interconnect

interoperable solutions
connecting smart homes,
buildings and grids

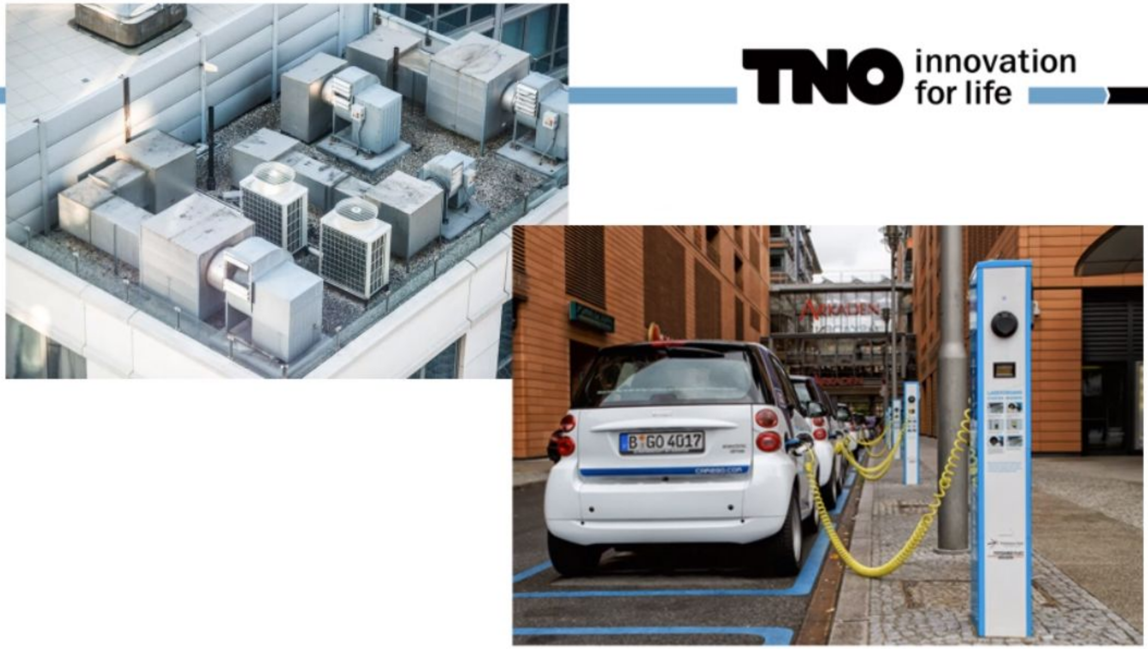
Learning and Reasoning over Smart Home Knowledge Graphs

Ronald Siebes, Victor de Boer, Roberto Reda, Roderick van der Weerdt

Interoperable solutions connecting smart homes, buildings and grids

KNX
DAIKIN
Vaillant
Wirelane
Miele
B/S/H/
Fachhochschule Dortmund
University of Applied Sciences and Arts
DFKI
Deutsches Forschungszentrum für Künstliche Intelligenz GmbH
UNIKASSEL VERSITÄT
ABB
Keo
OpenMotics
E.DSO
inetum.
Positive digital flow
CYBERGRID
POLITECNICO MILANO 1863
RSE
Ricerca Sistema Energetico
sensinov
ENGIE
ENEDIS
L'ELECTRICITE EN RESEAU
COSMOTE
HERON
VU
VRIJE UNIVERSITEIT AMSTERDAM
Cordium
eerste partner in sociaal wonen
3E
DuCoop
imec

LARGE VS. SMALL



Larger, industrial installations

- This is already being done
- Relatively high margin per unit
- Custom control technology developed for every unit
- Process is well known and very predictable

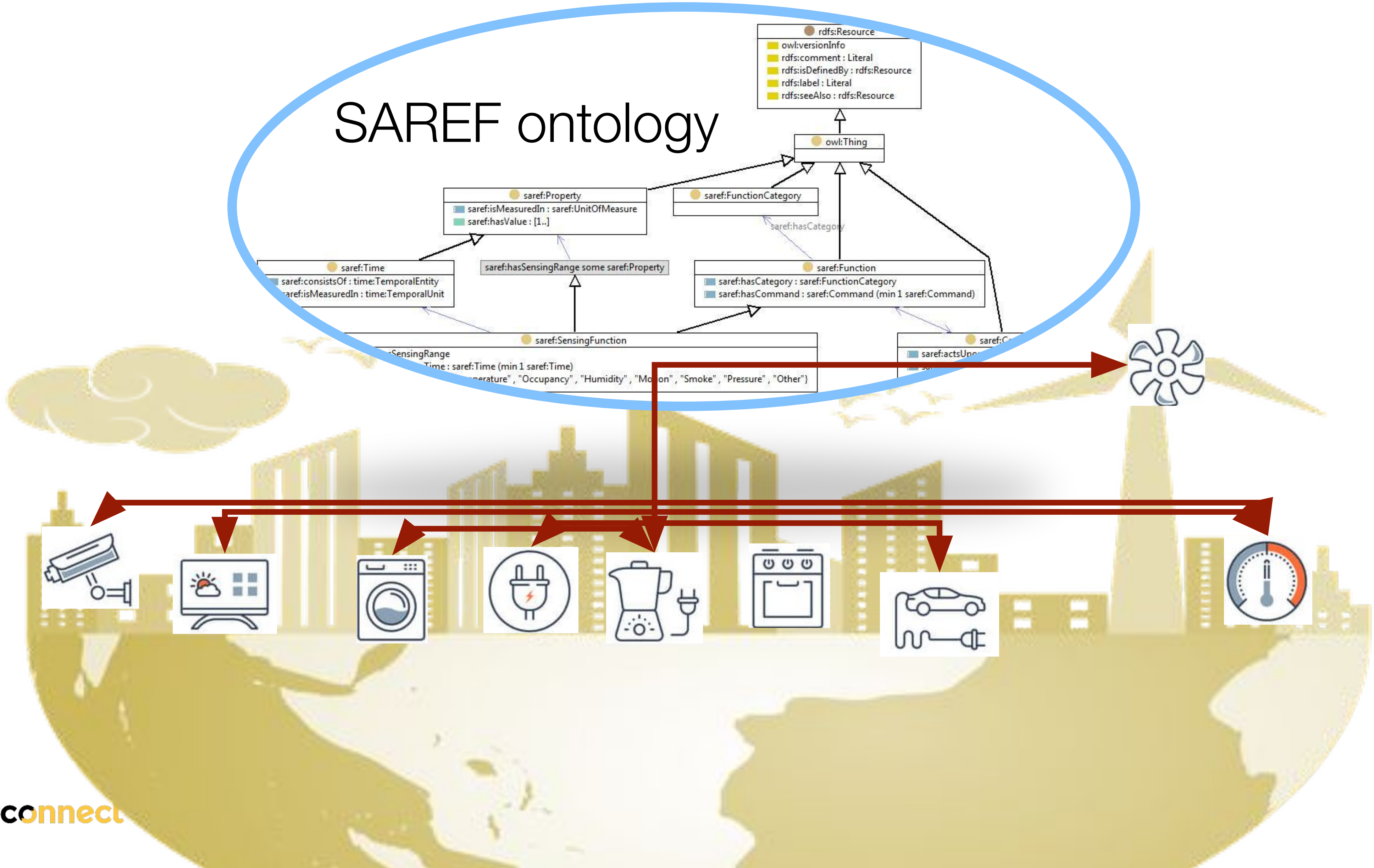
- Already profitable
- Standardization may improve efficiency

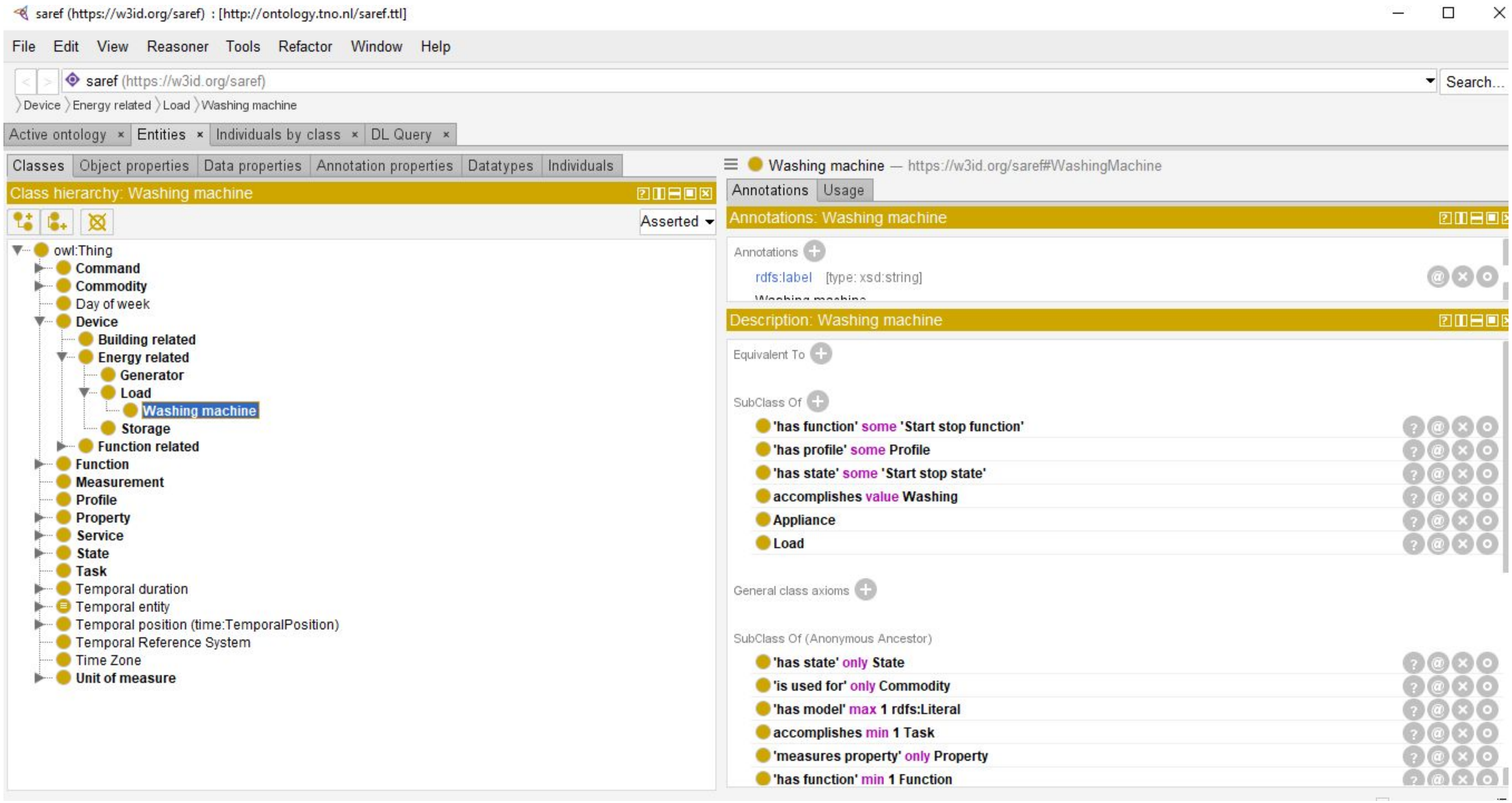
Smaller, residential/office devices

- Currently in development
- Many makes and models of devices providing energy flexibility
- Many units, but very low margin per unit
- Energy flexibility is made available by ‘best effort’; no guarantees

- If we want to scale up, we need to minimize installation and operational costs

SAREF ontology


rdfs:Resource
owl:versionInfo
rdfs:comment : Literal
rdfs:isDefinedBy : rdfs:Resource
rdfs:label : Literal
rdfs:seeAlso : rdfs:Resource
owl:Thing
saref:Property
saref:isMeasuredIn : saref:UnitOfMeasure
saref:hasValue : [1..]
saref:FunctionCategory
saref:hasCategory
saref:Time
saref:consistsOf : time:TemporalEntity
aref:isMeasuredIn : time:TemporalUnit
saref:hasSensingRange some saref:Property
saref:Function
saref:hasCategory : saref:FunctionCategory
saref:hasCommand : saref:Command (min 1 saref:Command)
saref:SensingFunction
SensingRange
Time : saref:Time (min 1 saref:Time)
erature" , "Occupancy" , "Humidity" , "Mo on" , "Smoke" , "Pressure" , "Other"}
saref:C
saref:actsUp



Class hierarchy: Washing machine

- owl:Thing
  - Command
  - Commodity
  - Day of week
  - Device
    - Building related
    - Energy related
      - Generator
      - Load
        - Washing machine
      - Storage
    - Function related
  - Function
  - Measurement
  - Profile
  - Property
  - Service
  - State
  - Task
  - Temporal duration
  - Temporal entity
  - Temporal position (time:TemporalPosition)
  - Temporal Reference System
  - Time Zone
  - Unit of measure

Washing machine — https://w3id.org/saref#WashingMachine

Annotations: Washing machine

Annotations

rdfs:label [type: xsd:string]

Washing machine

Description: Washing machine

Equivalent To

SubClass Of

- 'has function' some 'Start stop function'
- 'has profile' some Profile
- 'has state' some 'Start stop state'
- accomplishes value Washing
- Appliance
- Load

General class axioms

SubClass Of (Anonymous Ancestor)

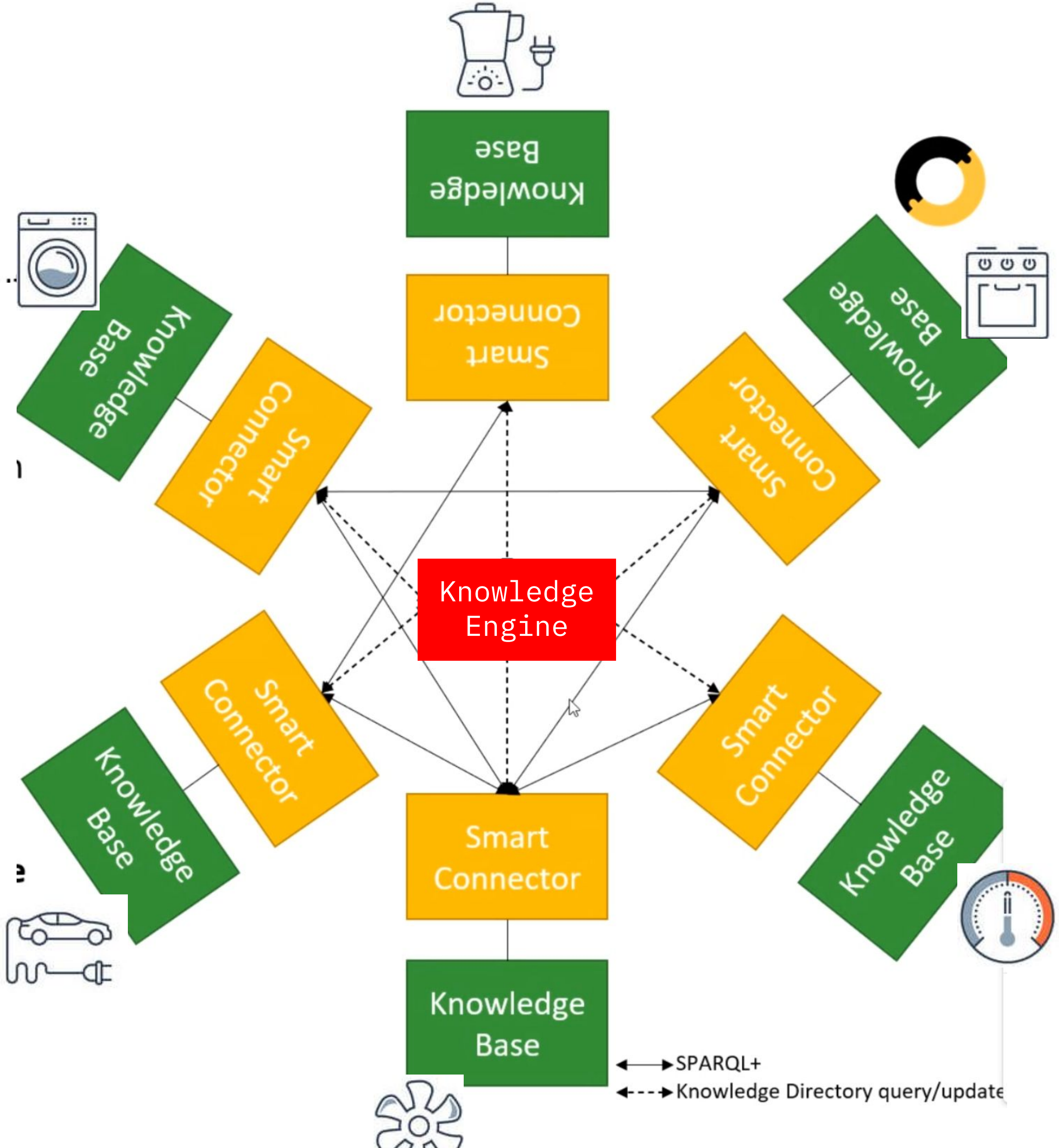
- 'has state' only State
- 'is used for' only Commodity
- 'has model' max 1 rdfs:Literal
- accomplishes min 1 Task
- 'measures property' only Property
- 'has function' min 1 Function

http://ontology.tno.nl/saref.ttl

Knowledge engine

Pub-Sub matchmaking on graph patterns

using the SAREF ontology



interconnect

Challenge: Machine Learning over Smart home KGs

Explainability:

using Jupyter Notebooks as RESTful service

(https://github.com/rsiebes/interconnect-explainability)

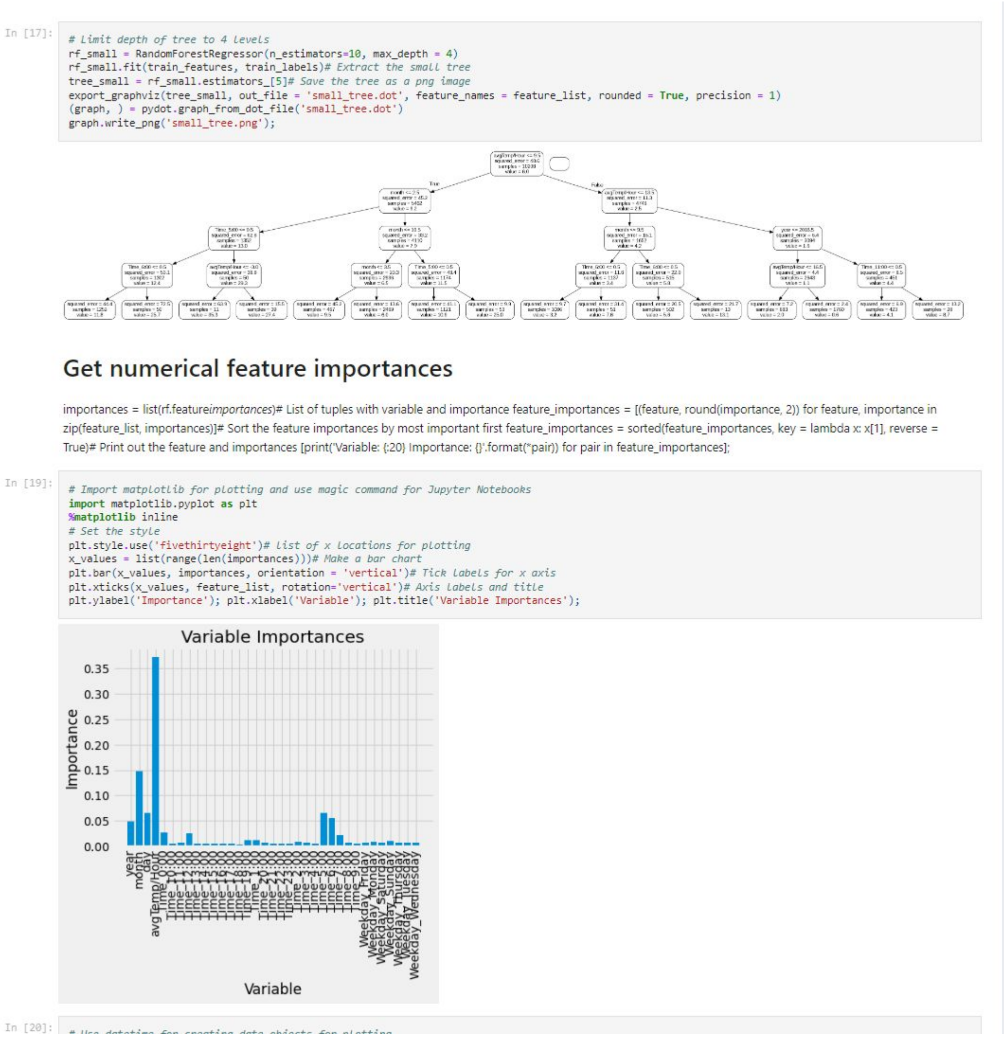
Heterogeneous data:

serialize SAREF types to encode the data to features

(e.g. ‘One-hot’ encoding, time-stamps to year, month, day, hour features etc)

Graph Learning:

Applying Graph Neural Networks for value prediction, classification, error detection



In [17]:

```
# Limit depth of tree to 4 levels
rf_small = RandomForestRegressor(n_estimators=10, max_depth = 4)
rf_small.fit(train_features, train_labels)# Extract the small tree
tree_small = rf_small.estimators_[5]# Save the tree as a png image
export_graphviz(tree_small, out_file = 'small_tree.dot', feature_names = feature_list, rounded = True, precision = 1)
(graph, ) = pydot.graph_from_dot_file('small_tree.dot')
graph.write_png('small_tree.png');
```

Get numerical feature importances

importances = list(rf.feature*importances*)# List of tuples with variable and importance feature_importances = [(feature, round(importance, 2)) for feature, importance in zip(feature_list, importances)]# Sort the feature importances by most important first feature_importances = sorted(feature_importances, key = lambda x: x[1], reverse = True)# Print out the feature and importances [print('Variable: {:20} Importance: {}'.format(*pair)) for pair in feature_importances];

In [19]:

```
# Import matplotlib for plotting and use magic command for Jupyter Notebooks
import matplotlib.pyplot as plt
%matplotlib inline
# Set the style
plt.style.use('fivethirtyeight')# list of x locations for plotting
x_values = list(range(len(importances)))# Make a bar chart
plt.bar(x_values, importances, orientation = 'vertical')# Tick labels for x axis
plt.xticks(x_values, feature_list, rotation='vertical')# Axis labels and title
plt.ylabel('Importance'); plt.xlabel('Variable'); plt.title('Variable Importances');
```

In [20]:

```
# Use datetime for creating data objects for plotting
```

Challenge: Machine Learning over Smart home KGs

Explainability:

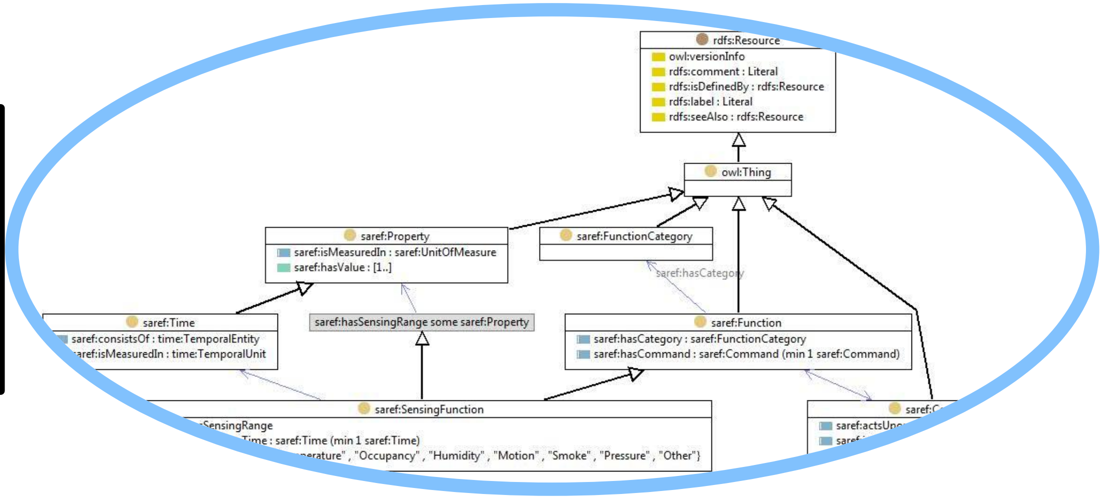
using Jupyter Notebooks as RESTful service

(https://github.com/rsiebes/interconnect-explainability)

Heterogeneous data:

serialize SAREF types to encode the data to features

(e.g. ‘One-hot’ encoding, time-stamps to year, month, day, hour features etc)


rdfs:Resource
owl:versionInfo
rdfs:comment : Literal
rdfs:isDefinedBy : rdfs:Resource
rdfs:label : Literal
rdfs:seeAlso : rdfs:Resource
owl:Thing
saref:Property
saref:isMeasuredIn : saref:UnitOfMeasure
saref:hasValue : [1..]
saref:FunctionCategory
saref:hasCategory
saref:Time
saref:consistsOf : time:TemporalEntity
aref:isMeasuredIn : time:TemporalUnit
saref:hasSensingRange some saref:Property
saref:Function
saref:hasCategory : saref:FunctionCategory
saref:hasCommand : saref:Command (min 1 saref:Command)
saref:SensingFunction
SensingRange
Time : saref:Time (min 1 saref:Time)
erature" , "Occupancy" , "Humidity" , "Motion" , "Smoke" , "Pressure" , "Other"}
saref:C
saref:actsUp
saref

Graph Learning:

Applying Graph Neural Networks for value prediction, classification, error detection

Challenge: Machine Learning over Smart home KGs

Explainability:

using Jupyter Notebooks as RESTful service

(https://github.com/rsiebes/interconnect-explainability)

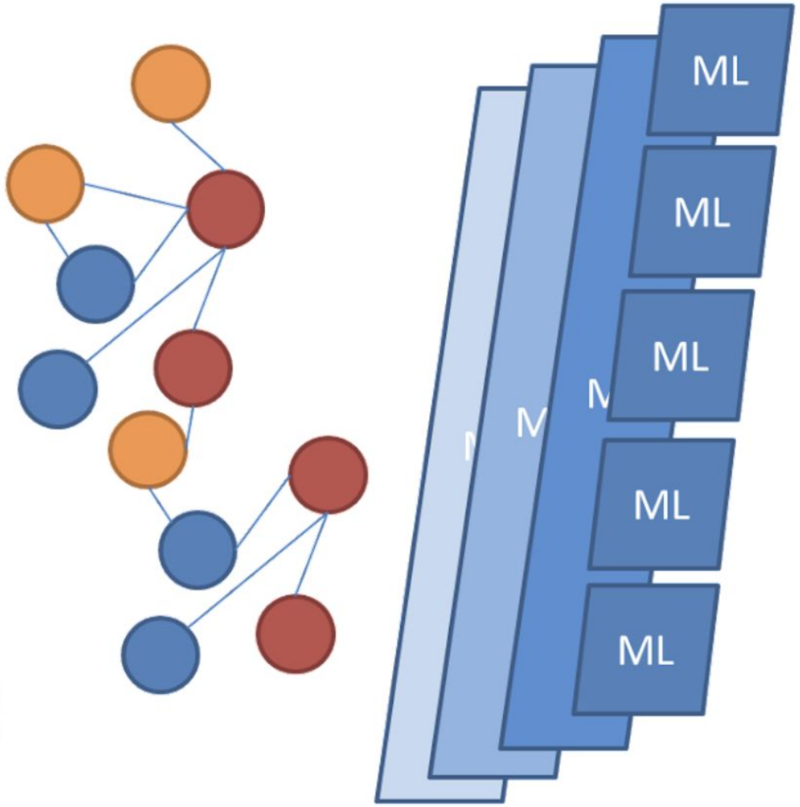
Heterogeneous data:

serialize SAREF types to encode the data to features

(e.g. 'One-hot' encoding, time-stamps to year, month, day, hour features etc)

Graph Learning:

Applying Graph Neural Networks for value prediction, classification, error detection



Challenge: Conversion to SAREF

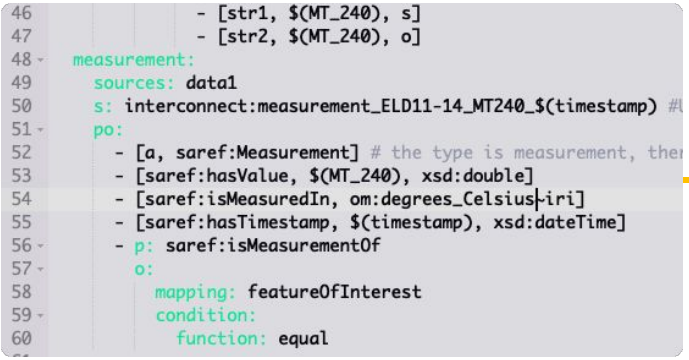
Datasets in RDF were not available

Created new KGs with YARRRML mappings

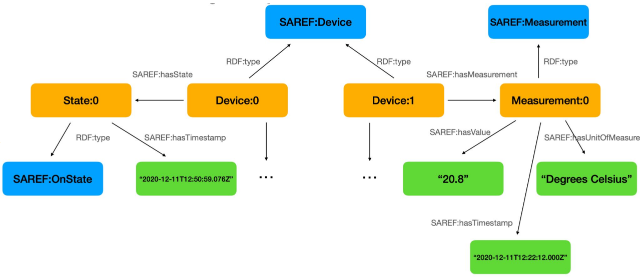
The KGs were validated to contain all the information available in the original dataset

Timestamp	State
2020-12-11T12:50:59.076Z	On
2020-12-11T13:05:23.546Z	Off
2020-12-11T13:20:45.789Z	Off
...	...

```
{
  "id":"1234.5678",
  "type":"Sensor",
  "value":
  {
    "Temperature":
    {
      "value":"20.8",
      "unit":"Degrees Celsius",
      "timestamp":"20201211122212"
    }
  }
}
```

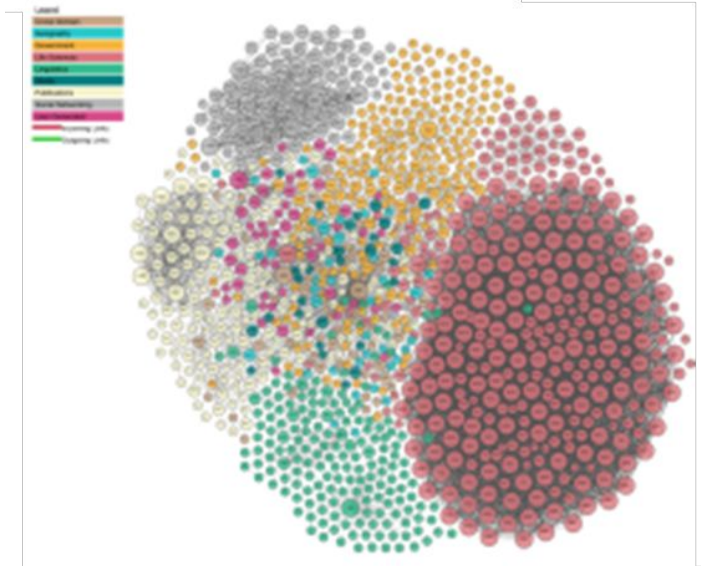


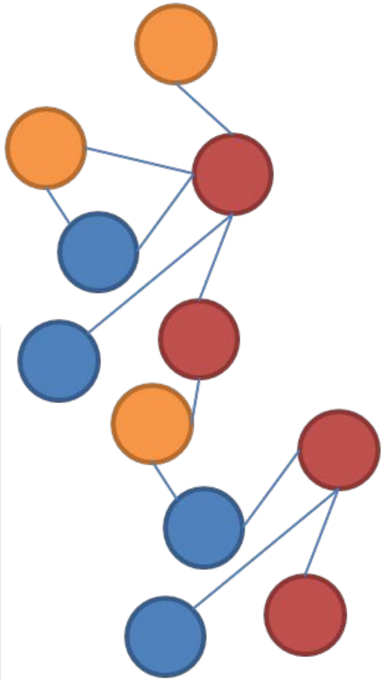
```
              - [str1, $(MT_240), s]
              - [str2, $(MT_240), o]
  measurement:
    sources: data1
    s: interconnect:measurement_ELD11-14_MT240_$(timestamp) #
    po:
      - [a, saref:Measurement] # the type is measurement, ther
      - [saref:hasValue, $(MT_240), xsd:double]
      - [saref:isMeasuredIn, om:degrees_Celsius~iri]
      - [saref:hasTimestamp, $(timestamp), xsd:dateTime]
      - p: saref:isMeasurementOf
        o:
          mapping: featureOfInterest
          condition:
            function: equal
```

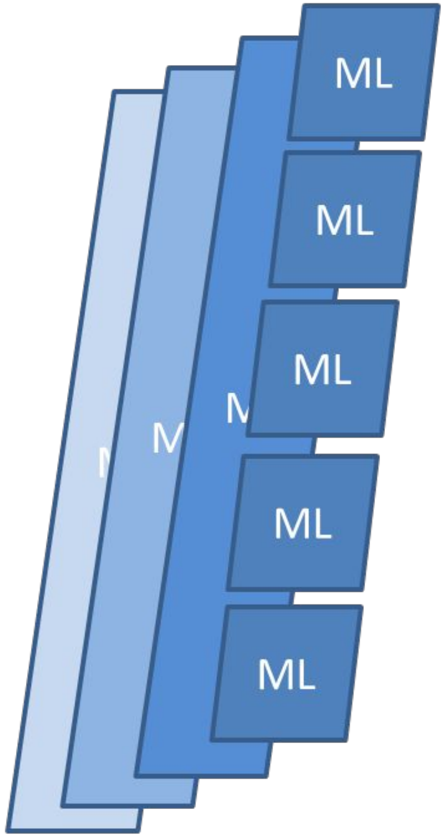


van der Weerdt R., de Boer V., Daniele L., Nouwt B. (2021) Validating SAREF in a Smart Home Environment. In: Garoufallou E., Ovalle-Perandones MA. (eds) Metadata and Semantic Research. MTSR 2020. Communications in Computer and Information Science, vol 1355. Springer, Cham. https://doi.org/10.1007/978-3-030-71903-6_4

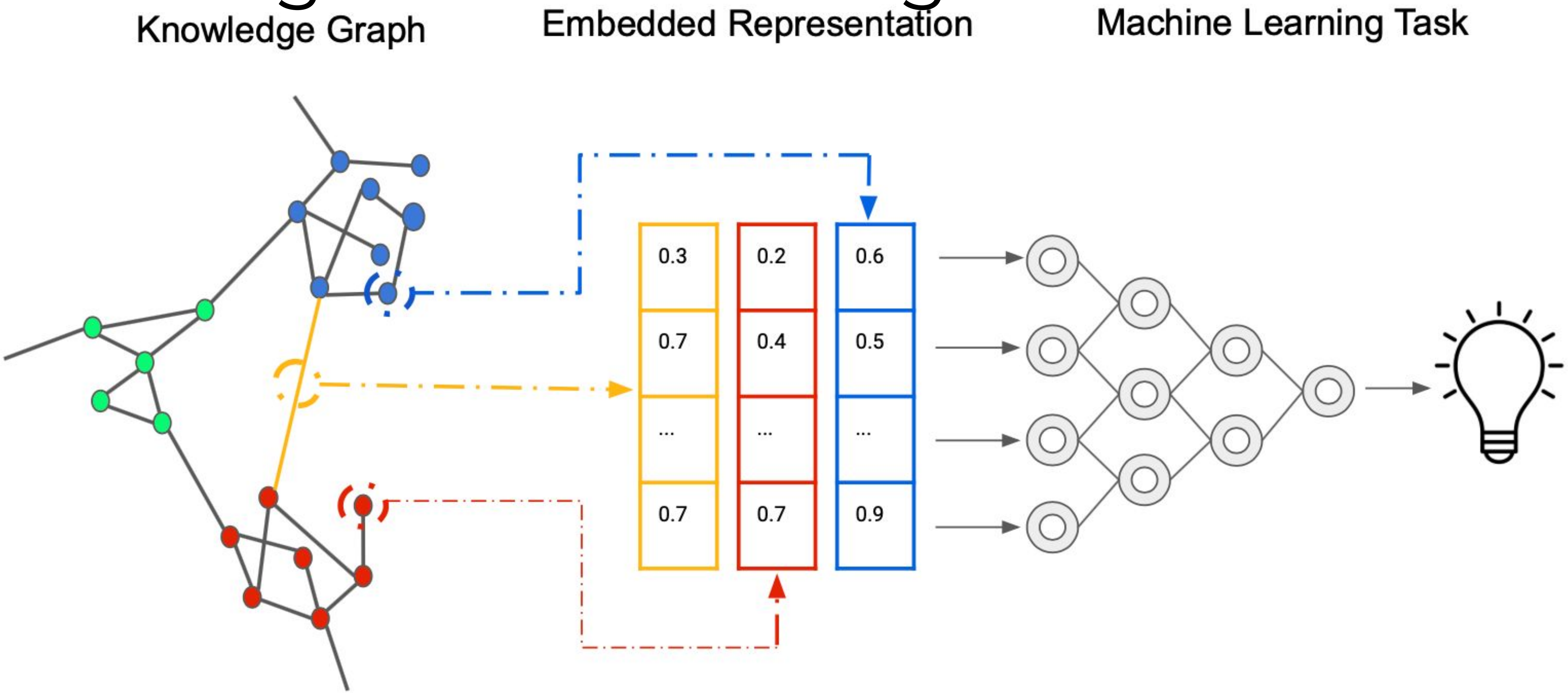
Learning over **Multimodal Knowledge Graphs** for IOT/Smart home







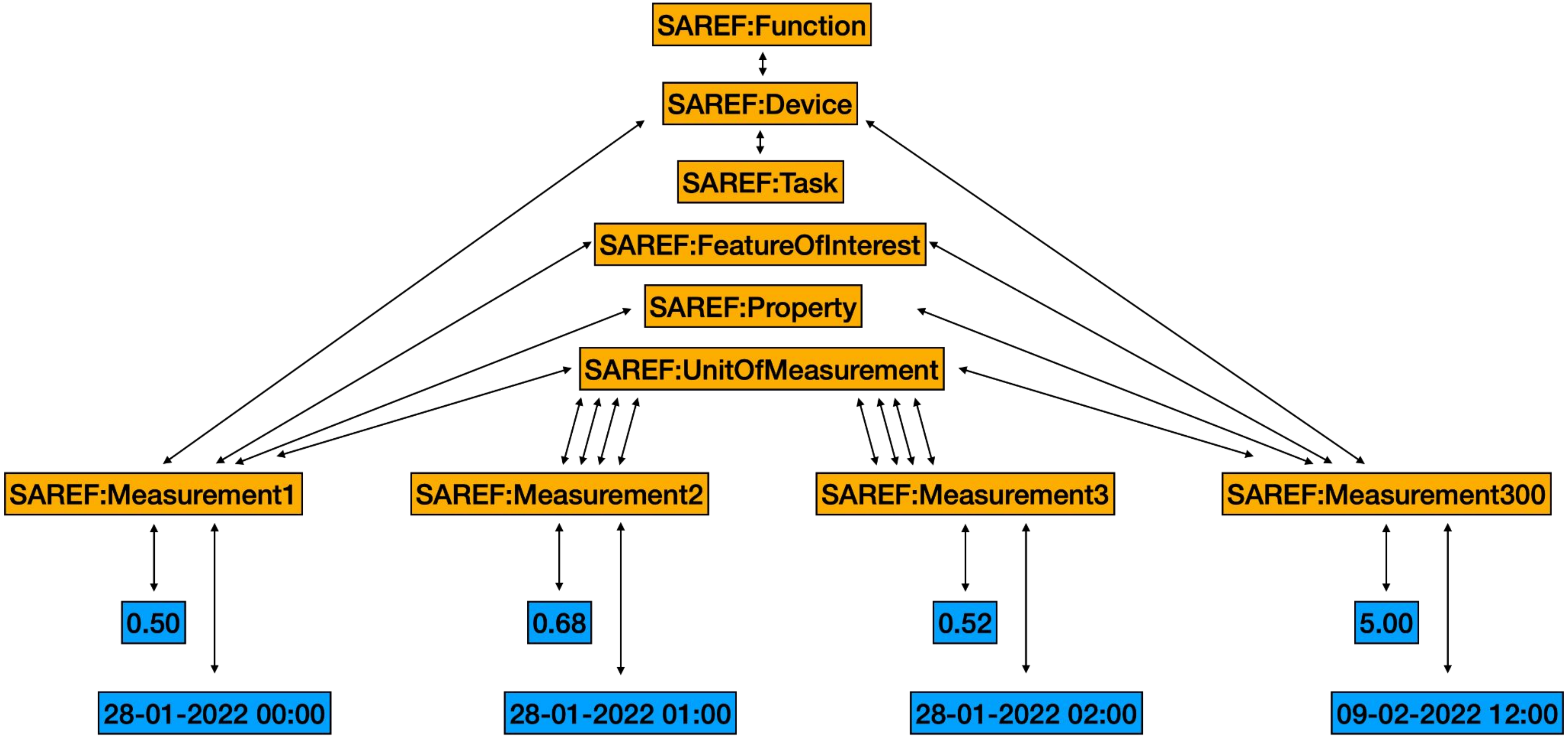
Creating the Embeddings



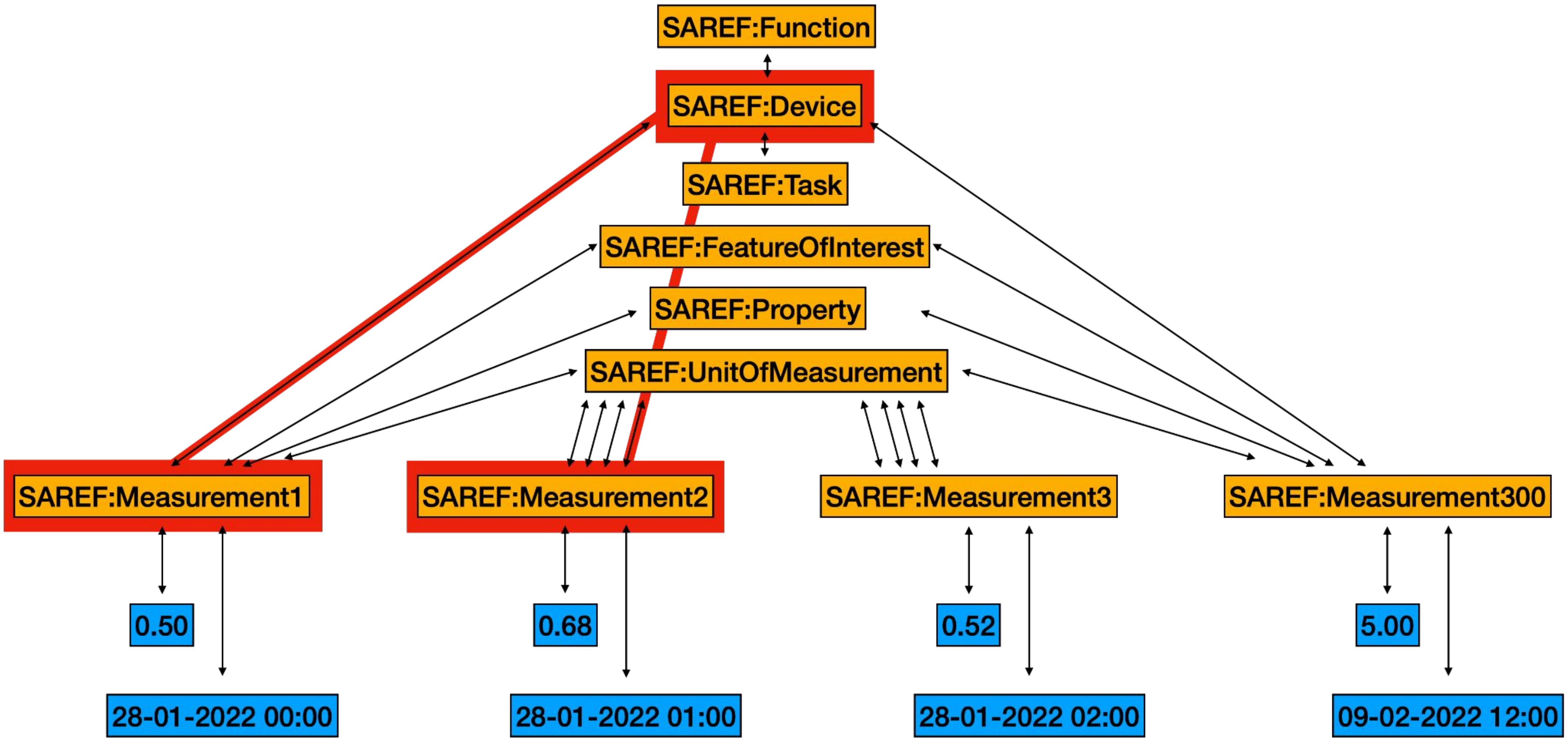
0.3	0.2	0.6
0.7	0.4	0.5
...	...	...
0.7	0.7	0.9

source: https://en.wikipedia.org/wiki/Knowledge_graph_embedding

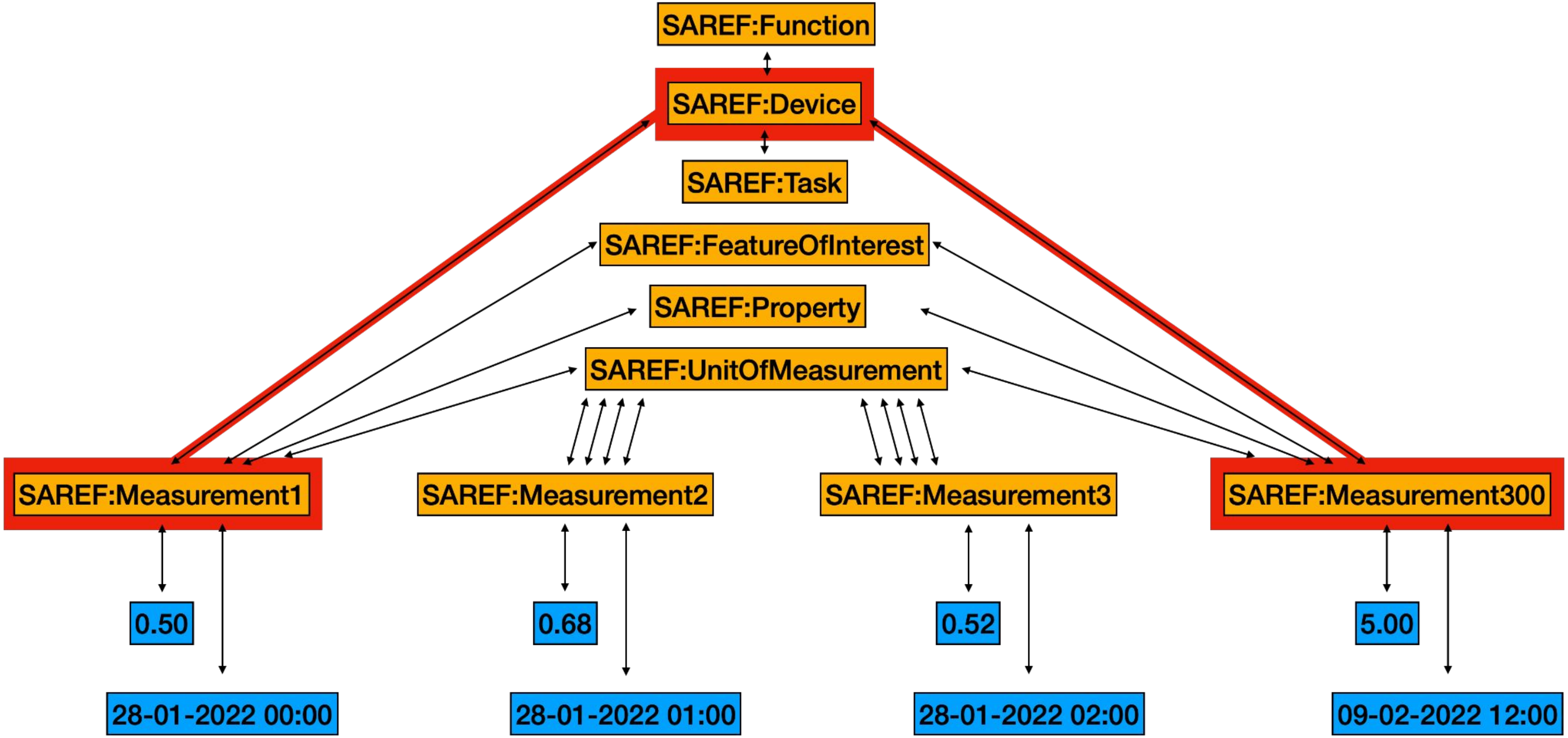
Experiments: Creating the Embeddings



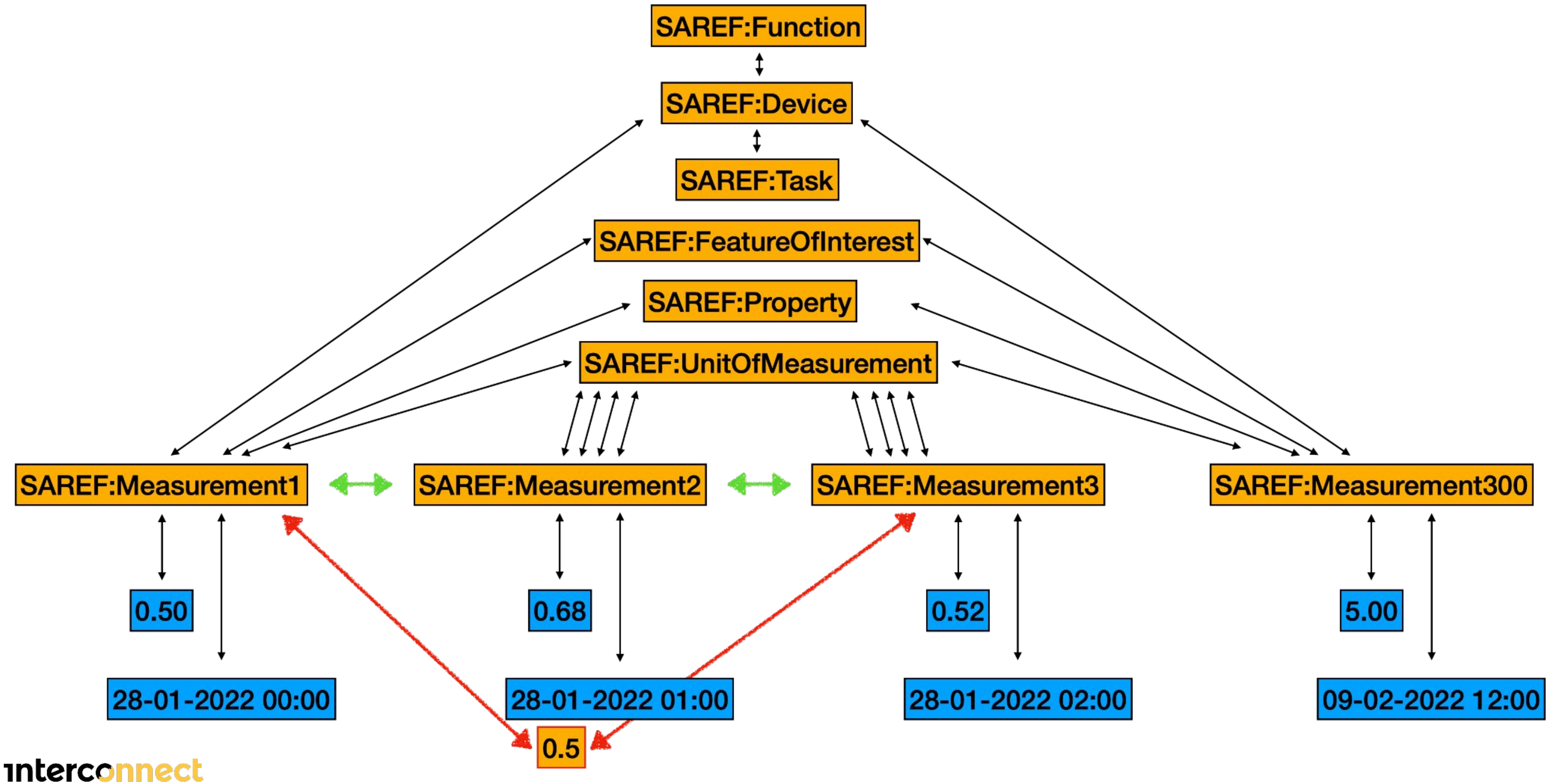
Experiments: Creating the Embeddings



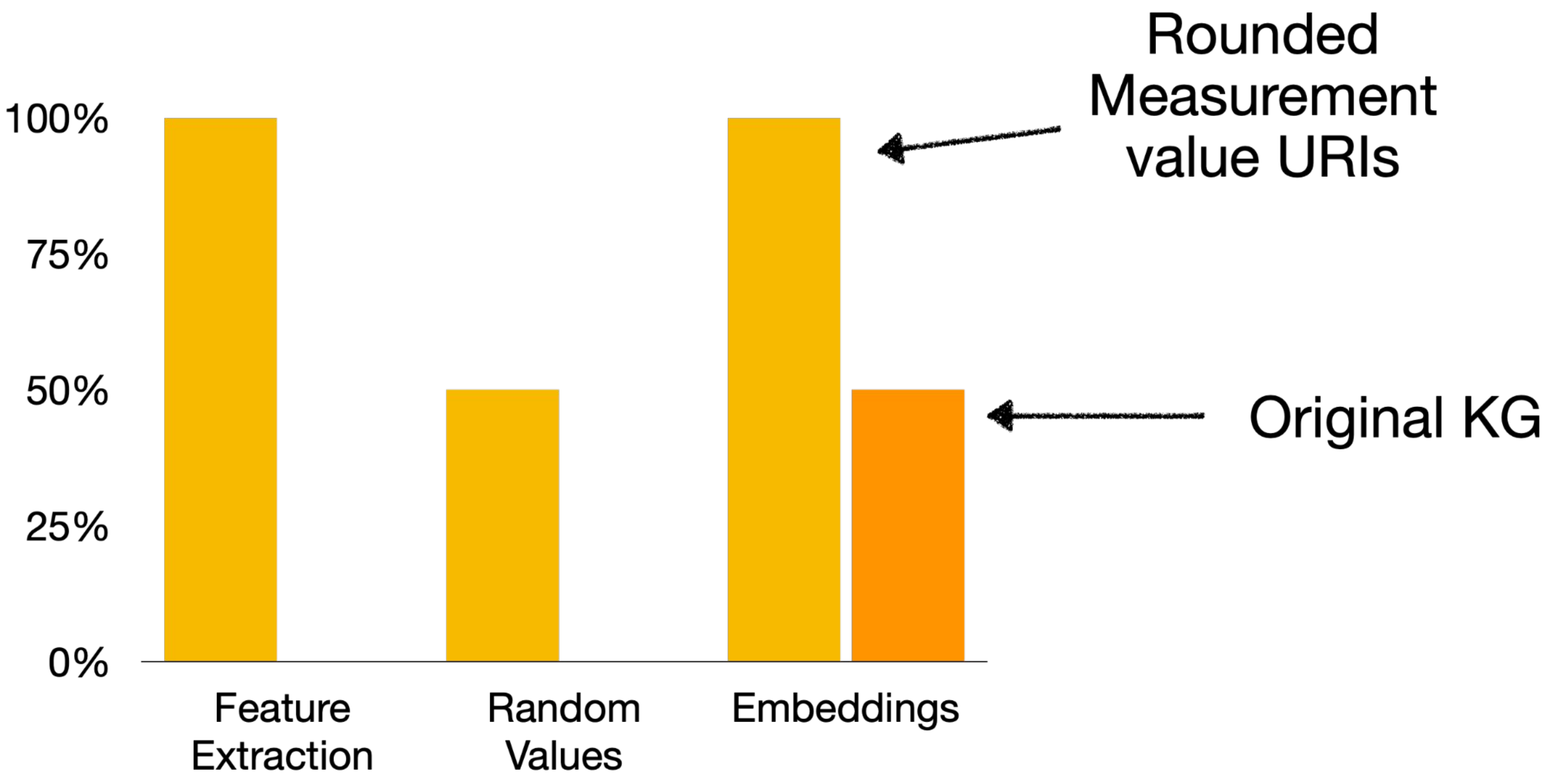
Experiments: Creating the Embeddings



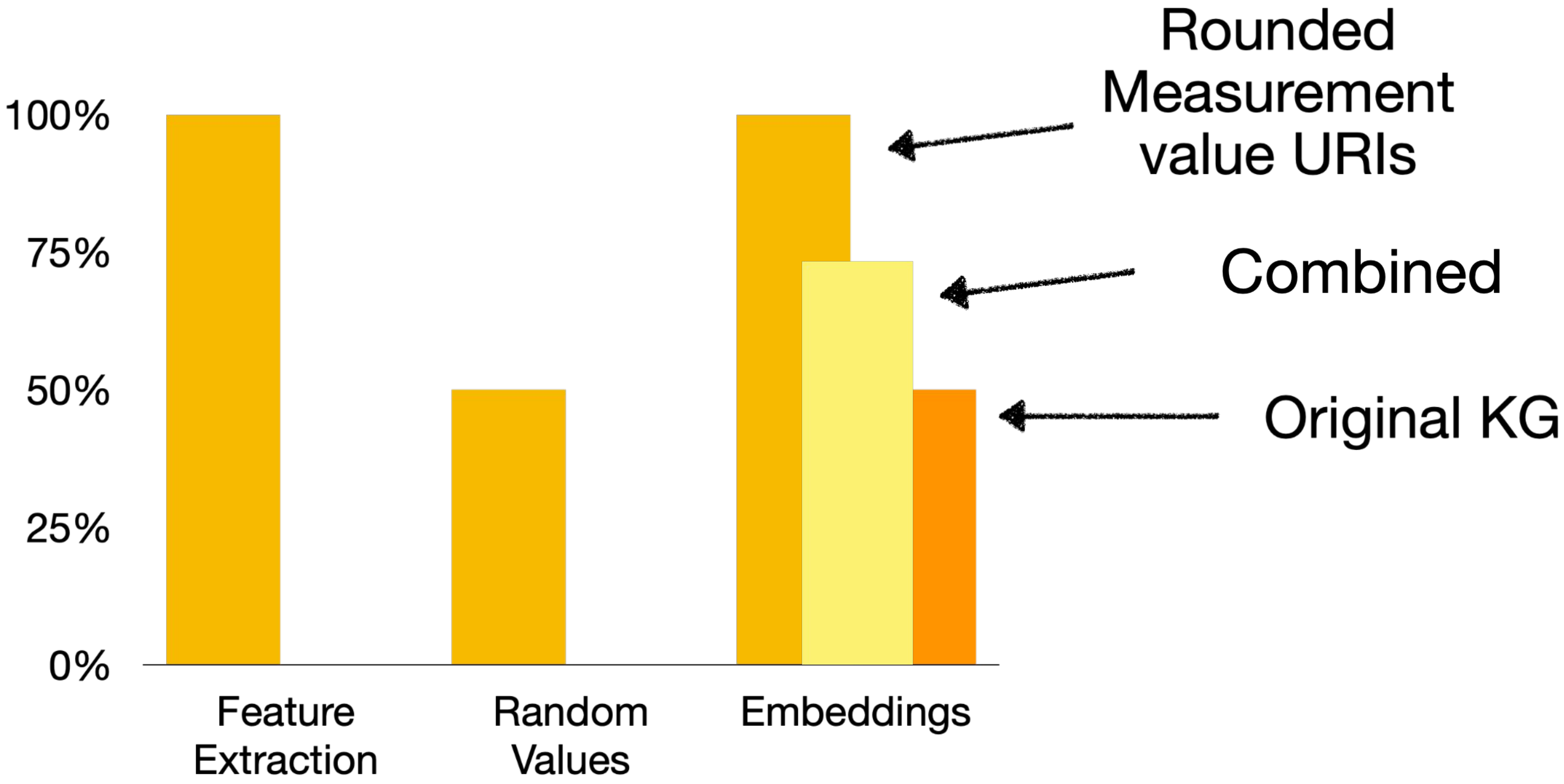
Experiments: Creating the Embeddings



Experiments: Creating the Embeddings



Experiments: Creating the Embeddings

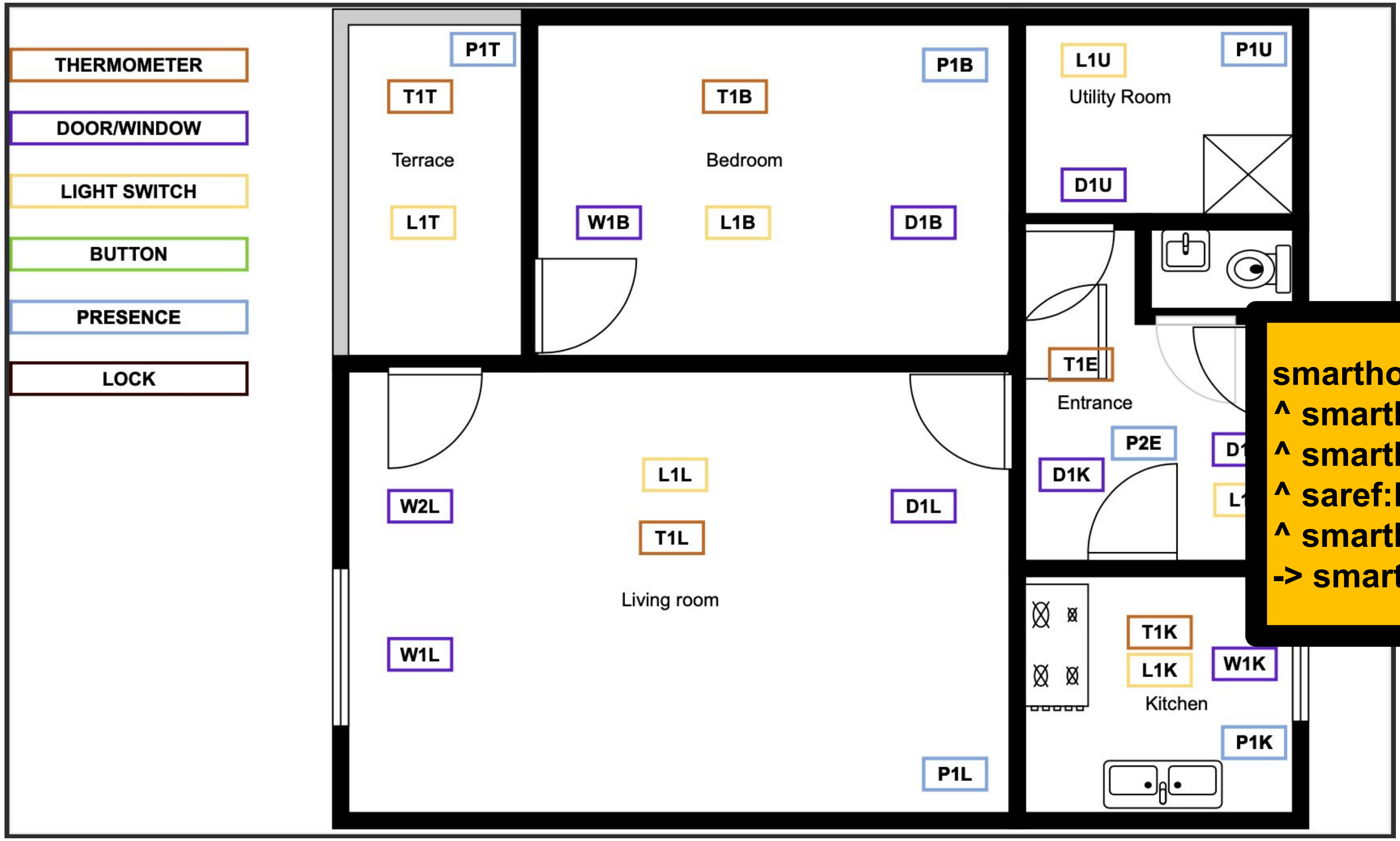


Rounded Measurement value URIs

Combined

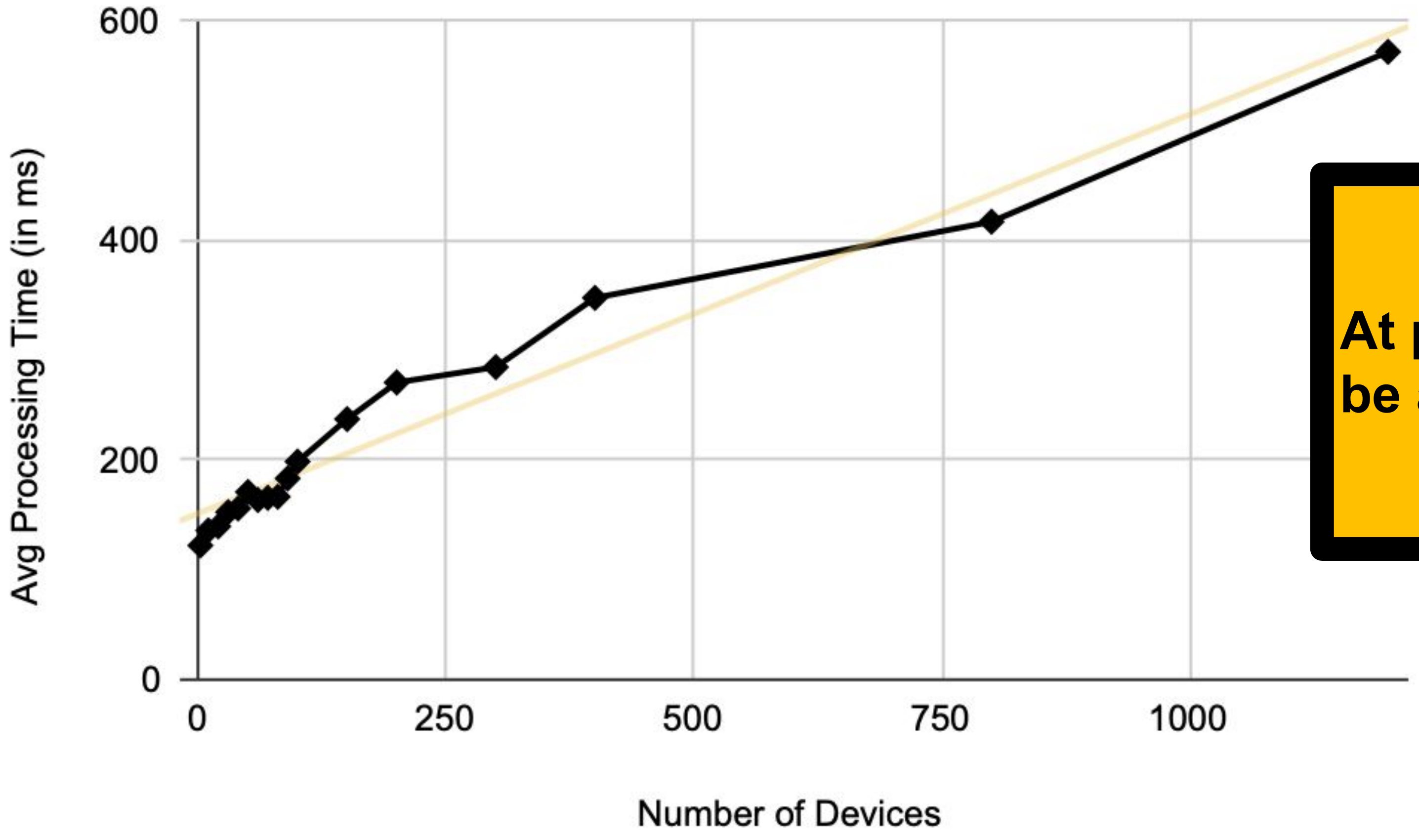
Original KG

Challenge: Smart home scenario handling using OWL and SWRL



smarthouse:VisitorAtTheDoor(Entrance)
^ smarthouse:Tenant(?tenant)
^ smarthouse:isLocatedIn(?tenant, ?room)
^ saref:LightSwitch(?light)
^ smarthouse:isLocatedIn(?light, ?room)
-> smarthouse:TemporaryRedLightColourState(?light)

Semantic Smart Home System : Performance



At present, real time performance can be achieved on mid-resource machines.

Seven large scale pilots leading to market driven deployments will be installed

Greece

Large residential community with smart appliances and EV integration

France

Residential & non-residential, with tertiary buildings and apartments

Portugal

Residential & geographically widespread tertiary buildings

Netherlands

Residential & non-residential buildings

Germany

Groups of residential buildings and hotels

Belgium

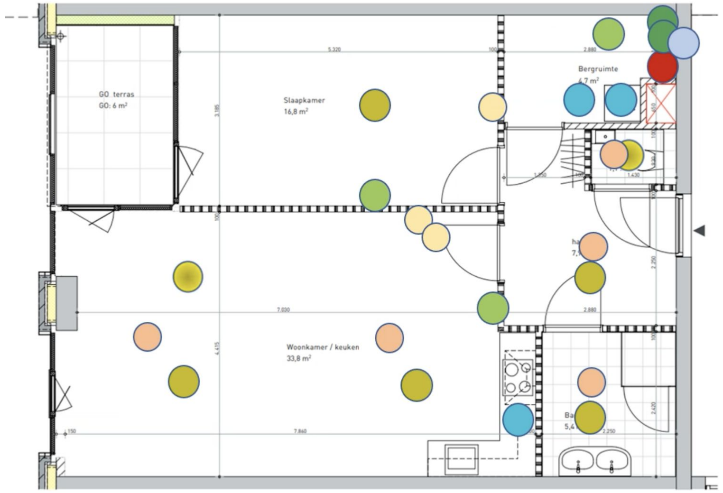
Residential and tertiary buildings in communities of multi-energy vectors

Italy

Residential social housing

Dutch Pilot - Next

99 apartments located in Eindhoven (Strijp-S)



Outfitted with smart devices (e.g. washing machines, door sensors,charging stations, climate systems etc)

iCity
SMART CONNECTIONS, QUALITY LIVING

Hyrde

recognize

VolkerWessels Telecom

TNO

VU VRIJE UNIVERSITEIT AMSTERDAM

Dutch Pilot - Videolab

Large office space

10 different types of smart devices

With most devices having multiple measurements

total of 1500 smart devices

Currently mapping two months worth of measurements

interconnect

interoperable solutions
connecting smart homes,
buildings and grids

Thank you!

Questions?

VU VRIJE UNIVERSITEIT AMSTERDAM

TNO

FINANCING

This project has received funding from the European Union's Horizon 2020 research and innovation programme under Grant agreement No 857237

PROJECT CONTACT

interconnect_project@inesctec.pt

DURATION

01.10.2019 / 30.09.2023

DISCLAIMER: The sole responsibility for the content lies with the authors. It does not necessarily reflect the opinion of the CNECT or the European Commission (EC). CNECT or the EC are not responsible for any use that may be made of the information contained therein.