

Bezvadu **S**ensoru **T**īkli

Semantiski sensoru tīkli

Reinholds Zviedris
Datorikas fakultāte
Latvijas Universitāte
07.12.2016.

Organizatorisks atgādinājums

- Datumi:
 - 06.01.2017. 13:00-15:00 – Melnā piektdiena
 - 11.01.2017. 14:30-16:00 – Eksāmens
 - 18.01.2017. 14:30-18:00 – Praktisko darbu projektu atrādīšana, atzīmju izlikšana

Semantika

Attiecība starp zīmēm un to jēdzienisko nozīmi, kas nav atkarīgas no to interpretēšanas un izmantošanas.



Attēls no: commons.wikimedia.org

Konteksts

- Visur esošā skaitļošana (Ubiquitous Computing)
- Lietu internets (Internet of Things)
- Lietu tīmeklis (Web of Things)
- Dati, kurus paredzēts:
 - publicēt (padarīt pieejamus citiem)
 - spēt vēlāk saprast un izmantot
 - mākoņskaitļošana (Cloud computing)

Saistītās tēmas

- Semantika, meta dati
- Saistītie dati un semantiskais tīmeklis (Linked Data and Semantic Web)
- Zinātne par datiem (Data Science)
- Lielie dati (Big Data)

Kas būtu nepieciešams...

- Lai savāktos datus varētu saprast, interpretēt un izmantot:
 - citi cilvēki?
 - paši pētnieki pēc kāda laika?

Semantisko sensoru tīklu tēmas

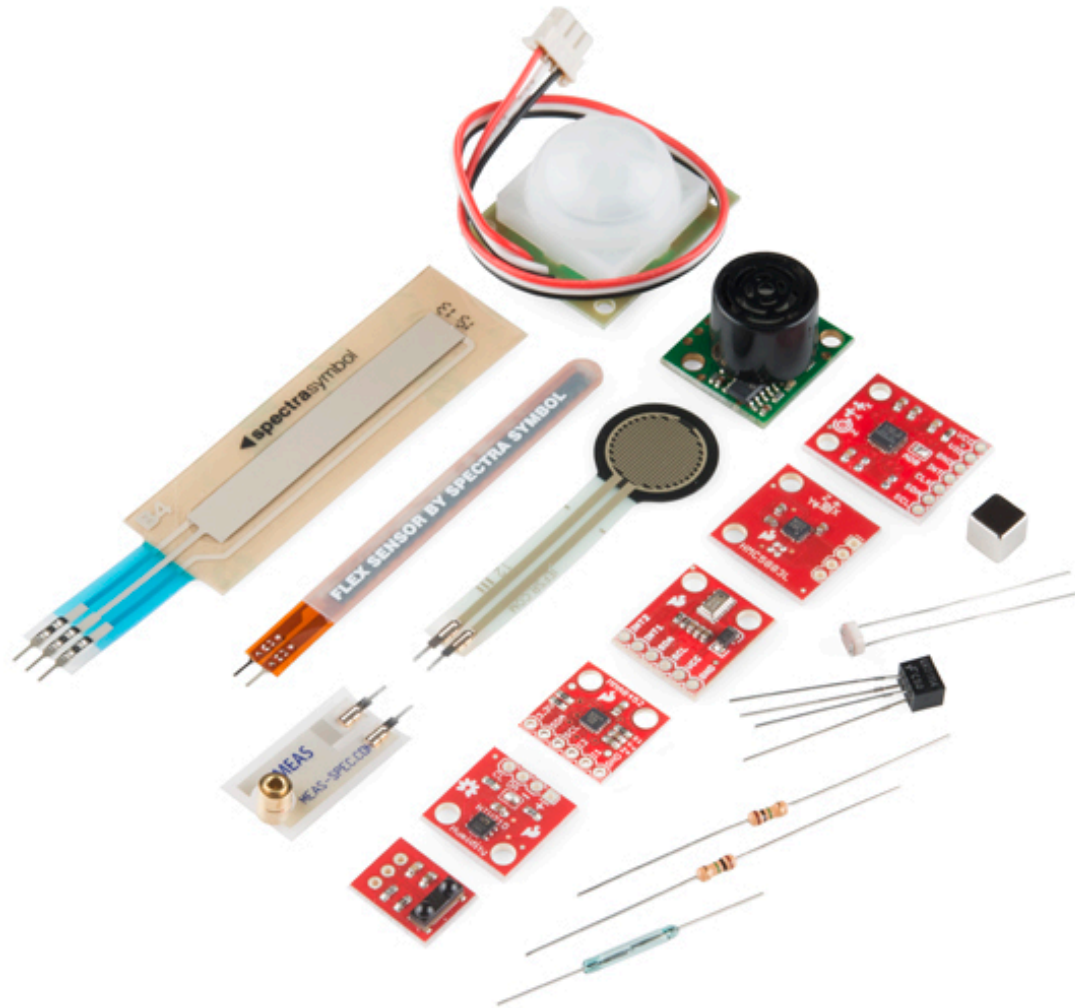
- Sensoru datu un meta datu avoti
 - Sensoru datu ieguve no μ C, telefona, bezvadu tīkliem, pūļa (crow sourcing)
 - Meta datu ieguve ar ievades metode vai no sensoru datiem
- Sensoru datu modelēšana un publicēšana
 - Sensoru datu ontolīgiskais attēlojums
 - Meta dati kā saistītie dati (Linked Data)
 - APIs
 - Vēsturisko datu publicēšana

Semantisko sensoru tīklu tēmas (2)

- Sensoru datu pielietojums
 - Sarežģītu notikumu apstrāde no sensoru tīklu datu plūsmas
 - Sensoru tīmekļa API izveide
 - Visur esošās skaitļošanas pielietojumi
 - Meklēšana sensoru datos

Atkāpe

Kas tad ir sensori?



Attēls no: sparkfun.com

Kā Jūs ar tiem sarunājaties?

- Ar vienkāršiem analogiem vai digitāliem signāliem
- Izmantojot vienkāršus protokolus, piemēram I²C
- Tie nesaprot HTTP vai TCP/IP
- Tie jāpieslēdz kādai “gudrākai” iekārtai, piemēram, mikrokontrolierim

OK, kā Jūs sarunājaties ar μ C?

- Izveidojot programmu, piemēram, MansOS vai TinyOS vai Arduino IDE
- Liekot tai izvadīt datus uz seriālā porta vai pa radio
- Pieslēdzot tīkla saskarni, iespējama arī saziņa caur HTTP

Ko mēs varam izmērīt ar sensoriem?

Vienkāršas fiziskas īpašības

- Spēks
- Spiediens
- Stiepe
- Deformācija
- Locīšanās
- Magnētiskais lauks
- Utt. utjp.

Vide

- Temperatūra
- Gaisa spiediens
- Mitrums
- Gaisma
- Gāzes koncentrācija
- Troksnis
- Utt. utjp.

Kur Jūs esat, kas Jums apkārt

- GPS
- Kompas
- Augstums
- Paātrinājums
- Attālums
- Skaņa
- Utt. utjp.

Cilvēks

- Elpošanas biežums
- Sirdspuksti
- Asinsspiediens
- Ādas temperatūra
- EEG
- Balss
- Utt. utjp.

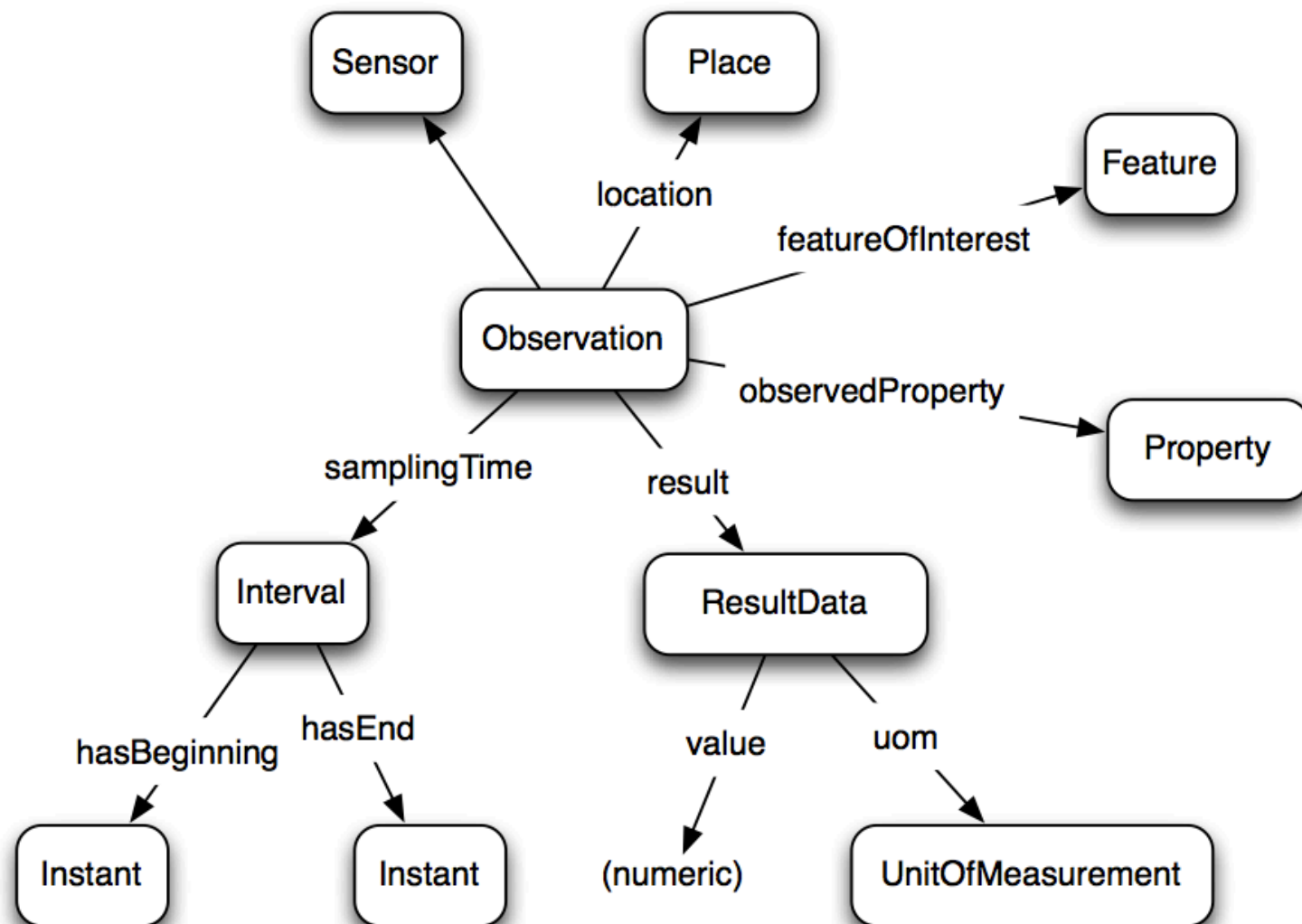
Sensoru lietojumu veidi

- Laika intervāla bāzētie lietojumi
 - Sensoru ievāc datus regulāros intervālos
- Notikumu bāzētie lietojumi
 - Sensori ievāc datus tikai notiekot konkrētiem notikumiem

Sensoru tīklu ontoloģijas

- Novērojumu un mērījumu modelis OWL, ko virza OGC Sensor Web Enablement Initiative
- Semantisko sensoru tīklu ontoloģija, ko virza W3C Semantic Sensor Network Incubator Group
- OntoSensor ontoloģija, kas uzbūvēta uz SUMO un ISO 19115 (ģeo meta dati)
- Daudzas citas

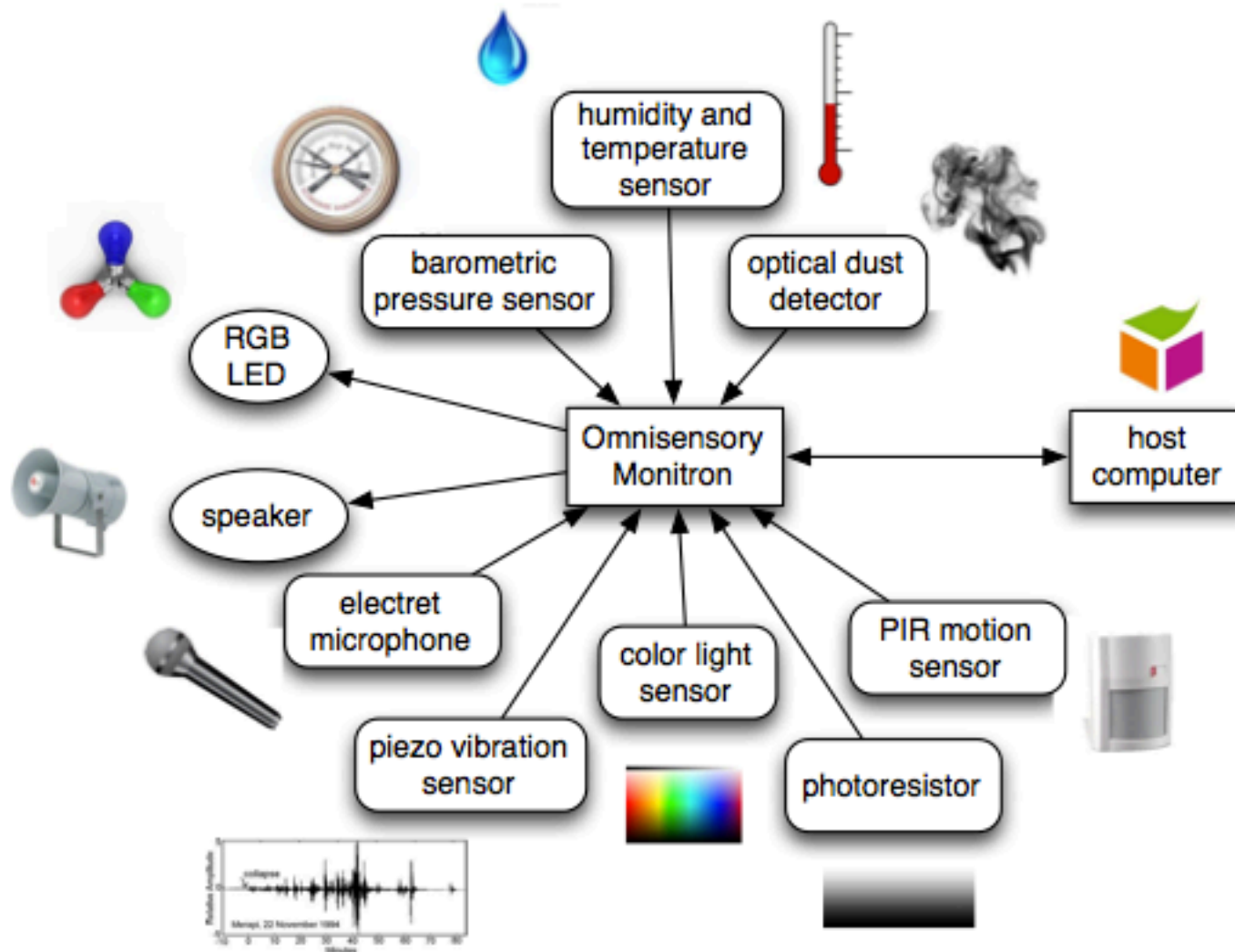
Novērojuma modelēšana



Piekluve datiem

- Ar RDF
- Izmantojot SPARQL

Piemērs: the Monitron

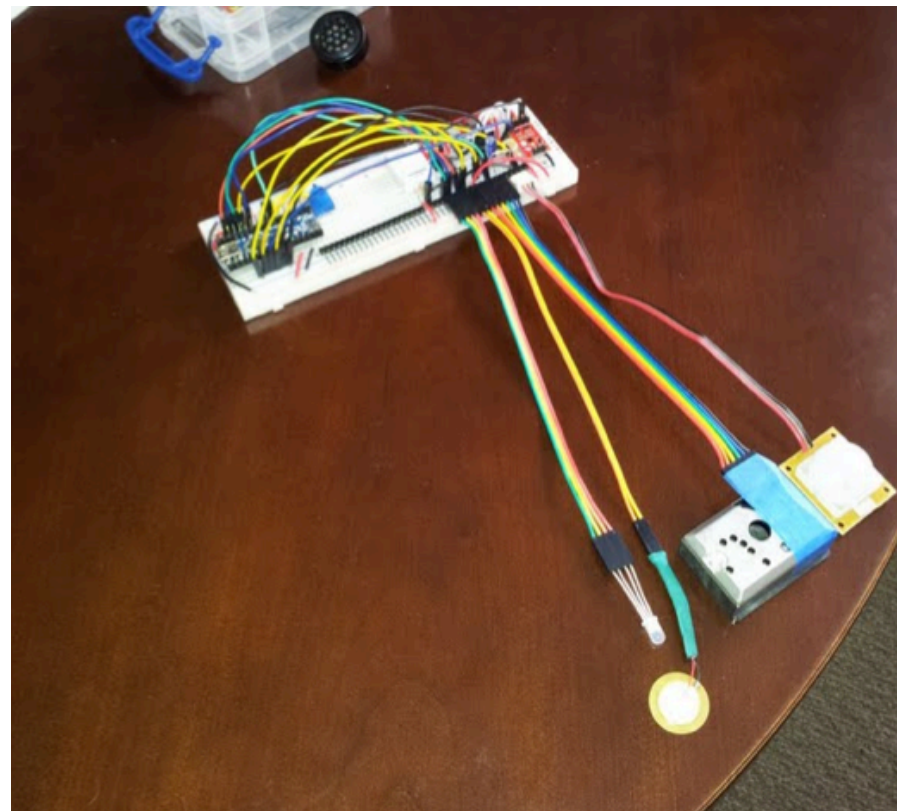


Semantic Synchrony arhitektūra

- Arduino Nano + sensori
- Sensoru dati tiek ievākti cikliski – novērojumi tiek veikti regulāros intervālos un dati tiek agregēti
- Novērojumi un kļūdu paziņojumi tiek nosūtīti uz bāzes datoru kā OSC ziņojumi
- Bāzes dators var nosūtīt paziņojumus Monitron iekārtai
- Programma uz bāzes datora pārveido saņemtos datus RDF formātā, pievienojot tos RDF plūsmai izmantojot RDFAgents, kā arī katrs novērojums tiek iekļauts RDF grafā

1. solis: uzbūvēt iekārtu

- Pasūtīt detaļas
- Sagriezt vadus
- Pielodēt te
- Uzlikt termoizlācību tur
- Gatavs!



2. solis: uzrakstīt programmatūru

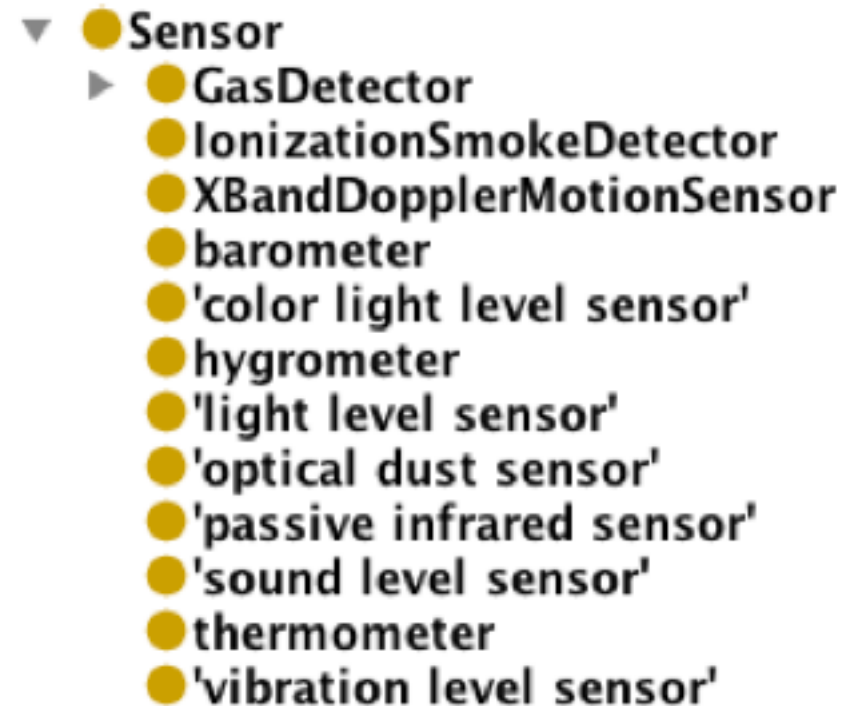
- Izvads izskatās sekojošs:

```
/om/system/time 1753A23
/om/sensor/7bb20610/vibr 1753A23 1754CEE 10000 0.003 0.022 0.021 0.000000
/om/sensor/md9745apzf/sound 1753A23 1754CEE 10000 0.254 0.792 0.507 0.001740
/om/sensor/photo/light 1753A23 1754CEE 10000 0.561 0.567 0.565 0.000020
/om/sensor/rht03/humid 1754CFC 1754D03 1 0.490 0.490 0.490 0.000000
/om/sensor/rht03/temp 1754CFC 1754D03 1 1.000 22.600 22.600 0.000000
/om/sensor/bmp085/press 1754D0B 1754D17 1 1.000 101209.000 101209.000 0.000000
/om/sensor/bmp085/temp 1754D0B 1754D17 1 1.000 23.500 23.500 0.000000
/om/sensor/gp2y1010au0f/dust 1754D1F 1754D29 1 0.160 0.160 0.160 0.000000
/om/sensor/sel0/motion 1753A23 1754D2E 1 0
```

- Katram novērojumam:
 - Sensora ID, laika intervāls, novērojumu skaits, minimālā vērtība, maksimālā vērtība, vidējā vērtība, starpība

3. solis: paplašināt sensoru ontoloģiju

- Jums nepieciešamas klases katram Jūsu iekārtas sensoram
- Katra izdod savādāku novērojumu
- Ar savādākām mērvienībām



4. solis: aprakstiet sensoru

- Katrs sensors ir Sensor apakšklases instance
- Iekārtai ir vairāk kā vienu sensors, ja tā mēra vairāk kā vienu kvantitāti (piemēram, temperatūra + spiediens)

```
:avago-adjd-s311-cr999_1 a m:ColorLightLevelSensor ;  
  rdfs:label "color light level sensor of Monitron #1" ;  
  rdfs:comment "an Avago Technologies ADJD-S311-CR999 miniature surface mount RGB digital  
color sensor which has been calibrated to output red, green, and blue color values from 0 to 1" .  
  
:bosch-bmp085_1_thermometer a m:Thermometer ;  
  rdfs:label "thermometer #1 of Monitron #1" ;  
  rdfs:comment "a Bosch BMP085 digital (barometric) pressure sensor which has been calibrated  
to output temperature values in degrees Celsius" .  
  
:bosch-bmp085_1_barometer a m:Barometer ;  
  rdfs:label "barometer of Monitron #1" ;  
  rdfs:comment "a Bosch BMP085 digital (barometric) pressure sensor which has been calibrated  
to output pressure values in pascals" .
```

5. solis: pievienojiet loģiku

- Kas jādara bāzes datoram, ja tas saņem nepilnu mērījumu?
- Katrs mērījums ir jāapstrādā savādāk
- Var izmanto Java klašu hierarhiju



6. solis: sāciet sūtīt plūsmu

@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

@prefix monitron: <http://fortytwo.net/2012/08/monitron#> .

@prefix om: <http://schemas.opengis.net/om/2.0/> .

@prefix time: <http://www.w3.org/2006/time#> .

@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

@prefix : <http://fortytwo.net/2012/08/universe#> .

:event-UXhu5B5

om:Procedure :bosch-bmp085_1_thermometer ;

om:featureOfInterest :room_1 ;

om:observationLocation :room_1 ;

om:observedProperty monitron:airTemperature ;

om:result [

om:uom monitron:degreesCelsius ;

om:value 23.3 ;

a om:ResultData] ;

om:samplingTime [

a time:Interval ;

time:hasBeginning [

a time:Instant ;

time:inXSDDateTime "2012-11-28T07:50:06.132-05:00"^^xsd:dateTime] ;

time:hasEnd [

a time:Instant ;

time:inXSDDateTime "2012-11-28T07:50:06.145-05:00"^^xsd:dateTime]] ;

a monitron:AirTemperatureObservation .

7. solis: nosūtiet ciklisko pieprasījumu

```
PREFIX :      <http://fortytwo.net/2012/08/universe#>
PREFIX m:     <http://fortytwo.net/2012/08/monitron#>
PREFIX om:    <http://schemas.opengis.net/om/2.0/>
PREFIX rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
```

```
SELECT ?result ?value WHERE {

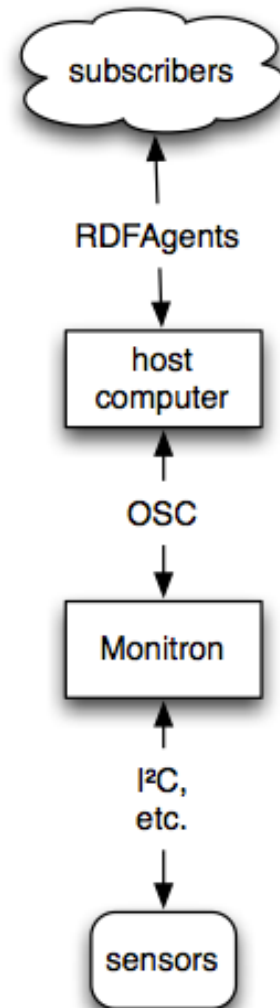
    ?obs rdf:type m:VibrationLevelObservation .

    ?obs om:result ?result .
    ?result om:value ?value .

    ?obs om:observationLocation ?loc .
    ?loc m:containedIn :winslowBuilding .

}
```

8. solis: saņemiet rezultātus!



Semantic Synchrony kods

- Pieejams šeit:
 - <https://github.com/joshsh/smsn>

Kopsavilkums

- Semantiskiem datiem, kas iegūti no sensoru tīkliem, ir nākotne

12. eseja (pēdējā)

Izdomājiet sensoru datu vākšanas piemēru, kur dati ir paredzēti lietošanai citiem, piemēram, publicēšanai tīmeklī utt.

Uzdevums – īsi aprakstiet savu piemēru (t.sk. vācamos datus). Kādu informāciju Jūs pievienotu sensoru rādījumiem, lai nodrošinātu, ka šie dati bez Jūsu iejaukšanās ir viegli saprotami un citiem interpretējami?

Termiņš: 14.12.2016. 10:00