SEMANTICS-DRIVEN MIDDLEWARE LAYER FOR BUILDING OPERATION ANALYSIS IN LARGE-SCALE ENVIRONMENTS

Adam Kučera, Tomáš Pitner

LAB OF SOFTWARE ARCHITECTURES
AND INFORMATION SYSTEMS

FACULTY OF INFORMATICS
MASARYK UNIVERSITY

lasaris

Outline

- **Introduction**
  - Facility management
  - Information systems in facility management
- **Motivation** and **Goals**
  - Use case: University campus of Masaryk University
- **Problem:** Automation data analysis
- **Method:** Automation data semantics and querying
- Results, Summary, Conclusions

Facility Management

- According to IFMA (International Facility management association): *„a profession that encompasses multiple disciplines to ensure functionality of the built environment by integrating people, place, process and technology*"
- FM ensures tasks, which are not part of organization‘s „core business“

IS in Facility Management

BIM

- BIM = Building Information Model
- Built environment, locations and devices
- Generally static data

CAFM

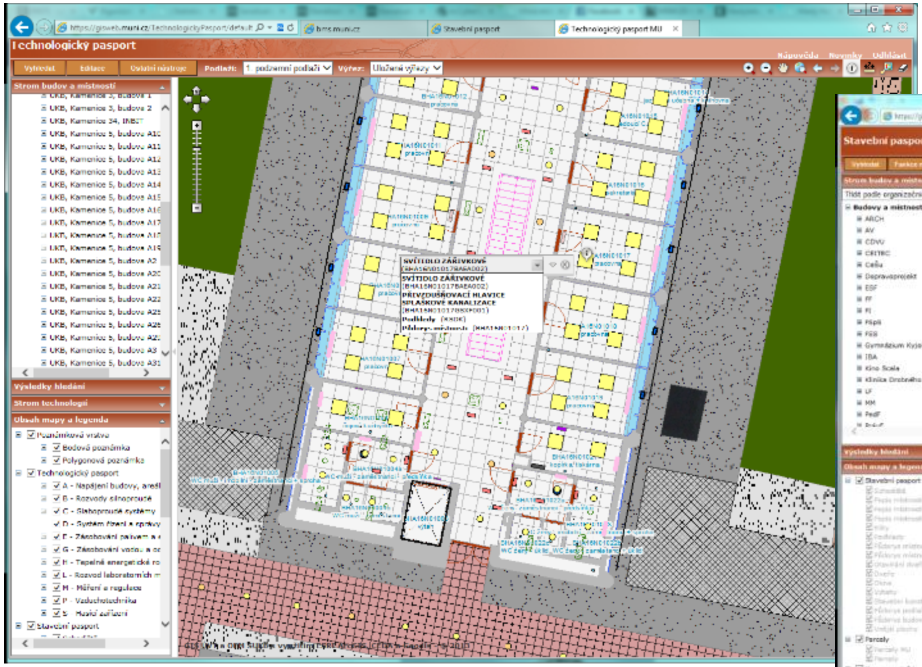
- CAFM = Computer Aided Facility Management Software
- Space management, Furniture, Maintenance, Energy management
- Dynamic data (e.g. financial, HR), uses BIM data
- Analysis & Reporting

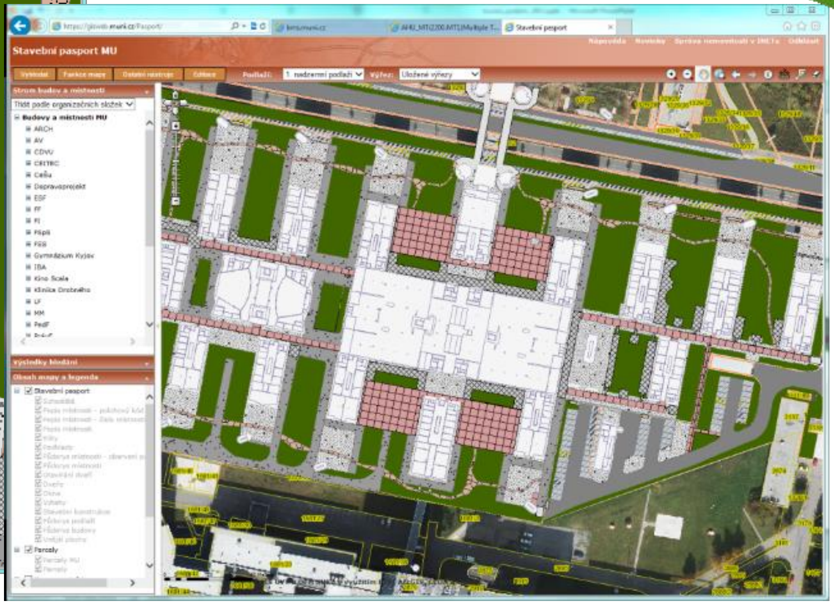
BMS

- BMS = Building Management System
- Remote monitoring and control of building automation systems
- Recent (present) and historical data from sensors and other devices

BIM – Building Information Model

- Digital representation of a building





Source: Authors

CAFM – Computer-Aided Facility Mgmt

- CAFM software supports:
  - Space management
  - Maintenance
  - Energy management
- Provides advanced analytical tools

Source: Archibus, Inc.

Smart buildings

- Devices in buildings connected to a network
  - Heaters
  - Air conditioning units (HVAC)
  - Lighting
  - Energy meters
  - ...
- Monitored and controlled remotely

Smart buildings – Approaches

Modern (Households & SOHO/IoT)

- „We have cheap computers, can we use them to control appliances?“
- Origins in ICT

Traditional (Large sites)

- „We have lot of devices in a building, can we facilitate the management?“
- Origins in civil engineering & electronics engineering

Smart buildings – Approaches

Households & SOHO/ IoT

- Examples:
  - Arduino
  - .NET Gadgeteer
  - Energomonitor
  - Nest/Google thermostat
- Relatively cheap

Large sites

- Technologies
  - Building Automation Systems
  - Building Management Systems
- Expensive
- Long device lifetime
- Compliance to standards

lasaris

Smart buildings – Approaches

Households & SOHO/IoT

- Devices using:
  - Operating system
  - Wi-Fi
  - HTTP
  - Web services
  - Cloud
  - M2M, Internet of Things
- Controlled by
  - Web interface
  - Smart phones

Large sites

- Devices using
  - Microcontrollers
  - Serial bus (RS232,RS485), Ethernet, TCP/IP
  - Specialized automation protocols
- Controlled by
  - Dedicated desktop applications
  - Web interface

lasaris

Smart buildings – Approaches

Households & SOHO/IoT

Source: Google, Inc.

- ARM Cortex A8
- 40 MB flash

Large sites

Source: Delta Controls, Inc.

- CPU 25 MHz
- 128 kB RAM
- 1 MB flash

lasaris

Smart buildings – BAS & BMS

- **BAS** = Building Automation System
- **BMS** = Building Management System
- Used mostly at large sites
- Ensures automated operation of building technologies:
  - *HVAC*
  - *Lighting*
  - *Safety & Security systems (Fire alarm, Access control)*
  - *Elevators*
  - *Energy monitoring*

lasaris

Smart buildings – BAS & BMS

- Remote monitoring and control
- Integration of different systems
- User interface
- Alarming
- Archiving
- Regulation algorithms
- Scheduling
- Cooperation

lasaris

BMS – PLCs

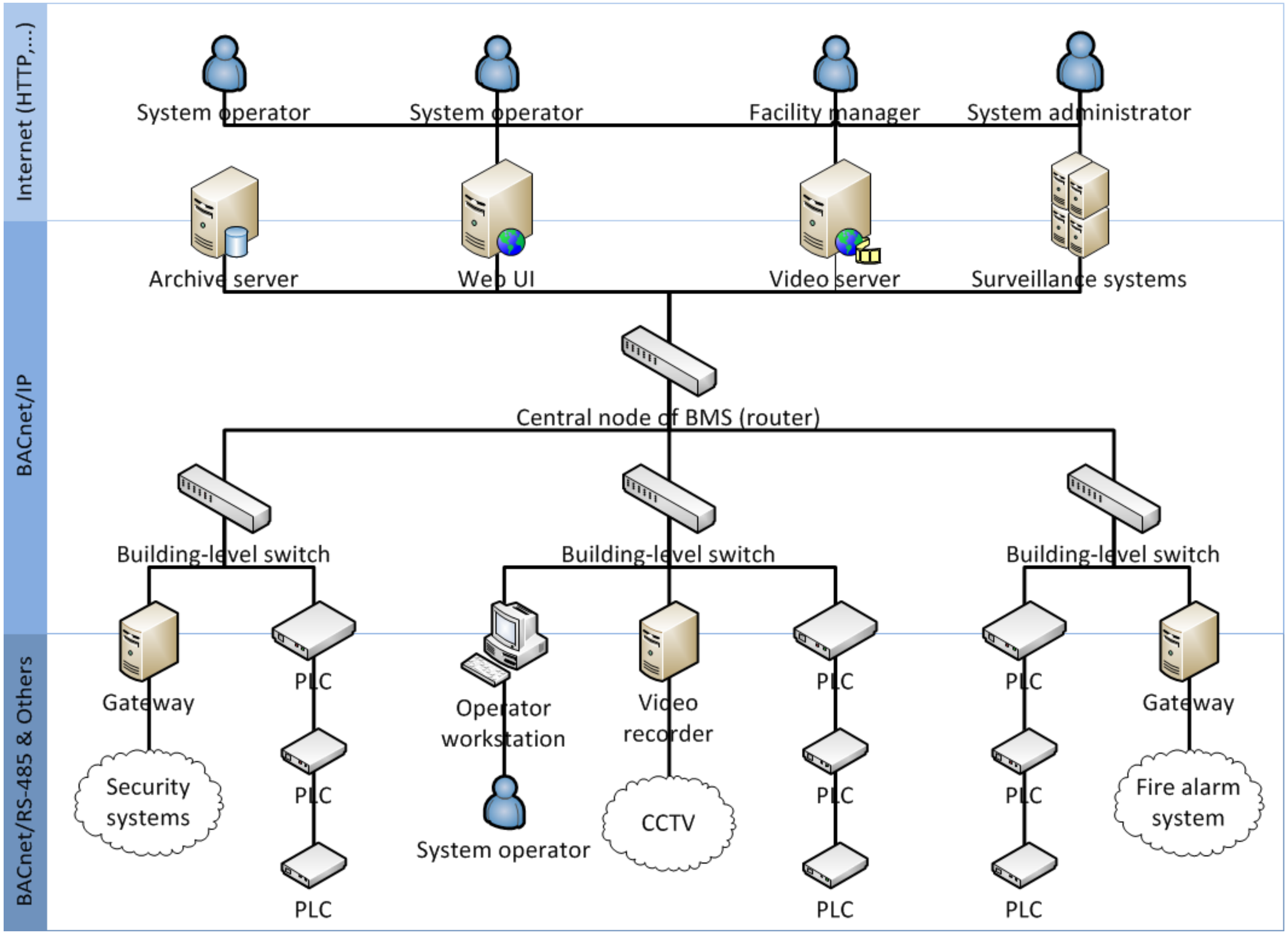
- **PLC** = Programmable logical controller
- Specialized computer for automation
- Provides various types of input and outputs
  - **Analog inputs** –e.g. temperature, humidity, pressure sensors
  - **Analog output** – e.g. valve opening
  - **Digital (discrete) inputs** – e.g. motion sensor
  - **Digital (discrete) outputs** – e.g. fan speed, relay control
- Programmable by specialized tools & languages

BMS – PLCs

Source: OFM SUKB MU

Source: siemens.com

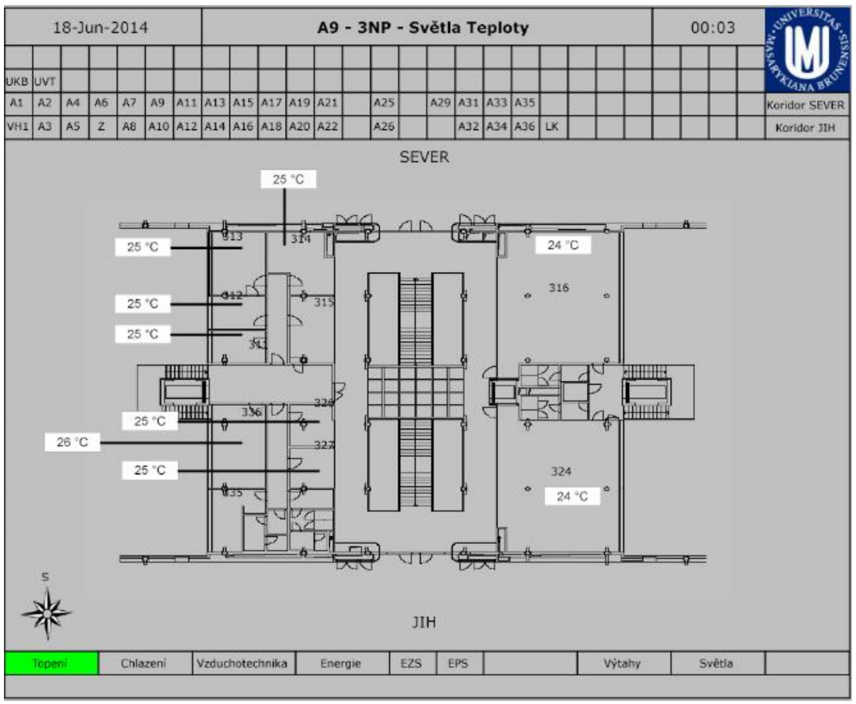
BMS – Structure

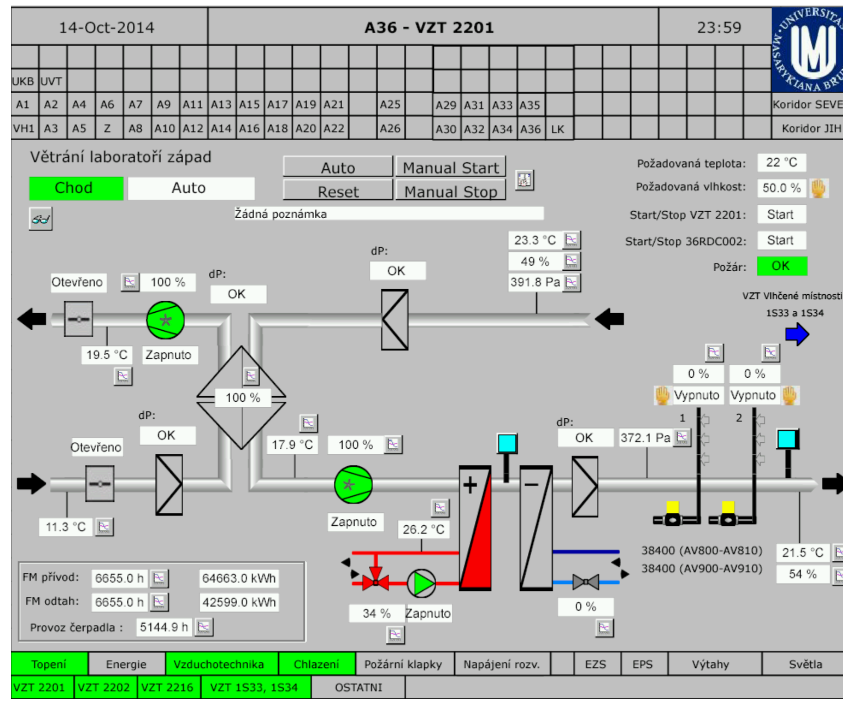


Source: Authors

lasaris

BMS – UI

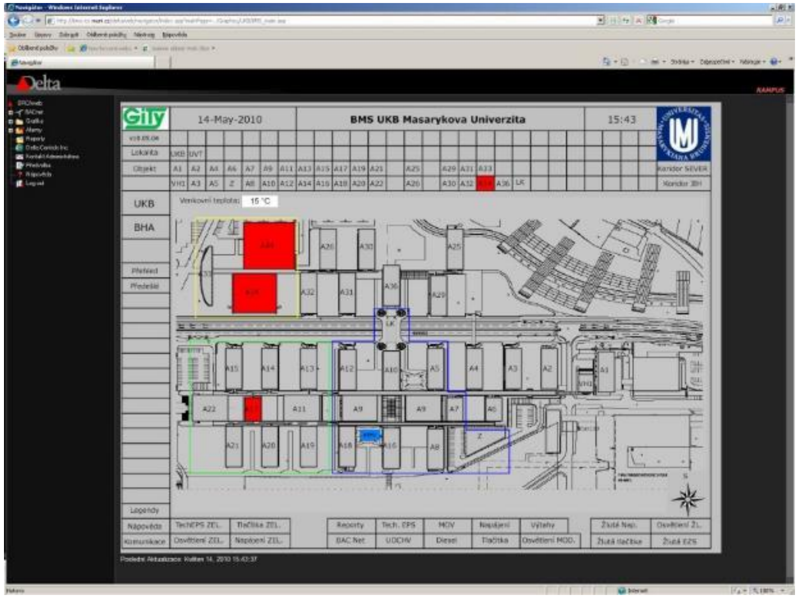




Source: OFM SUKB MU

lasaris

BMS – UI



Source: OFM SUKB MU

Motivation – Use case

- **Goal:** Examining building operation **performance** and **efficiency** using BMS data
- **Use case:** BMS of Masaryk University (40 buildings, 150 000 data points)


Source: muni.cz

lasaris

Motivation – Analytical capabilities

BMS	CAFM
Sensor data	Financial data
High detail	Low detail
Recent data	Delayed data
Simple applications	Complex applications

How much does the electricity consumption differ across the campus?

How much energy is consumed by air conditioning?

What are the average room temperatures?

BMS vs. Big Data

- Volume **does not** apply
  - 150 000 data points, Up to 10GB of useful data/year
- Velocity **does not** apply
  - Polling frequency: minutes
  - Change of Value (e. g. 1°C)
- Variety **does** apply (partially)
  - Structured data
  - Undifferentiated data types (Temperature, Humidity, Setpoint,...)
- Variability & Veracity **do not** apply
  - Data are consistent, credible and of high quality

lasaris

Problem – Complexity

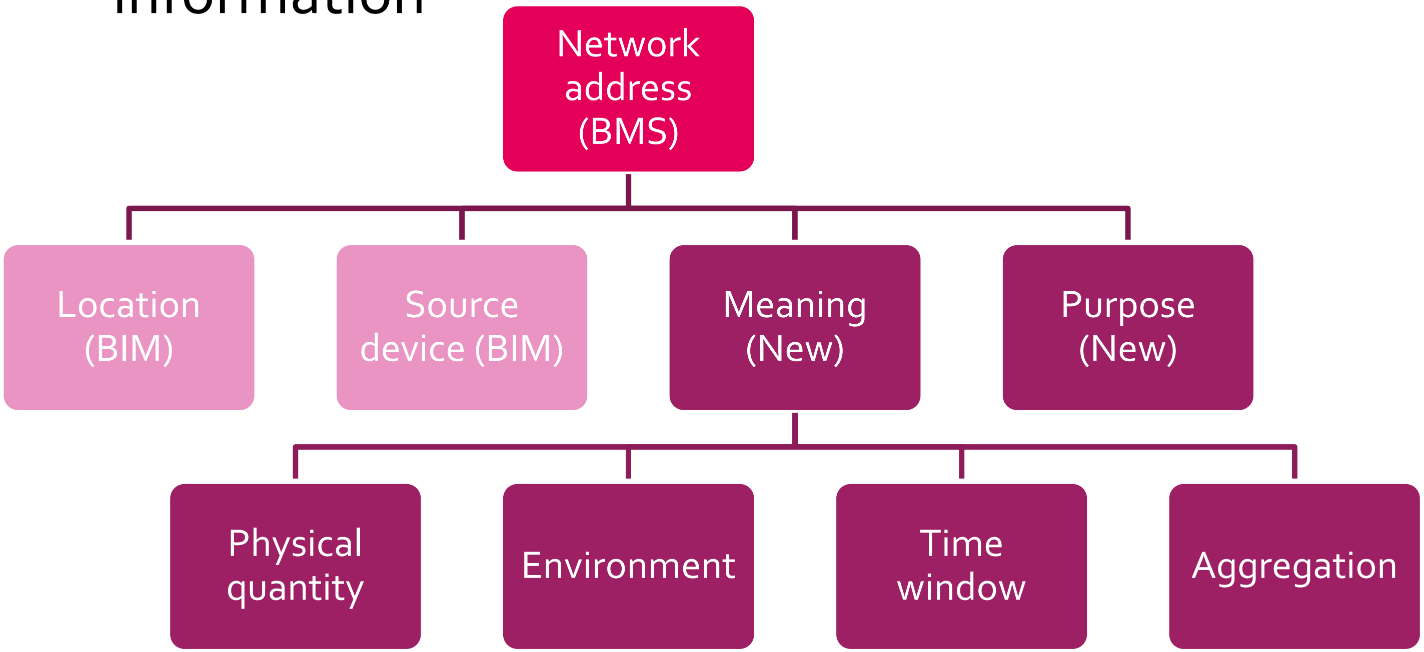
- Application development tasks:
  - Data access (automation protocols, OLTP)
  - Data selection, grouping & aggregation
  - Analytical methods
  - User Interface

Problem – Unsuitable semantics

- Data points **identified by** network **address** in BMS
  - BACnet protocol: 25104.AI101
- Data point properties carry **limited semantics**:
  - Object type (Analog/Binary/..., Input/Output/Variable/...)
  - Engineering units
- **Missing relation** to the physical world:
  - Location
  - Source device
  - Measuring environment (air, water,...)
  - ...

Aims & Methods – New semantics

- New approach to analysis of BMS data
  - Network **addresses are not used** as identifiers
  - Universal model relates **BMS** and **BIM** and also adds new information



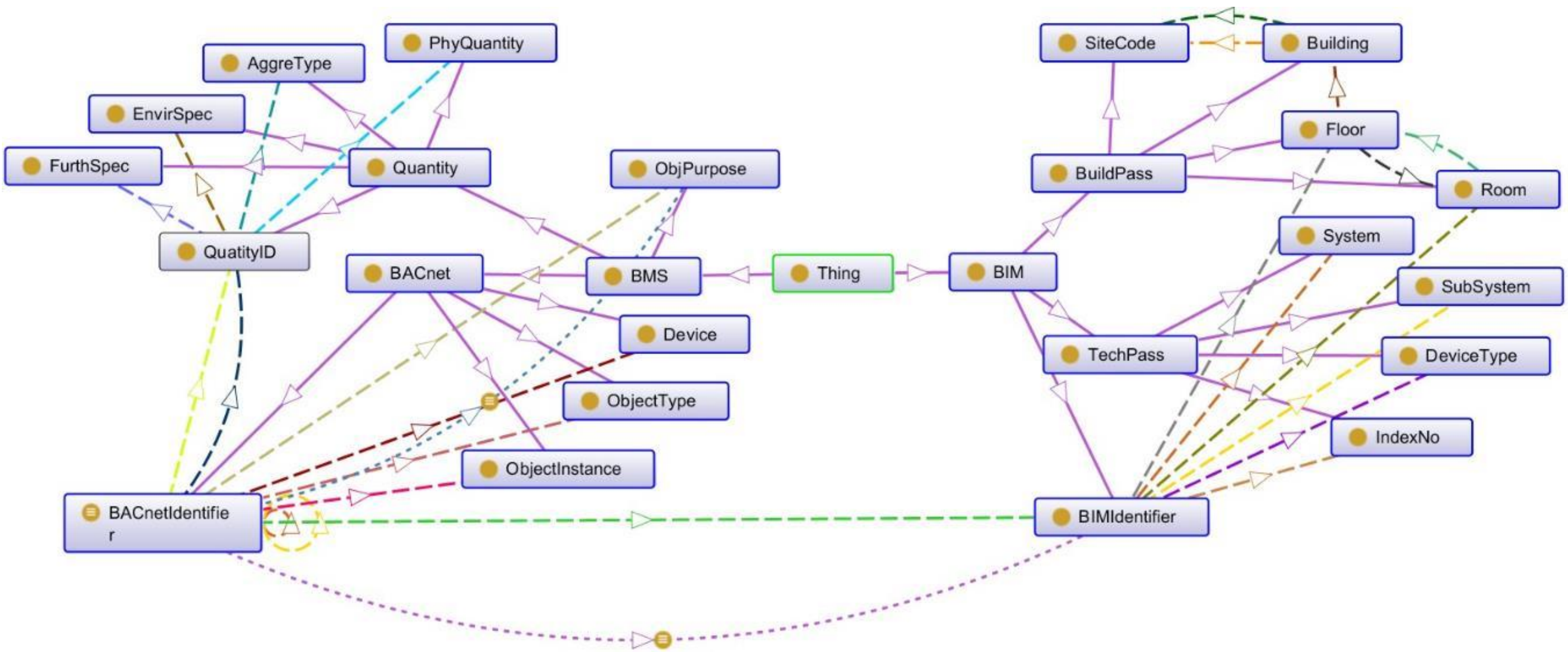
Aims & Methods – Ontology

- New semantics of BMS data can be described by **Ontology language**
- **OWL** –Web Ontology Language (W3C)
  - Designed for **Semantic web & Linked Data**
  - Based on **RDF** (Resource Definition Framework)
  - **„Subject-Predicate-Object“**

Aims & Methods – Existing ontologies

- **Upper** ontologies – describe general concepts accross domains (not used in our use case)
- Semantic Sensor Network ontology – unsuitable
  - Uses upper ontology as a base
  - Complicated querying
  - Focuses on different concepts
    - SSN: Relation between observation and obtained value
    - BMS: Relation between source device and value, description of measured value

Aims & Methods – Ontology



Source: Muhammad Asfand-e-yar, FI MU

Aims & Methods – Ontology querying

- Ontology repositories can be queried using specialized query languages (**SPARQL**)

Query

```
Select ?De ?Ot ?Oi Where { ?idnet1 Abstract:isCharacterizedBy ?Ot. ?idnet1 Abstract:isConnectedWith ?Oi. ?idnet1 Abstract:isCompriseOf ?De
{Select ?idnet1 Where {?idnet1 Abstract:hasMeaningOf ?idQua1.
{ Select ?idQua1
Where {?idQua1 Abstract:hasPhyQuantity Abstract:PhT. ?idQua2 Abstract:isCheckedWith Abstract:AtK.
 ?idQua3 Abstract:isMeasuredIn Abstract:EnA. ?idQua4 Abstract:hasSomeMore Abstract:Fu14.
 FILTER (?idQua1 = ?idQua2 && ?idQua2 = ?idQua3  && ?idQua3 = ?idQua4)}}.
?idnet2 Abstract:isGettingDataFrom ?idBim1.
{ Select  ?idBim1 Where {?idBim1 Abstract:hasSpecific Abstract:Ro001a. ?idBim2 Abstract:hasParticular Abstract:FIN01.  FILTER (?idBim1 = ?idBim2)}}
?idnet3 Abstract:isRepresentedAs Abstract:OpPresentInputValue.
FILTER(?idnet1 = ?idnet2 && ?idnet1 = ?idnet3)}}.}
```

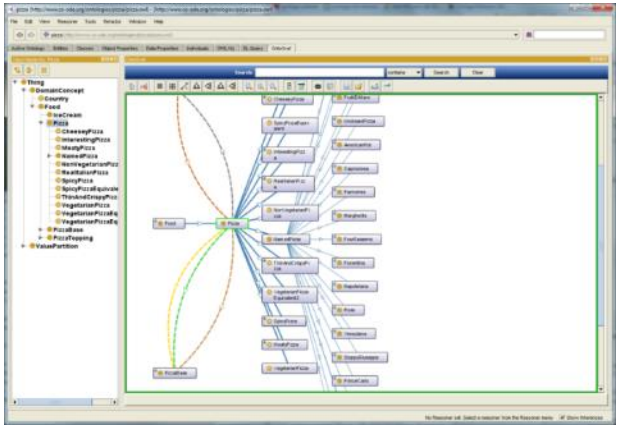
SPARQL

Source: Muhammad Asfand-e-yar, FI MU

lasaris

Aims & Methods – Ontology tools

- **Protégé** – Open source ontology editor
- **Apache Jena** - Open Source ontology framework
  - OWL/RDF Java API
  - **SPARQL** engine
  - **TDB** - Native (noSQL) persistent **triplestore**
  - **Fuseki** – standalone RESTful web server



Source: http://protegewiki.stanford.edu/

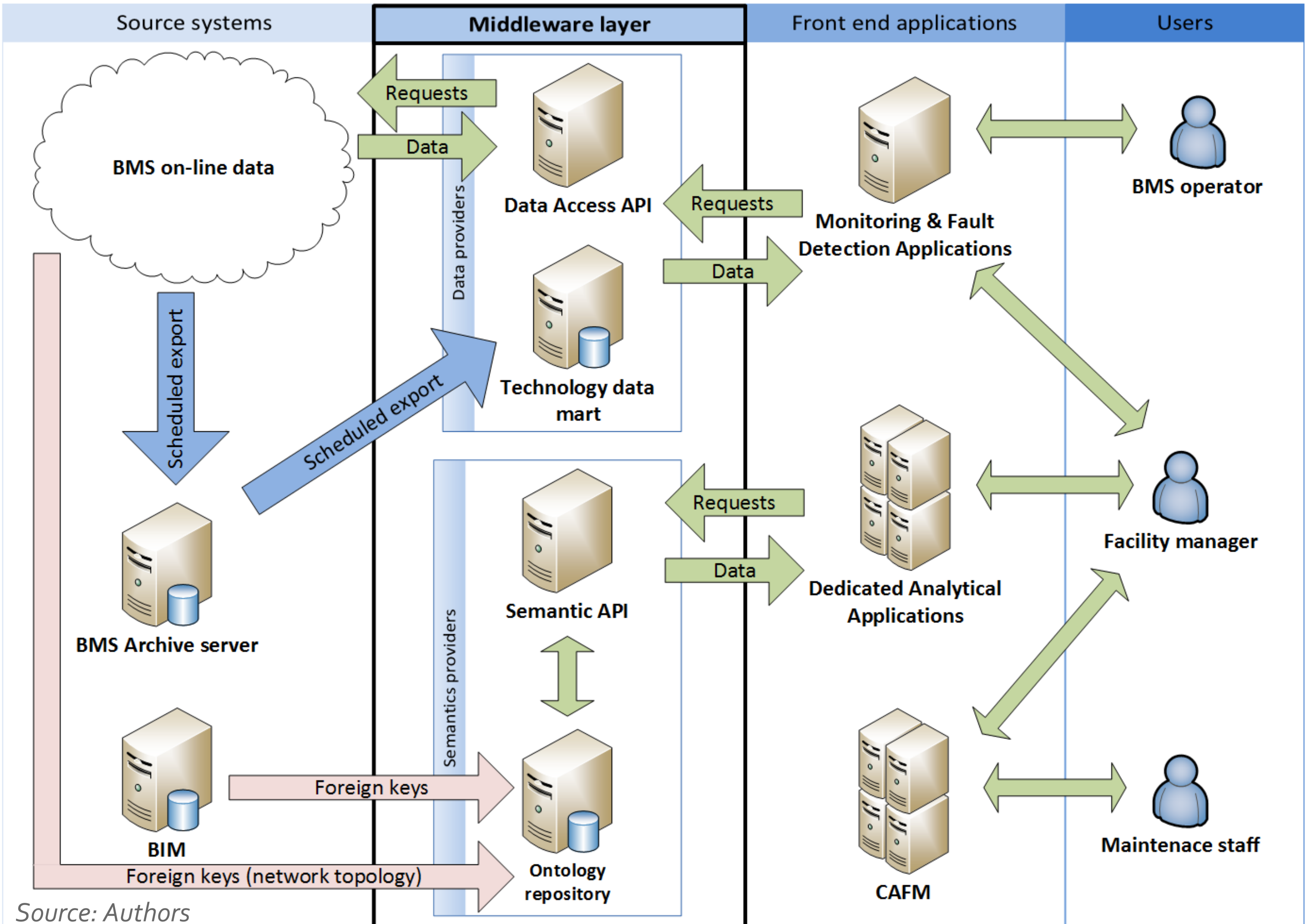
lasaris

Aims & Methods – APIs

- Simplification of application **development** & **integration**
- **Data access** APIs
- **Semantic** API
  - **Encapsulating** OWL & SPARQL
  - Domain-specific **operators** – aggregation, grouping & filtering according to:
    - Location
    - Source device
    - Meaning
    - ...
  - Ready-to-use **functions** for frequent queries

lasaris

Aims & Methods – Middleware layer



Source: Authors

Query examples

1. Semantic query

Location: *Campus Bohunice; Building A11*
Grouping: *Per floor*
Measured value: *Room temperature*
Source device: *Temperature sensor*
Data type: *Historical data*
Desired output: *Network address*

2. Semantic result

N01: {11400.TL5, 11500.TL5, 11600.TL1}
N02: {12100.TL5, 12300.TL3, 12400.TL5}
N03: {12500.TL1, 12600.TL1, 12800.TL1}

3. Data query

Data points: *Semantic result data*
Aggregate: *temporal AVG*
Period: *09/2014 – 1/2015*
Aggregation Window: *1 day*

4. Data result

N01: { {2014-09-01, 23.8}, {2014-09-02, 24.8}, {2014-09-03, 25.1}, {2014-09-04, 24.7}, …
N02: { … }
N03: { … }

Query examples

1. Semantic query
Data type: *Input; Output; User defined value*
Influenced value: *Room temperature*
Influenced location: *Room 231 at building UCB-A11*
Desired output: *{Source device (with Location); Network address; Data type; Meaning (quantity) }*

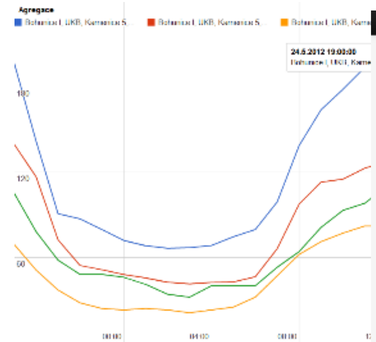
2. Semantic result
{ Pump in UCB-A11-1S05, 10200.AO1, Output, Pump mode (on/off) }
{ Temperature sensor in UCB-A11-1S05, 10200.AI5, Input, Water temperature }
{ Application controller in UCB-A11-1S07, 10000.AV4, User defined value, Setpoint temperature }

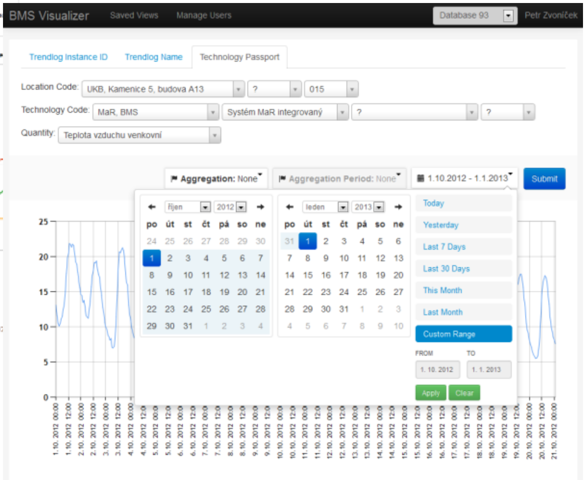
3. Data query
Data points: *Semantic result data*
Aggregate: - *(present value)*

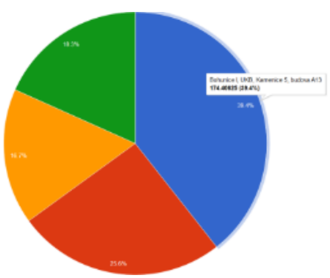
4. Data result
{ Pump in UCB-A11-1S05; ON }
{ TS in UCB-A11-1S05, 76,5 °C }
{ AC in UCB-A11-1S07, 22 °C }

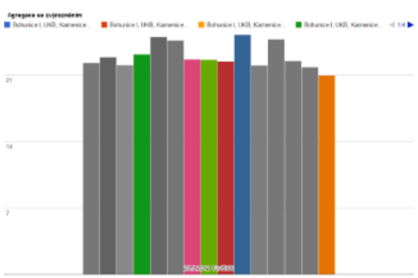
Results

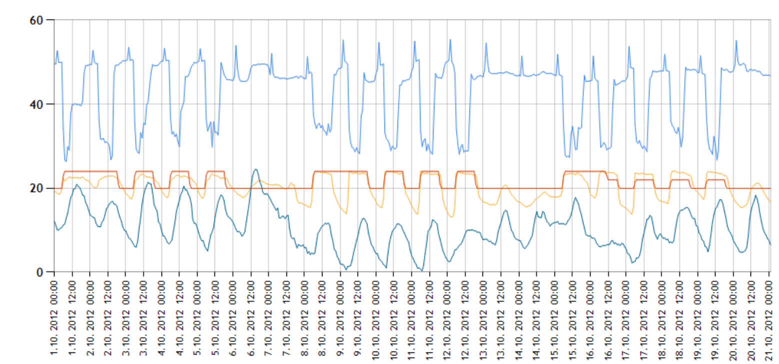
- Architecture design
- End-user applications
- Data access API
- Semantic model











Source: Authors, Petr Zvoníček, FI MU

Summary & Conclusion

- **Area:** Building operation analysis using data from automation systems
- **Aims:**
  - Provide new semantics to BMS data
  - Simplify development of analytical tools
- **Method:** Middleware layer
  - Semantic information – Integrating BMS and BIM
  - Data access
- **Evaluation:** Implementation of benchmarks defined in *EN 15 221: Facility Management*

lasaris