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

Georgios Bouloukakis & Ajay Kattepur

April 24, 2023

Examples of data schemas and property graphs for the Louvre Museum use case.

1 Physical Spaces

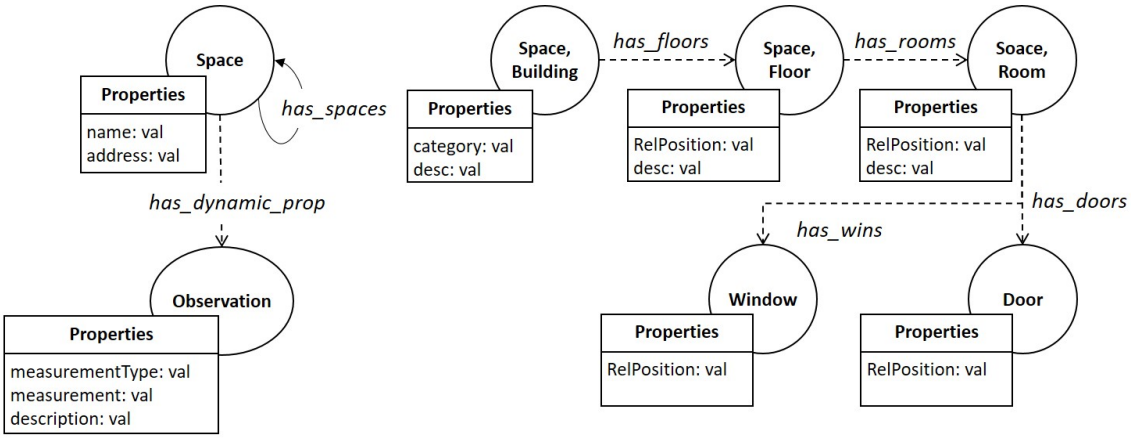

Figure 1: A data schema for Physical spaces.

Table 1: A data schema for Physical spaces.

T_N = {Space, Observation, Building, Floor, Room, Window, Door}
T_E = {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}

2 Devices

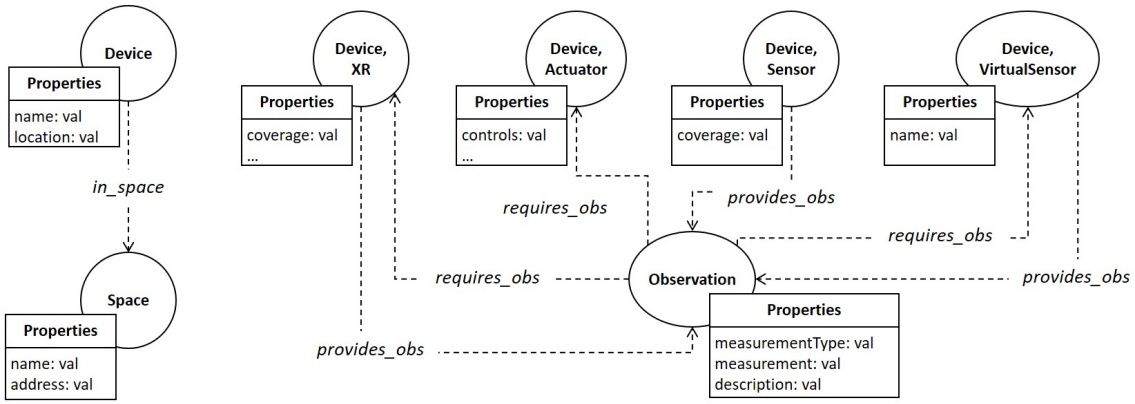

Figure 2: A data schema for Devices.

Table 2: A data schema for Devices.

T_N = {Device, Space, XR, Actuator, Sensor, VirtualSensor, Observation}
T_E = { in_space, provides_obs, requires_obs }

β (Device, name) = val
β (Device, location) = val
β (Space, name) = val
β (Space, address) = val
β (XR, coverage) = val
β (Actuator, controls) = val
β (Sensor, coverage) = val
β (VirtualSensor, name) = val
β (Observation, measurementType) = val
β (Observation, measurement) = val
β (Observation, description) = val

δ (Device, Space) = {in_space}
δ (XR, Observation) = { provides_obs}
δ (Observation, XR) = { requires_obs}
δ (Observation, Actuator) = { requires_obs}
δ (Sensor, Observation) = { provides_obs}
δ (Observation, VirtualSensor) = { requires_obs}
δ (VirtualSensor, Observation) = { provides_obs}

3 Metaspaces

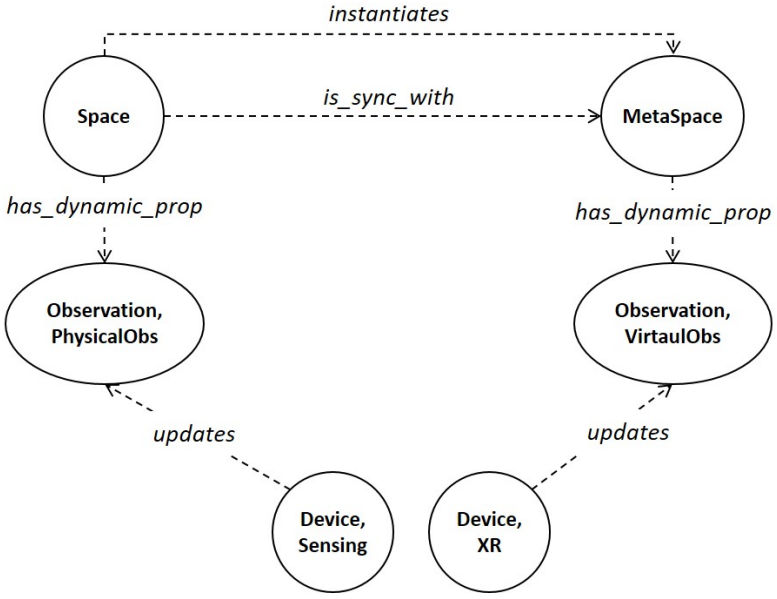

Figure 3: A data schema for Metaspaces.

Table 3: A data schema for Metaspaces.

T_N = {Space, MetaSpace, PhysicalObs, VirtualObs, Sensing, XR}
T_E = { instantiates, in_sync_with, has_dynamic_prop, updates }

δ (Space, MetaSpace) = {instantiates }
δ (Space, MetaSpace) = {in_sync_with}
δ (Space, PhysicalObs) = {has_dynamic_prop}
δ (MetaSpace, VirtualObs) = {has_dynamic_prop}
δ (Sensing, PhysicalObs) = { updates}
δ (Device, VirtualObs) = { updates}

4 VirtualPerson, MetaPerson

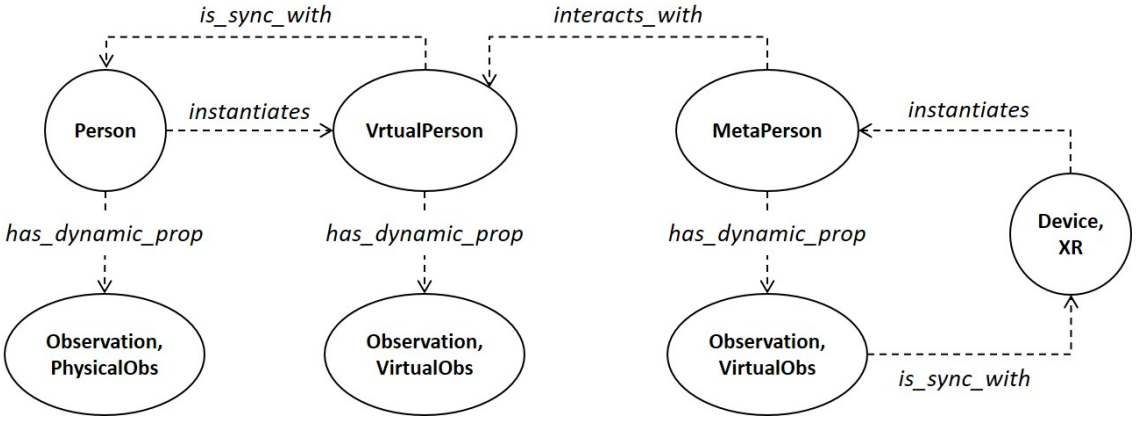

Figure 4: A data schema for Virtualperson and Metaperson.

Table 4: A data schema for Virtualperson and Metaperson.

T_N = {Person, VirtualPerson, MetaPerson, PhysicalObs, VirtualObs, XR}
T_E = {instantiates, in_sync_with, interacts_with, has_dynamic_prop}

δ (Person, VirtualPerson) = {instantiates }
δ (VirtualPerson, Person) = {in_sync_with}
δ (MetaPerson, VirtualPerson) = {interacts_with}
δ (XR, MetaPerson) = {instantiates}
δ (Person, PhysicalObs) = {has_dynamic_prop}
δ (VirtualPerson, VirtualObs) = {has_dynamic_prop}
δ (MetaPerson, VirtualObs) = {has_dynamic_prop}
δ (VirtualObs, XR) = {in_sync_with}

5 Metaverse Interactions

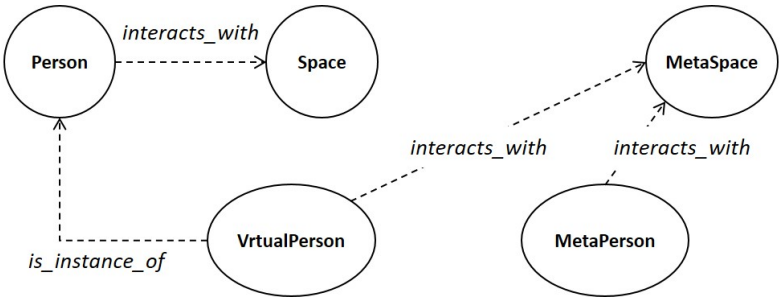

Figure 5: A data schema for Metaverse Interactions.

Table 5: A data schema for Metaverse Interactions.

T_N = {Person, VirtualPerson, Space, MetaSpace, MetaPerson}
T_E = {interacts_with, is_instance_of}

δ (Person, Space) = {interacts_with}
δ (VirtualPerson, Person) = {is_instance_of}
δ (VirtualPerson, MetaSpace) = {interacts_with}
δ (MetaPerson, MetaSpace) = {interacts_with}

6 Louvre Museum Visit Property Graph

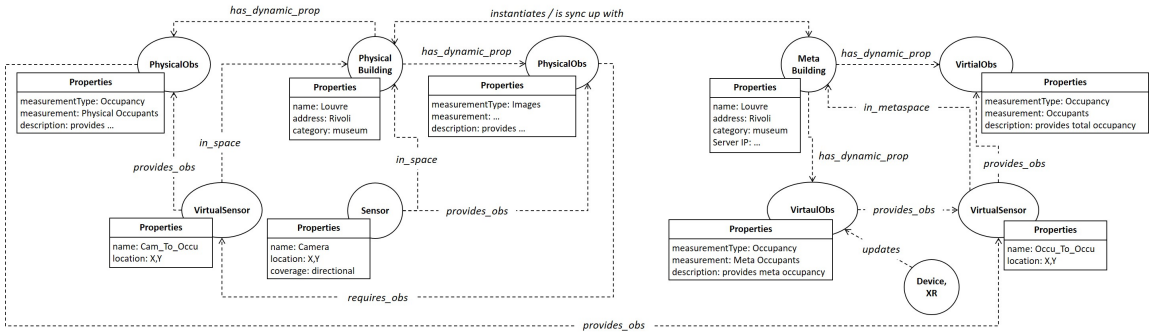

Figure 6: Property Graph for Measuring Physical and Virtual Occupancy in Louvre.

Table 6: Property Graph for Measuring Physical and Virtual Occupancy in Louvre.

$N = \{n_1, n_2, n_3, n_4, n_5, n_6, n_7, n_8, n_9, n_{10}\}$
$E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7\}$

$\lambda(n_1) = \{\text{Building}\}$;
$\sigma(n_1, \text{id}) = \text{'id'}$; $\sigma(n_1, \text{type}) = \text{'abstract'}$;
$\sigma(n_1, \text{name}) = \text{'Louvre'}$; $\sigma(n_1, \text{address}) = \text{'Rivoli'}$;
$\sigma(n_1, \text{Desc}) = \text{''}$; $\sigma(n_1, \text{category}) = \text{'museum'}$;

$\lambda(n_2) = \{\text{Observation}\}$;
$\sigma(n_2, \text{id}) = \text{'id'}$; $\sigma(n_2, \text{type}) = \text{'physical'}$;
$\sigma(n_2, \text{measurementType}) = \text{'images'}$; $\sigma(n_2, \text{measurement}) = \text{'val'}$;
$\sigma(n_2, \text{description}) = \text{'provides'}$;

$\lambda(n_3) = \{\text{Observation2}\}$;
$\sigma(n_3, \text{id}) = \text{'id'}$; $\sigma(n_3, \text{type}) = \text{'physical'}$;
$\sigma(n_3, \text{measurementType}) = \text{'occupancy'}$; $\sigma(n_3, \text{measurement}) = \text{'val'}$;
$\sigma(n_2, \text{description}) = \text{'provides'}$;

$\lambda(n_4) = \{\text{VirtualSensor}\}$;
$\sigma(n_4, \text{id}) = \text{'id'}$; $\sigma(n_4, \text{type}) = \text{'virtual'}$;
$\sigma(n_4, \text{name}) = \text{'Cam\_To\_Occu'}$; $\sigma(n_4, \text{location}) = \text{'X,Y'}$

$\lambda(n_5) = \{\text{Sensor}\}$;
$\sigma(n_5, \text{id}) = \text{'id'}$; $\sigma(n_5, \text{type}) = \text{'physical'}$;
$\sigma(n_5, \text{coverage}) = \text{'val'}$

$\lambda(n_6) = \{\text{MetaBuilding}\}$;
$\sigma(n_6, \text{id}) = \text{'id'}$; $\sigma(n_6, \text{type}) = \text{'abstract'}$;
$\sigma(n_6, \text{name}) = \text{'Louvre'}$; $\sigma(n_6, \text{address}) = \text{'Rivoli'}$;
$\sigma(n_6, \text{Desc}) = \text{''}$; $\sigma(n_6, \text{category}) = \text{'museum'}$;

$\lambda(n_7) = \{\text{VirtualObservation}\}$;
$\sigma(n_7, \text{id}) = \text{'id'}$; $\sigma(n_7, \text{type}) = \text{'virtual'}$;
$\sigma(n_7, \text{measurementType}) = \text{'Occupancy'}$;
$\sigma(n_7, \text{measurement}) = \text{'Occupants'}$;
$\sigma(n_7, \text{description}) = \text{'provides total occupancy'}$;

$\lambda(n_8) = \{\text{VirtualObservation2}\}$;
$\sigma(n_8, \text{id}) = \text{'id'}$; $\sigma(n_8, \text{type}) = \text{'virtual'}$;
$\sigma(n_8, \text{measurementType}) = \text{'occupancy'}$;
$\sigma(n_8, \text{measurement}) = \text{'MetaOccupants'}$;
$\sigma(n_8, \text{description}) = \text{'provides meta occupancy'}$;

$\lambda(n_9) = \{\text{VirtualSensor}\}$;
$\sigma(n_9, \text{id}) = \text{'id'}$; $\sigma(n_9, \text{type}) = \text{'virtual'}$;
$\sigma(n_4, \text{name}) = \text{'Occu\_To\_Occu'}$; $\sigma(n_4, \text{location}) = \text{'X,Y'}$

$\lambda(n_{10}) = \{\text{XR}\}$;
$\sigma(n_{10}, \text{id}) = \text{'id'}$; $\sigma(n_{10}, \text{type}) = \text{'device'}$;

$\rho(e_1) = (n_1, n_3)$; $\lambda(e_1) = \{\text{has\_dynamic\_property}\}$;
$\rho(e_2) = (n_1, n_2)$; $\lambda(e_2) = \{\text{has\_dynamic\_property}\}$;
$\rho(e_3) = (n_4, n_1)$; $\lambda(e_3) = \{\text{in\_space}\}$;
$\rho(e_4) = (n_5, n_1)$; $\lambda(e_4) = \{\text{in\_space}\}$;
$\rho(e_5) = (n_5, n_2)$; $\lambda(e_5) = \{\text{provides\_obs}\}$;
$\rho(e_6) = (n_2, n_4)$; $\lambda(e_6) = \{\text{requires\_obs}\}$;
$\rho(e_7) = (n_4, n_3)$; $\lambda(e_7) = \{\text{provides\_obs}\}$;
$\rho(e_8) = (n_1, n_6)$; $\lambda(e_8) = \{\text{instantiates}\}$;
$\rho(e_9) = (n_6, n_1)$; $\lambda(e_9) = \{\text{in sync up with}\}$;
$\rho(e_{10}) = (n_3, n_9)$; $\lambda(e_{10}) = \{\text{provides obs}\}$;
$\rho(e_{11}) = (n_6, n_8)$; $\lambda(e_{11}) = \{\text{has dynamic prop}\}$;
$\rho(e_{12}) = (n_6, n_7)$; $\lambda(e_{12}) = \{\text{has dynamic prop}\}$;
$\rho(e_{13}) = (n_9, n_6)$; $\lambda(e_{13}) = \{\text{in metaspace}\}$;
$\rho(e_{14}) = (n_9, n_7)$; $\lambda(e_{14}) = \{\text{provides obs}\}$;
$\rho(e_{15}) = (n_8, n_9)$; $\lambda(e_{15}) = \{\text{provides obs}\}$;
$\rho(e_{16}) = (n_{10}, n_8)$; $\lambda(e_{16}) = \{\text{updates}\}$;