DONNA: A Data Model for Enabling Extensible and Efficient Metaverse Applications

Georgios Bouloukakis

Télécom SudParis, Institut Polytechnique de Paris, France.

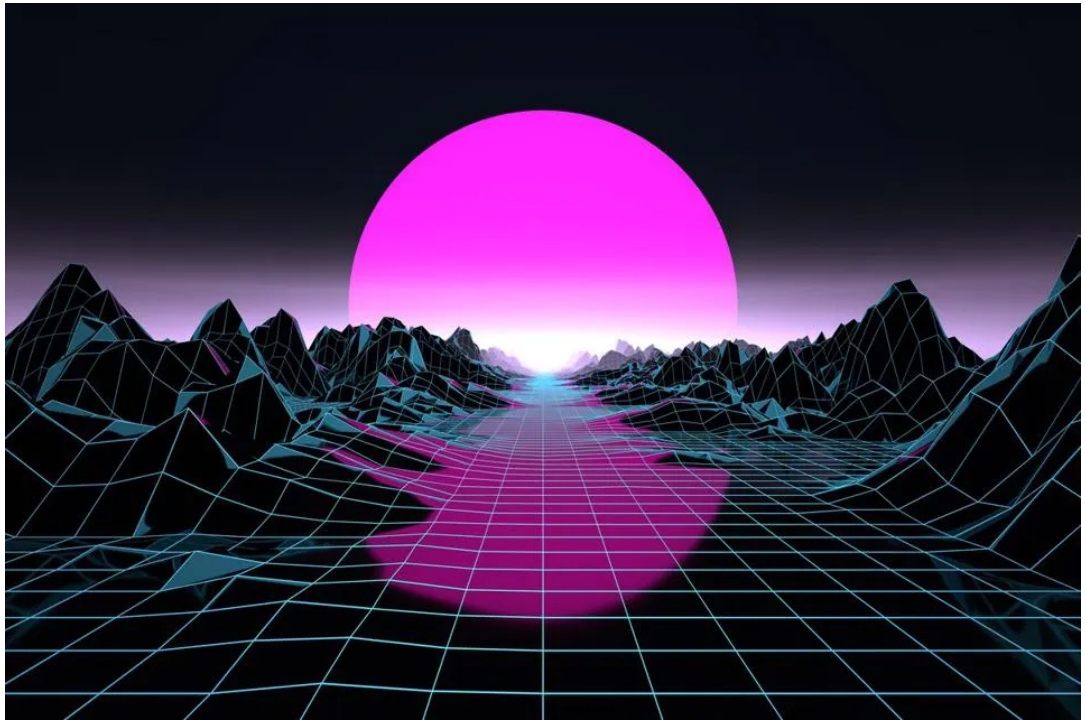
Ajay Kattepur

Ericsson Research Artificial Intelligence, Bangalore, India.

IEEE MetaCom 2023

Motivation

- The Metaverse has the potential to disrupt multiple industries
- What is missing ?
  - Comprehensive data model that captures the interactions between the physical and virtual worlds
- DONNA: A Data Model for Enabling Extensible and Efficient Metaverse Applications:
  - Data model of interactions between physical space, virtual spaces, sensors, devices, physical participants and avatars
  - Property graph schemas to formally represent the physical and virtual worlds
  - Demonstrated over a virtual museum visit use case



Metaverse: Use cases and Challenges

Metaverse

The Metaverse building blocks:

- *Environment*: The real physical space is reproduced in digital form with additions
- *Avatars*: The avatar should reflect the real persona and map emotions, sensory actions and reactions
- *Devices*: Head mounted devices that support virtual/augmented reality applications
- *Infrastructure*: 5G technologies, Edge computing and intelligent resource allocation

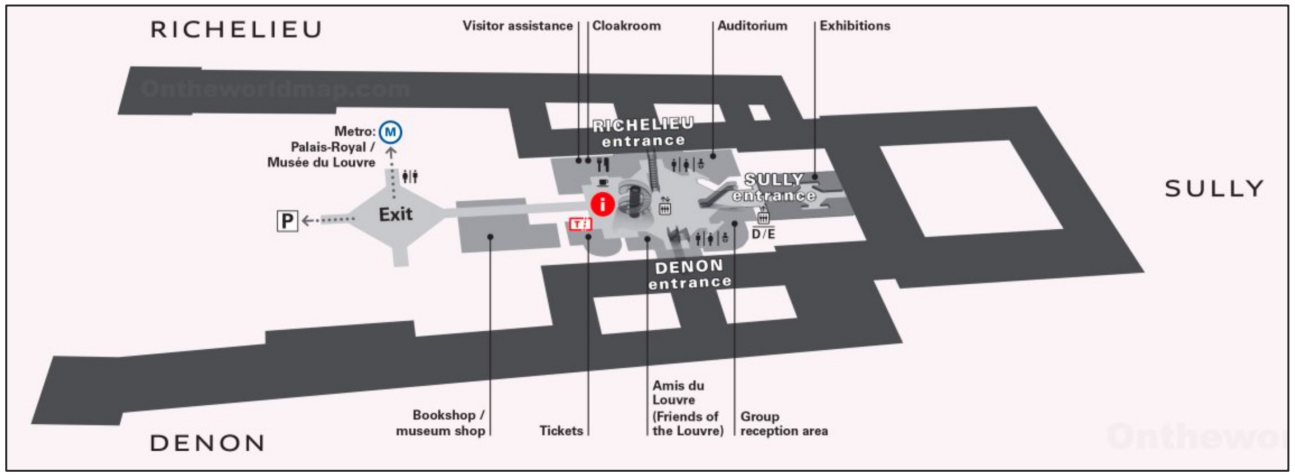
Park and Kim, “A metaverse: Taxonomy, components, applications, and open challenges," IEEE Access, 2022.

Louvre Museum Visit

Louvre museum has three sectors (Richlieu, Sully, Denon) and spans five floors with tens of rooms.

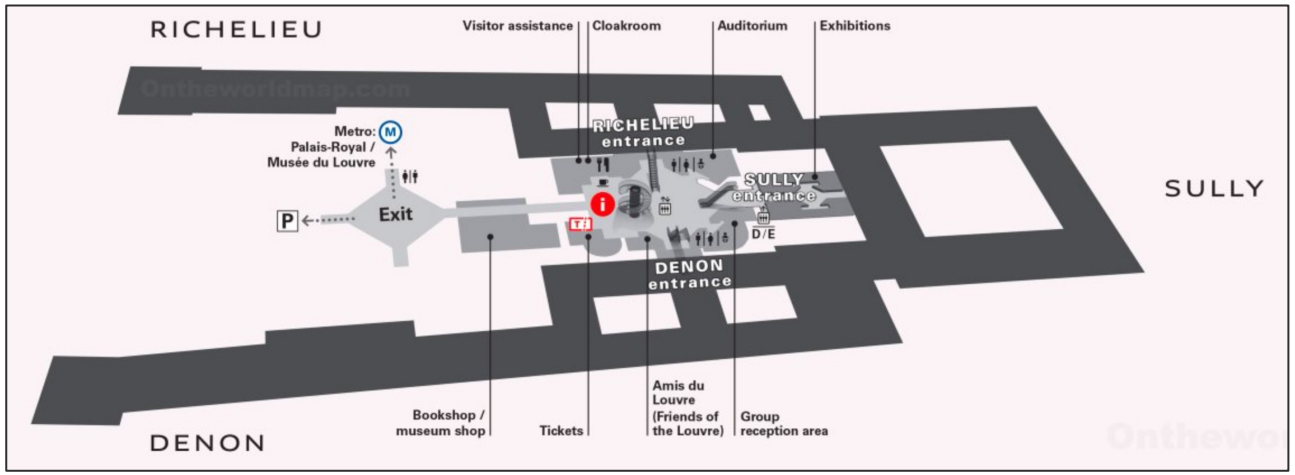
Requirements

- An accurate depiction of the floor plan, sections and rooms within the physical museum
- Virtual visit participants may want to see the physically present crowd
- Virtual world participants may want exclusive access
- There may be temporary exhibits, events and author signings that must also be replicated in the Metaverse
- Changes in the environment (lighting, crowd) must be accurately mapped in the virtual world



Challenges

- *Semantic Mapping*: Mapping and synchronization of properties and features that can change (location, lighting)
- *Standard Data Models*: Data schemas, properties and knowledge shared between elements of the physical/virtual world
- *Interactions*: Haptic feedback, avatar observations and interactions are to be captured to create a seamless experience
- *Synchronization*: Virtual/augmented devices and sensor readings are to be synchronized in real-time
- *Intelligent Rendering*: Advances in XR integrated with positioning and prediction algorithms to provide superior QoE



Knowledge Graphs

Property Graph

A property graph is a type of graph model where relationships not only are connections but also have a label and properties

1. *Nodes*: the entities in the graph. Nodes can be tagged with zero to many text labels representing their type.
2. *Edges*: the directed links between nodes. While edges are directed, they can be navigated and queried in either direction.
3. *Properties*: the key-value pairs associated with a node or with an edge.

Hogan and Blomqvist,"Knowledge graphs," ACM Comput. Surv., 2021.

Property Graph. A property graph is a tuple $G = (N, E, \rho, \lambda, \sigma)$ where:
– N is a finite set of nodes;
– E is a finite set of edges such that E has no elements in common with N;
– $\rho : E \to N \times N$ is a function that associates each edge with a pair of nodes;
– $\lambda : (N \cup E) \to (\mathcal{L})^+$ is a function that associates a node/edge with a set of labels from $\mathcal{L}$ (for a set X, $(X)^+$ is the set of all finite subsets of X);
– $\sigma : (N \cup E) \times P \to (\mathcal{V})^+$ is a function that associates nodes/edges with a property value from $\mathcal{V}$.

Property Graph Schema

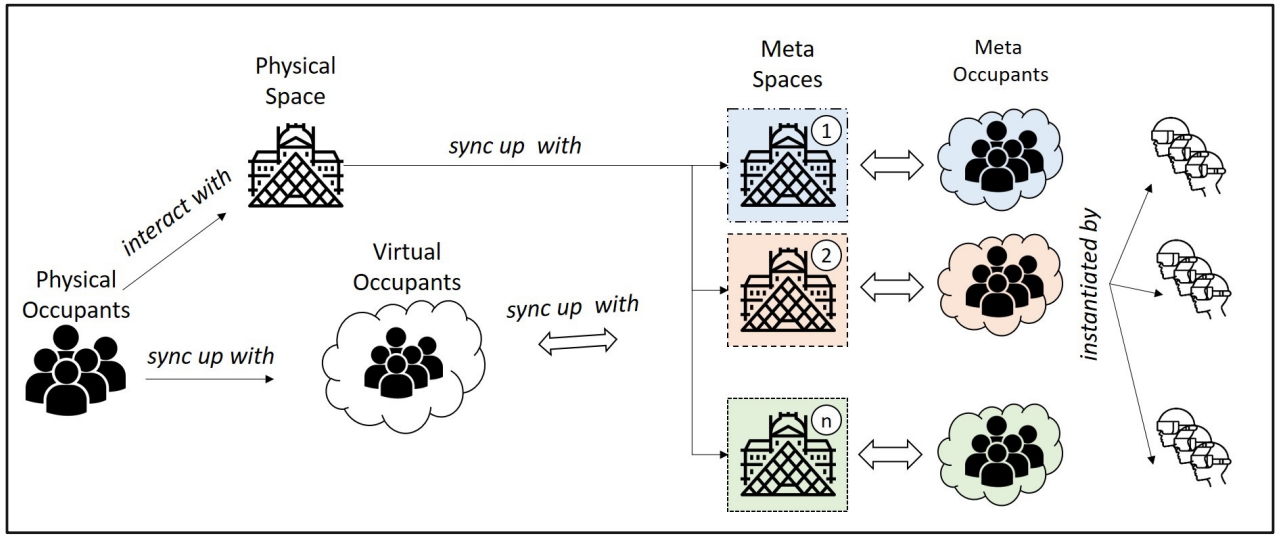
- A data schema is a data modeling feature that allows to describe the structure of the data and enforce its consistency
- Graph schema allows to define the graph structure by specifying the types of nodes, the types of edges, and the properties for such types
- Consistency checks may be done on property graph instances in relation to the schema

Property Graph Schema. With T is a finite set of datatypes (e.g., String, Integer), a property graph schema is a tuple $S = (T_N, T_E, \beta, \delta)$ where:
– $T_N \subset \mathcal{L}$ is a finite set of labels representing node types;
– $T_E \subset \mathcal{L}$ is a finite set of labels representing edge types;
– $\beta : (T_N \cup T_E) \times P \to T$ is a function that defines the properties for nodes and edge datatypes;
– $\delta : (T_N, T_N) \to (T_E)^+$ is a function that defines the edge types allowed between a pair of node types.

DONNA: Metaverse Data Models

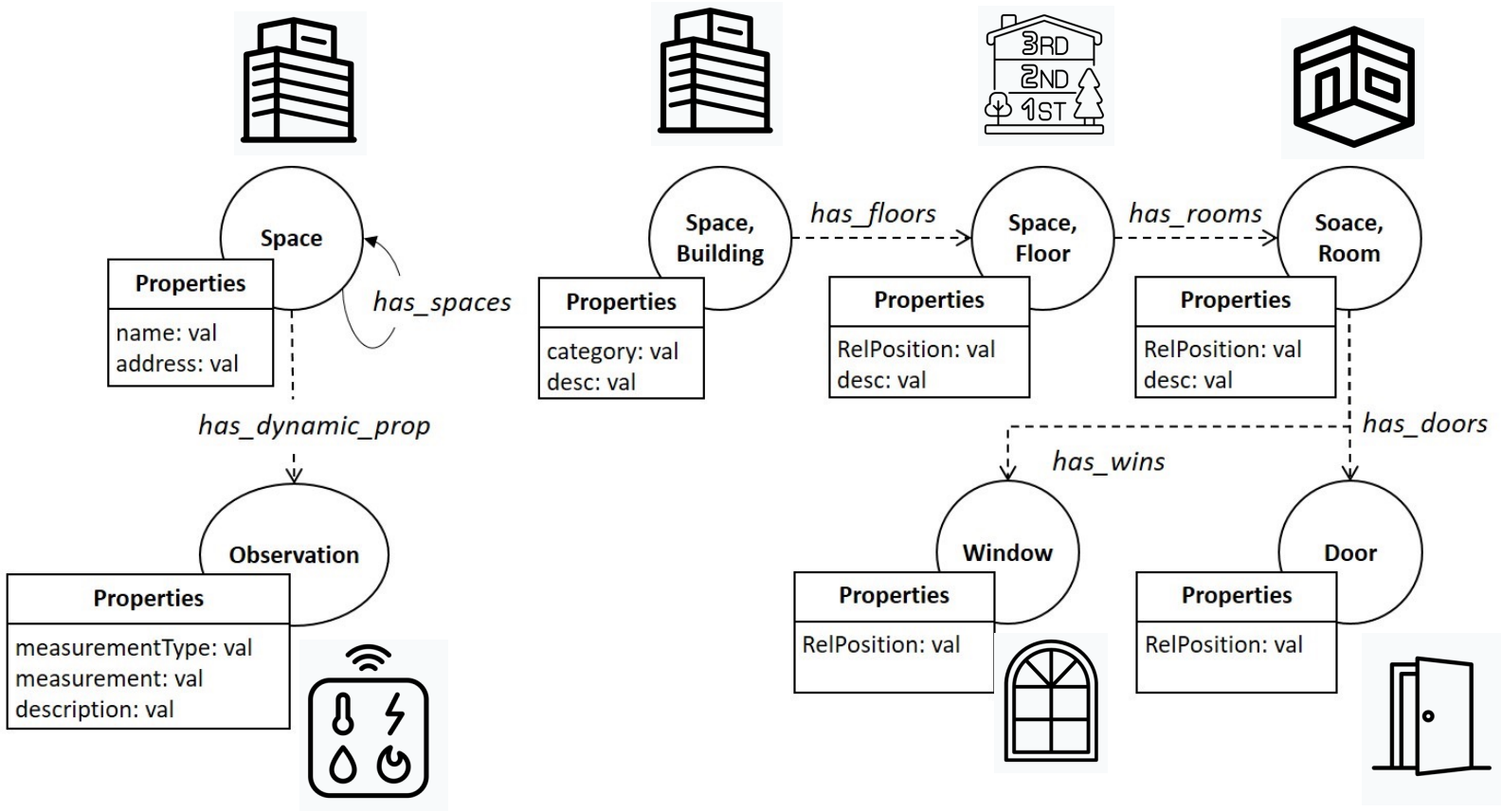
Metaverse Ecosystem

- A physical space may host a Physical Occupant that also represents a real entity used to create instances of occupant
- A Meta Space is an entity type that can be initially instantiated by relying on a Physical Space instance
- Virtual Occupant instances that correspond to the Physical Space Occupants
- Apart from the static properties (floors, rooms, objects, etc.), dynamic properties (physical occupants, space conditions, etc.) must be presented
- Meta Occupant as an entity that can instantiated using Extended Reality (XR) devices – the occupant is not present in the physical world

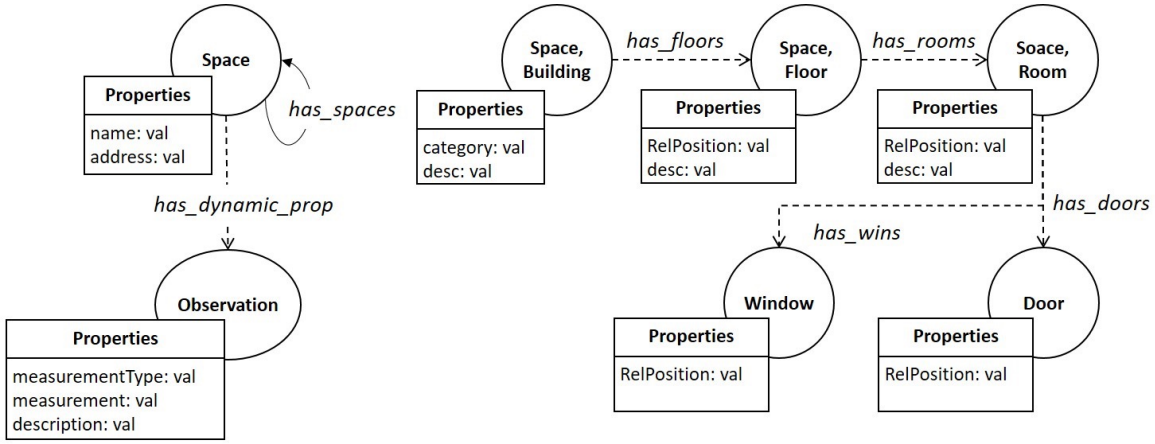


Physical Space and Person

- Spaces have properties which include unique identifiers.
- The spaces may be described using sensor observations or static knowledge (blueprint, maps)
- Of particular interest are dynamic properties that are to be periodically monitored and synchronized onto the Metaverse



Physical Spaces and Person



An expanded version of the schema:
https://github.com/SAMSGBLab/metaverse--donna

Table 1: A data schema for Physical spaces.

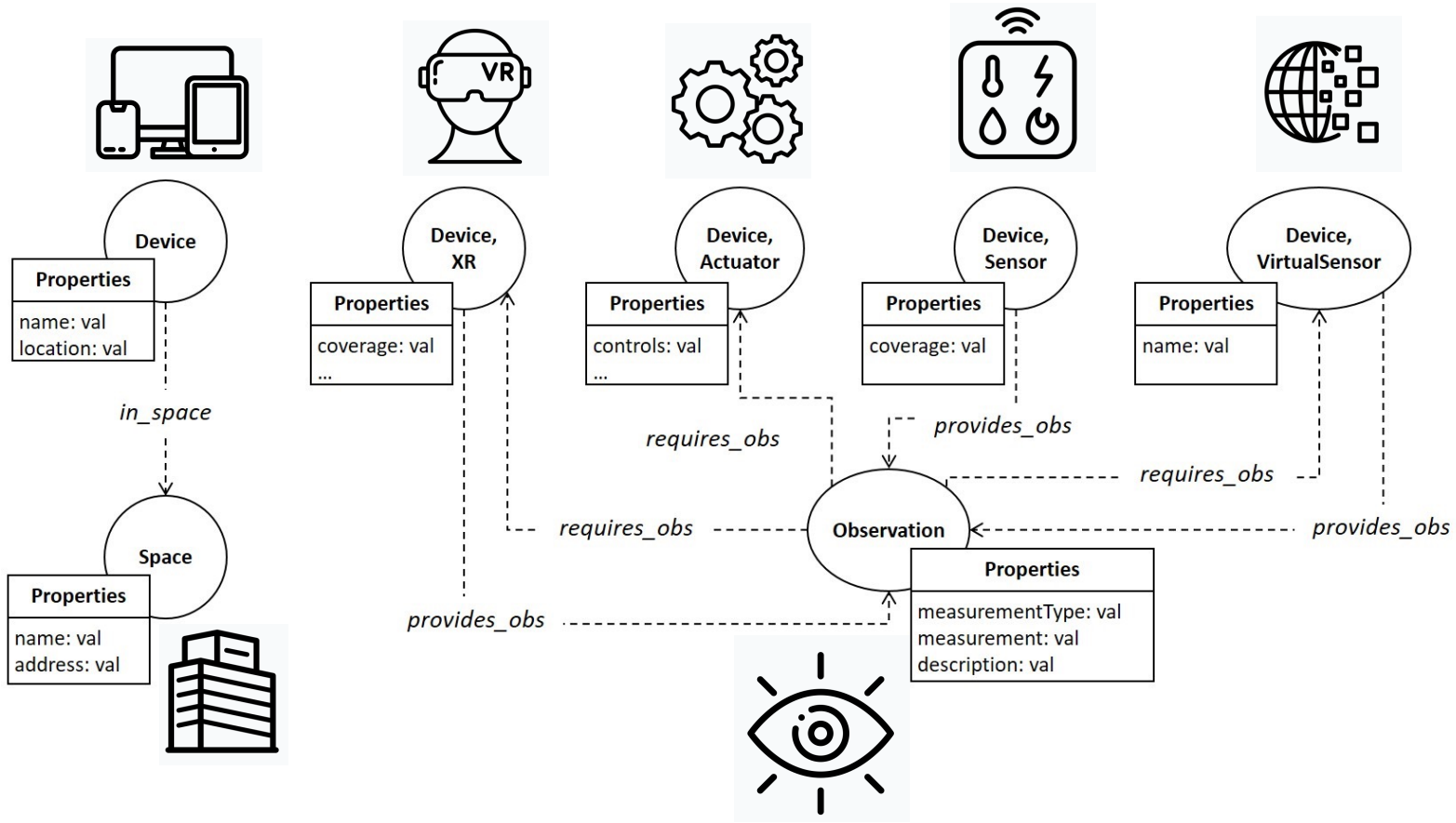
$T_N = \{\text{Space, Observation, Building, Floor, Room, Window, Door}\}$
$T_E = \{\text{has\_spaces, has\_dynamic\_prop, has\_floors, has\_rooms, has\_doors, has\_wins}\}$

β (Space, name) = val
β (Space, address) = val
β (Observation, measurementType) = val
β (Observation, measurement) = val
β (Observation, description) = val
β (Building, categorty) = val
β (Building, desc) = val
β (Floor, RelPosition) = val
β (Floor, desc) = val
β (Room, RelPosition) = val
β (Room, desc) = val
β (Window, RelPosition) = val
β (Door, RelPosition) = val

δ (Space, Space) = {has_spaces}
δ (Space, Observation) = {has_dynamic_prop}
δ (Building, Floor) = {has_floors}
δ (Floor, Room) = {has_rooms}
δ (Room, Window) = {has_wins}
δ (Room, Door) = {has_doors}

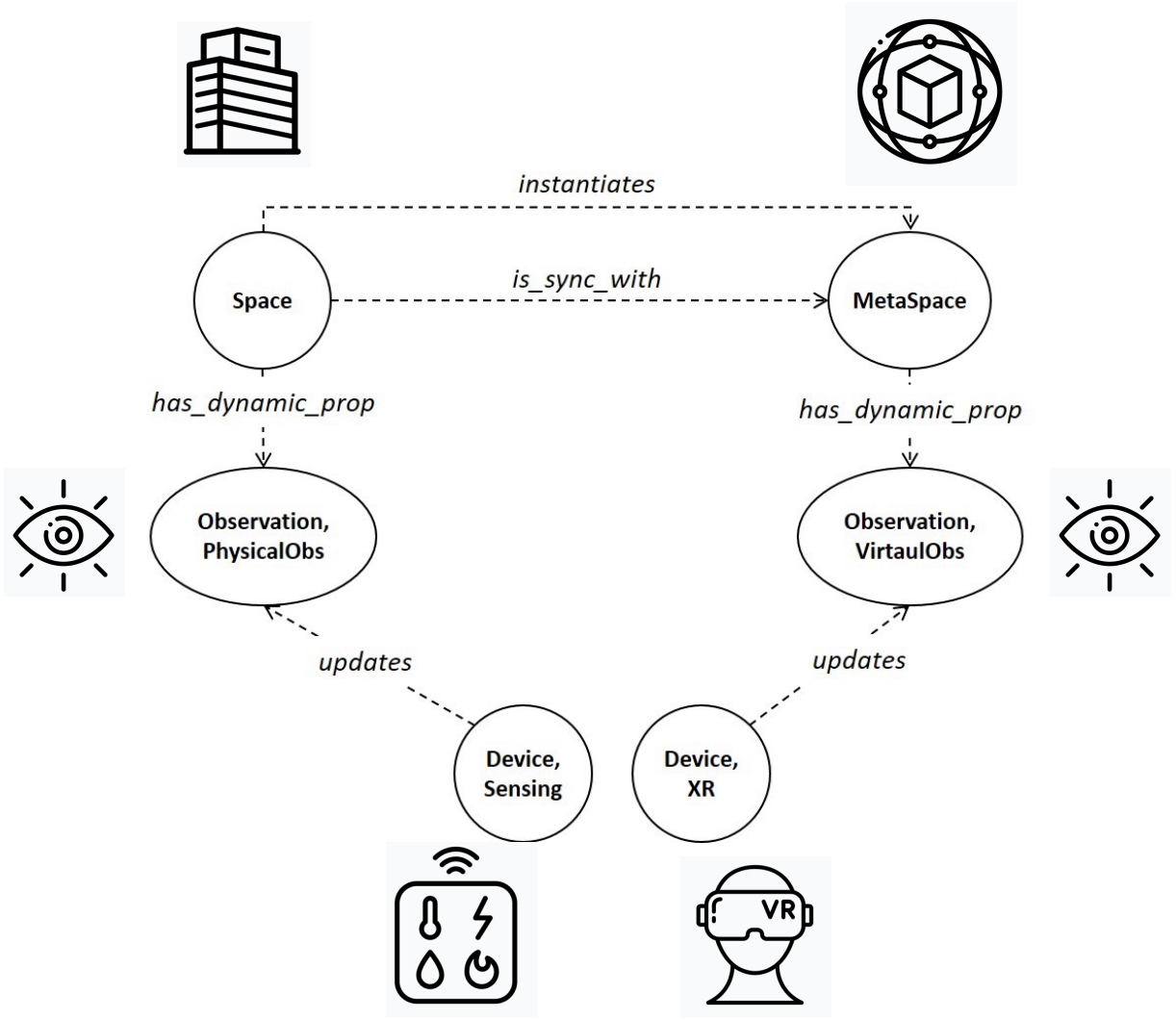
Devices

- Devices such as *sensor* are producers of observations
- The *actuator* devices used to make changes in the environment
- The XR devices act as both producers and consumers of data
- *Virtual sensors* that can combine physical sensor readings or provide virtual sensing within the Metaverse



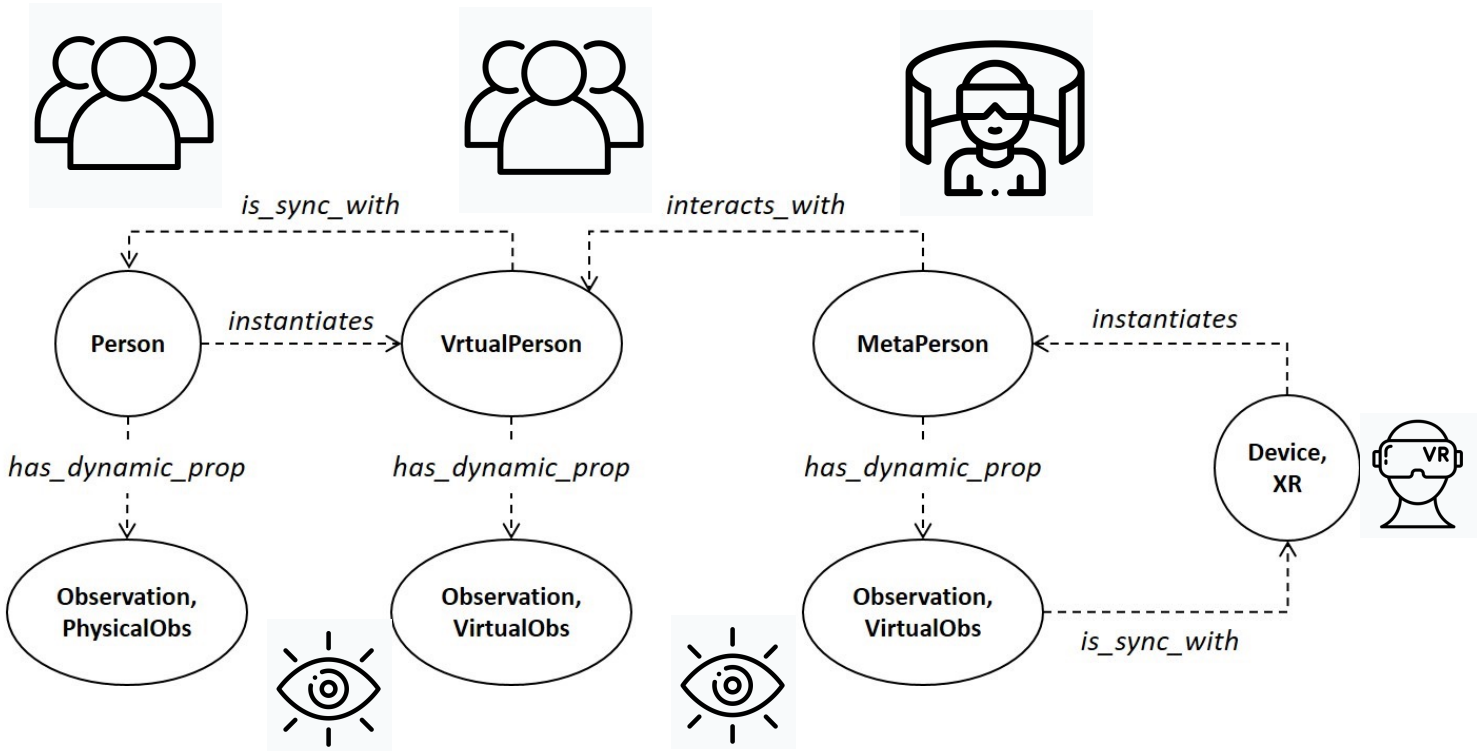
Space and MetaSpace

- The physical Space instantiates a Meta Space and keeps in synchronization with changes.
- Sensors observe the physical space and update the observations
- Meta Space has VirtualObs that are updated (e.g., changes in the number of participants, topology in the Metaverse)



Person and MetaPerson

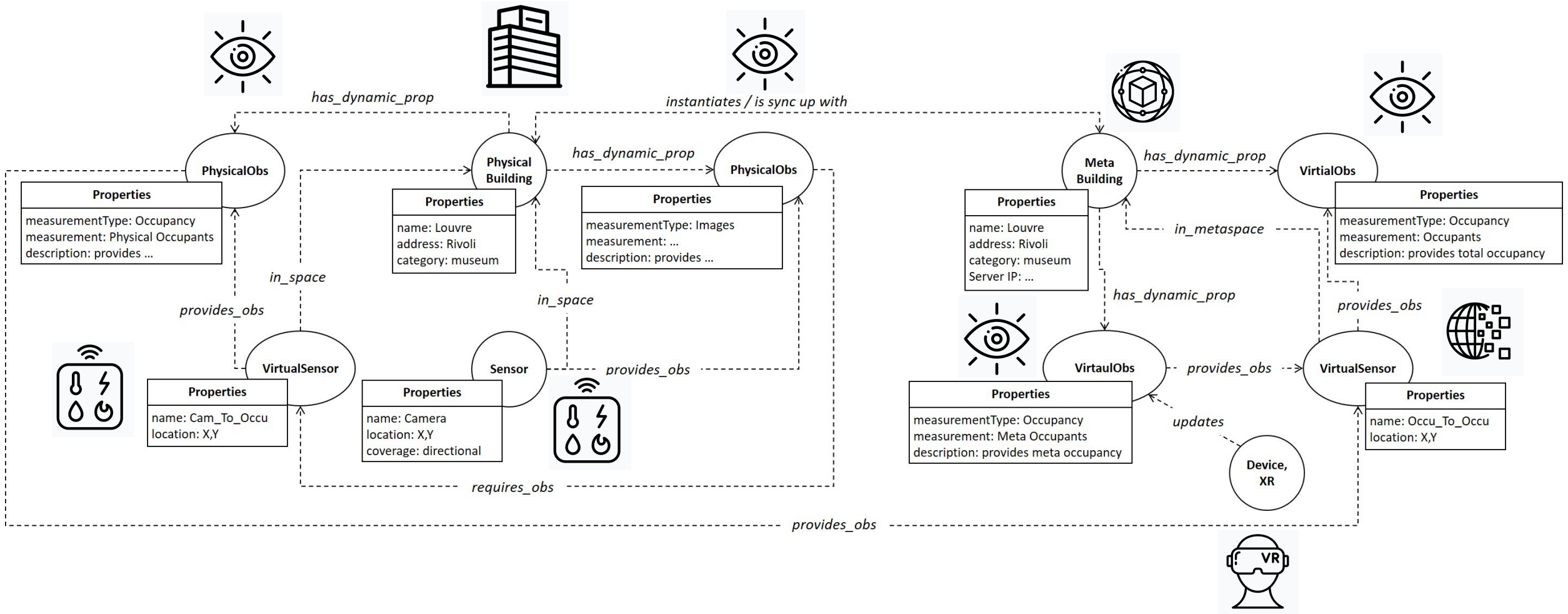
- The physical Person instantiates the Virtual Person and keeps in synchronization with changes.
- The Meta Space has other participants, which we refer to as Meta Persons (accessing the Metaverse via XR devices).
- Interactions and events from the Meta person's space should also be instantiated.



An expanded version of the schema can be accessed here:
https://github.com/SAMSGBLab/metaverse--donna

Louvre Use Case

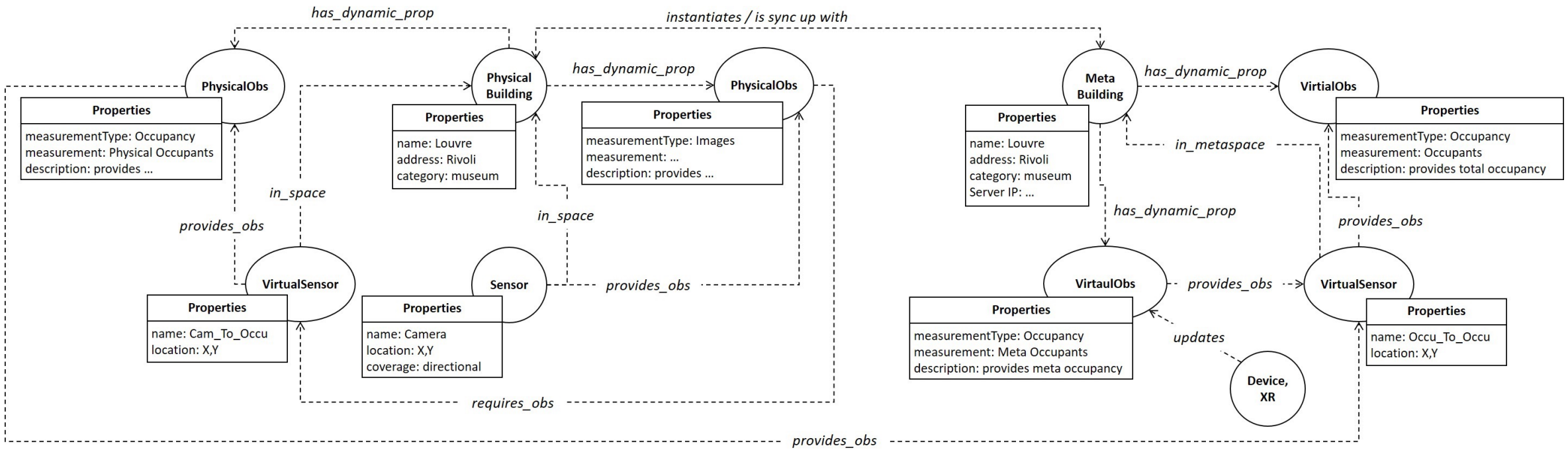
Louvre Use Case: Occupancy



Louvre Use Case

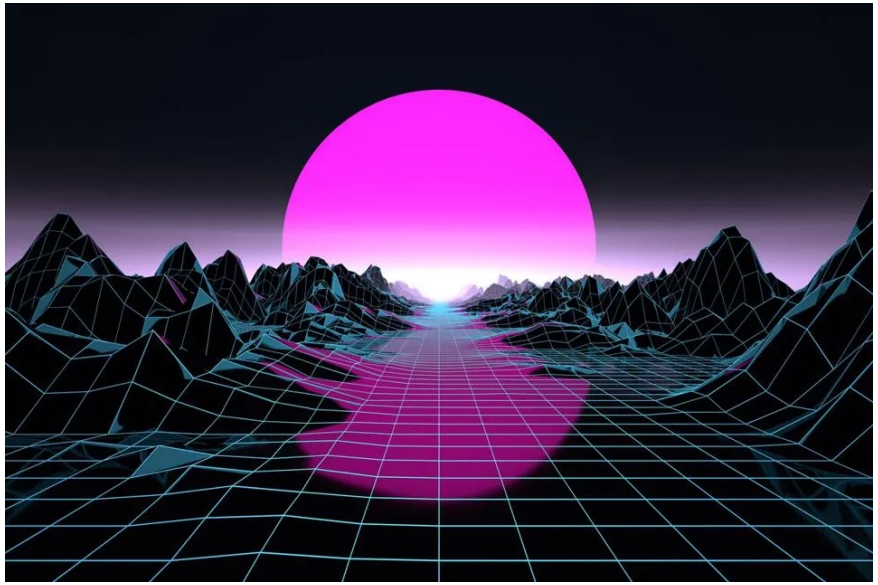
Note that temporary events or changes in crowd distribution are captured and rendered dynamically

1. visit an exact replica of the Louvre museum including physical participants
2. create a cordoned off area any only meet virtual participants with the Louvre background
3. a combination of the cases



Extensions

- *Portability*: Defining data models facilitates the deployment of Metaverse applications across real spaces (implemented via NGSI-LD)
- *Intelligent data processing:* Defining generic properties of smart spaces (occupancy, location, etc.) enables advanced learning methodologies such as federated learning or on swarm intelligence
- *Superior synchronization*: The DONNA data model helps to reduce the data load that must be synchronized between instances



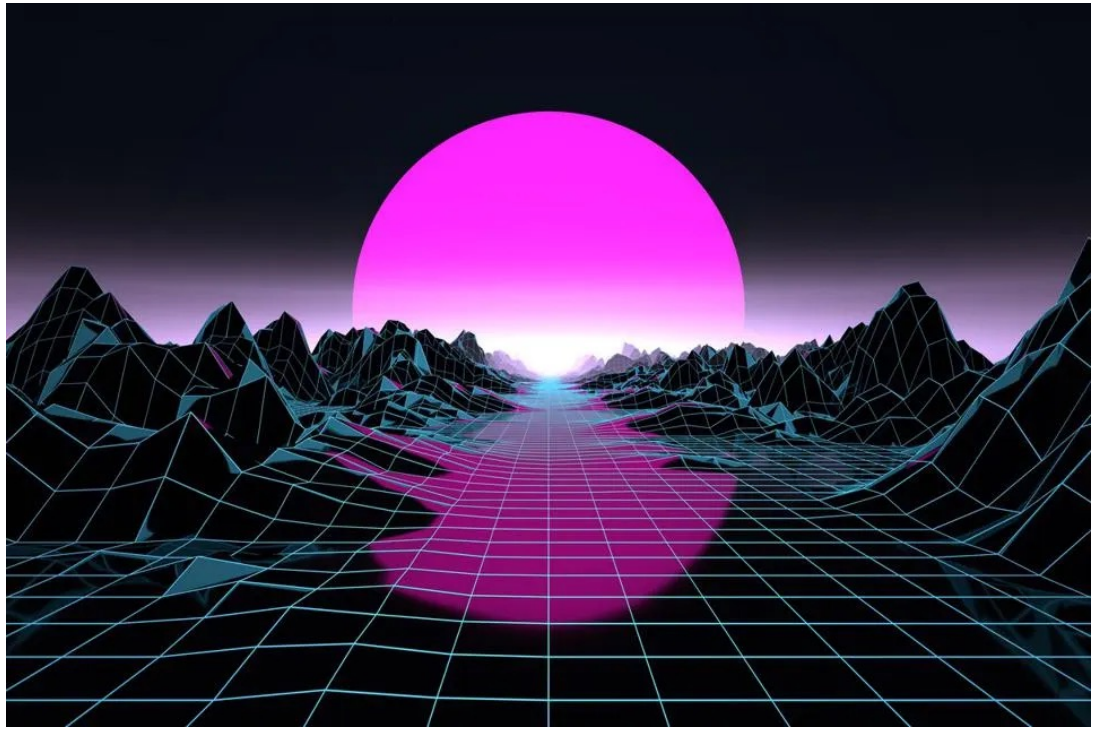
https://github.com/FIWARE/tutorials.NGSI-LD

Conclusions

Conclusions

- Metaverse: critical to specify data models for interoperability and extensibility
- DONNA:
  - Via property graphs, models on physical spaces, devices, meta-spaces, physical participants, virtual participants and property changes
  - The extensibility of the formalism is demonstrated over a virtual visit to the Louvre museum
  - The data models allow developers to semantically annotate interactions and efficiently synchronize between physical/virtual worlds

Future Directions: Implement with NGSI-LD and perform experimental evaluation



ajay.kattepur@ericsson.com