Markus Stocker
Åbo Akademi • Turku • 04.04.2013

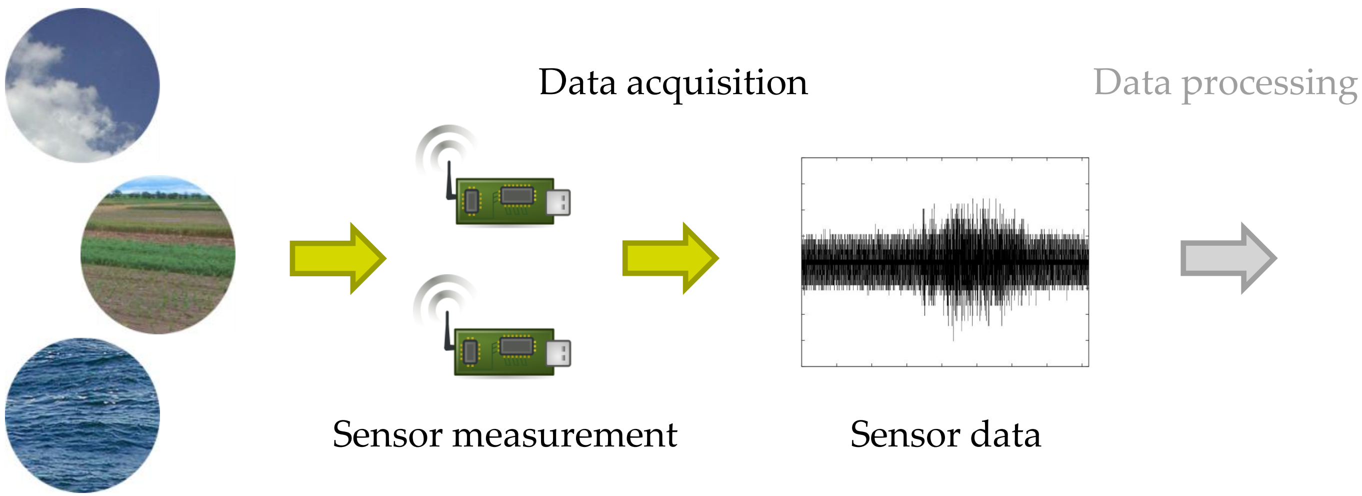
# Wavellite: A software framework for the representation of knowledge about real-world phenomena observable by a sensor network

UNIVERSITY OF
EASTERN FINLAND

# About (me)

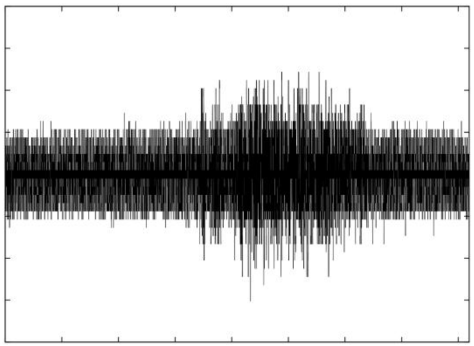
- PhD student at UEF in environmental technology (informatics)
  - Also collaborating with Åbo Akademi
- MSc student at UEF in environmental science
- Background in informatics (MSc UZH, 2007)
- Interned with HPL Bristol (UK) and Clark & Parsia (USA) on various projects related to the semantic web
- In Kuopio since summer 2009
- Music, sports, cooking, photography, reading

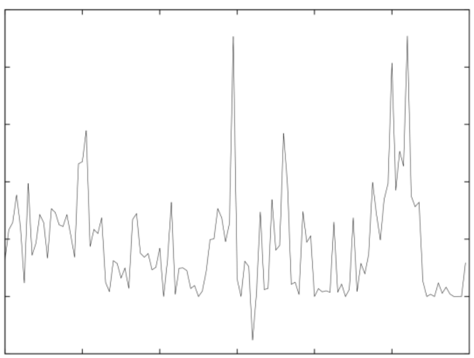
# Introduction
# Environmental monitoring (using sensors)




Real-world phenomena
and their properties (e.g. the
temperature of indoor air)

# Introduction
# Sensor data makes *no* sense (almost)



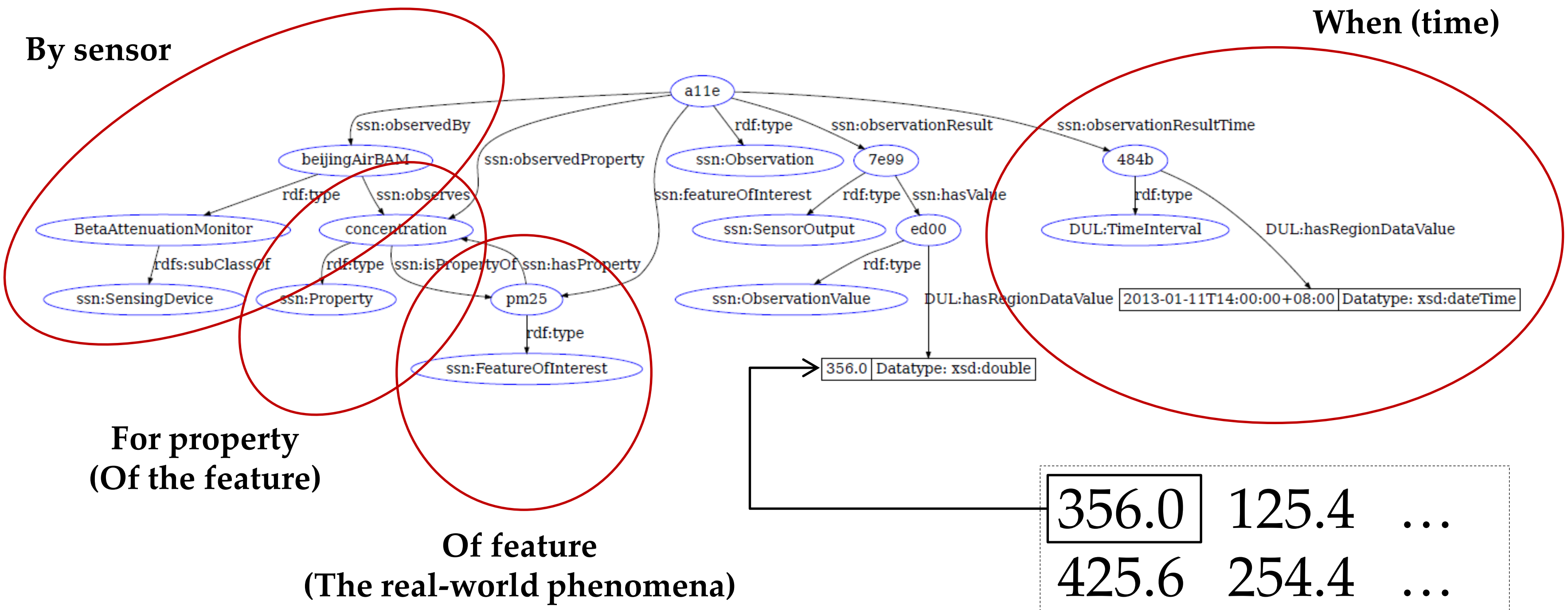



```
  7.3316302167824074e+005  6.7295417637024639e+002  0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000
0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000
1.1248345678061637e+003  7.6451655783892170e+002  6.6489916883398178e+002  5.0996188060151417e+002  5.3037337771849309e+002
2.5990152166875021e+002  2.6354791501170808e+002  2.6316883341227492e+002  3.3696093295963192e+002  3.2708190669952091e+002
3.9391899945110441e+002  3.2426846166900322e+002  2.0141200942601449e+002  1.4700101128227615e+002  4.0928587811142940e+001
1.2831620686413419e+001  0.0000000000000000e+000
  7.3316303024305555e+005  5.9954073981185900e+002  0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000
0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000
8.1548383245909952e+002  6.4755047038178782e+002  4.7342333956288417e+002  5.0497438804107304e+002  4.8678378833469156e+002
2.1558831592426984e+002  2.4469781600550951e+002  2.4637216889127018e+002  3.9338495014282574e+002  2.8801709503956096e+002
4.1110643983031497e+002  2.8396322218576239e+002  2.1375040618051182e+002  1.4957123471247169e+002  3.7975823102376701e+001
2.0382122539321763e+001  1.6943213617983506e+001
  7.3316303880787035e+005  6.4135476612374202e+002  0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000
0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000
7.9725650443997608e+002  4.0630047999881816e+002  3.5247269226184869e+002  3.7902085695621594e+002  2.8805461752290518e+002
1.8132905435068977e+002  2.6028214884093450e+002  1.6268304363454374e+002  3.3463583010398571e+002  2.0169500653114130e+002
4.6380154662366067e+002  2.6412454349776425e+002  2.0127541006127689e+002  1.3503244293786514e+002  2.3154902150570045e+001
7.6393422067327084e+000  1.7201363075124494e+001
  7.3316304687500000e+005  6.1036574617458143e+002  0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000
0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000
7.2736971215268670e+002  3.9684227323502876e+002  3.4903661050199901e+002  3.4531808153726166e+002  3.0686816846877008e+002
1.5193881810750614e+002  2.4133238139702240e+002  2.4228704440975204e+002  2.8825745672488563e+002  2.0412591958392611e+002
4.0962054312850012e+002  2.8308360749792962e+002  2.2063151348349507e+002  1.2405735268505646e+002  1.0751097756079318e+001
1.1881689141194544e+001  1.3547412000218912e+001
  7.3316305520833330e+005  6.4313854345743607e+002  0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000
0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000
8.0539618874151654e+002  3.8626777986640963e+002  2.8741638600588004e+002  2.5488895887853258e+002  2.6289312410222209e+002
1.6973869269222212e+002  1.7064856680396792e+002  2.6528938061547774e+002  2.7436008754172127e+002  2.3416327840132379e+002
4.5037513571727550e+002  3.4357911665879971e+002  2.1968465764114876e+002  1.0707033953281511e+002  4.2106918027684735e+001
0.0000000000000000e+000  2.5075705754016887e+001
  7.3316306354166672e+005  6.9018149795028319e+002  0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000
0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000  0.0000000000000000e+000
1.0122740400440272e+003  4.7588170401009631e+002  3.6163701941444793e+002  3.8492799229831303e+002  3.1633325134373962e+002
1.8305150227202395e+002  2.2529123956614126e+002  2.0836586101380522e+002  3.1634735061210313e+002  2.5690547494848136e+002
3.6567505629777969e+002  3.0461779733996269e+002  1.8490080930780519e+002  1.1544775603278390e+002  4.1043905610232025e+000
1.3740708174213207e+001  2.9403296138810457e+001
```

# Introduction
# … and

- There is lots of it, including in environmental science
- Typically automatically sampled, sometimes at “high frequency”
- National Ecological Observatory Network (NEON, USA)
  - Bioclimate, biodiversity, biogeochemistry, ecohydrology, infectious disease, land use change
  - ~60 observatory sites in 20 eco-climatic domains over 30 years
  - 539 ”basic calibrated data products”[1]
  - ”Billions of data points collected by thousands of sensors and hundreds of people”[2]
- Measuring vehicle-induced vibration on a road section
  - 3 vibration sensors at 2 kHz; 15 billion measurements a month

# Introduction
# Self-describing sensor data




| | | |
|---|---|---|
| 356.0 | 125.4 | … |
| 425.6 | 254.4 | … |
| 375.8 | 85.7 | … |
| … | … | … |

+ quality, uncertainty, location, …

# Introduction
# The problem

- More sensors → more data
  - Very likely
- More data → more information/knowledge?
  - Desirable but (more) challenging
  - If we continue developing and improving
    - Processing, storage, and retrieval of sensor data
    - Acquisition of information and knowledge from sensor data
    - Integration, reuse, and sharing of data, information, and knowledge
- Currently,
  - Large gap between sensor data and abstract information/knowledge
  - Generic systems to visualize (spatiotemporal) sensor data
  - *Ad hoc* systems to make more sense of sensor data

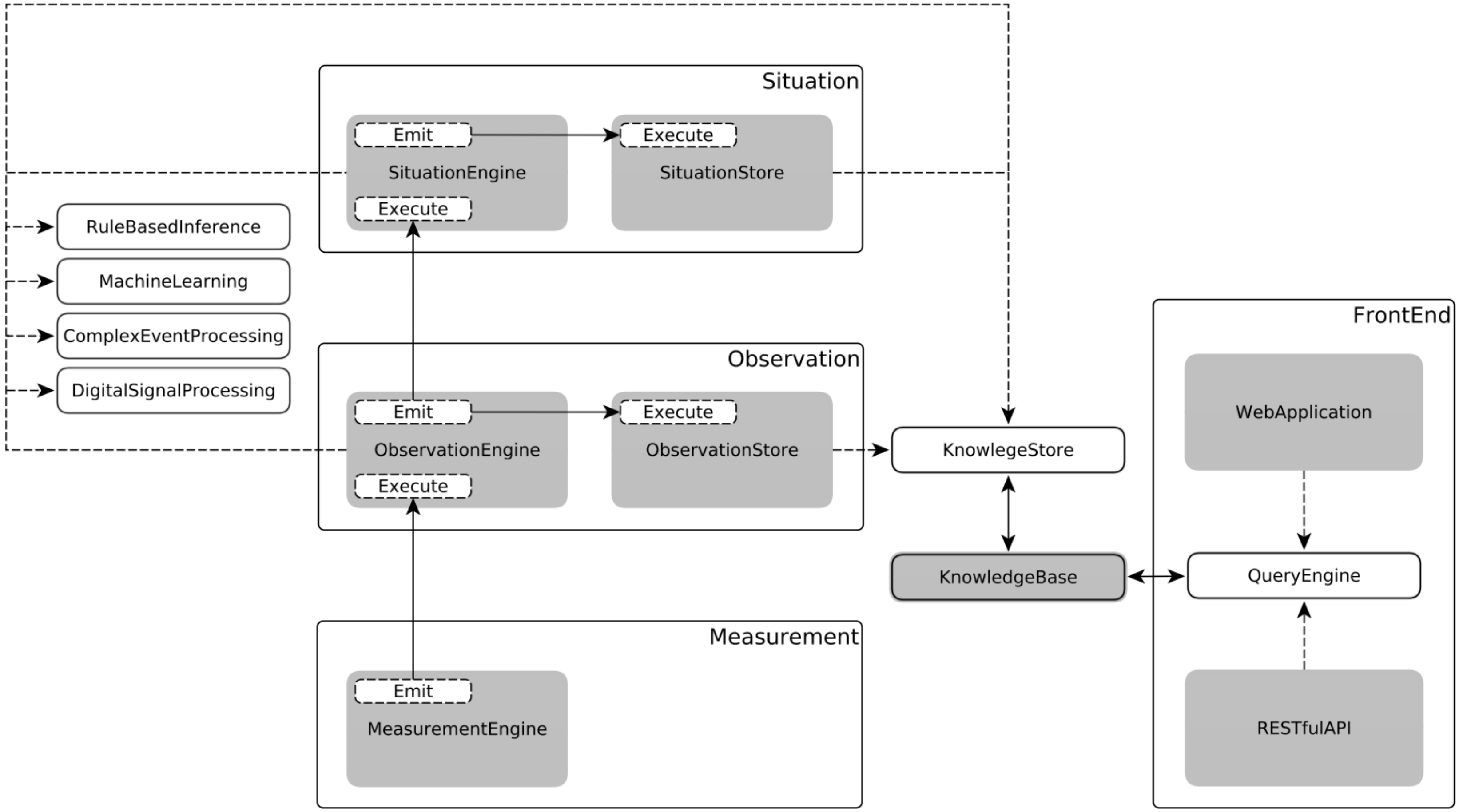
# Wavellite
# Overview

- Software framework for
  - Continuous, real-time, and distributed
  - Representation of knowledge (for real-world phenomena)
  - Acquired from sensor data
- Software interface to represented knowledge for
  - Users via Web application
  - Software via RESTful API

# Wavellite Overview

- Manage sensor data *and* its metadata
  - Sensor data *and* the when-by-for-of (self-describing sensor data)
  - In other words, sensor data *and* information on
    - when is the sampling time
    - the sensor by which sampling occurs
    - the property for which measurement is performed
    - and the feature of which the property is measured
  - For instance, data sampled now by this beta attenuation monitor for the concentration of outdoor $PM_{2.5}$
- Represent knowledge for real-world phenomena observed by sensors
  - Events, episodes, scenes, changes: more generally situations
  - For instance, events of unhealthy exposure to $PM_{2.5}$

# Wavellite Architecture



# Wavellite Implementation

- Wavellite builds on Storm for distributed processing
- Storm is a distributed real-time computation system[3]
- Storm topology
  - Consists of nodes and streams
  - Nodes can implement “arbitrarily” complex logic
  - Can be deployed on single machine or cluster
- By building on Storm, Wavellite
  - Components (engines and stores) are Storm nodes
  - Measurement, observation, situation streams are Storm streams
- Knowledge base is a third-party software (Stardog RDF database)
- Front end is implemented in Spring Framework

# Wavellite
# Measurement layer

- Measurement engine
  - Software wrapper for (one or more) sensors
    - More generally, a source of data (e.g. sensing device, databases, files, …)
    - Very heterogeneous
  - Responsible for the retrieval of sensor data
  - Parsing may be necessary
  - Generates measurement(s) forwarded via stream(s)
  - Support relevant communication protocol(s) and data format(s)
  - Domain specific

# Wavellite
# Observation layer

- Observation engine
  - Subscribes to measurement stream(s)
  - Semantic enrichment of measurement data
    - Add the when-by-for-of to sensor data
    - Turn measurements into observations
  - Forward observations via stream(s)
  - Uses domain terminology defined in the knowledge base
    - Describe your sensors, features, properties, locations (types)
  - Domain terminology extends from the SSN ontology
    - Generic/reusable terminology for sensor networks and data
- Observation store
  - Subscribes to Observation stream(s)
  - Takes care to store observations to knowledge base

# Wavellite
# Situation layer

- Situation engine
  - Subscribes to Observation stream(s)
  - Represents situations acquired from observations
  - May use DSP, ML, CEP, … for acquisition
  - Uses domain terminology defined in the knowledge base
    - Describe your situation types, relevant objects in situations, …
  - Domain terminology extends from the STO
    - Generic/reusable terminology for situations
    - Based on Situation Theory[4]
- Situation store
  - Subscribes to Situations stream(s)
  - Takes care to store situations to knowledge base

# Wavellite Modules

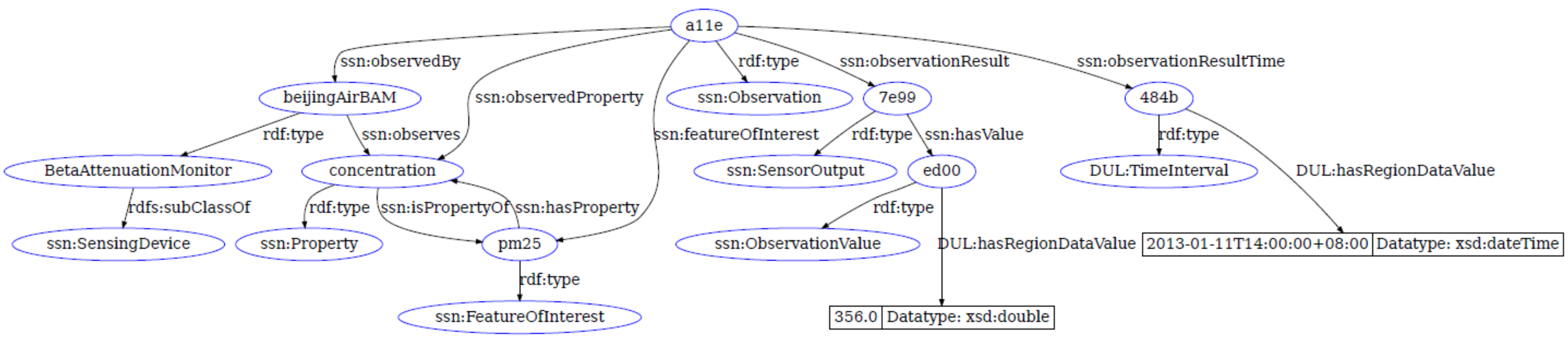
- Provide services to Wavellite components, for instance
  - FFT of signal observed over time
  - Pattern classification
  - Storage to knowledge base
  - Query from knowledge base
- Implement logic to acquire knowledge
  - Learning or reasoning
  - Typically domain specific
  - Implementation in Java (possibly also JVM script languages)
- Typically software libraries

# Example BeijingAir

- Sensor data acquired from @bejingair Twitter stream
- U.S. Embassy in Beijing runs a BAM for PM2.5 monitoring
- One status update every hour, e.g.

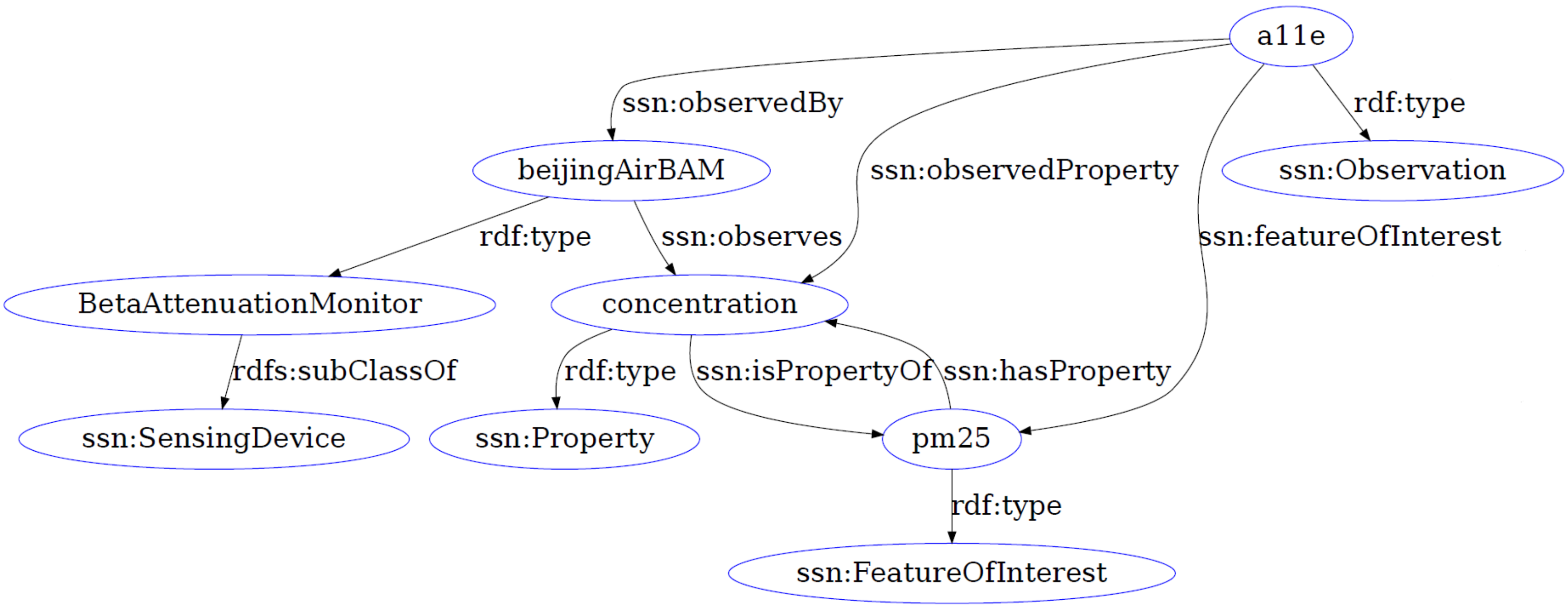

- Statuses -> Measurement -> Observations -> Situations
- Situations of unhealthy exposure (> 65.5 μg/m$^3$; 24-hour)

# Example
# BeijingAir (Observation)

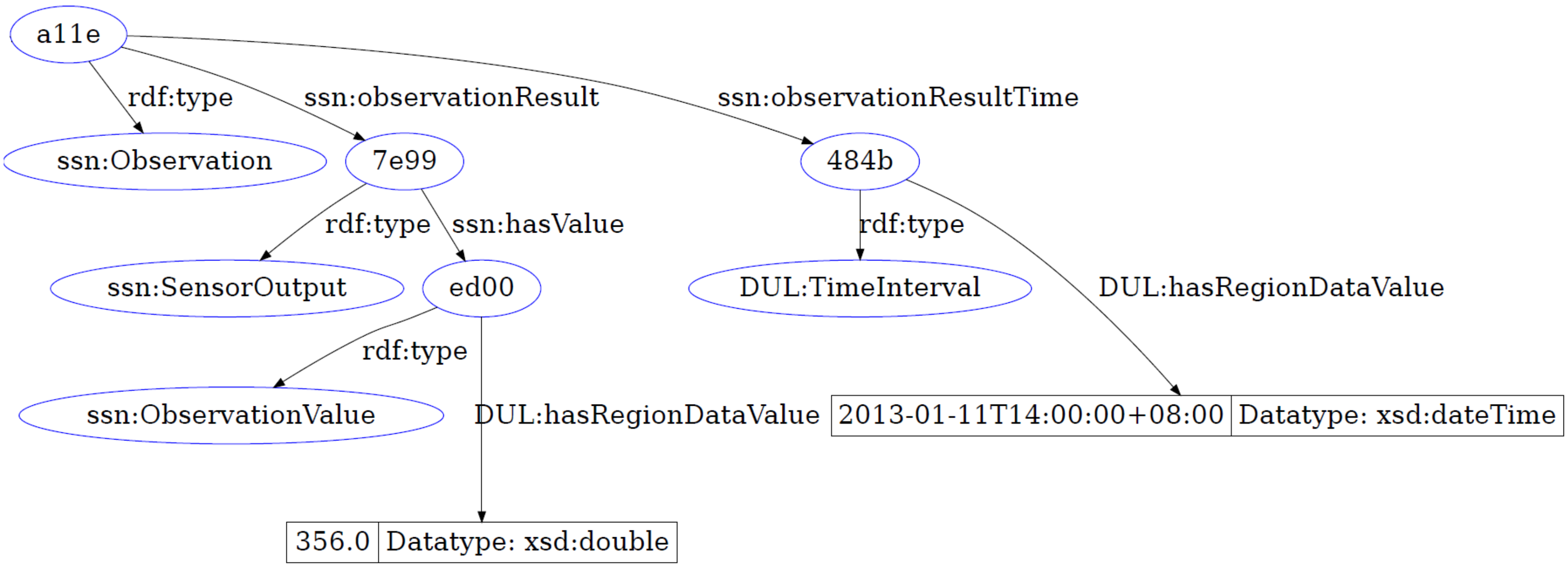



UNIVERSITY OF EASTERN FINLAND

# Example
# BeijingAir (Observation metadata)

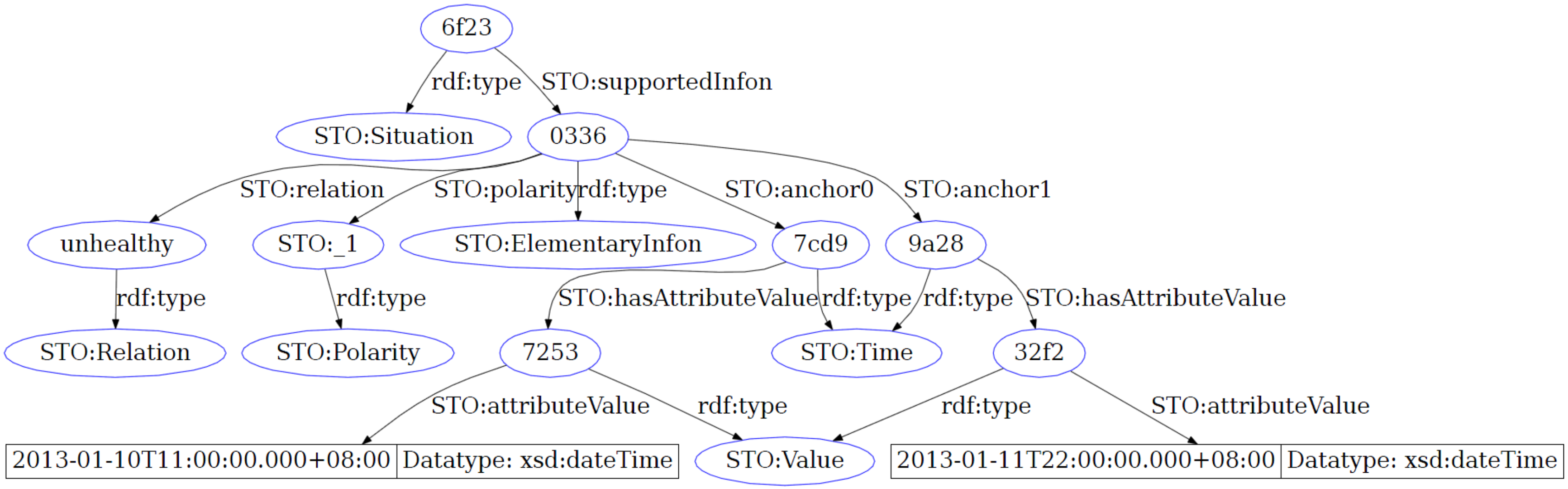


# Example
# BeijingAir (Observation data)




UNIVERSITY OF EASTERN FINLAND

# Example
# BeijingAir (Situation)



# Wavellite
# Demo

- Use Web application to visualize
  - Observations by the BAM operated by the U.S. Embassy to Beijing
  - Visualize situations of (unhealthy) exposure
- Show RESTful API

# Future work

- Continue developing Wavellite functionality
  - Core, Web application, RESTful API
  - For instance
    - Summary statistics view for queried observations
    - Support for KB-driven knowledge acquisition tasks
- Apply Wavellite to new domains
  - Environmental science
  - Smart grids
  - Monitoring of industrial machinery
- Write a PhD thesis

# Related work

- Terminologies to describe sensor networks and sensor data[5]
- Utilize SSN ontology[6] or STO[7,8,9]
  - But not both in a hybrid manner
- Extract from sensor data physiological properties of athletes[10]
  - Representation using XML, not expressive ontology language
- Generic architecture to extract information form sensor network[11]
  - Three-layered architecture
  - Bridging sub symbolic layer (measurement) with symbolic layer
  - Via conceptual layer that implements a metric space (similarity)
- System that can be queried for high-level events[12,13]
  - Does not require handling of sensor data
- Ontology-based environmental information systems[14,15]

# Conclusions

- More and more sensors; more and more sensor data
- Are we keeping up with knowledge acquisition?
- Concerns
  - Large gap between sensor data and abstract knowledge
  - Complex knowledge acquisition tasks
  - Automated representation of acquired knowledge
- Wavellite aims at tackling some of these concerns
- Generic framework extensible with domain specific
  - Software that implements sensor data and knowledge acquisition
  - Modelling of sensors, observations, situations

# References

[1] http://www.neoninc.org/science/data

[2] http://www.neoninc.org/news/big-data-part-ii

[3] http://storm-project.net/

[4] Devlin, K. (1991): Logic and information. Cambridge University Press

[5] Compton, M., Henson, C., Neuhaus, H., Lefort, L., Sheth, A., 2009. A Survey of the Semantic Specification of Sensors, in: 2nd International Workshop on Semantic Sensor Networks, at 8th International Semantic Web Conference.

[6] Barnaghi, P., Ganz, F., Henson, C., Sheth, A., 2012. Computing Perception from Sensor Data. Technical Report. knoesis.org.

[7] De Maio, C., Fenza, G., Furno, D., Loia, V., 2012. Swarm-based semantic fuzzy reasoning for situation awareness computing, in: Fuzzy Systems (FUZZ-IEEE), 2012 IEEE International Conference on, pp. 1--7.

[8] Doulaverakis, C., Konstantinou, N., Knape, T., Kompatsiaris, I., Soldatos, J., 2011. An Approach to Intelligent Information Fusion in Sensor Saturated Urban Environments, in: Intelligence and Security Informatics Conference (EISIC), 2011 European, pp. 108--115.

[9] Fenza, G., Furno, D., Loia, V., Veniero, M., 2010. Agent-based Cognitive approach to Airport Security Situation Awareness, in: Proceedings of the 2010 International Conference on Complex, Intelligent and Software Intensive Systems, IEEE Computer Society. pp. 1057--1062.

[10] Conroy, K., May, G., Roantree, M., Warrington, G., Cullen, S.J., Mc-Goldrick, A., 2011a. Knowledge acquisition from sensor data in an equine environment, in: Proceedings of the 13th international conference on data warehousing and knowledge discovery, Springer-Verlag, Berlin, Heidelberg. pp. 432--444.

[11] Gaglio, S., Gatani, L., Lo Re, G., Ortolani, M., 2007. Understanding the Environment Through Wireless Sensor Networks, in: Proceedings of the 10th Congress of the Italian Association for Articial Intelligence on AI*IA 2007: Artificial Intelligence and Human-Oriented Computing, Springer-Verlag, Berlin, Heidelberg. pp. 72--83.

[12] Liu, J., Zhao, F., 2005. Towards semantic services for sensor-rich information systems, in: Broadband Networks, 2005. BroadNets 2005. 2nd International Conference on, pp. 967--974.

[13] Whitehouse, K., Zhao, F., Liu, J., 2006. Semantic Streams: A Framework for Composable Semantic Interpretation of Sensor Data, in: Römer, K., Karl, H., Mattern, F. (Eds.), Wireless Sensor Networks. Springer Berlin / Heidelberg. volume 3868 of LNCS, pp. 5--20.

[14] Wanner, L., Vrochidis, S., Tonelli, S., Mossgraber, J., Bosch, H., Karppinen, A., Myllynen, M., Rospocher, M., Bouayad-Agha, N., Bügel, U., Casamayor, G., Ertl, T., Kompatsiaris, I., Koskentalo, T., Mille, S., Moumtzidou, A., Pianta, E., Saggion, H., Serani, L., Tarvainen, V., 2011. Building an Environmental Information System for Personalized Content Delivery, in: Hrebícek, J., Schimak, G., Denzer, R. (Eds.), Environmental Software Systems. Frameworks of eEnvironment. Springer Boston. volume 359 of IFIP Advances in Information and Communication Technology, pp. 169--176.

[15] Wei, W., Barnaghi, P., 2009. Semantic Annotation and Reasoning for Sensor Data. Smart Sensing and Context , 66--76.

# Appendix
# Situation Theory

- A situation, *s*, supports (one or more) infon(s) $\sigma$
  - Formally, $s \vDash \sigma$
- An infon is a tuple $\langle\langle R, a_i, \ldots a_m, 0/1 \rangle\rangle$ where
  - *R* is a relation
  - $a_i, \ldots a_m$ are objects appropriate to the relation *R*
  - And *0/1* is the polarity, whereby *1* means that $a_i, \ldots a_m$ "stand in" *R*
- Example
  - Situation of "unhealthy exposure to $PM_{2.5}$ in Kuopio over two days"
  - $s \vDash \langle\langle \textit{unhealthy-exposure}, PM_{2.5}, \textit{Kuopio}, 2013\text{-}04\text{-}15, 2013\text{-}04\text{-}16, 1 \rangle\rangle$
- Developed by Barwise and Perry, extended by Devlin
- Relates to Situational Awareness by Endsley

# Appendix
# Ontology

- An ontology formally represents knowledge as a set of concepts within a domain, and the relationships between pairs of concepts [Wikipedia: Ontology (information science)]
- A document that describes concepts and relationships of a domain; such documents are "written" by means of a (ontology) language
- An ontology is external to, and shared among, software systems
- Software systems commit to one (or more) ontology by adopting the defined terminology and semantics
- Key technology in knowledge representation and reasoning
- Developed within artificial intelligence research
- Used in many domains, e.g. bioinformatics (see Gene Ontology)

# Appendix
# Semantic Sensor Network (SSN) ontology

- Vocabulary for the representation of knowledge for sensors, properties, features, observations, …
- Adopted at the Wavellite observation layer
- Domain-specific sensors and measured properties of features are accommodated by extending from SSN ontology
- Examples
  - BetaAttenuationMonitor *subClassOf* ssn:SensingDevice
  - beijingAirBAM *isA* BetaAttenuationMonitor
  - pm25 *isA* ssn:FeatureOfInterest
  - concentration *isA* ssn:Property

# Appendix
# Situation Theory Ontology (STO)

- Vocabulary for the representation of knowledge for situations
- Borrows from Situation Theory
- Adopted at the Wavellite situation layer
- Domain-specific situations and relations are accommodated by extending from STO
- Example
  - AirPollutionEvents *subClassOf* sto:Situation
  - unhealthy-exposure *isA* sto:Relation