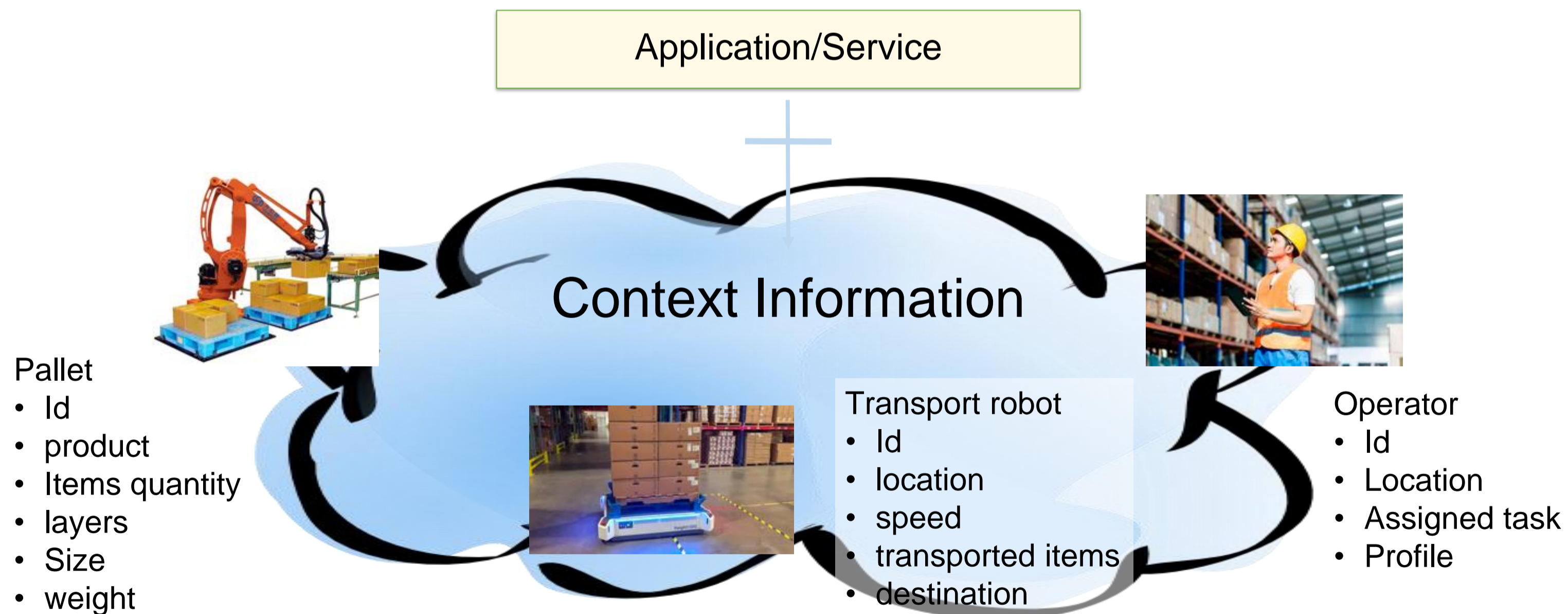
Context Broker

- NGSI-LD API servers are usually referred as Context Brokers

- A Context Broker is associated to a transport end point

- A Context Broker does not necessarily hold the data you are looking for but "knows" how that data can be obtained. Strictly speaking, they provide access to data (using the NGSI-LD API)

Application/Service

ETSI NGSI-LD API

Context Information

Pallet
- Id
- product
- Items quantity
- layers
- Size
- weight

Transport robot
- Id
- location
- speed
- transported items
- destination

Operator
- Id
- Location
- Assigned task
- Profile

FIWARE
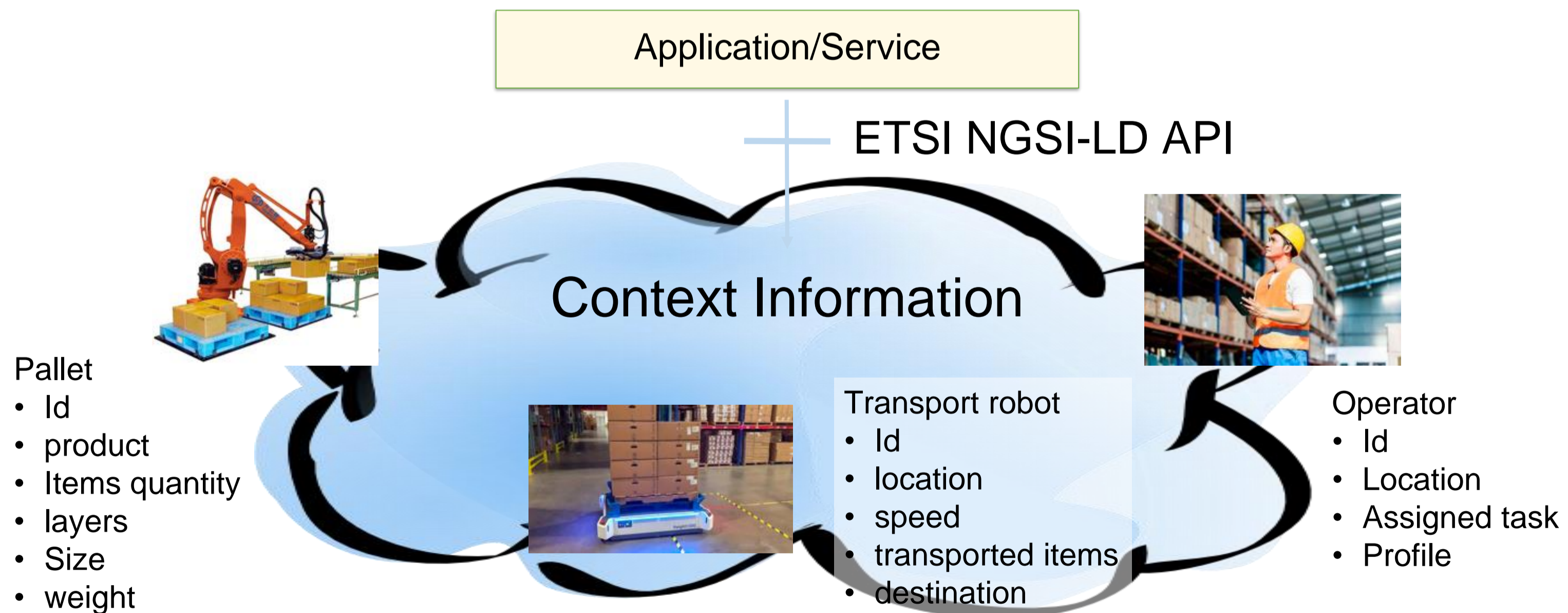
Data Exchange API: ETSI NGSI-LD

Application/Service

FIWARE NGSI API
(NGSIv2 → NGSI-LD)

Context Broker

Ambulance
- license plate
- location
- speed
- equipment
- current alert

Alert
- alert id
- emergency level
- description
- location
- patient id

Surgery Room
- id
- hospital
- equipment
- surgery plan

Surgery Room
- id
- hospital
- equipment
- surgery plan

Patient
- id
- location
- patient record
- treatment

Doctor
- id
- specialties
- location
- visits plan

Entities (Digital Twins)

Hospital
- name
- address
- total patient rooms
- occupied patient rooms
- total surgery rooms

Attribute

Ambulance
- License plate
- location
- speed
- equipment
- current alert
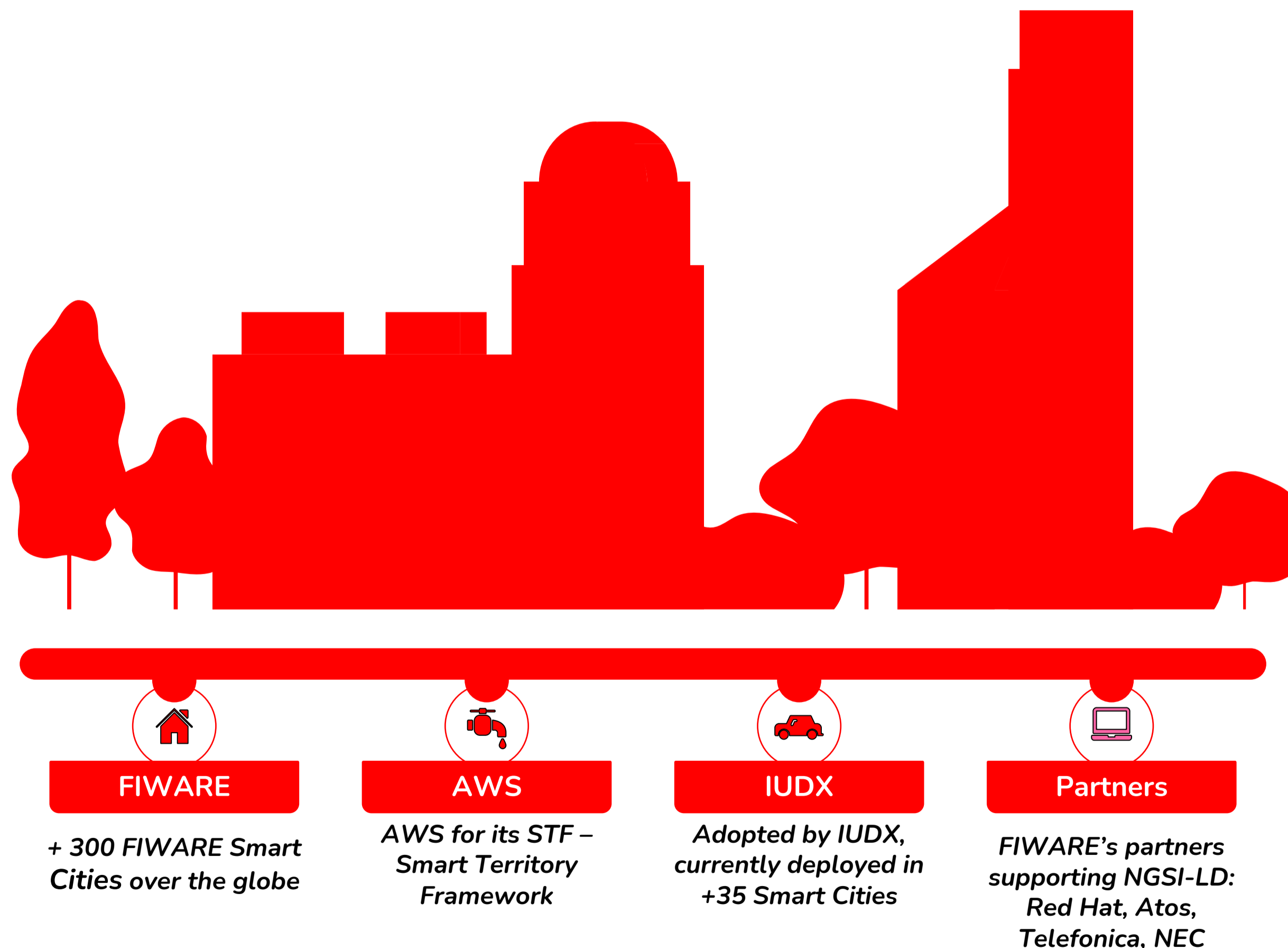
FIWARE

NSGI-LD

Next Generation Service Interface-Linked Data (NGSI-LD) is an open **standard** for context information management developed to facilitate the exchange of information between applications in the context of the Internet of Things (IoT).

FIWARE	AWS	IUDX	Partners
+ 300 FIWARE Smart Cities over the globe	*AWS for its STF – Smart Territory Framework*	*Adopted by IUDX, currently deployed in +35 Smart Cities*	*FIWARE's partners supporting NGSI-LD: Red Hat, Atos, Telefonica, NEC*

JSON-LD

- **JSON** is the data exchange format. However, it is not so easy for machines to read. The attributes of a JSON can have different meanings even among humans themselves.

- For example, consider a JSON that has "*name*" as an attribute. For a person that attribute could mean the person's native name. For another, it could mean the person's stage name. For yet another the username of the person.

- What has been done is to define a **JSON extension** in which it is possible to **annotate** a piece of JSON with **additional information**.

- Instead of having the *name* attribute **without context (JSON)** we can make that **attribute a URI** rather than a simple key-value pair. In this way, via the URI that name attribute actually represents the person's real name: **easier for people and machines to understand it**.

NGSI-LD entity's **type** serves as a representation of the entity's **class or category**.

In NGSI-LD, the **attributes** of an entity represent the **characteristics** or **properties** of the entity. In adherence to the Entity-Attribute-Value (EAV) model, an entity in NGSI-LD consists of:
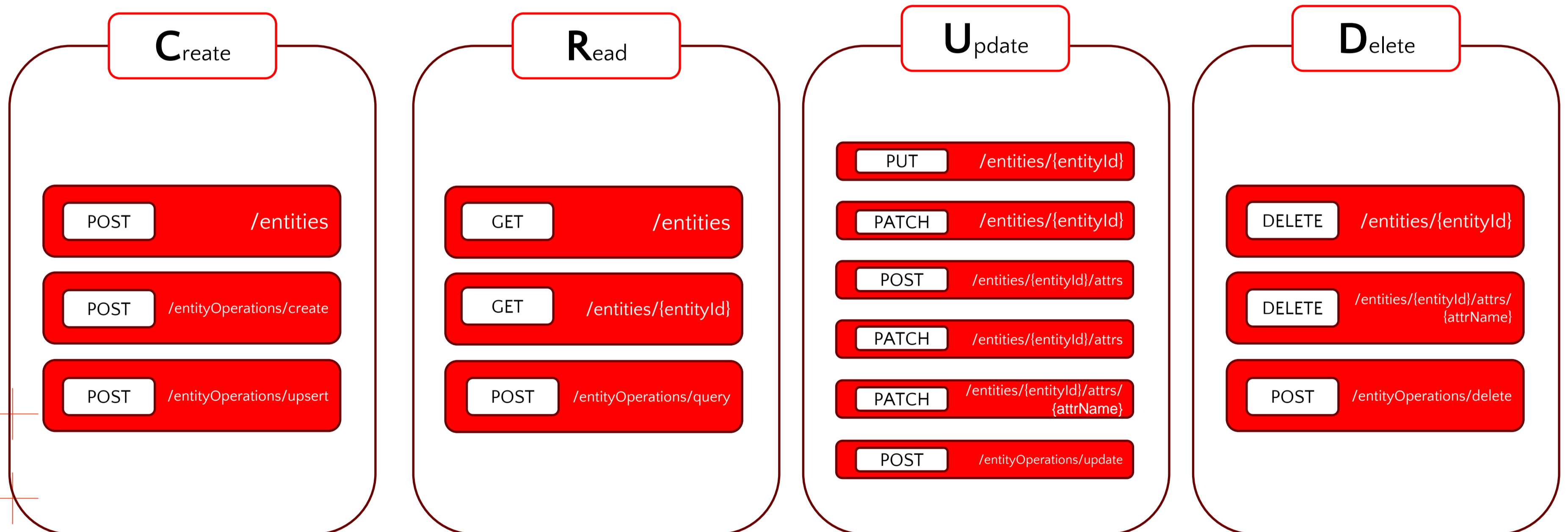
- Entity: A tangible object in the physical world, such as a car or weather station, capable of being defined and tracked.
- **Attribute**: Descriptors that define an entity's specific characteristics.
- **Value**: The current measurement or state of an attribute.
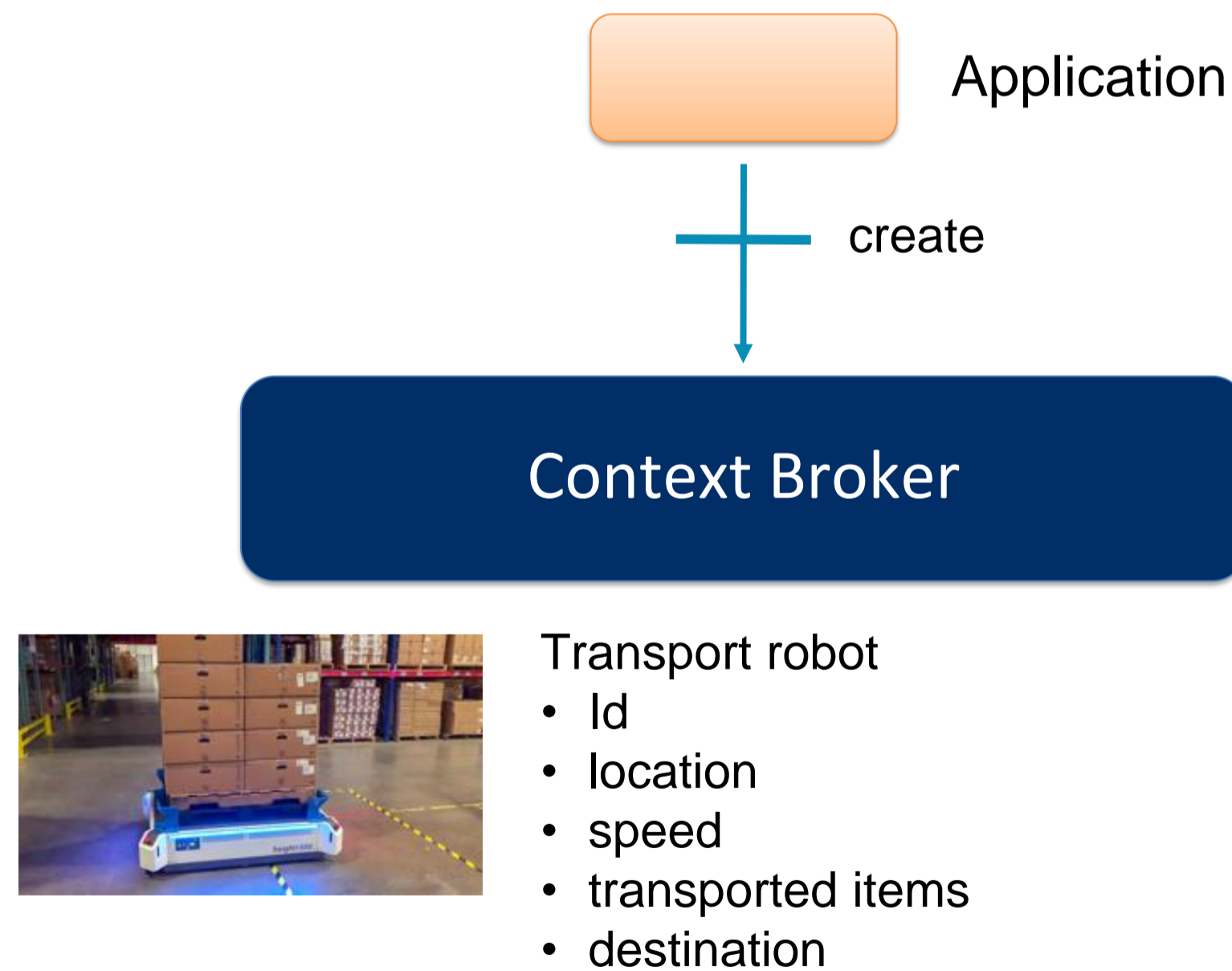
tenant

Entity — **has** — Attribute

1 n

id   type   scope

type   name   value
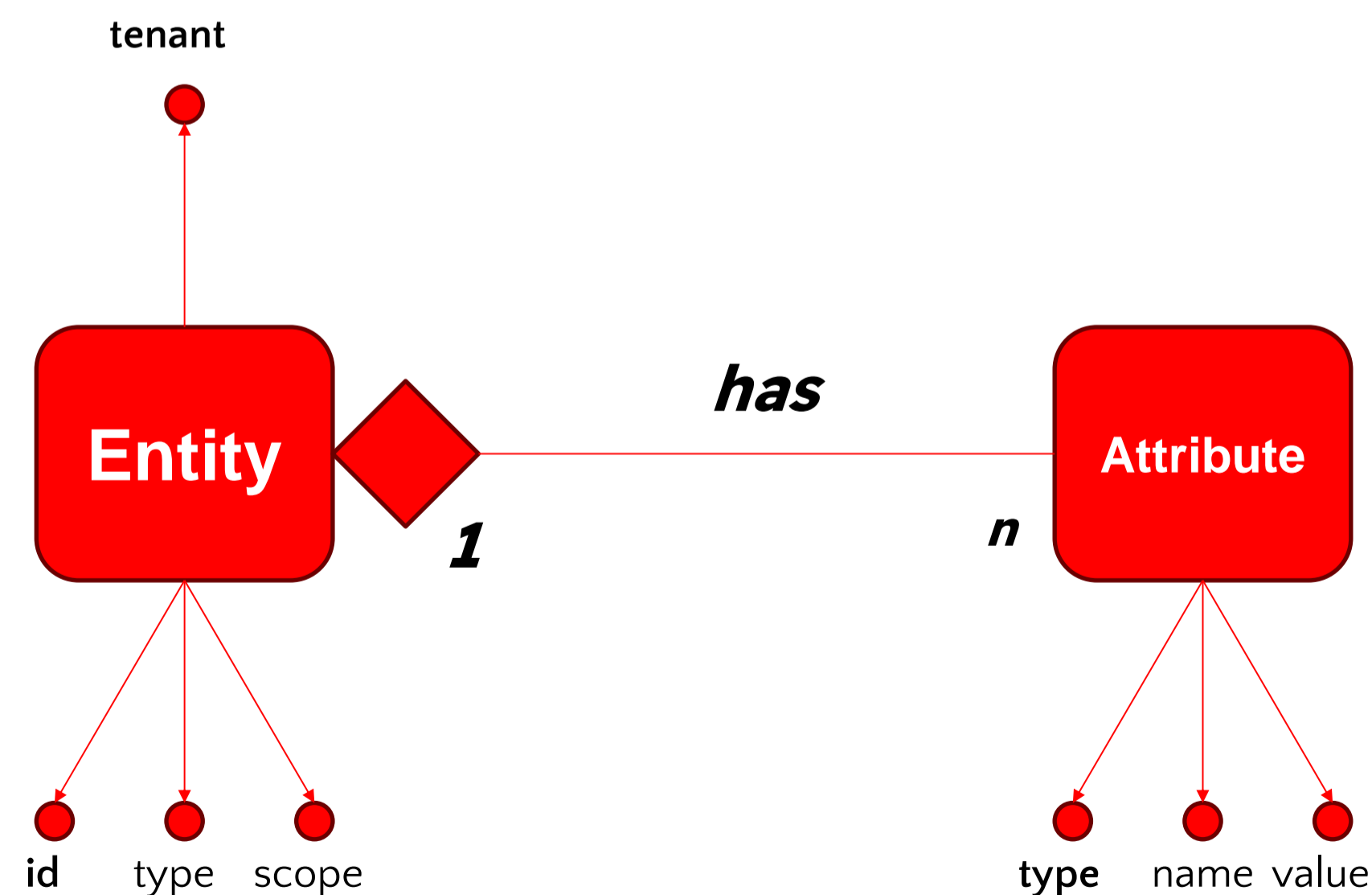
NGSI-LD – CRUD on Entities/Attributes

The Base Path of an NGSI-LD **Context Broker** is:

https://<YOUR_DOMAIN> **/ngsi-ld/v1**

Create

POST	/entities
POST	/entityOperations/create
POST	/entityOperations/upsert

Read

GET	/entities
GET	/entities/{entityId}
POST	/entityOperations/query

Update

PUT	/entities/{entityId}
PATCH	/entities/{entityId}
POST	/entities/{entityId}/attrs
PATCH	/entities/{entityId}/attrs
PATCH	/entities/{entityId}/attrs/{attrName}
POST	/entityOperations/update

Delete

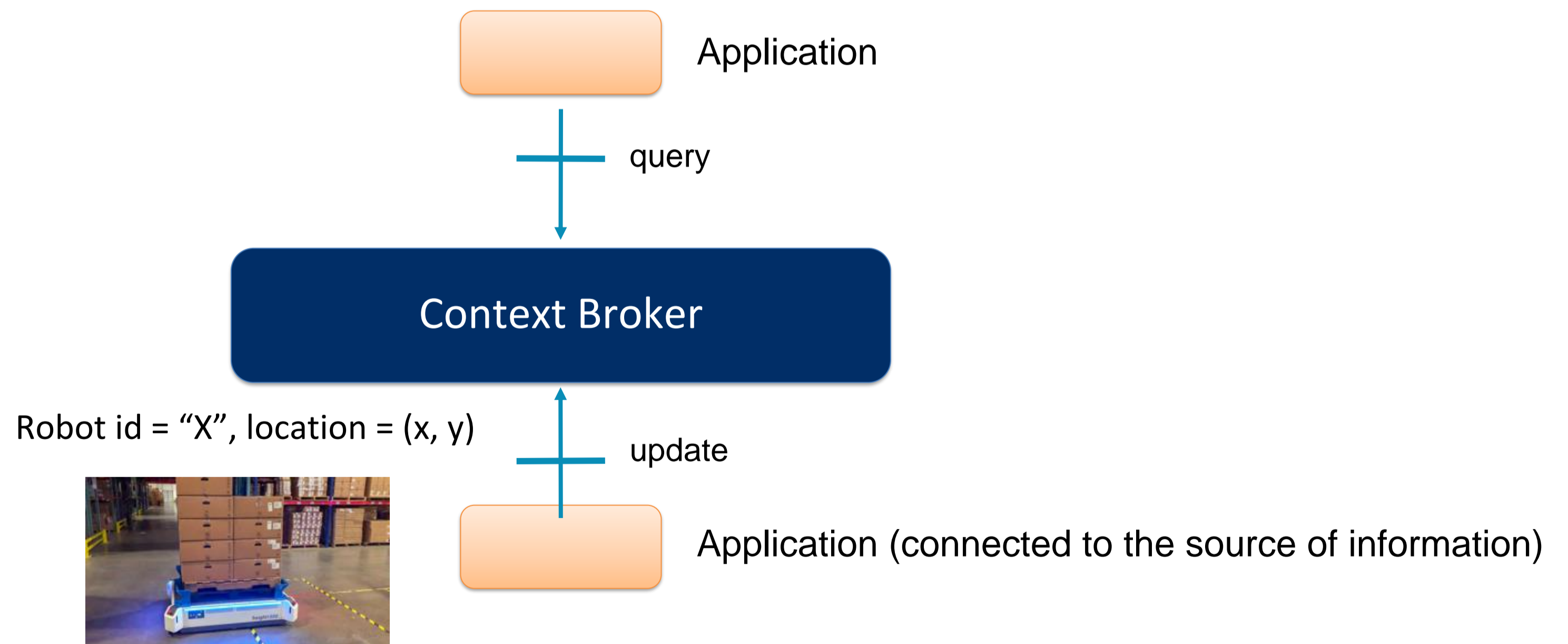| DELETE | /entities/{entityId} |
| DELETE | /entities/{entityId}/attrs/{attrName} |
| POST | /entityOperations/delete |

Creation of entities

- Application can create entities and give initial values to their attributes (properties/relationships)

- Multiple entities can be created in a single request

Application

create

Context Broker

Transport robot
- Id
- location
- speed
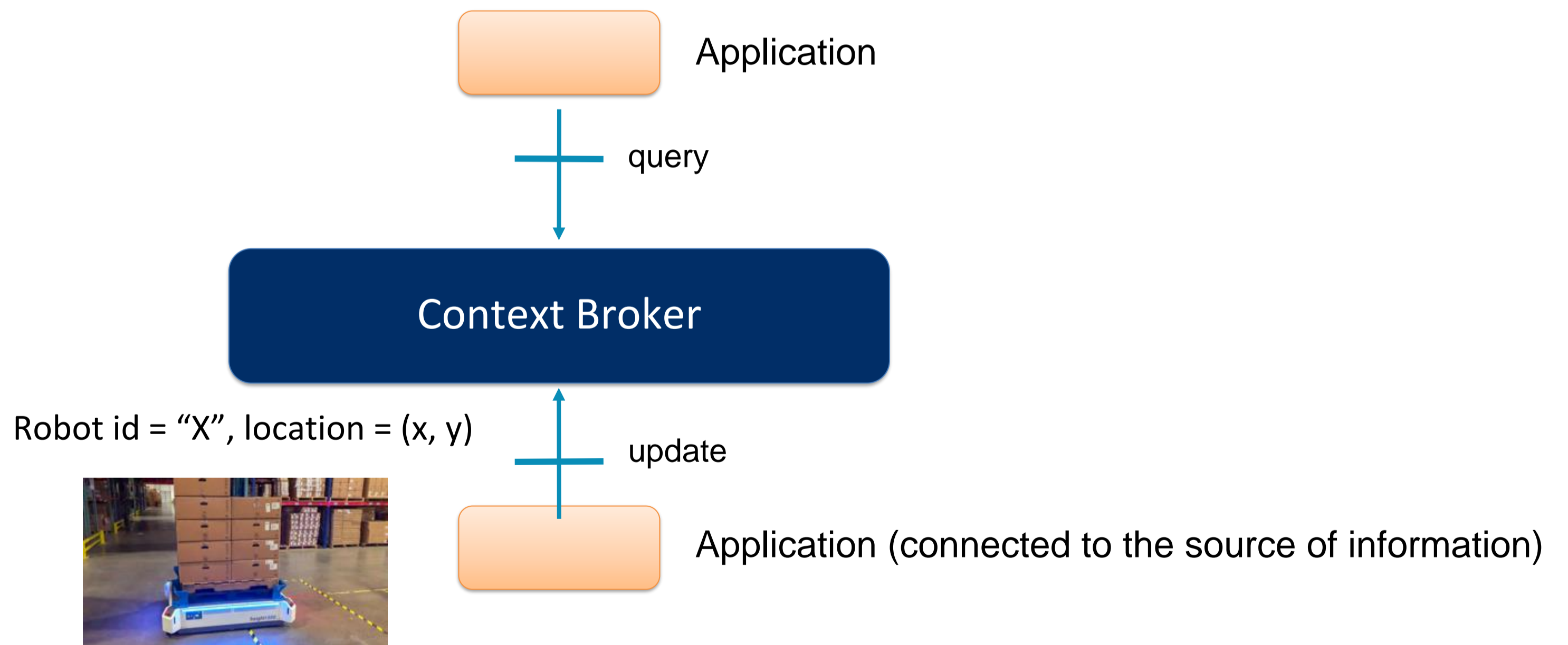- transported items
- destination

FIWARE

Updates on entities

- You can update an attribute of a given entity with a request (simplest case)

- You can update a set of attributes on one or more entities in a single request

- Attributes can be added to a given entity type (effectively extending the underlying data model) at any time without the need to re-initiate

Application

query

Context Broker

Robot id = "X", location = (x, y)

update

Application (connected to the source of information)
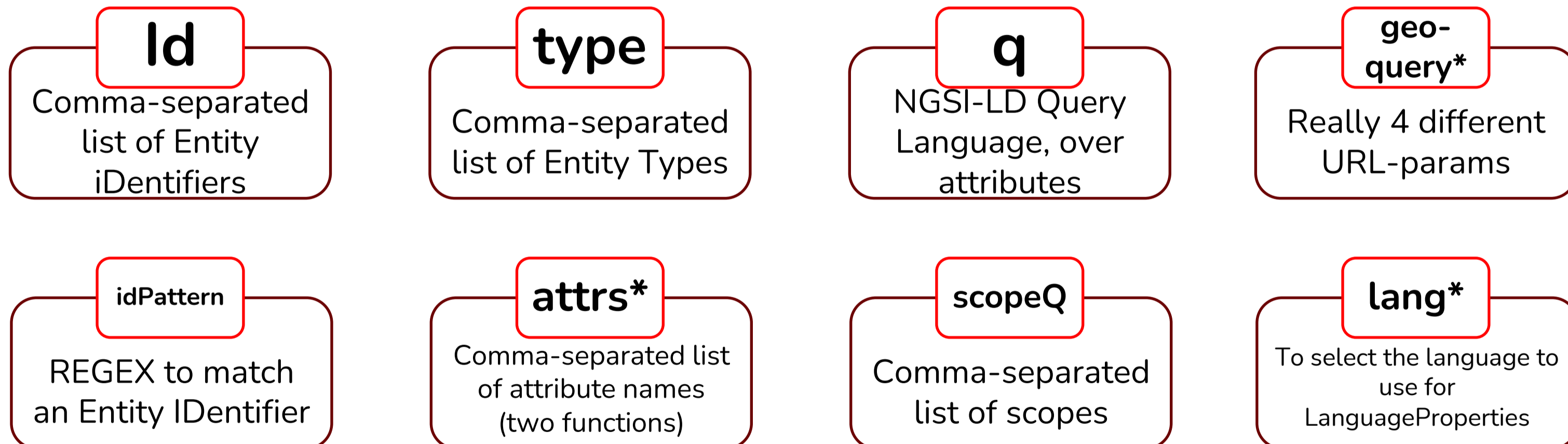
Queries on entities

- You can query for the value of an attribute of a given entity with a request (simplest case)

- You can query for a set of attributes on one or more entities in a single request

- You can specify filters/conditions on your query to refine what entities you are querying information about

Application

query

Context Broker

Robot id = "X", location = (x, y)

update

Application (connected to the source of information)
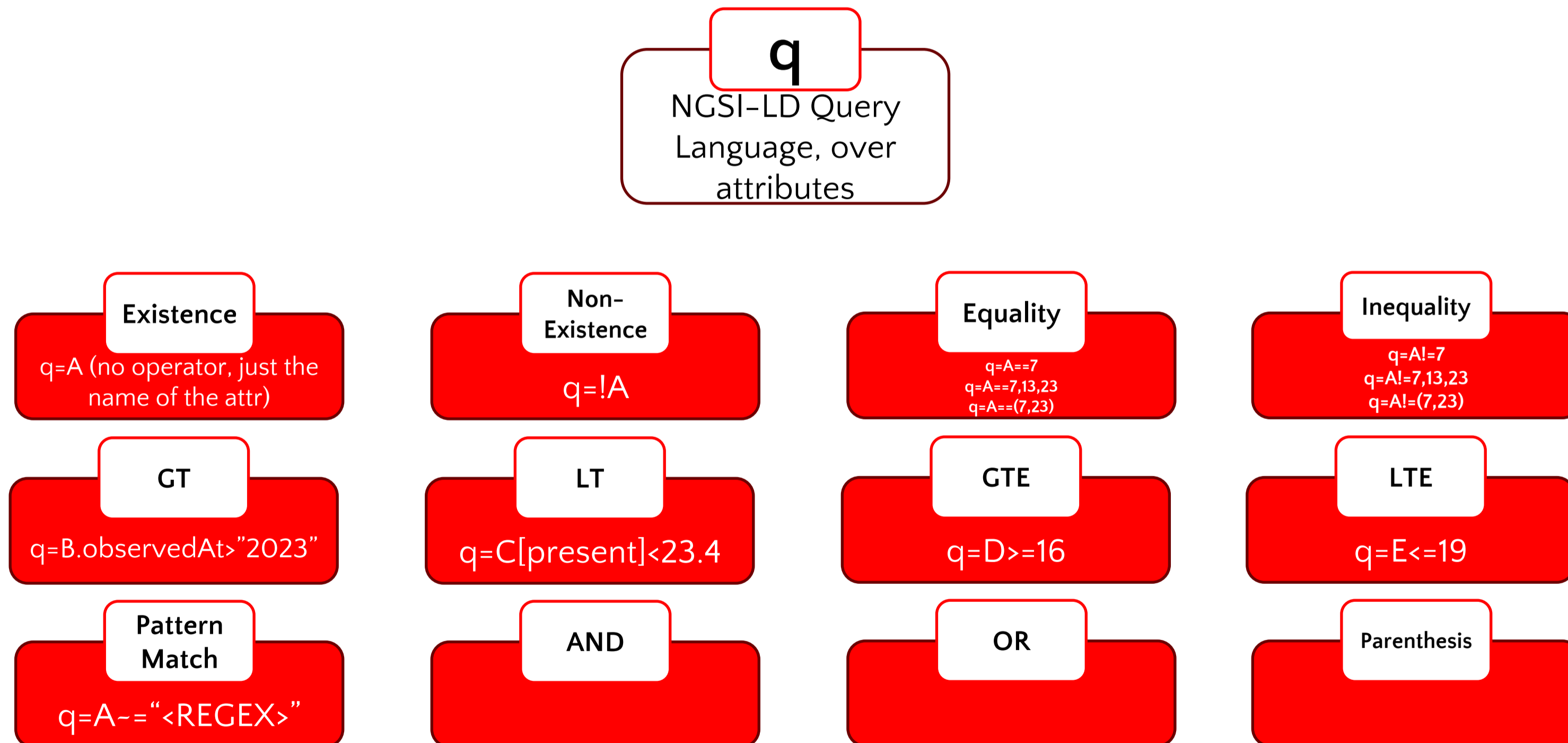
FIWARE

NGSI-LD – Options for Querying Entities

Filters are criteria or parameters used to restrict and select specific entities or information during a query of entities.

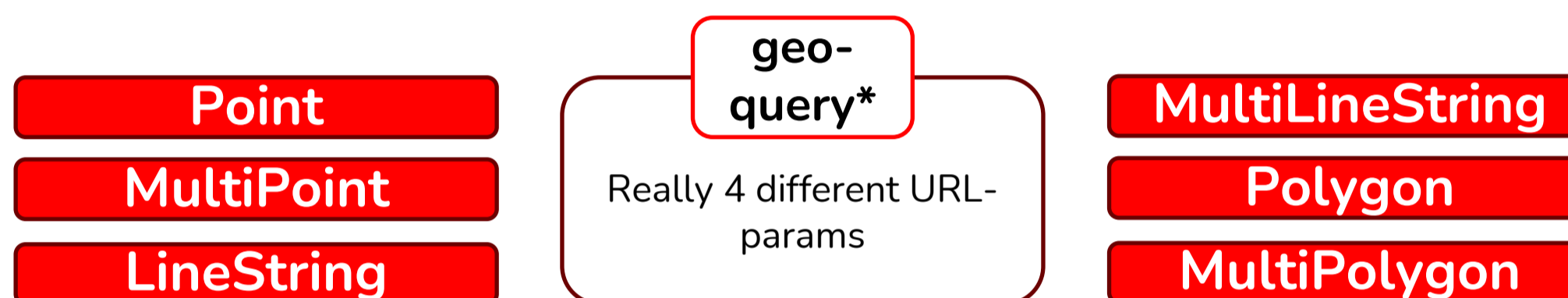When using a "POST Query", where they are included in the body of the request (*).

Id
Comma-separated list of Entity iDentifiers

type
Comma-separated list of Entity Types

q
NGSI-LD Query Language, over attributes

geo-query*
Really 4 different URL-params

idPattern
REGEX to match an Entity IDentifier

attrs*
Comma-separated list of attribute names (two functions)

scopeQ
Comma-separated list of scopes

lang*
To select the language to use for LanguageProperties

NGSI-LD – Query Language: Operators

The "NGSI-LD Query Language" is a query language used in the context of NGSI-LD to **perform advanced queries on entity attributes**.

q
NGSI–LD Query Language, over attributes

Existence
q=A (no operator, just the name of the attr)

Non-Existence
q=!A

Equality
q=A==7
q=A==7,13,23
q=A==(7,23)

Inequality
q=A!=7
q=A!=7,13,23
q=A!=(7,23)

GT
q=B.observedAt>"2023"

LT
q=C[present]<23.4

GTE
q=D>=16

LTE
q=E<=19

Pattern Match
q=A~="<REGEX>"

AND

OR

Parenthesis

NGSI-LD – Query Language: Operators

The "geo-query" is a functionality within NGSI-LD that allows **spatial queries to be performed to select entities based on geographical criteria**. This functionality is particularly useful when managing geographical or geospatial location data.

Point	**geo-query***	**MultiLineString**
MultiPoint	Really 4 different URL-params	**Polygon**
LineString		**MultiPolygon**

Each geometry type has its own "<u>**georels**</u>" that specifies the **geographical relationship** between the entity and the specified point. (e.g., "near;minDistance==X" for "Point"). See: <u>https://www.rfc-editor.org/rfc/rfc7946</u>.

Example (GET)

/entities?geoProperty=location&geometry=Point&coordinates=[1,2]&georel=near;maxDistance==100
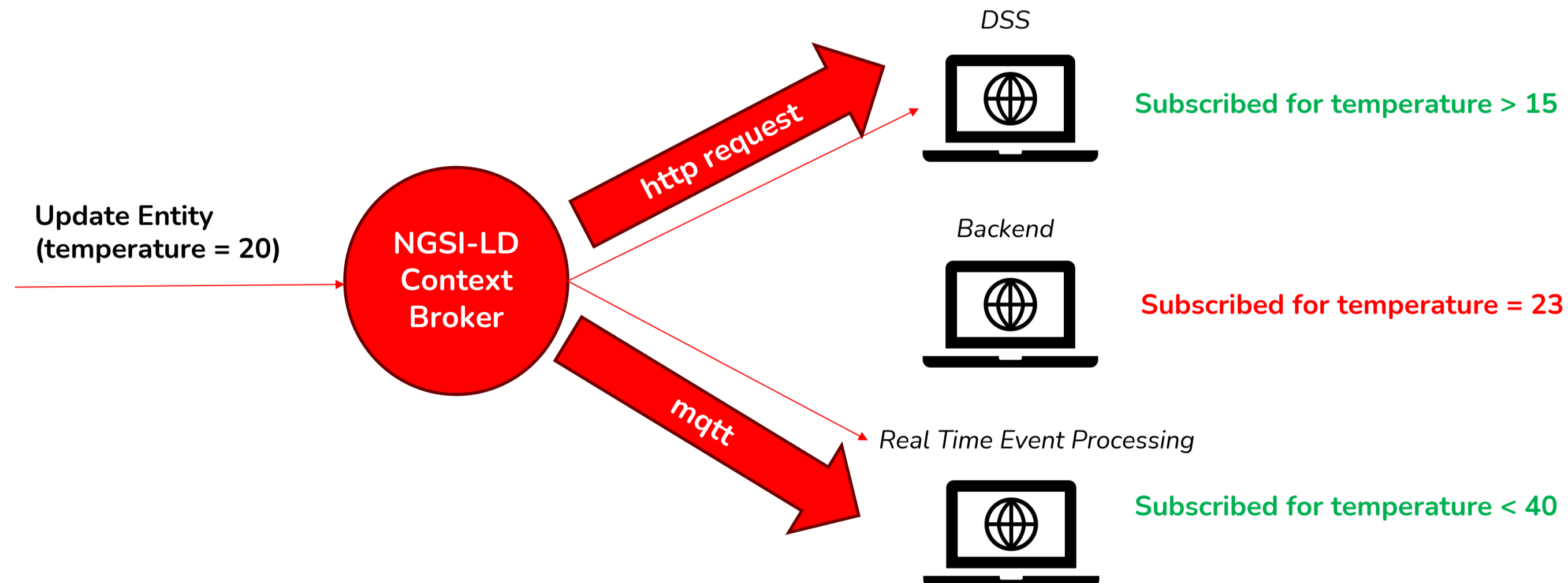
Example (POST)

```
{
  "geo-query": {
    "geometry": {
      "type": "Point",
      "coordinates": [longitude, latitude]
    },
    "georel": "near",
    "maxDistance": distance
  }
}
```

31

NGSI-LD – Subscriptions

Subscriptions are mechanisms that allow users **to receive real-time notifications when information associated with entities meets certain specified conditions**.

A Subscription defines "what to get notified for" and "where and in what format to send the notification". The notification are sent by CB **via HTTP or MQTT Protocols**.

NGSI-LD – Subscriptions (Example)

TEMA

```
curl -L -X POST 'http://localhost:1026/ngsi-ld/v1/subscriptions/' \
-H 'Content-Type: application/ld+json' \
-H 'NGSILD-Tenant: openiot' \
--data-raw '{
  "description": "Notify me of low feedstock on Farm:001",
  "type": "Subscription",
  "entities": [{"type": "FillingLevelSensor"}],
  "watchedAttributes": ["filling"],
  "q": "filling>0.6;filling<0.8;controlledAsset==%22urn:ngsi-
ld:Building:farm001%22",
  "notification": {
    "attributes": ["filling", "controlledAsset"],
    "format": "keyValues",
    "endpoint": {
      "uri": "http://tutorial:3000/subscription/low-stock-farm001",
      "accept": "application/json"
    }
  },
    "@context": "http://context/ngsi-context.jsonld"
}'
```
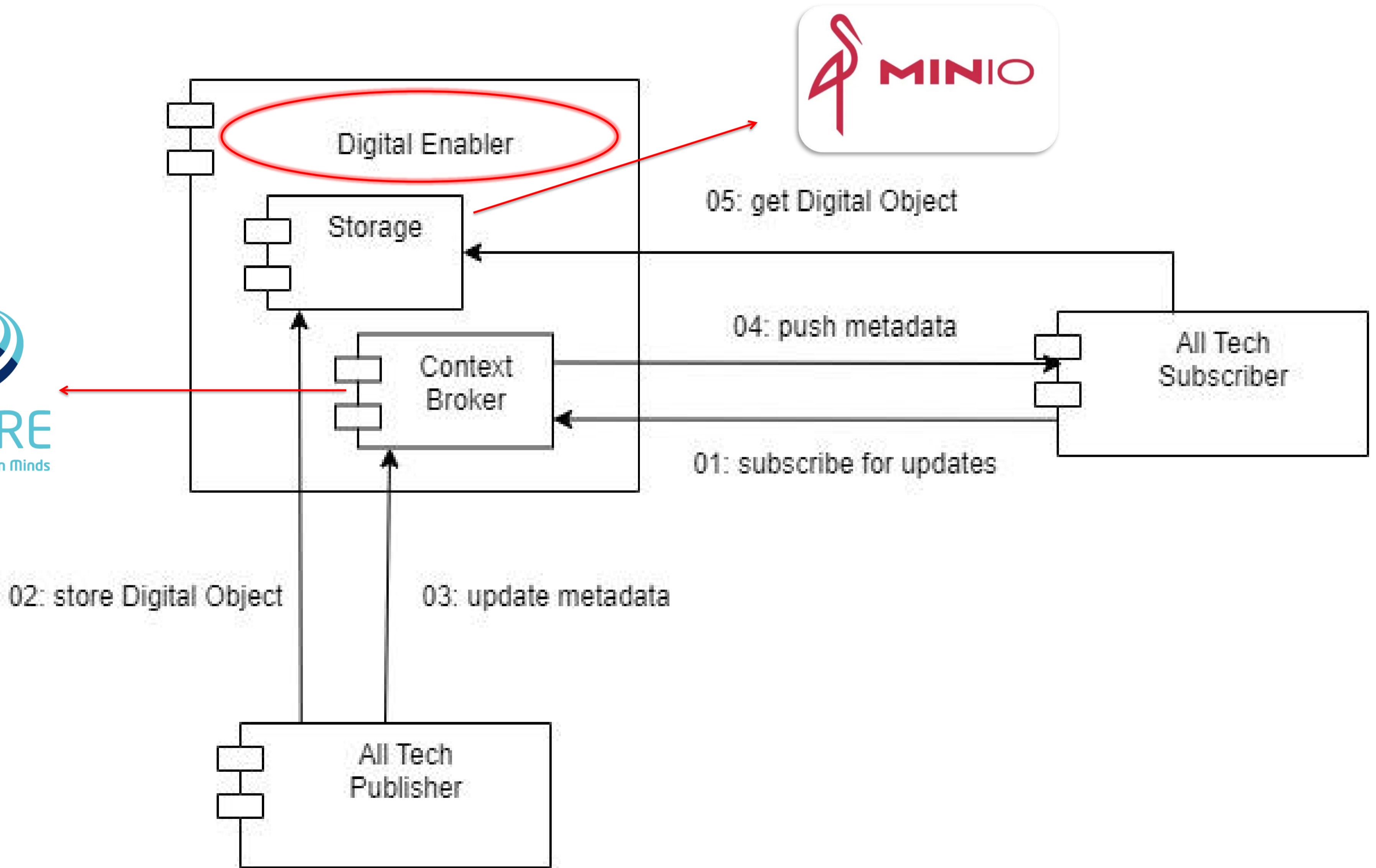
URL for Subscription

Trigger Condition(s)

Target

Data type to be recieved

33

Components adopted in TEMA to support Natural Disasters management

What is MinIO

MinIO is a **high performance**, **software defined**, **distributed object storage** server, designed for peta-scale data infrastructure.

It was built from scratch with the **private cloud** as its target.
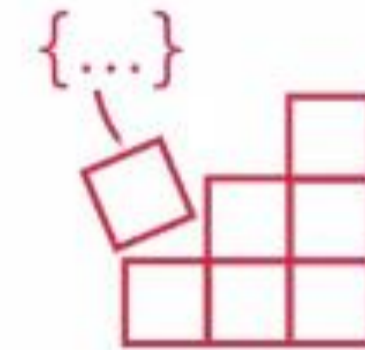
MINIO SERVER **MINIO CLIENT** **MINIO SDK**

MINIO

MinIO Deployment Use Cases

Big Data/Machine Learning environments

HDFS replacements

High performance data lake/warehouse infrastructure

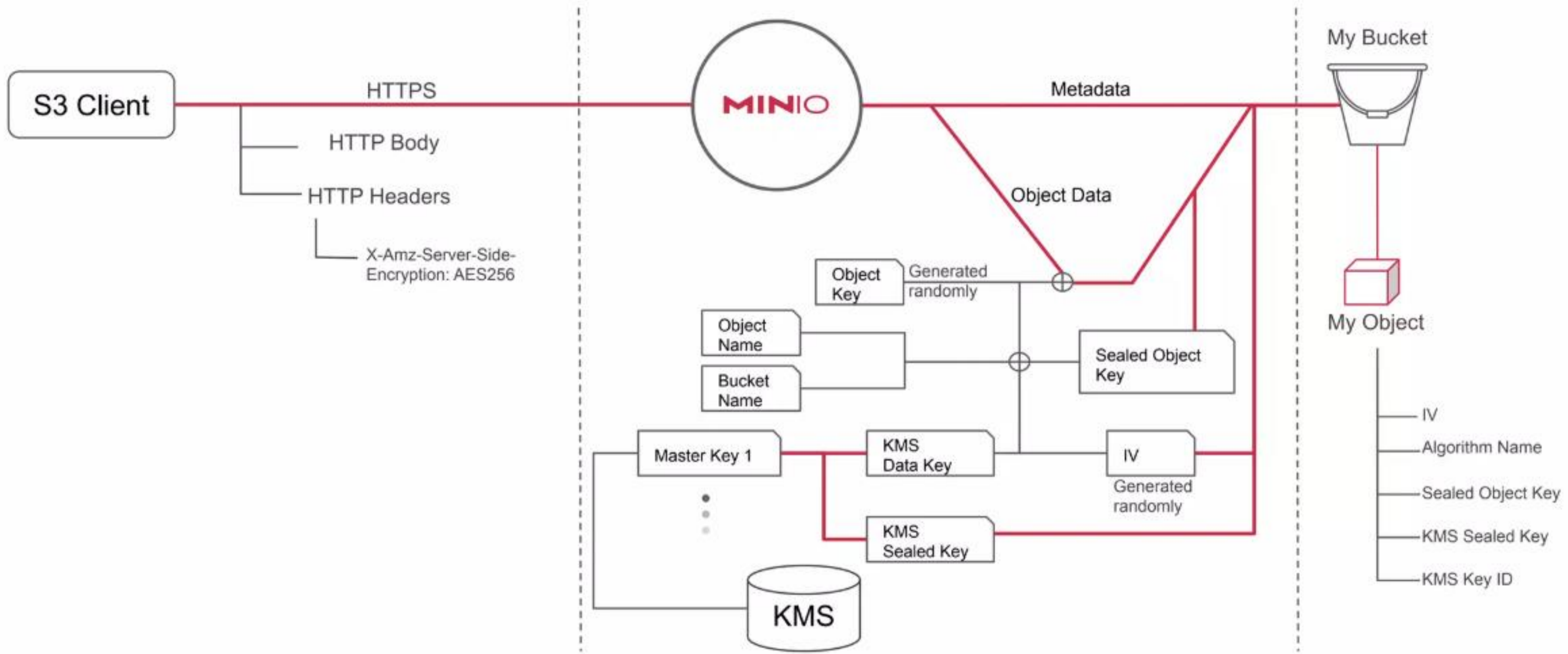Cloud native applications (replacing file + block)

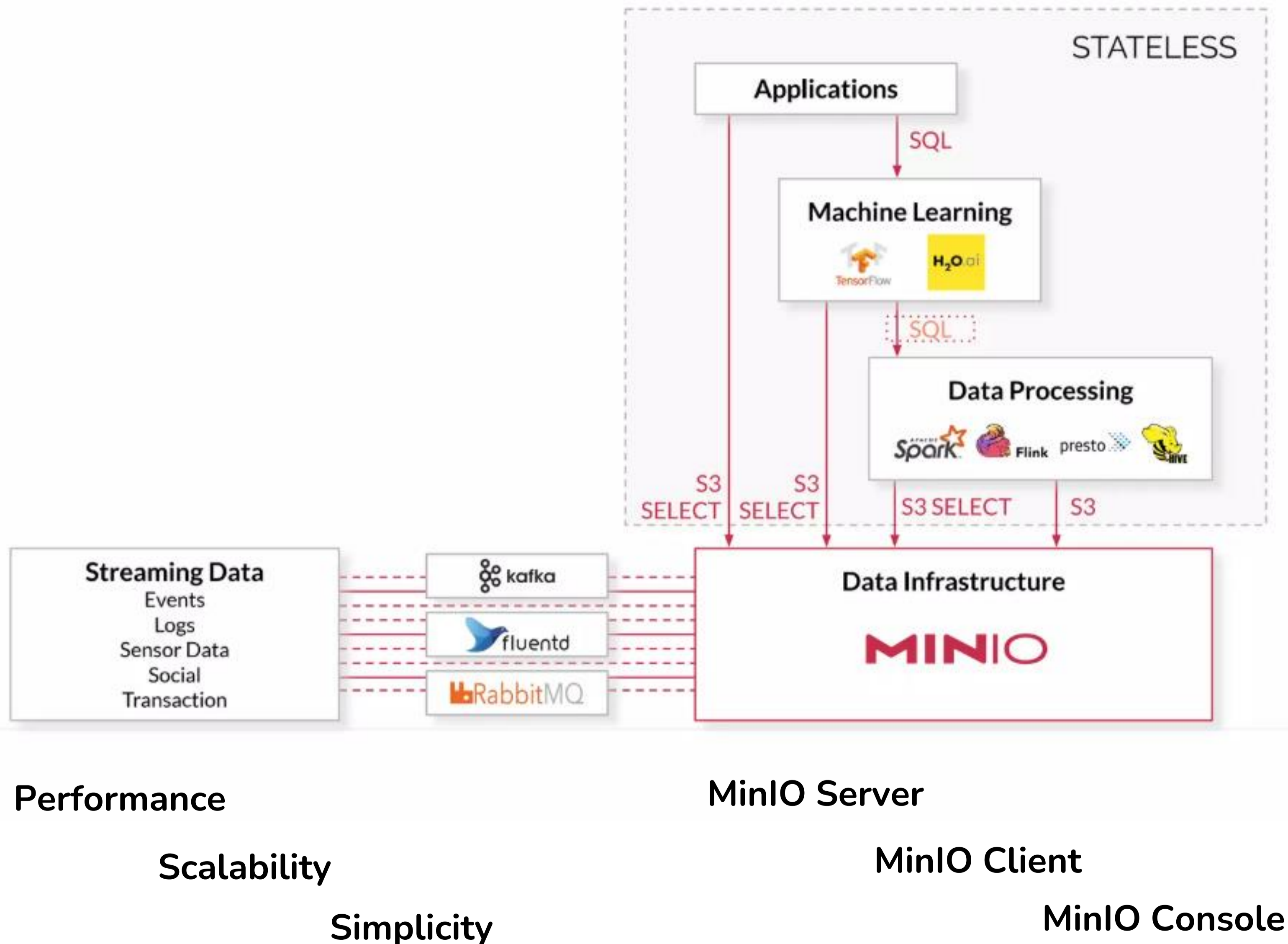Multi-cloud environments (portability)

Endpoint for streaming workloads

By deploying for performance, long term archival storage and disaster recovery are effectively free.

MINIO

MinIO Encryption - SSE-S3

At the Center of a Modern Architecture

Performance

Scalability

Simplicity

MinIO Server

MinIO Client

MinIO Console

TEMA

TRUSTED
EXTREMELY PRECISE
MAPPING AND PREDICTION
FOR EMERGENCY
MANAGEMENT

Thank you for your attention!

FRANCESCA D'AGRESTI

Engineering Ingegneria informatica S.p.A.

francesca.dagresti@eng.it