

# A semantic event approach to enrich device-free indoor localization data

Thomas Hassan  
Fano Ramparany  
Grégoire Lefebvre  
thomas.hassan@orange.com  
fano.ramparany@orange.com  
gregoire.lefebvre@orange.com  
Orange Labs Services  
Meylan, France

## ABSTRACT

Device-free indoor localization technologies can achieve fine-grained localization and have the advantage of being non-intrusive. However, interpreting the low-level data produced by these technologies can be challenging. In this paper, we investigate the application of stream reasoning technologies to enrich and exploit such data. In particular, we are interested in the semantic fusion of events coming from various ambient sensors (door sensors, smart plugs, motion sensors ...), including a ceiling-mounted Pyroelectric InfraRed (PIR) sensor array that provides accurate multi-user positioning on a 2D plane. We show how existing stream reasoning technologies can be used to detect complex events, such as the movements and actions of multiple users in a real environment. These events represent useful information which can later be used for human-centered use cases, such as detecting activities or abnormal user behavior.

## CCS CONCEPTS

• **Human-centered computing** → **Ubiquitous and mobile computing systems and tools**; • **Computing methodologies** → **Temporal reasoning**; **Spatial and physical reasoning**; • **Computer systems organization** → *Real-time system architecture*;

## KEYWORDS

Indoor localization, smart-home, complex events

## ACM Reference Format:

Thomas Hassan, Fano Ramparany, and Grégoire Lefebvre. 2018. A semantic event approach to enrich device-free indoor localization data. In *International Conference on Smart Objects and Technologies for Social Good (Goodtechs '18)*, November 28–30, 2018, Bologna, Italy. ACM, New York, NY, USA, 6 pages. <https://doi.org/10.1145/3284869.3284898>

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

*Goodtechs '18*, November 28–30, 2018, Bologna, Italy

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-6581-9/18/11...\$15.00

<https://doi.org/10.1145/3284869.3284898>

## 1 INTRODUCTION

The promise of indoor localization technologies is to provide new assistance solutions in applications such as comfort services, health-care, monitoring and security. While a majority of indoor localization systems focus on using a device (e.g. a smartphone or wearable sensor) to locate and identify inhabitants, device-free localization has the advantage of being non-intrusive and non-stigmatizing. On the downside, device-free approaches make interpreting sensor data more challenging: data is generally more sparse, incomplete and/or uncertain.

Semantic web technologies allow transforming sensor data into highly expressive information, which can encapsulate contextual information of the environment. Semantic approaches were extensively investigated in the domain of contextual modeling and pervasive computing [Kolozali et al. 2014]. In the last decade, these semantic approaches benefited from the advances in the domain of real-time RDF(Resource Description Framework) stream processing, which evolved into stream reasoning and semantic CEP(Complex Event Processing). Stream reasoning allows combining sensor data with background knowledge, and performing high-level event detection [Mileo et al. 2017].

Our approach is developed in a device-free environment, where ambient sensors (e.g. door sensors, smart plugs, motion sensors ...) are combined with a PIR sensor array<sup>1</sup> to monitor the locations and actions of users. We aim to use stream reasoning to infer valuable information from the fusion of this data. In particular, the detection of recurrent trajectory patterns give information about the behavior of the users [Renso et al. 2013], and enhanced (i.e. semantic) trajectories can be used to infer user activity or detect anomalies in their behavior. The contributions of this paper are the following:

- An architecture to derive complex events from trajectory and ambient sensing devices, and an ontology to model these events along with contextual information. The ontology extends previously established ontologies.
- Use cases of device-free user tracking using stream reasoning, including movement/stop detection and action tracking. To the extent of our knowledge, these use cases were not addressed in an indoor context : they have been investigated only recently in outdoor use cases [Dejonghe et al. 2017] [Alirezaie et al. 2017], and non-vision user tracking is an

<sup>1</sup><http://www.irlynx.com/>

ongoing challenging task [Liu et al. 2018] [Renoux et al. 2017].

- A preliminary implementation of these use cases, in the form of queries, based on the C-SPARQL engine and the GeoSPARQL language. We believe these use cases represent an essential step towards detecting more complex events such as the detection of ADL (Activities of Daily Living), from low-level localization and context information.

The rest of the paper is divided into 3 sections. Next section discusses related work, in the fields of stream reasoning and event detection on localization data. Section 3 describes the proposed approach, including our ontology and how it is used to model smart-home data. Section 4 defines the use cases and their implementation with the C-SPARQL engine. A final section draws conclusions and describes ongoing and future developments of the approach.

## 2 RELATED WORK

Semantic modeling for the detection of complex events is an ongoing research issue. In [Okeyo et al. 2014], authors combines ontological and temporal knowledge representation formalisms to provide powerful representation capabilities for activity modeling. They also present methods to develop temporal entailment rules to support the interpretation and inference of composite activities. [Marroquin et al. 2018] defines an ontology to describe events extracted from a network of smart cameras in a smart-building. The ifcOWL<sup>2</sup> ontology is used along other standard ontologies to model the smart-building environment. Contextual knowledge of the state of the users and the building is deduced from the results of computer vision algorithms, and extended with the result of reasoning processes based on SWRL rules. [Renoux et al. 2017] describes a context recognition approach, where probabilistic reasoning and knowledge modeling are used to count the number of people in a smart-home. The approach uses multiple ambient sensing devices and prior knowledge such as the environment's topology, sensor types, and the features they monitor.

On top of contextual modeling, localization and trajectory analysis provide essential information in the detection of meaningful events [Renso et al. 2013][Cumin et al. 2018] [Ying et al. 2011] Semantic trajectories stem from modeling trajectories in more expressive ways using semantic web technologies. Combined with contextual data these trajectories can help interpreting behavior patterns [Ye et al. 2015]. Semantic trajectory analysis was first applied to GPS coordinates data, as this type of data is massively available, e.g. in city traffic or harbor management. In [Vandecasteele et al. 2014] semantic trajectories of moving vessels are constructed in 4 steps using loosely-coupled modules: trajectory aggregation, trajectory segmentation (detection of movement and stops), semanticization, and complex event detection. In the last step, handcrafted SWRL rules are defined to detect patterns of interest such as a vessel entering a forbidden fishing area (geospatial analysis is done using SWRL built-ins).

Although semantic web technologies are suited to sensor data and trajectory modeling, they lack temporal semantics and real-time querying and reasoning [Ye et al. 2015] [Riboni et al. 2011]. Thus, techniques such as stream reasoning were used to mitigate the

limitations of standard semantic web technologies. There is a common effort towards defining a standard language<sup>3</sup> and processing frameworks dedicated to stream reasoning. First attempts at implementing stream reasoning frameworks include C-SPARQL [Barbieri et al. 2010] and EP-SPARQL [Anicic et al. 2011]. Although these tools allow processing and querying RDF streams, they are still limited and do not represent a complete Semantic CEP engine (SCEP) [Ye et al. 2015]. Recent experimental implementations towards a complete SCEP framework include [Tommasini and Della Valle 2017] and [Gillani et al. 2017], that include an extended syntax for complex events operators.

[Le-Phuoc et al. 2016] proposes to use a semantic CEP engine along with a semantic spatial engine to query a large-scale graph of things. The approach is implemented as a lambda architecture combining semantic and big data tools (hadoop-based). Examples of application include city transport heatmaps. In [Alirezaie et al. 2017] and [Keskiarkkä 2017], authors describe stream reasoning methods to create and query complex event in smart-home health monitoring use cases. The stream reasoning layer filters and aggregates sensor events described with SSN, DUL, and GeoSPARQL. The results of these steps feed an ASP (Answer Set Programming) solver to infer new knowledge. In [Dejonghe et al. 2017], stream reasoning is used to detect suspicious/illegal vessel activity. To achieve this goal, they integrate spatial information in the event processing architecture, using C-SPARQL, GeoSPARQL and a spatial-enabled triple store. [Patroumpas et al. 2017] also investigate complex event processing for maritime monitoring of vessel trajectories, although the approach doesn't rely on semantic web technologies. They show how a complex event processing approach enables the detection of useful trajectory patterns such as short pause, long stops, turns, or missing data points.

Previous work investigated the benefits of integrating semantic and stream reasoning/CEP technologies to detect meaningful events in smart environments. However to the extent of our knowledge the analysis of user movement in indoor environments with these methods hasn't been addressed. In a device-free setting in particular, this type of analysis represents useful information to detect areas of interest within a space, anomalies in user behavior, and along with contextual data infer the activities of users.

## 3 APPROACH DESCRIPTION

Several ambient sensors are distributed around a demonstration room: motion detection sensors which indicate presence, door/window/fridge sensors which detect their state, smart plugs which detect if electric devices are being used (e.g coffee maker), and the PIR localization sensor. The demonstration room is depicted in figure 1. All sensors feed the stream reasoning engine, through the MQTT and Websocket protocols. The PIR array localization sensor is fixed on the ceiling, and transmits localization data of each target (person) under it. The localization payload contains the target ID along with its geospatial coordinates. Each target keeps a consistent ID as long as it stays within the sensor range (approximately 3 meters), without an explicit identification mechanism.

<sup>2</sup><http://openbimstandards.org/standards/ifcowl/>

<sup>3</sup>Documented at <https://www.w3.org/community/rsp/> and <https://github.com/streamreasoning/RSP-QL>

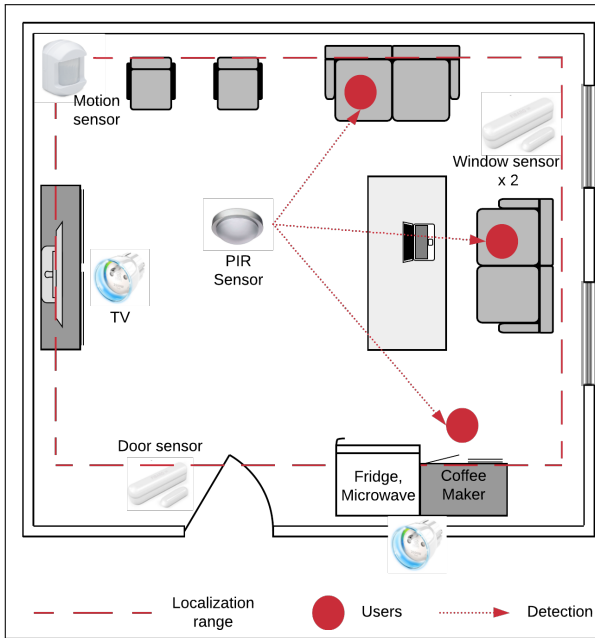


Figure 1: Top-view of the demonstration room

Similarly to previous work, we reuse and extend existing ontologies to construct a model which allows describing the events required in our use cases. The ontologies that form the core of our model are the standard ontologies SSN/SOSA [Compton et al. 2012], DUL, and FOAF. SSN is used to model data related to sensor observations and actuations, while DUL and FOAF are used to model agents (users) and their locations. Along with these ontologies, the Building<sup>4</sup> and Zone<sup>5</sup> modules of the SEAS [Lefrançois 2017] ontology are used to model the smart-home environment. SEAS is an exhaustive ontology dedicated to smart appliances and energy management, that is aligned with the SSN ontology. The GeoSPARQL language and vocabulary are also used to model geometrical data (including room shapes, as well as sensors and users coordinates). Alignments between these ontologies were already established<sup>6</sup>.

The expressiveness of the resulting ontology corresponds to the *SRIN* DL fragment. The namespace for this prototype ontology is <http://orange.smart.home#>, with the corresponding **home** prefix. Table 1 defines ontology prefixes which are used in most use cases of the next section (omitting common prefixes *rdf*, *rdfs*, *owl*, *ex*, *xsd*).

Figure 2 shows an example graph pattern which depicts assertions using these ontologies, including trajectory modeling, represented in the Protégé editor<sup>7</sup>. We emphasize a few concepts and properties that were added to the ontology, to model composite and complex event based on localization and speed information extracted from the PIR sensor array. For the segmentation of trajectories (i.e. stop and move), four properties are defined:

<sup>4</sup><https://ci.mines-stetienne.fr/seas/BuildingOntology-1.0>

<sup>5</sup><https://ci.mines-stetienne.fr/seas/ZoneOntology>

<sup>6</sup><https://www.w3.org/ns/ssn/dul>, <https://ci.mines-stetienne.fr/seas/SSNAlignment>, <https://www.w3.org/TR/sdw-bp/>. Alignment of DUL and FOAF can be made using the Agent concept : <http://dbpedia.org/ontology/Agent>.

<sup>7</sup><https://protege.stanford.edu/>

Table 1: Redundant prefixes

spatial	<a href="http://jena.apache.org/spatial#">http://jena.apache.org/spatial#</a>
geo	<a href="http://www.opengis.net/ont/geosparql#">http://www.opengis.net/ont/geosparql#</a>
sosa	<a href="http://www.w3.org/ns/sosa/">http://www.w3.org/ns/sosa/</a>
foaf	<a href="http://xmlns.com/foaf/0.1/">http://xmlns.com/foaf/0.1/</a>
dul	<a href="http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#">http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#</a>
home	<a href="http://orange.smart.home#">http://orange.smart.home#</a>
cs	<a href="http://larkc.eu/csparql/sparql/jena/ext#">http://larkc.eu/csparql/sparql/jena/ext#</a>

**home: movedFrom**, **home: movedUntil**, **home: stoppedFrom** and **home: stoppedUntil**. The property **home: closeTo** identifies a geographic and temporal proximity correlation between two **sosa: observation**, as illustrated in the first use case below. Although there are existing ontologies to describe semantic trajectories, we employ straightforward definitions because they are sufficient for our use cases.

#### 4 USE CASES AND IMPLEMENTATION

Table 2 defines the system use case targeted by our system. These use cases are categorized by their use of localization information, speed, and composability, i.e. whether they use only observable knowledge or knowledge inferred by another query. For the latter, the results of the first query are used as a data source for the second.

Table 2: Use cases

System use case	Localization	Speed	Composability
Action detection	✓	X	X
User stopped	X	✓	X
User stopped in area	✓	X	✓
Moved & acted	✓	X	✓
Hybrid tracking	✓	X	X

We choose the C-SPARQL engine for the implementation of these use cases, because it supports SPARQL 1.1 queries, which is useful to construct composite and complex events [Keskiärrkkä 2017]. Also, its modularity allows to integrate spatial functionalities, which is required for some use cases. The Jena spatial SPARQL functions<sup>8</sup> are used in our implementation, but another implementation can be integrated<sup>9</sup>. The architecture of the approach is described in Figure 3. The raw data streams are parsed and transformed into RDF representations accordingly to the semantic model, then the RDF streams are read by the stream reasoning engine. To detect complex events, low-level events such as movement segmentation are first detected by the engine (1). The stream of results is then fed back to the engine as another source of observations (2). From raw and low level event streams, the engine can detect more complex events (3), and store the results in a knowledge base for archiving or further analysis (4). Virtuoso<sup>10</sup> is used as the back-end knowledge base in our prototype. The queries below highlight examples use cases of

<sup>8</sup><https://jena.apache.org/documentation/query/spatial-query.html>

<sup>9</sup>E.g. using Parliament <http://parliament.semwebcentral.org/>

<sup>10</sup><https://virtuoso.openlinksw.com/>

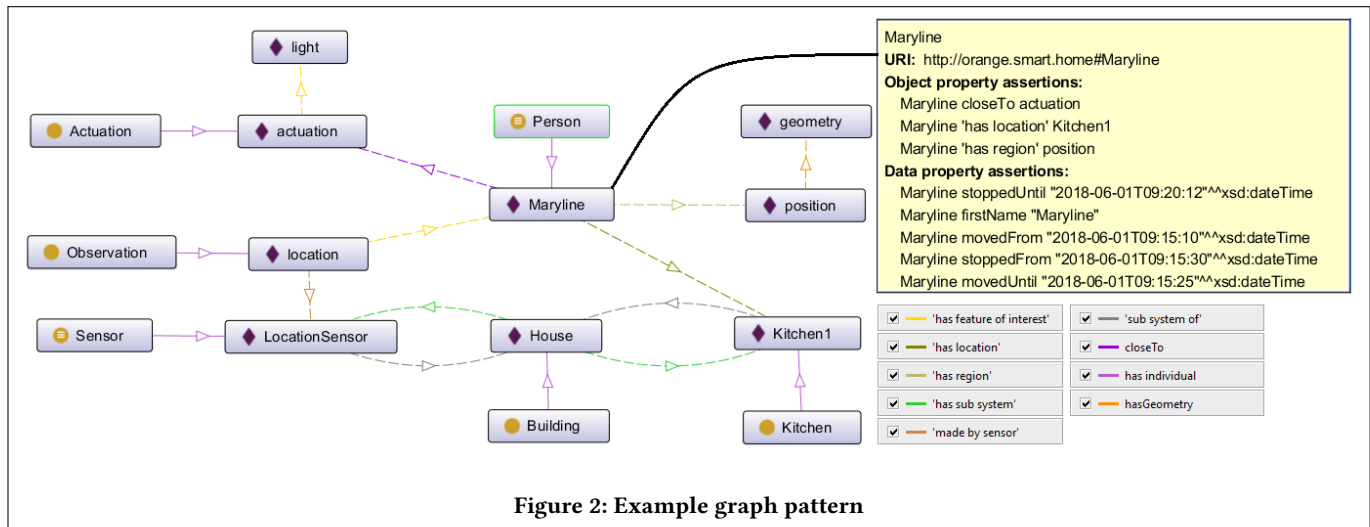


Figure 2: Example graph pattern

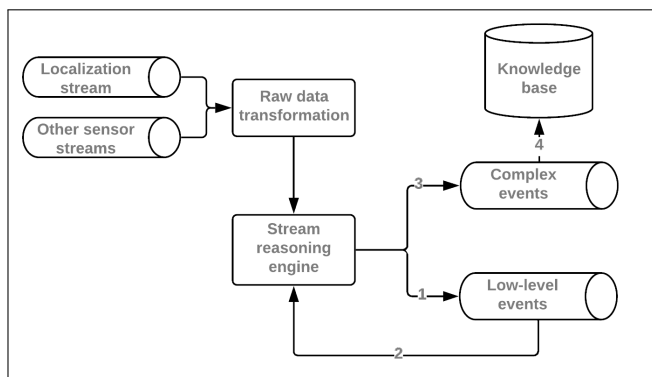


Figure 3: Architecture of the system

our approach. Queries are defined with to the C-SPARQL syntax: each query is a query executed continuously over one or multiple streams (*FROM STREAM*), on a specific time window (*[RANGE ...]*).

```
REGISTER QUERY actionDetection AS
CONSTRUCT {?user home:closeTo ?y}
FROM STREAM <ex:observationStream>
[RANGE 5s TUMBLING]
WHERE {?x a sosa:Observation .
?y a sosa:Observation .
?x sosa:hasFeatureOfInterest ?user.
?user a foaf:Person .
?user dul:hasRegion ?pos .
?pos geo:hasGeometry ?geo .
?y sosa:madeBySensor ?sensor .
?sensor rdfs:label ?labelSens .
?geo spatial:nearby(latitudeOfSensor
longitudeOfSensor radius 'm')
FILTER(?labelSens = "ex:sensor")};
```

Listing 1: Action detection by geographic proximity of observations

In Listing 1 , the query returns the users that were geographically close to an observation coming from a specific sensor, e.g. a fridge opened or a coffee-maker used. The selection is represented as a basic graph pattern : sensor observations are modeled according to the SOSA and GeoSPARQL ontologies. The *dul:hasRegion* property is used to link the user with its observed position. The Jena *spatial:nearby* function is used, but could be replaced another spatial function. We assume the name ("*ex:sensor*") and location (*latitudeOfSensor* and *longitudeOfSensor*) of the sensor are provided. As a concrete example, to restrict the search to the users who open the fridge, the FILTER clause could be : **FILTER(?labelSens = "ex:fridgeBreakRoom")**. The sequence of events is not explicit but the 5 seconds tumbling window ensures the temporal proximity of the two events.

```
REGISTER QUERY targetStopped AS
CONSTRUCT {
?user home:stoppedFrom ?minTime .
?user home:stoppedUntil ?maxTime
}
FROM STREAM <ex:observationStream>
[RANGE 1m STEP 20s]
WHERE {
SELECT distinct ?user (MAX(?time) AS
?maxTime) (MIN(?time) AS ?minTime)
WHERE {
?x a sosa:Observation .
?x sosa:hasResultTime ?time .
?x sosa:hasFeatureOfInterest ?user.
?x sosa:observedProperty <ex#userSpeed> .
?x sosa:hasSimpleResult ?speed
}
GROUP BY ?user
HAVING (AVG(?speed) < maxSpeed &&
(MAX(?time) - MIN(?time)) > ?minTime)} ;
```

Listing 2: User stopped

Listing 2 illustrates movement or stop detection using the speed of the target. A nested query allows to aggregate and segment distinct user's movements. The query can include a minimum time span restriction such as shown in the *HAVING* clause, i.e. the minimum amount of time that triggers a stop detection.

```
REGISTER QUERY userStoppedInZone AS
CONSTRUCT {?user home:stoppedIn ?geo}
FROM STREAM <ex:userStopsStream> [RANGE 10s
TUMBLING]
FROM STREAM <ex:observationStream> [RANGE 10s
TUMBLING]
WHERE {
  ?user home:stoppedFrom ?start .
  ?user home:stoppedUntil ?end .
  ?x a sosa:Observation .
  ?x sosa:hasFeatureOfInterest ?user .
  ?x sosa:observedProperty <ex#userSpeed> .
  ?user dul:hasRegion ?pos .
  ?pos geo:hasGeometry ?geo .
  ?geo spatial:withinBox(latMin longMin
latMax longMax)
}
```

Listing 3: User stopped in Zone

Listing 3 uses the result of a stop detection query, and localization observations. The *spatial:withinBox* function is used to restrict the stop detection to a rectangular area, assuming the area coordinates are provided (*latMin, lonMin, latMax, lonMax*).

```
REGISTER QUERY moveAction AS
SELECT *
FROM STREAM <ex:observationStream>
[RANGE 20s TUMBLING]
FROM STREAM <ex:actionStream>
[RANGE 1m STEP 20s]
FROM STREAM <ex:moveStream>
[RANGE 1m STEP 20s]
WHERE {
  ?x a sosa:Observation .
  ?x sosa:hasFeatureOfInterest ?user .
  ?user home:closeTo ?y .
  ?y a sosa:Actuation .
  ?y sosa:hasFeatureOfInterest <ex#Light> .
  ?user home:movedUntil ?end .
  ?user dul:hasRegion ?pos .
  ?pos geo:hasGeometry ?geo .
  ?geo spatial:withinBox(latMin longMin
latMax longMax)
FILTER (
  cs:timestamp(?user home:movedUntil ?end) <
  cs:timestamp(?user home:closeTo ?y) &&
  cs:timestamp(?user home:closeTo ?y) <
  cs:timestamp(?user dul:hasRegion ?pos))
};
```

Listing 4: Movement and action event sequence

Listing 4 represents a complex event composed of movement and action tracking. The objective is to detect a sequence of events: a movement followed by an action, followed by a localization of the user in a rectangular area like in the previous use case. This use case can be used to detect simple activities, e.g. moving, (*?user home:movedUntil ?end*), turning off the lights and sitting on the couch. The sequence of events could include a stop detection following the localization of the user (*?pos geo:hasGeometry ?geo*). The C-SPARQL *cs:timestamp* function is used to define the sequence of events. As stated in section 2, the recommended *RSP-QL* syntax includes an actual *SEQ* operator<sup>11</sup>, which would be a better fit to detect complex events, but frameworks that implement this kind of operator don't match other requirements such as spatial reasoning (see section 2).

```
REGISTER QUERY trackingHybrid AS
SELECT ?posX ?locY ?locZ
FROM STREAM <ex:observationStream>
[RANGE 20s TUMBLING]
FROM STREAM <ex:actuationStream>
[RANGE 20s TUMBLING]
WHERE {{
  ?x a sosa:Observation .
  ?x sosa:hasFeatureOfInterest ?user .
  ?x madeBySensor ?PIRSensor .
  ?PIRSensor a home:locationSensor .
  ?user dul:hasRegion ?posX .
}
UNION {
  ?y a sosa:Actuation .
  ?y sosa:hasFeatureOfInterest ?window .
  ?window a home>window .
  ?window dul:hasLocation ?locY .
}
UNION {
  ?z a sosa:Observation .
  ?z madebySensor ?presenceSensor .
  ?platform sosa:hosts ?presenceSensor .
  ?presenceSensor rdf:type home:presenceSensor .
  ?presenceSensor dul:hasLocation ?locZ
}
FILTER (?COUNT(distinct ?user) = 1)} ;
```

Listing 5: Hybrid tracking

Listing 5 tackles user tracking without localization information, using static knowledge about the sensors positions. When the user is alone in the environment (e.g. for elderly monitoring), ambient sensors can complement the localization observations when the user is outside of its range, since he is the source of any observation or actuation in the environment. This is particularly important in a real environment, that is most certainly not fully equipped with localization devices. The query uses specific concepts to model a window (*home>window*) which can extend the *seas:System* class of the SEAS ontology, and to sensor types that can extend the *sosa:Sensor* class. The locations of the window and the sensor are returned along the positions observed by the PIR sensor.

<sup>11</sup>This operator is common in complex event processing engines

To conclude on the applicability of these queries, Figure 4 depicts a typical sequence of events. 1-2: the user enters the room and triggers the door sensor (action -> move). 2-3: a movement is followed by activation of the coffee maker (move -> action). 4: the user enters the couch area. 5: a user stop is detected within the couch area (stop in area). 6-7-8: a sequence of movement (6), presence detection (7) and movement (8) is detected when the user picks up the TV remote. After these events have been detected and stored, the complete sequence of actions of the user and his trajectory can be built.

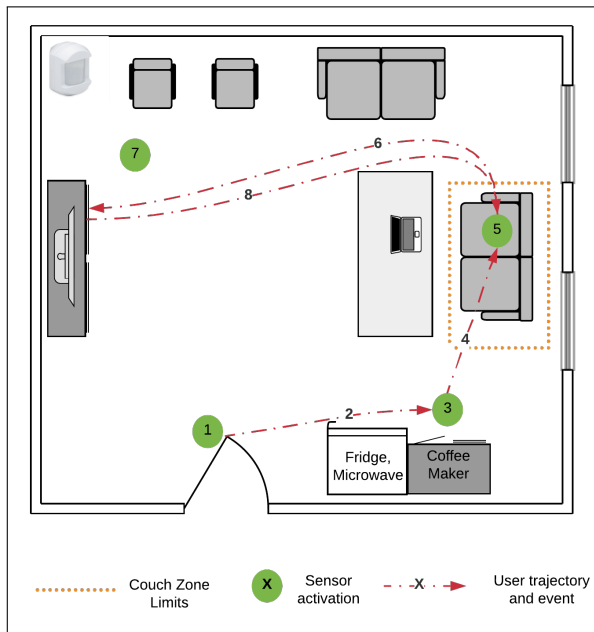


Figure 4: Sequence of action summary

## 5 CONCLUSION AND FUTURE WORK

In this paper we presented an approach to construct meaningful events occurring in a device-free sensing environment, using stream reasoning technologies and a dedicated ontology. We specifically addressed the exploitation of localization data towards detecting complex events, which serve as meaningful information for use cases such as elderly monitoring. The approach has been implemented and is being validated in a real environment. This work is still in progress: we expect to provide an experimental evaluation of the system in future work. More specifically we are studying the performance of the approach for event detection, i.e. the error rate and the event/second throughput. Finally, we also intend to show the applicability of the approach in multi-user scenarios, and extend the approach to deal with uncertain and partial data.

## REFERENCES

- Marjan Alirezaie, Jennifer Renoux, Uwe Köckemann, Annica Kristofferson, Lars Karlsson, Eva Blomqvist, Nicolas Tsiftes, Thimo Voigt, and Amy Loutfi. 2017. An Ontology-Based Context-Aware System for Smart Homes: E-Care@ Home. *Sensors* 17, 7 (2017), 1586.
- Darko Anicic, Paul Fodor, Sebastian Rudolph, and Nenad Stojanovic. 2011. EP-SPARQL: a unified language for event processing and stream reasoning. In *Proceedings of the 20th international conference on World wide web*. ACM, 635–644.
- Davide Francesco Barbieri, Daniele Braga, Stefano Ceri, EMANUELE DELLA VALLE, and Michael Grossniklaus. 2010. C-SPARQL: a continuous query language for RDF data streams. *International Journal of Semantic Computing* 4, 01 (2010), 3–25.
- Michael Compton, Payam Barnaghi, Luis Bermudez, Raúl García-Castro, Oscar Corcho, Simon Cox, John Graybeal, Manfred Hauswirth, Cory Henson, Arthur Herzog, et al. 2012. The SSN ontology of the W3C semantic sensor network incubator group. *Web semantics: science, services and agents on the World Wide Web* 17 (2012), 25–32.
- Julien Cumin, Fano Ramparany, Grégoire Lefebvre, and James L. Crowley. 2018. Fusion de Décisions Par Lieu Pour Reconnaître l'Activité Humaine et Sa Prise En Compte Dans La Gestion de Contexte. *Modélisation et utilisation du contexte* 18, 1 (Jan. 2018). <https://doi.org/10.21494/ISTE.OP.2018.0207>
- Alexander Dejonghe, Femke Ongenaë, Stijn Verstichel, and Filip De Turck. 2017. C-GeoSPARQL: Streaming GeoSPARQL Support on C-SPARQL. In *Joint Proceedings of the 2nd RDF Stream Processing (RSP 2017) and the Querying the Web of Data (QuWeDa 2017) Workshops Co-Located with 14th ESWC 2017 (ESWC 2017)*. 1–10.
- Syed Gillani, Antoine Zimmermann, Gauthier Picard, and Frédérique Laforest. 2017. A Query Language for Semantic Complex Event Processing: Syntax, Semantics and Implementation. *Semantic web journal* (2017).
- Robin Keskiärrä. 2017. *Towards Semantically Enabled Complex Event Processing*. Vol. 1782. Linköping University Electronic Press.
- Sefki Kolozali, Maria Bermudez-Edo, Daniel Puschmann, Frieder Ganz, and Payam Barnaghi. 2014. A Knowledge-Based Approach for Real-Time IoT Data Stream Annotation and Processing. *IEEE*, 215–222. <https://doi.org/10.1109/ITings.2014.39>
- Danh Le-Phuoc, Hoan Nguyen Mau Quoc, Hung Ngo Quoc, Tuan Tran Nhat, and Manfred Hauswirth. 2016. The Graph of Things: A Step towards the Live Knowledge Graph of Connected Things. *Web Semantics: Science, Services and Agents on the World Wide Web* 37-38 (March 2016), 25–35. <https://doi.org/10.1016/j.websem.2016.02.003>
- Maxime Lefrançois. 2017. Planned ETSI SAREF Extensions Based on the W3C&OGC SOSA/SSN-Compatible SEAS Ontology Patterns. In *Proceedings of Workshop on Semantic Interoperability and Standardization in the IoT, SIS-IoT*, (Amsterdam, Netherlands).
- Tong Liu, Xiaomu Luo, and Zhuoqian Liang. 2018. Enhanced Sparse Representation-Based Device-Free Localization with Radio Tomography Networks. *Journal of Sensor and Actuator Networks* 7, 1 (Feb. 2018), 7. <https://doi.org/10.3390/jsan7010007>
- Roberto Marroquin, Julien Dubois, and Christophe Nicolle. 2018. Ontology for a Panoptes Building: Exploiting Contextual Information and a Smart Camera Network. *Semantic Web Journal* (2018), 24.
- Alessandra Mileo, Minh Dao-Tran, Thomas Eiter, and Michael Fink. 2017. Stream Reasoning. In *Encyclopedia of Database Systems 2017*, M. Tamer Özsu and Ling Liu (Eds.). Encyclopedia of Database Systems, Vol. Volume. Springer-Verlag, New York, USA. <http://doras.dcu.ie/21771/>
- George Okeyo, Liming Chen, and Hui Wang. 2014. Combining Ontological and Temporal Formalisms for Composite Activity Modelling and Recognition in Smart Homes. *Future Generation Computer Systems* 39 (Oct. 2014), 29–43.
- Kostas Patrourmpas, Elias Alevizos, Alexander Artikis, Marios Vodas, Nikos Pelekis, and Yannis Theodoridis. 2017. Online event recognition from moving vessel trajectories. *Geoinformatica* 21, 2 (2017), 389–427.
- Jennifer Renoux, Marjan Alirezaie, Lars Karlsson, Uwe Köckemann, Federico Pecora, and Amy Loutfi. 2017. Context Recognition in Multiple Occupants Situations: Detecting the Number of Agents in a Smart Home Environment with Simple Sensors. In *Workshop on Knowledge-Based Techniques for Problem Solving and Reasoning (KnowProS'17)*.
- Chiara Renso, Miriam Baglioni, Jose António F. de Macedo, Roberto Trasarti, and Monica Wachowicz. 2013. How You Move Reveals Who You Are: Understanding Human Behavior by Analyzing Trajectory Data. *Knowledge and Information Systems* 37, 2 (Nov. 2013), 331–362. <https://doi.org/10.1007/s10115-012-0511-z>
- Daniele Riboni, Linda Pareschi, Laura Radaelli, and Claudio Bettini. 2011. Is ontology-based activity recognition really effective?. In *Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 2011 *IEEE International Conference on*. IEEE, 427–431.
- Riccardo Tommasini and Emanuele Della Valle. 2017. Yasper 1.0: Towards an RSP-QL Engine.. In *International Semantic Web Conference (Posters, Demos & Industry Tracks)*.
- Arnaud Vandecasteele, Rodolphe Devillers, and Aldo Napoli. 2014. From Movement Data to Objects Behavior Using Semantic Trajectory and Semantic Events. *Marine Geodesy* 37, 2-Special Issue: Special Issue on Coastal and Marine Geographic Information Systems (2014), 126–144.
- Juan Ye, Stamatia Dasiopoulou, Graeme Stevenson, Georgios Meditskos, Efstratios Kontopoulos, Ioannis Kompatsiaris, and Simon Dobson. 2015. Semantic Web Technologies in Pervasive Computing: A Survey and Research Roadmap. *Pervasive and Mobile Computing* 23 (Oct. 2015), 1–25. <https://doi.org/10.1016/j.pmcj.2014.12.009>
- Josh Jia-Ching Ying, Wang-Chien Lee, Tz-Chiao Weng, and Vincent S Tseng. 2011. Semantic trajectory mining for location prediction. In *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, 34–43.