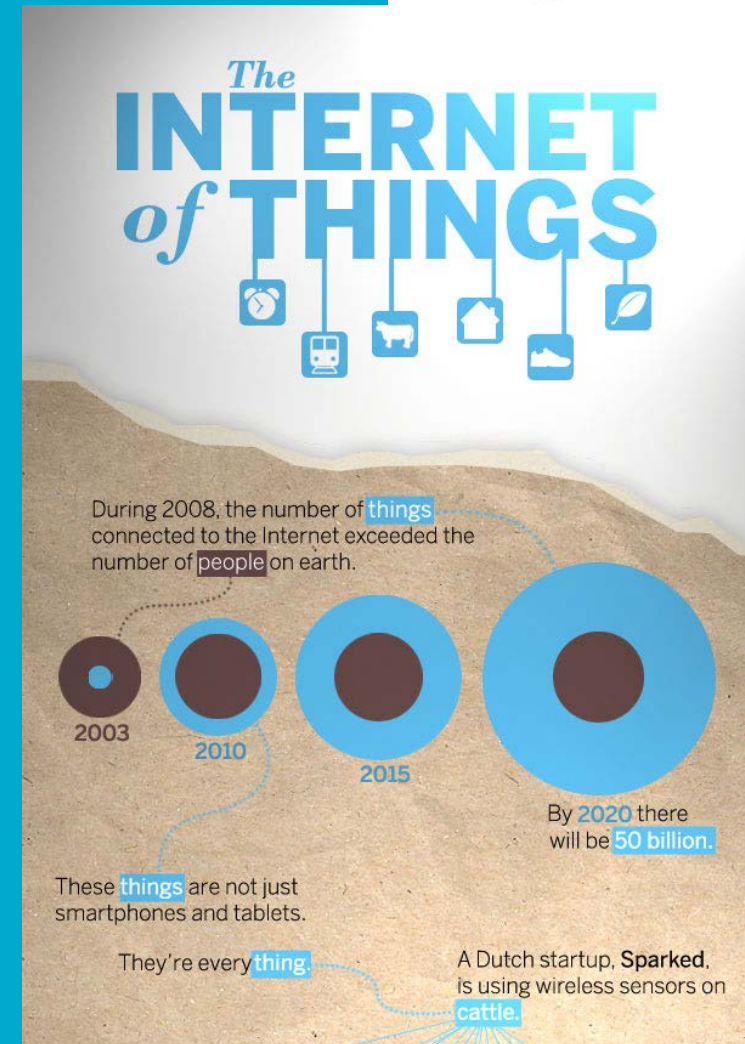


Context-aware Sensor Search, Selection and Ranking Model for Internet of Things Middleware

Charith Perera, Arkady Zaslavsky,
Peter Christen, Michael Compton
and Dimitrios Georgakopoulos

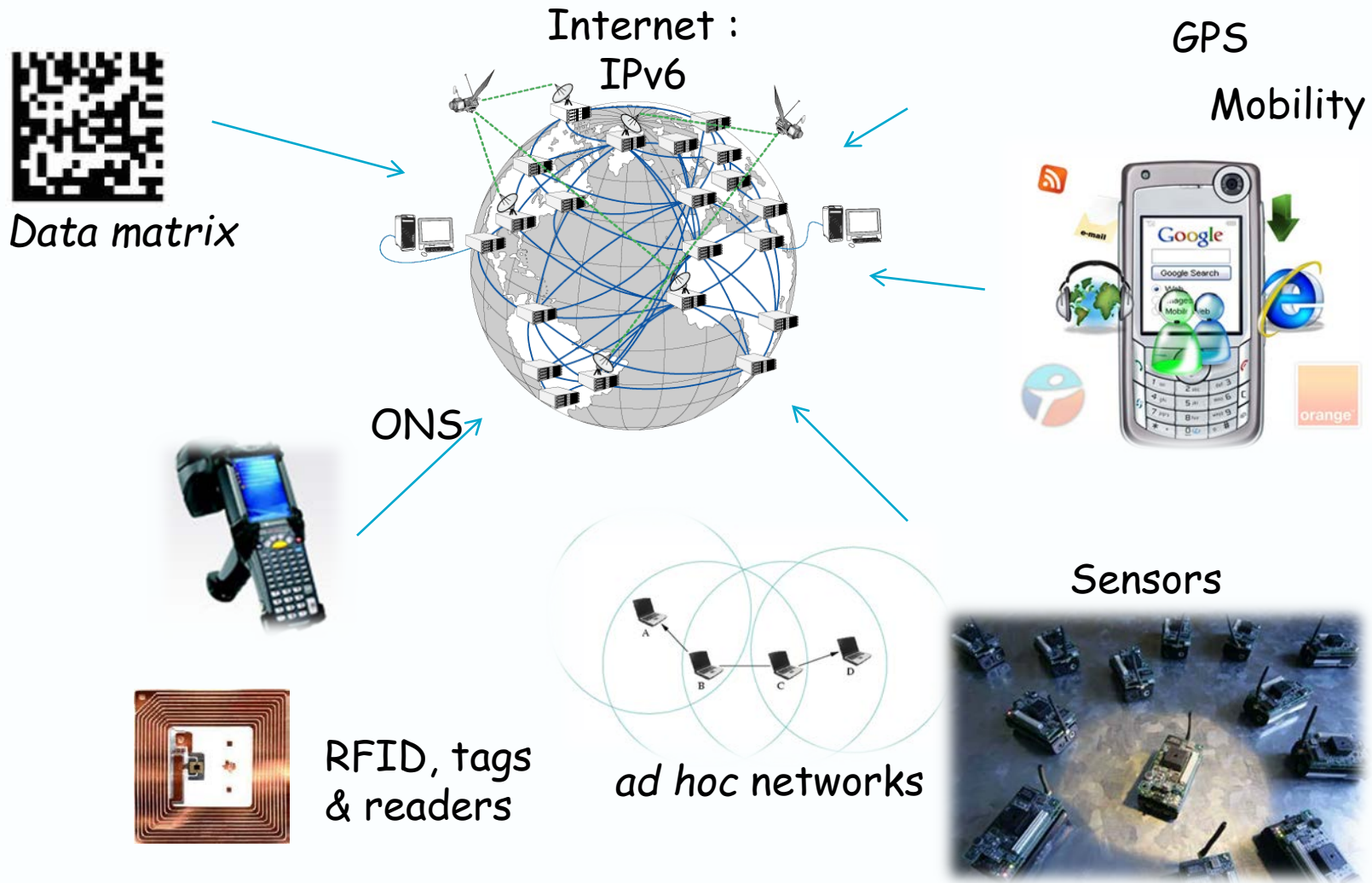


Outline

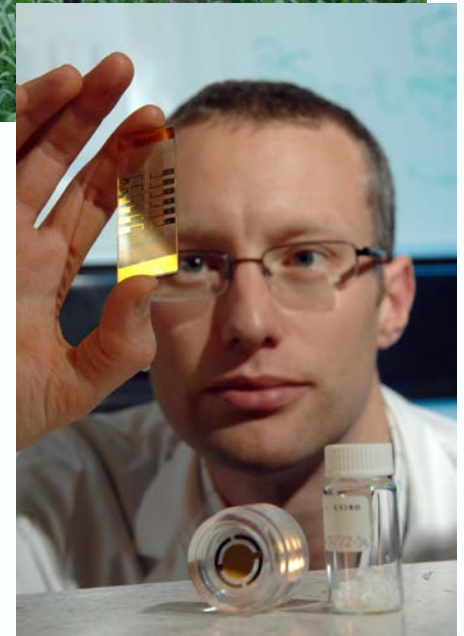
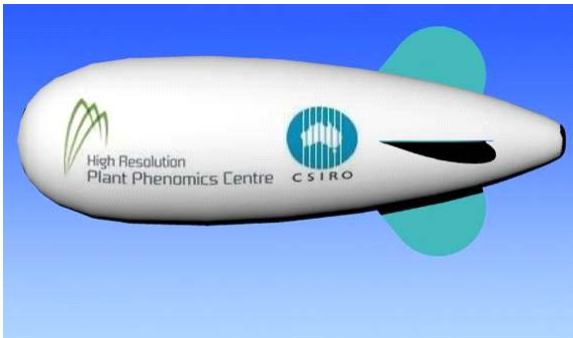
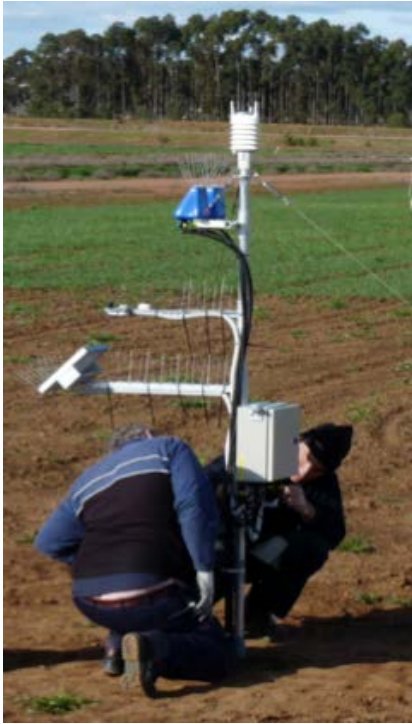
- Setting the IoT scene
- Motivating scenarios
- CASSARA Model & Tool
- Conclusion



Rather : a network of converging networks



CSIRO Things – Sensors, cameras, nanosensors on the ground, ocean, autonomous vehicles & airships



Other Things – Other Smart Internet Connected Objects



Nike shoe sensor



CSIRO virtual fence



Stick on RFIDs



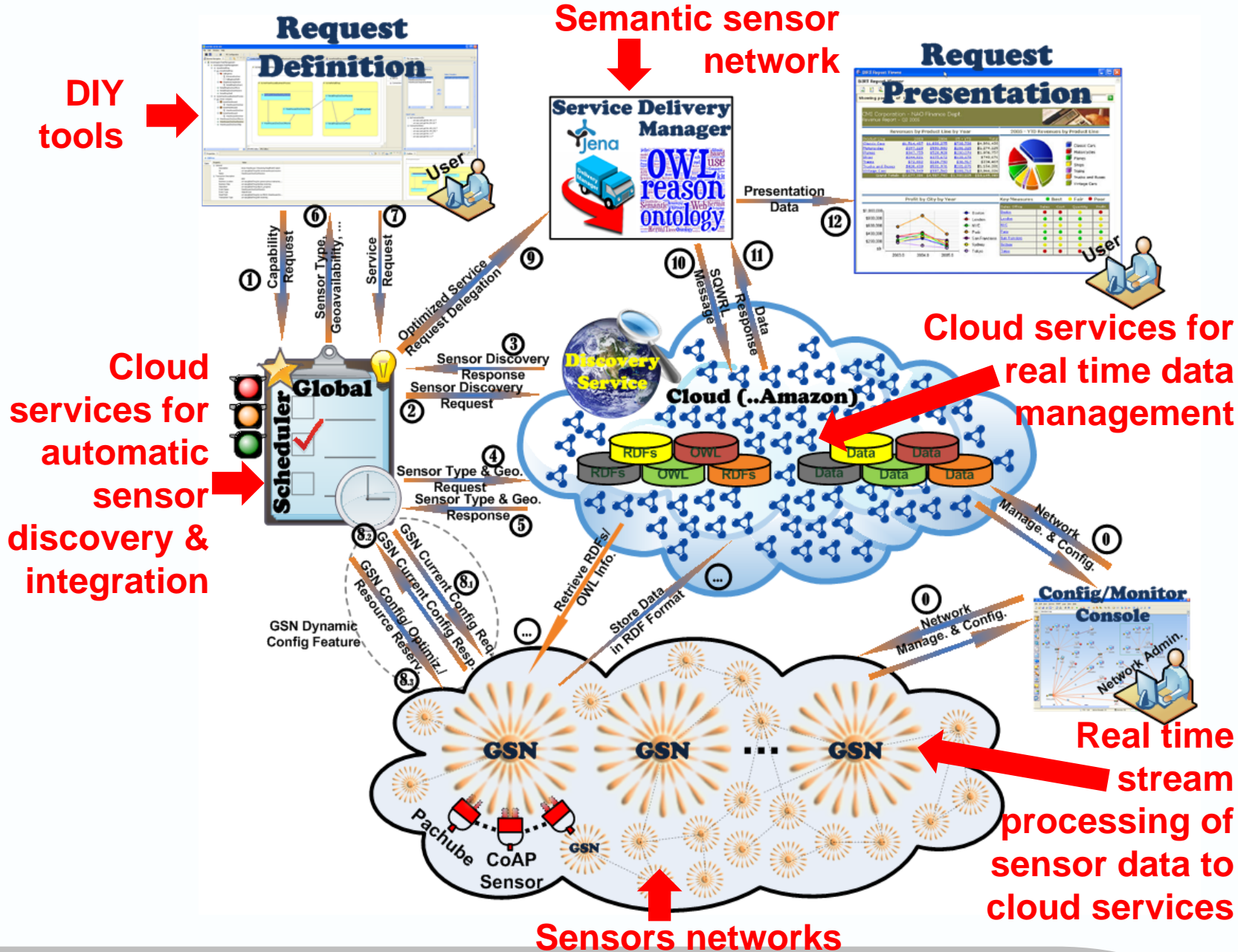
Olinda radio



Smart meter



Proteus pill



Motivating Scenario

- An office building just has been renovated. The owner wants to evaluate dust concentration, places which require careful cleaning and deploys massive amounts of low-cost sensors
- Not much point in collecting and processing values from all 1000s sensors
- From which sensors you would like to collect data ??? ← **PROBLEM**
- **What factors matters ?? (YES location is the most critical, but what else ??)**
- Assume: if there 20 sensors within 10m² and user wants only 4 sensors, how to select the BEST 4 sensors
- What is meant by BEST → **BEST means most suitable to user needs**
- Examining context information allows to select the BEST sensors

EXAMPLE set of context information related to sensor selection

availability, accuracy, reliability, response time, frequency, sensitivity, measurement range, selectivity, precision, latency, drift, resolution, detection limit, operating power range, system (sensor) lifetime, battery life, security, accessibility, robustness, exception handling, interoperability, configurability, user satisfaction rating, capacity, throughput, cost of data transmission, cost of data generation, data ownership cost, bandwidth, and trust.

What MATTERS to you MOST ?

Give more priority to them

Background:

No existing system provide such sensor search functionality ☹

Microsoft Research **SensorMap** (b)

Go To Location | Manage Views | Manage Visualizations | Time Traveler | View Permalink | Sign In

2D 3D Road Aerial Bird's eye Labels

Weather WebCam

Search

Location Data Sparql endpoint

Choose sensor

Sensor bikehire

Location City

Lat 15.6230368 Long 60.8203125

Manage Views

Popular Views

- Seattle SwissEx:Genepi
- SwissEx:Wannengrat
- NWSP:Singapore
- NTHU:Taiwan

Filter Sensors by Type

- ☒ Temperature
- ☒ Rotating Video Camera
- ☒ Traffic

Filter Sensors by Search

Filter by Search

Save Current View

Save View

Saved Views

You have no saved SensorMap views.

A SensorMap saved view saves your SensorMap settings (e.g., map location and sensor types being viewed).

Tags: a SensorMap view

© 2012 Microsoft Corporation | Terms of Use | Privacy | About

Help | Contact Us

Semantic Sensor Networks (SSN)

Goals and Outcomes

The goal of the W3C Incubator Activity XG was to develop an SSN ontology for sensor discovery and dynamic integration of:

- Heterogeneous sensors and other internet connected objects (ICOs)
- All data produced by such sensors and other ICOs
- Different sensor networks

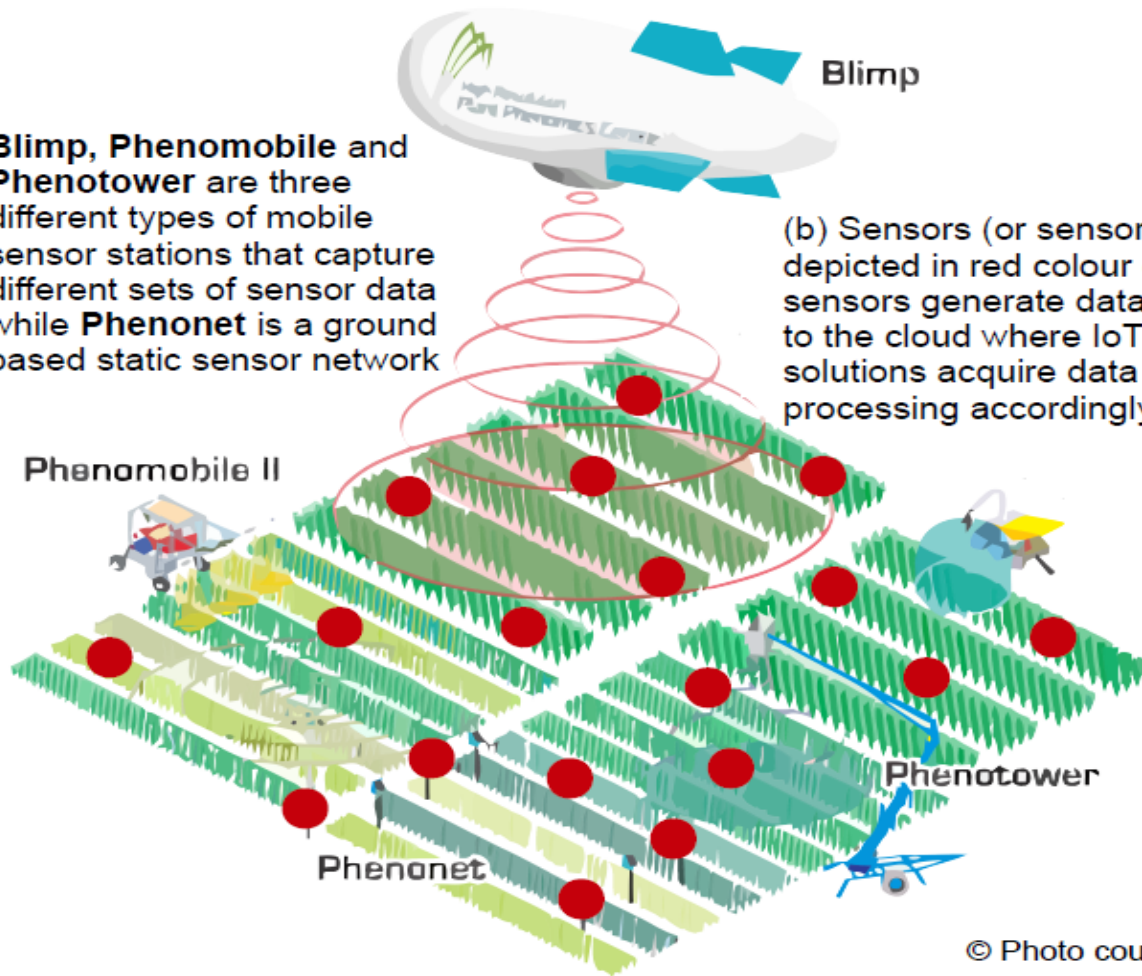
Results

- SSN ontology for description of sensors and sensor networks
- Extended OGC's Sensor Model Language (SensorML) and four Sensor Web Enablement (SWE) languages, to support such semantic annotations

Sensor-based monitoring in digital agriculture

Blimp, Phenomobile and Phenotower are three different types of mobile sensor stations that capture different sets of sensor data while **Phenonet** is a ground based static sensor network

(b) Sensors (or sensor stations) are depicted in red colour dots. These sensors generate data and upload to the cloud where IoT middleware solutions acquire data and do the processing accordingly.

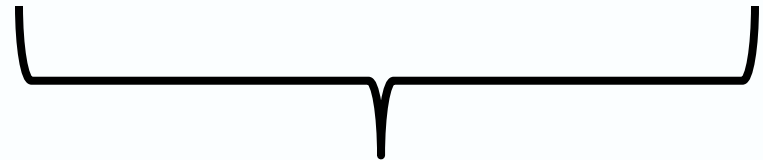


© Photo courtesy of CSIRO

CASSARAM concepts

Data Models

We extended the Semantic Sensor Network Ontology (SSNO) as follows:



This is how we extended the SSNO (orange colour)

We normalize $[0,1]$ the context information accordingly using min-max ranges.
(e.g. accuracy 74 means 0.74)

We generated 1 millions synthetic sensor data descriptions (individuals).
Similar to the one example depicted in green color above

Algorithm 1 Execution Flow of CASSARAM

Require: (\mathbb{O}) , (\mathbb{P}) , (\mathbb{Q}) , (\mathbb{N}) , $(\mathbb{S}_{Results})$, $(\mathbb{S}_{Indexed})$, (\mathbb{M}) .

```
1: Output:  $\mathbb{S}_{Results}$ 
2:  $\mathbb{S}_{Filtered} \leftarrow queryOntology(\mathbb{O}, \mathbb{Q})$ 
3: if  $cardinality(\mathbb{S}_{Filtered}) < \mathbb{N}$  then
4:   return  $\mathbb{S}_{Results} \leftarrow \mathbb{S}_{Filtered}$ 
5: else
6:    $\mathbb{P} \leftarrow captureUserPriorities(\mathbb{UI})$ 
7:    $\mathbb{M} \leftarrow \text{Plot Sensors in Multidimensional Space}(\mathbb{S}_{Results})$ 
8:    $\mathbb{S}_{Indexed} \leftarrow calculateCPWI(\mathbb{S}_{Results}, \mathbb{M})$ 
9:    $\mathbb{S}_{Results} \leftarrow rankSensors(\mathbb{S}_{Indexed})$ 
10:   $\mathbb{S}_{Results} \leftarrow selectSensors(\mathbb{S}_{Results}, \mathbb{N})$ 
11:  return  $\mathbb{S}_{Results}$ 
12: end if
```

Phase 1: Search

Users express their priorities using GUI tool that generates the SPARQL

The screenshot shows the CASSARA Tool interface. It includes a 'Total number of Sensors available' set to 100 and a 'Scale' set to 10. The 'Context Properties' section lists various sensor attributes with sliders and input fields for priority and ideal value. The 'Sensors ranked according to user preferences' section displays two tables of sensor data.

Context Properties:

Property	Low Priority	High Priority	Ideal Value
Availability	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0.5
Accuracy	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0.0
Reliability	<input type="checkbox"/>	<input type="checkbox"/>	0.0
Response time	<input type="checkbox"/>	<input type="checkbox"/>	0.0
Frequency	<input type="checkbox"/>	<input type="checkbox"/>	0.0
Sensitivity	<input type="checkbox"/>	<input type="checkbox"/>	0.0
Measurement range	<input type="checkbox"/>	<input type="checkbox"/>	0.0
Selectivity	<input type="checkbox"/>	<input type="checkbox"/>	0.0
Precision	<input type="checkbox"/>	<input type="checkbox"/>	0.0
Latency	<input type="checkbox"/>	<input type="checkbox"/>	0.0
Drift	<input type="checkbox"/>	<input type="checkbox"/>	n/a

Sensors ranked according to user preferences:

SensorID	Index
Sensor29	0.08644940351837434
Sensor54	0.08965059628780471
Sensor13	0.09584703846280879
Sensor88	0.10299887460391781
Sensor71	0.10912509075430306
Sensor1	0.11548635317974802
Sensor22	0.1238915114279971
Sensor75	0.1271692996649516
Sensor69	0.13291448180045895
Sensor20	0.1333482742538481

Generated SPARQL Query:

```
select ?sensor ?availability ?accuracy where{ ?sensor ssn:hasMeasurementCapability ?sensorcapa. ?sensorcapa ssn:hasMeasurementProperty ?property1. ?property1 ssn:hasDataValue ?availability. ?property1 ssn:type ssn:Availability. ?sensorcapa ssn:hasMeasurementProperty ?property2. ?property2 ssn:hasDataValue ?accuracy. ?property2 ssn:type ssn:Accuracy. FILTER ( 0.5 < ?availability ) FILTER ( ?availability < 0.9 ) }
```

Additional Sensor Results:

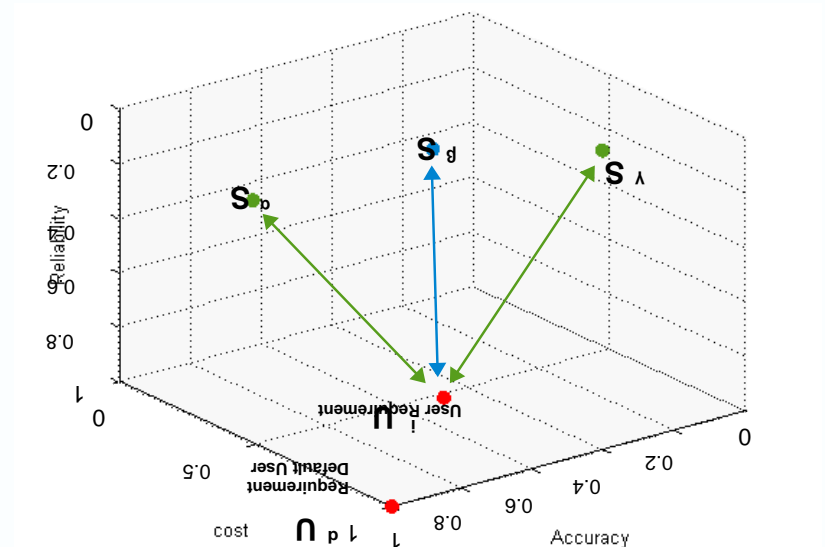
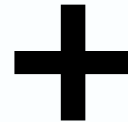
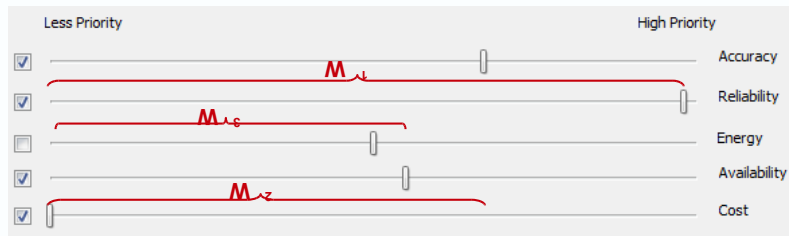
SensorID	Index
Sensor83	0.13682406679371756
Sensor11	0.13709185367997892
Sensor28	0.1565319746092264
Sensor84	0.17457376908372016
Sensor47	0.18888507156144413
Sensor10	0.19904706273430364
Sensor37	0.2000330495421254
Sensor98	0.2019538431710982
Sensor72	0.20293068161877337
Sensor96	0.20697359268729346
Sensor86	0.22055680592662125
Sensor93	0.22413106187035575
Sensor26	0.22428155197660563

Buttons: Show SPARQL, Select Sensors



```
select ?sensor ?availability ?accuracy ?reliability ?responsetime where{ ?sensor ssn:hasMeasurementCapability
?sensorcapa. ?sensorcapa ssn:hasMeasurementProperty ?property1. ?property1 ssn:hasDataValue ?availability. ?property1
ssn:type ssn:Availability. ?sensorcapa ssn:hasMeasurementProperty ?property2. ?property2 ssn:hasDataValue ?accuracy.
?property2 ssn:type ssn:Accuracy. ?sensorcapa ssn:hasMeasurementProperty ?property3. ?property3 ssn:hasDataValue
?reliability . ?property3 ssn:type ssn:Reliability. ?sensorcapa ssn:hasMeasurementProperty ?property4. ?property4
ssn:hasDataValue ?responsetime . ?property4 ssn:type ssn:ResponseTime.}
```

Phase 2: Index



Generate similarity index by combining context information and user priorities. (i.e. smaller the index, closer to the user preferred request)

Phase 3: Rank

Sort the sensors using indexes

(i.e. smaller the index, closer to the user preferred request)

OR one can use the inverse $(1-x)$ as illustrated in the GUI: no difference

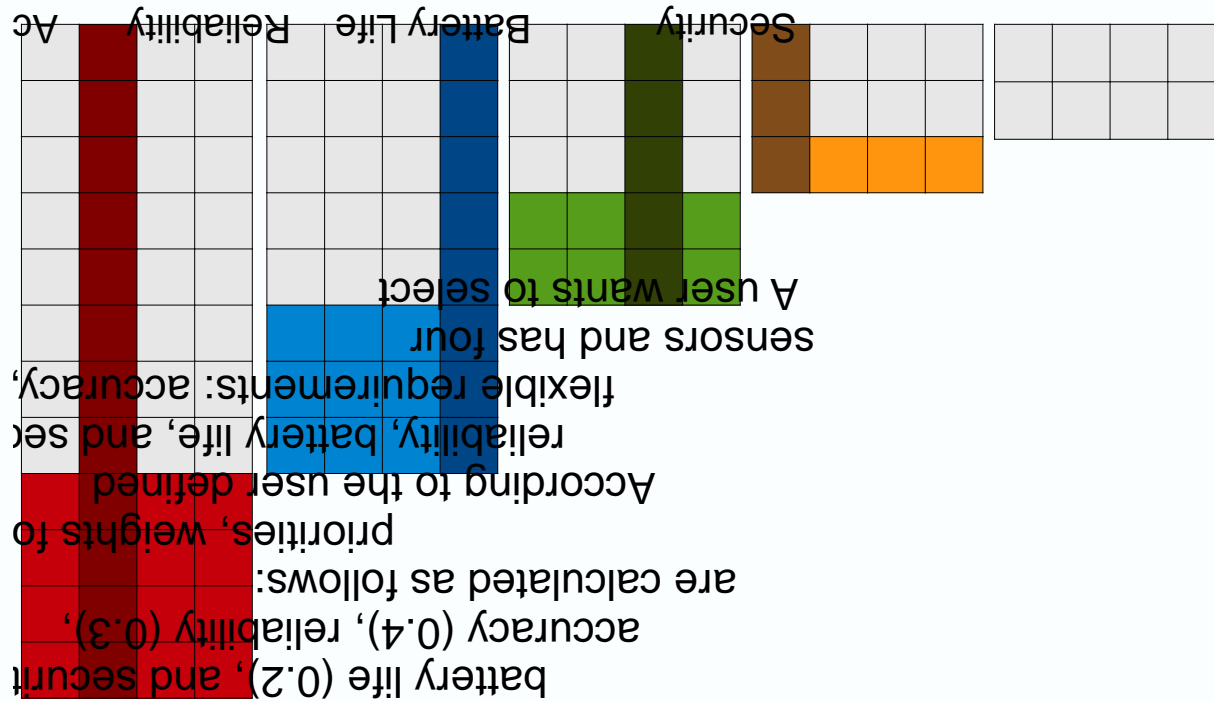


Phase 4: Select

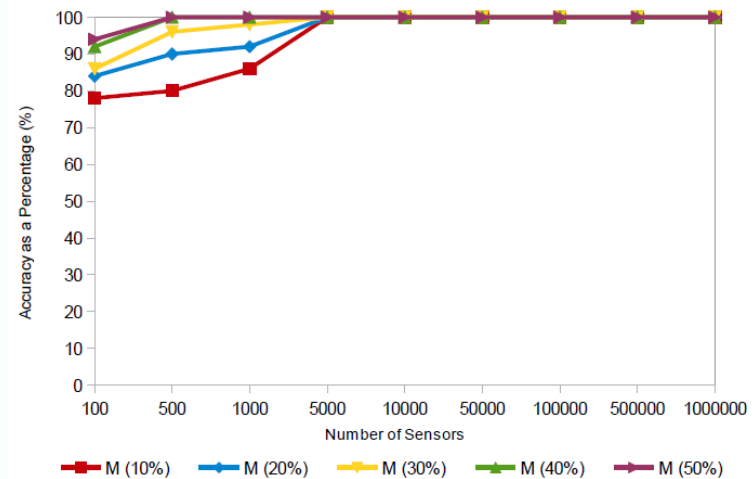
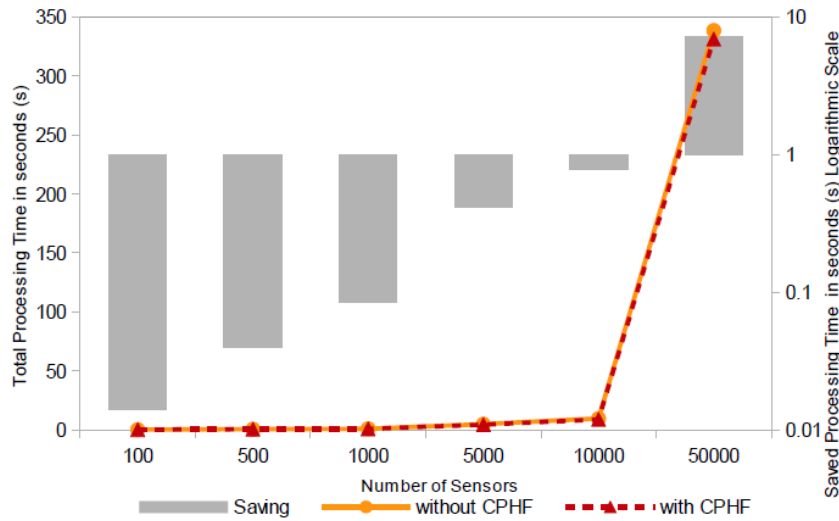
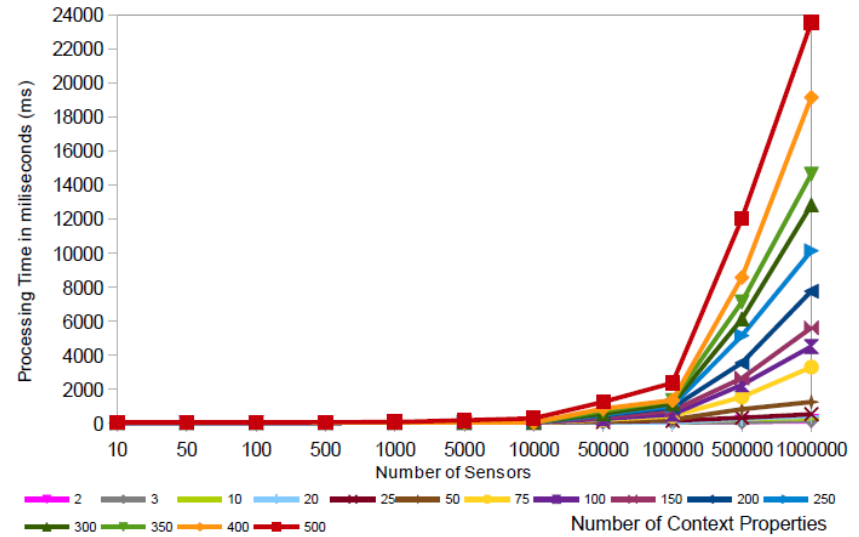
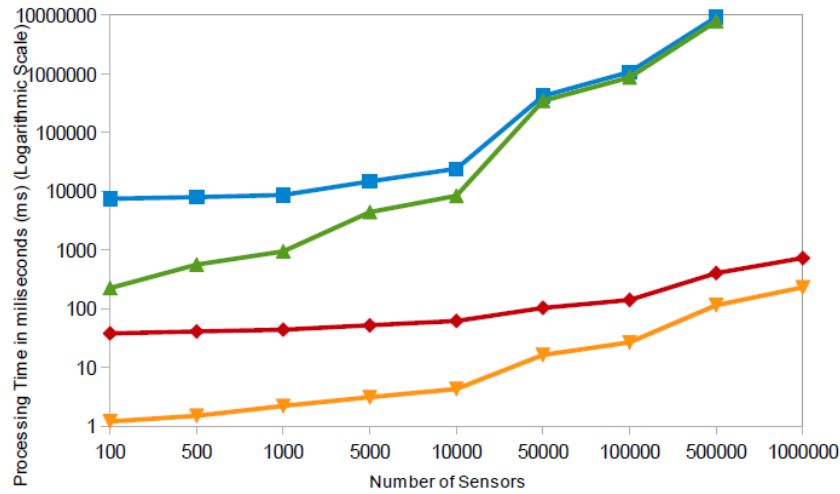
select the top-k sensors from the sorted list

Extended Features I

Comparative Priority-based Heuristic Filtering (CPHF)



Evaluation



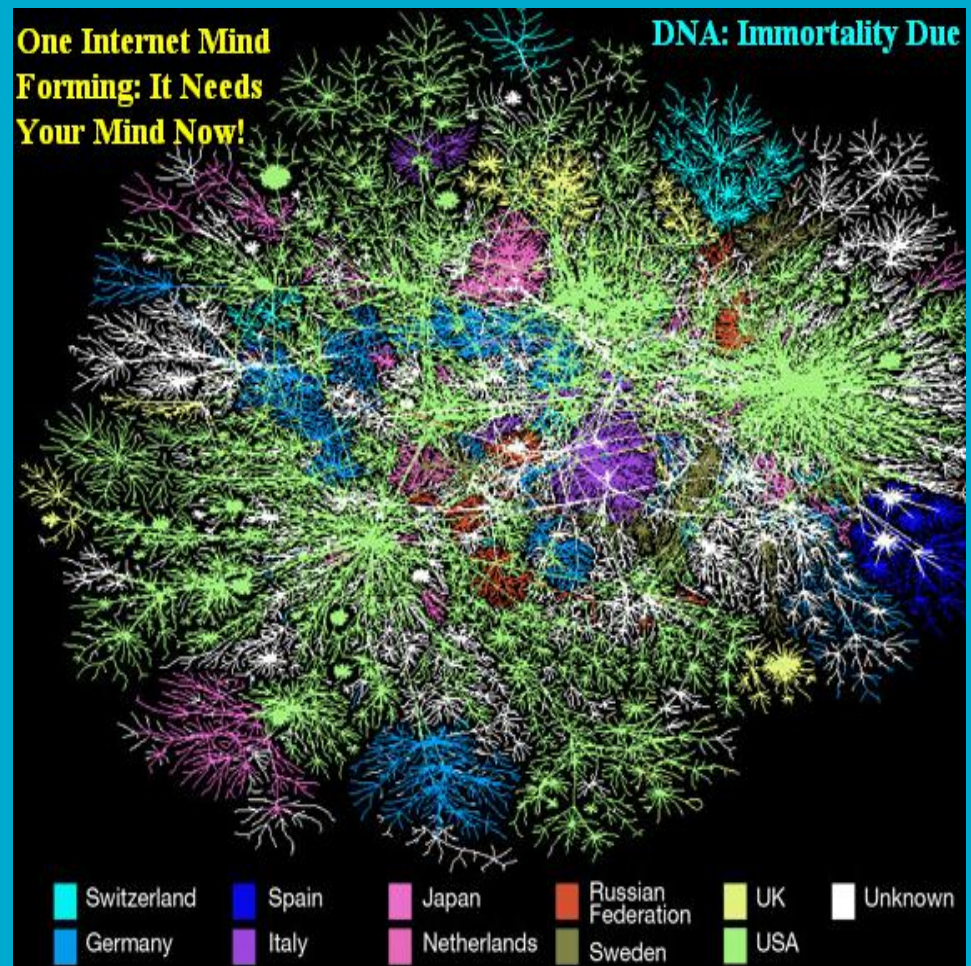
Conclusions

- Context-based framework for IoT sensors
- Extended SSNO
- CASSARAM
- CASSARA tool for prioritising sensor properties
- Comparative Priority-based Heuristic filtering
- Prototype development, performance and efficiency measurement

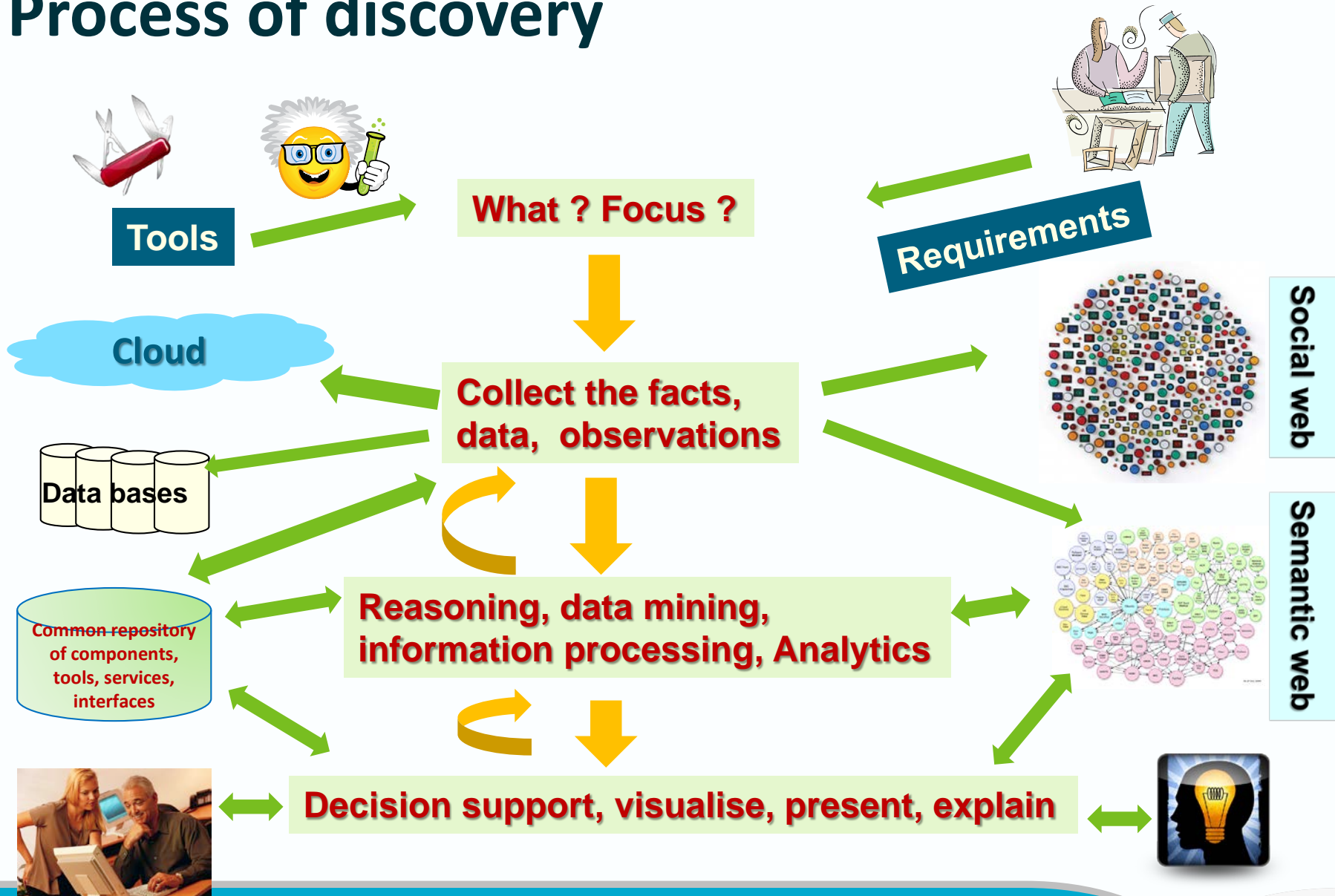
Thank you !

Dr Arkady Zaslavsky, Professor
Senior Principal Research Scientist
Science Leader in Semantic
Data Management
Phone: 02 6216 7132
Email: arkady.zaslavsky@csiro.au

www.csiro.au



Process of discovery





Semantic Sensor Networks (SSN)

W3C Incubator Activity XG (2009-11)

<http://www.w3.org/2005/Incubator/ssn>



kno.e.sis

Chairs:

- Commonwealth Scientific and Industrial Research Organization, Australia
- Kno.e.sis Lab, Wright State University, USA

Members:

- Ericsson, USA
- Boeing, USA
- Fundacion CTIC, Spain

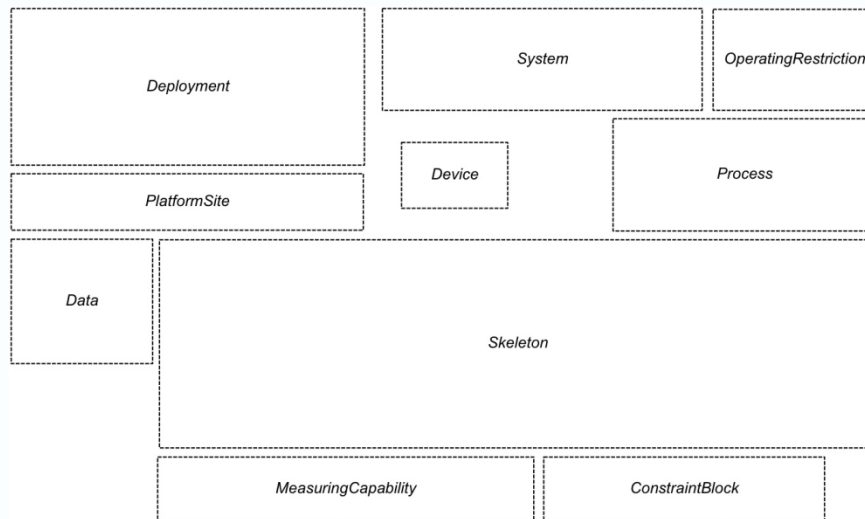
Members (continued):

- National University of Ireland (NUIG), Digital Enterprise Research Institute (DERI), Ireland
- University of Surrey, UK
- Universidad Politécnica de Madrid, Spain
- Fraunhofer Gesellschaft, Germany
- Pennsylvania State University, USA
- The Open University, UK
- University of Southampton, UK
- Monterey Bay Aquarium Research Institute, USA

....

SSN Ontology structure and uses

The SSN ontology consist of several ontology modules



The ontology can be used for a focus on any (or a combination) of a number of perspectives:

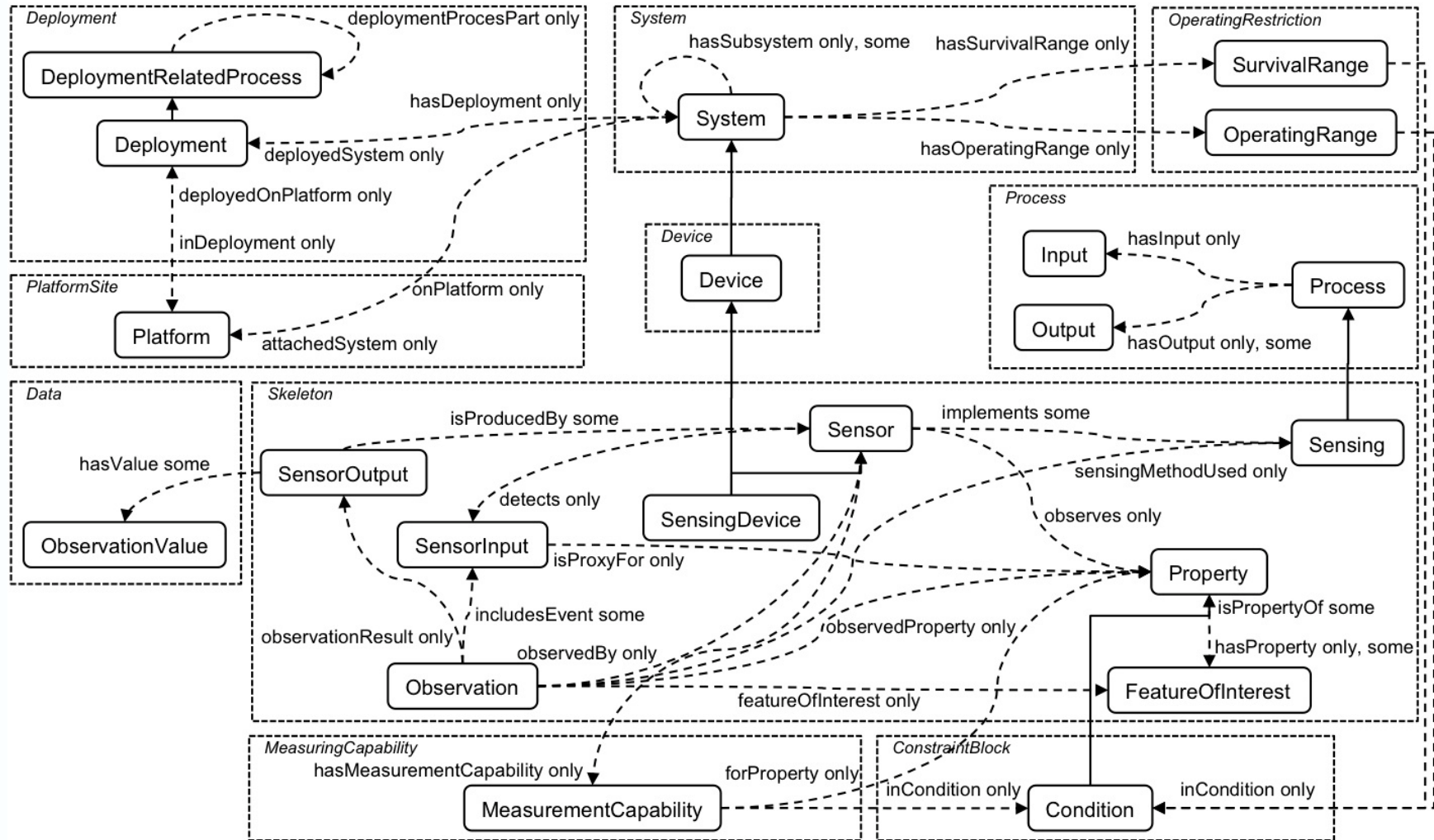
A sensor perspective, with a focus on what senses, how it senses, and what is sensed

A data or observation perspective, with a focus on observations and related metadata

A system perspective, with a focus on systems of sensors, or

A feature and property perspective, with a focus on features, properties of them, and what can sense those properties

The SSN Ontology



Adoptions of the SSN Ontology (more recently)



Linked Sensor Data

- <http://knoesis.wright.edu/>



EU FP6 SPITFIRE

- <http://spitfire-project.eu/>



EU FP7 Exalted project

- <http://www.ict-exalted.eu/>



EU FP7 SemSorGrid4Env

- <http://www.semsorgrid4env.eu/>



EU FP7 OpenIoT

- <http://www.openiot.eu/>