Institut
Mines-Télécom

Hes·so

Haute Ecole Spécialisée
de Suisse occidentale

Fachhochschule Westschweiz

University of Applied Sciences and Arts
Western Switzerland

A Web-of-Things Gateway for KNX Networks

Gérôme Bovet[1, 2] – Jean Hennebert[2, 3]

[1]Telecom ParisTech France

[2]University of Applied Sciences Western Switzerland

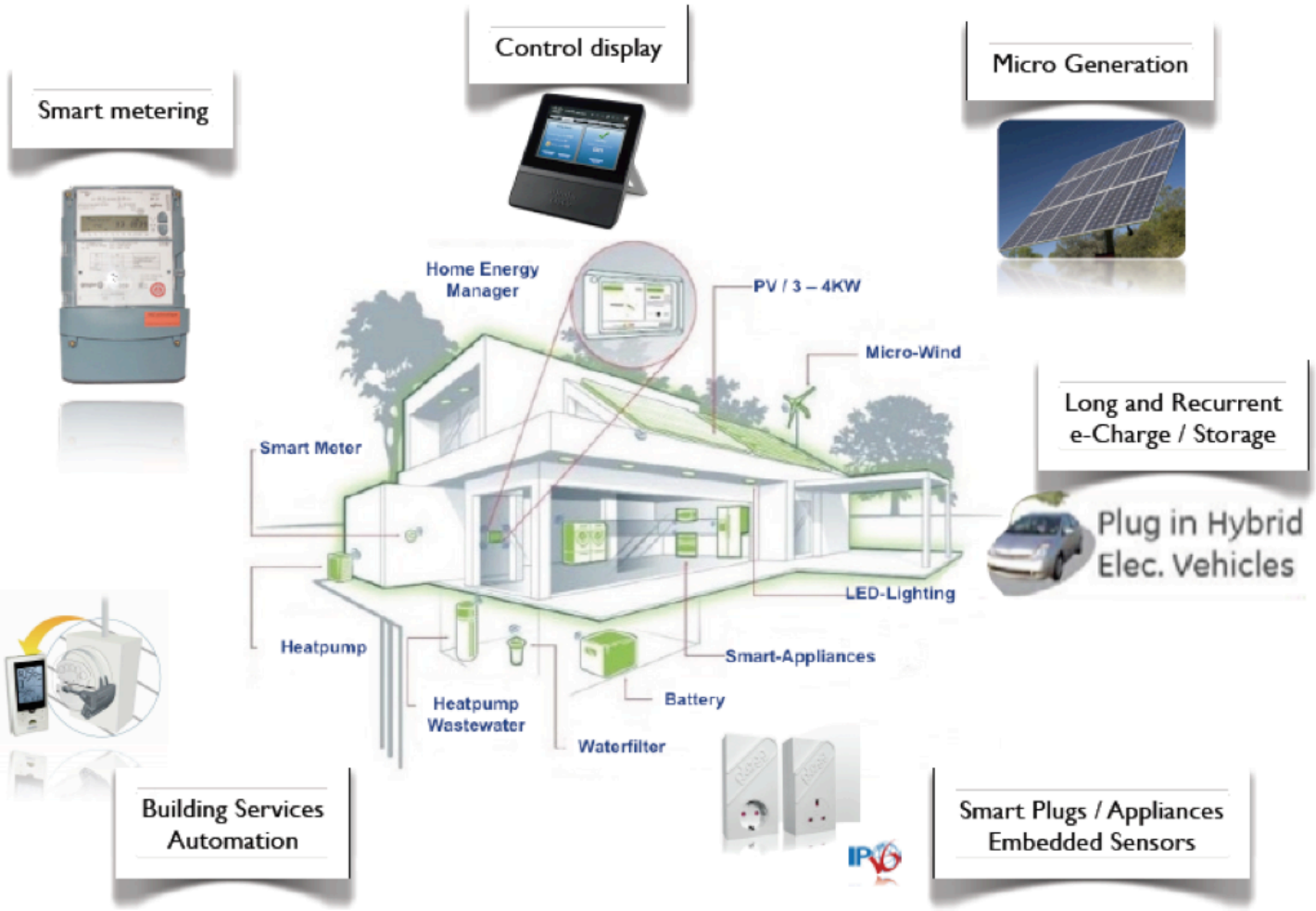
[3]University of Fribourg Switzerland

gerome.bovet@hefr.ch

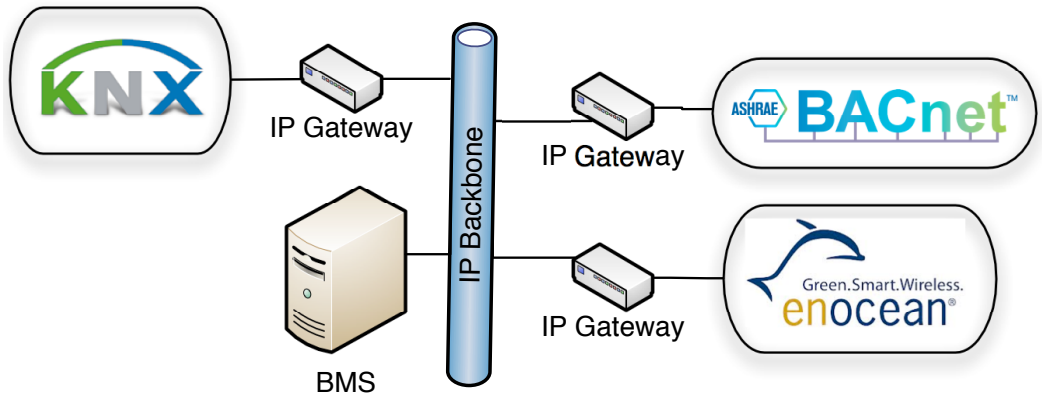
Outline

- **Introduction**
- **KNX network**
- **Web-of-Things**
- **Gateway**
  - Our approach
  - Mapping KNX to REST APIs
  - The API
- **Implementation**
- **Evaluation**
- **Conclusion and future works**

Introduction – General context



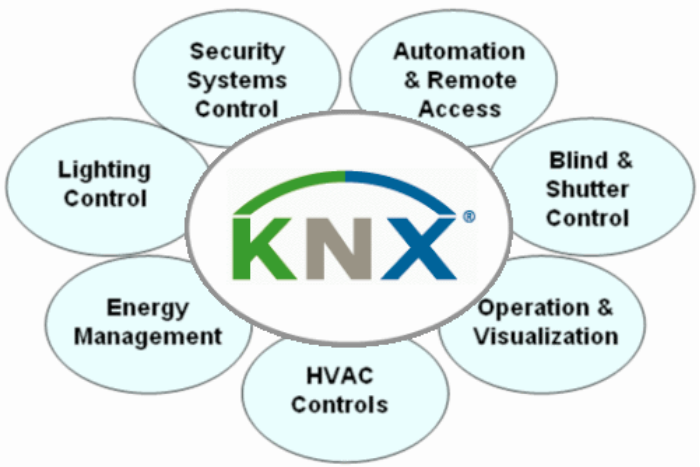
Introduction - Problematic



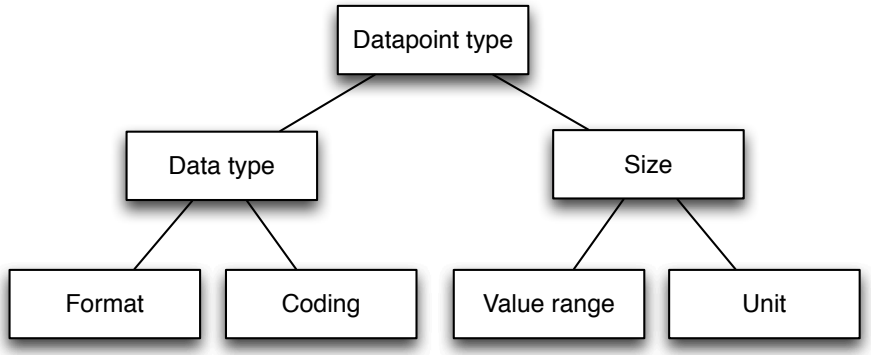
- **Simple encapsulation of telegrams in IP packets**

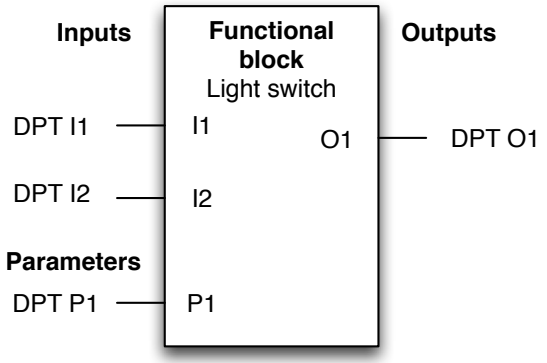
- ***No standard application protocol !***
  - *Implementation of each network in the BMS*
  - *Huge integration costs*
  - *High maintenance effort*

The KNX network



- **Various types of physical connections (TP, Power line, Ethernet, RF)**
- **KNX interworking**
  - Interoperability between manufacturers
  - Basing on functional blocks
  - Standardization of formats

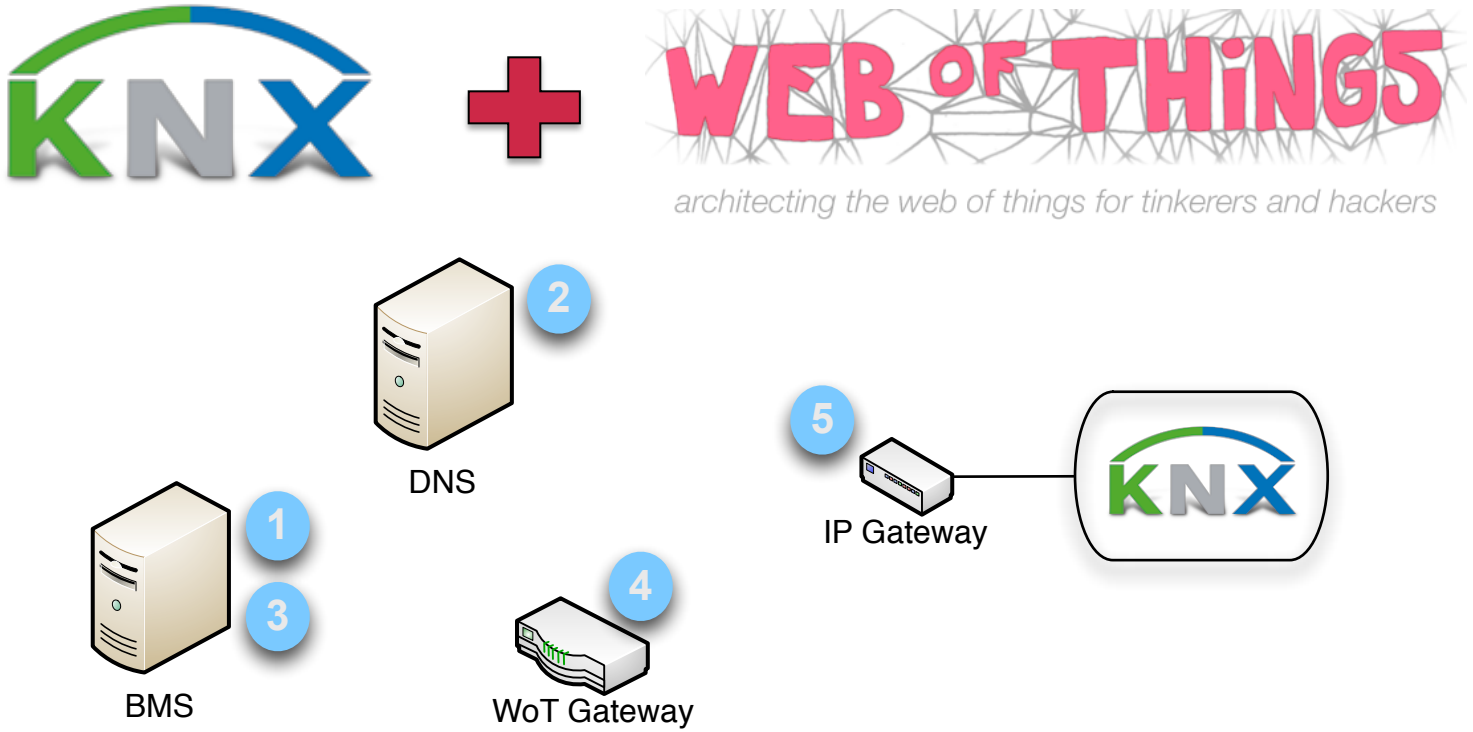




The Web-of-Things

- **WoT is an application layer for accessing resources**
  - Represents resources with URLs (e.g. http://air.kitchen.home/temperature/celsius)
  - Uses the REST paradigm (HTTP verbs)
    - GET for reading
    - POST for updating
- **Every device is embedding a Web server exposing a REST API**
- **Notification of events with callbacks**
  - HTTP
  - WebSockets (TCP connection)

The gateway approach - Principle



1. Client calls http://motion.kitchen.home
2. DNS server answers with IP of WoT gateway
3. Client performs the HTTP request
4. WoT gateway maps URL to KNX group
5. IP gateway forwards telegram to KNX network

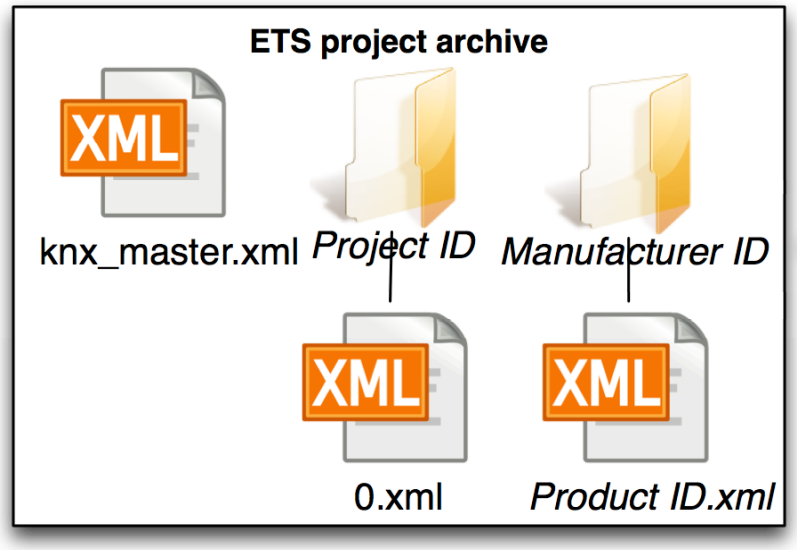
The gateway approach - Benefits

- Hiding the KNX complexity
- Simple to use
- Open & based on Web standards
- Solving the heterogeneity problem

Mapping KNX to REST APIs [1]

- **How to map KNX datapoints to RESTful services ?**
  - Active discovery of devices and datapoints too complex ☹
  - Most installations are configured with ETS ☺

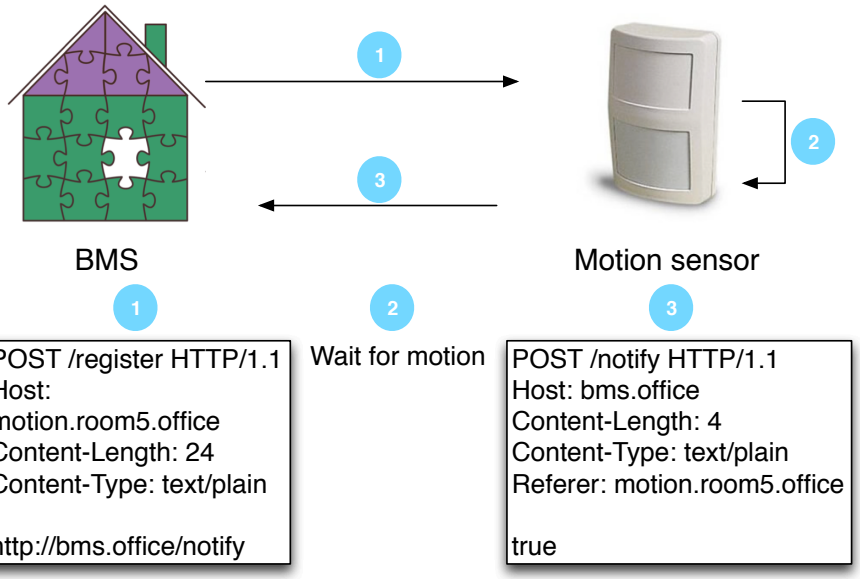
- **Using the knowledge stored in ETS**
  - Compressed file
  - Not protected
  - XML representation of
    - Building structure
    - Devices
    - Datapoints



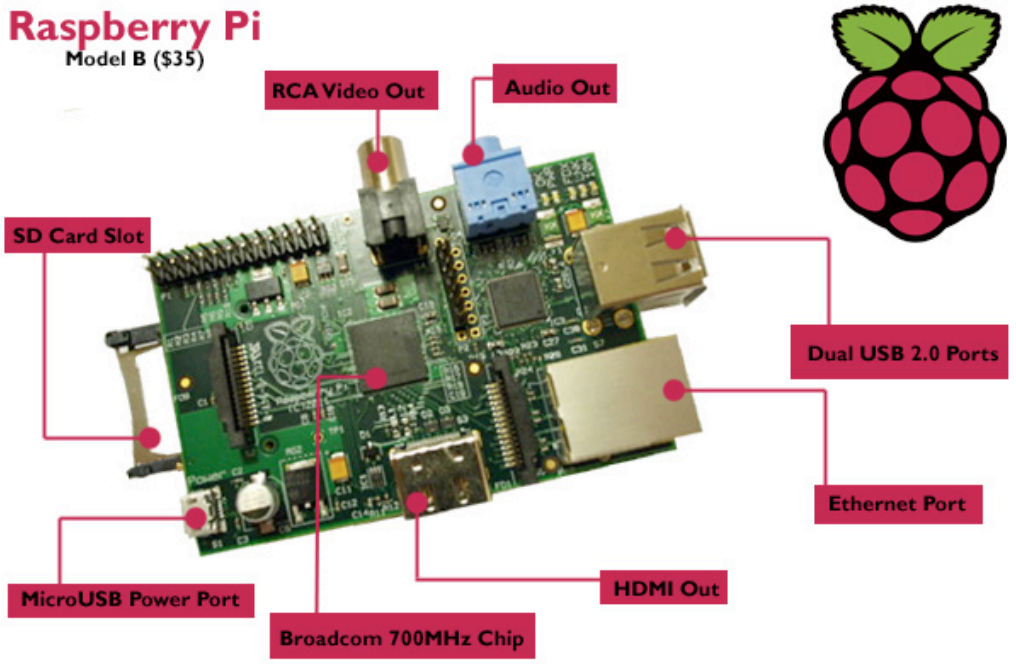
Mapping KNX to REST APIs [2]

- **Performing a XSL transformation**
  - ```
    <datapoint stateBased="true" name="Heating" desc="Status" mainNumber="1"
    priority="Low" actionName="DPT_Switch" actionDesc="on/off"
    dptDesc="1-bit" dptBitsSize="1" location="Office005.ground.LESO">
       <knxAddress type="group">6195</knxAddress>
    </datapoint>
    ```
- **Mapping rule**

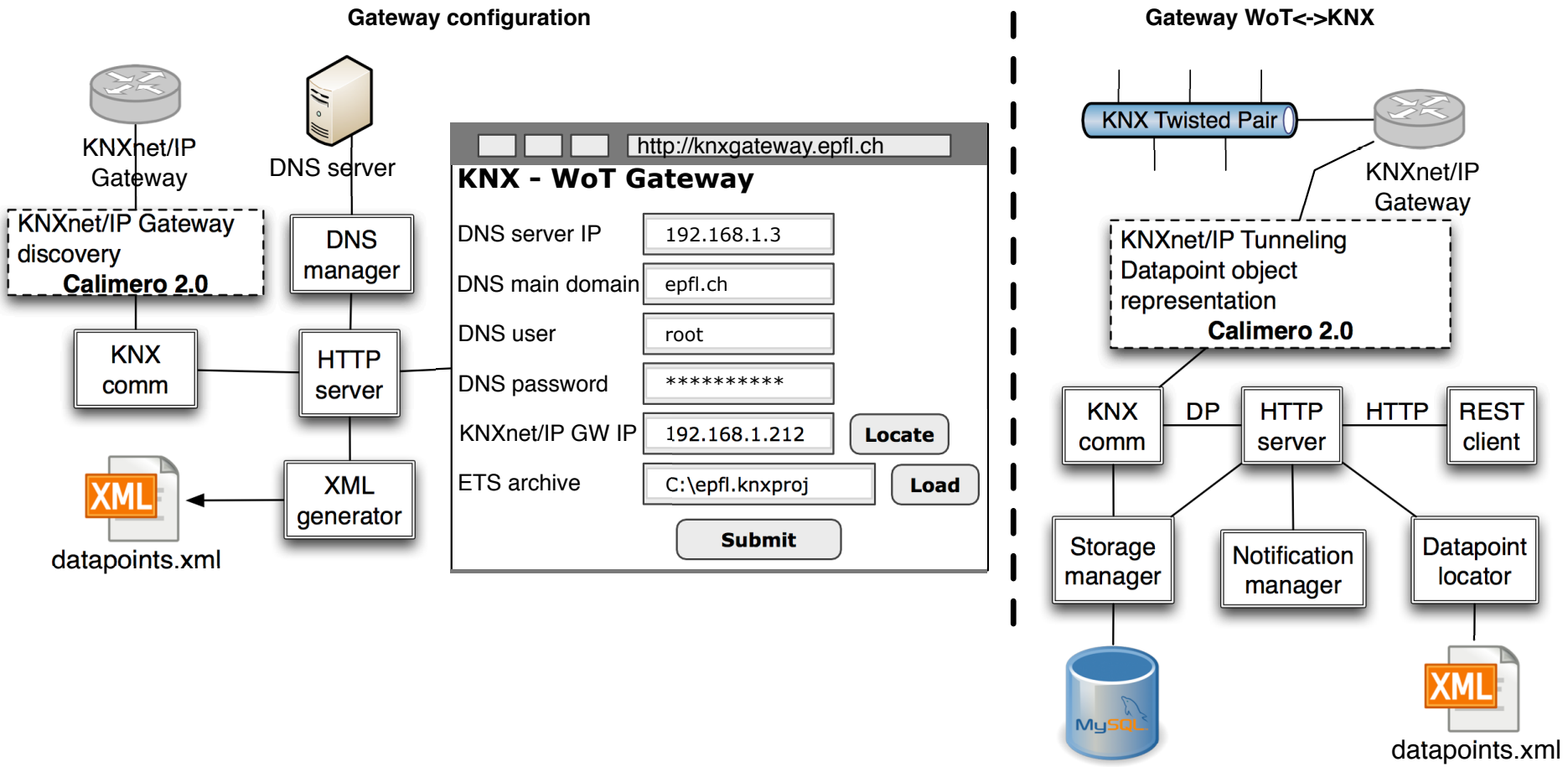
  http://<GROUP NAME>.<LOCATION>.<ORGANIZATION DOMAIN>/<DATAPOINT>

  http://heating.office005.ground.leso.epfl.ch/dpt_switch
- **HTTP requests**
  - GET for reading
  - POST for changing
- ***Devices seem like IP ones***
- ***URL reflecting the physical structure of the building***
    ```

Gateway API - Discovery

Building discovery

- http://<LOCATION>.<ORGANIZATION DOMAIN>
- http://ground.leso.epfl.ch
-
```
{”name”:”office005”,
“isGroup”:false,
“url”:http://office005.ground.leso.epfl.ch}
```


Datapoint discovery

- http://<LOCATION>.<ORGANIZATION DOMAIN>/*
- http://office005.ground.leso.epfl.ch/*
-
```
{"datapoint_info":"1-bit”,
"datapoint_type":"DPT_Switch”,
"description":"on/off”,
"bits_size":1,
"datapoint_number":"1.001”,
"url":"http://heating.office005.ground.leso.epfl.ch/dpt_switch"}
```

Gateway API - Events

Events (reactive BMS)

- Registration (HTTP POST)
  - http://<GROUP NAME>.<LOCATION>.<ORGANIZATION DOMAIN>/<DATAPOINT>**/register**
- Unregistration (HTTP POST)
  - http://<GROUP NAME>.<LOCATION>.<ORGANIZATION DOMAIN>/<DATAPOINT>**/unregister**



Gateway API - Storage

Storage (proactive BMS)

- Announcement (HTTP PUT)
  - http://<GROUP NAME>.<LOCATION>.<ORGANIZATION DOMAIN>/<DATAPOINT>**/storage/add**
  - Payload contains the number of days data should be stored
- Suppression (HTTP DELETE)
  - http://<GROUP NAME>.<LOCATION>.<ORGANIZATION DOMAIN>/<DATAPOINT>**/storage/remove**
  - Gateway shrinks data
- Retrieval (HTTP GET)
  - http://<GROUP NAME>.<LOCATION>.<ORGANIZATION DOMAIN>/<DATAPOINT>**/storage?days=X**
  - http://<GROUP NAME>.<LOCATION>.<ORGANIZATION DOMAIN>/<DATAPOINT>**/storage?from=X&to=Y**
  - ```
    {
       "storage": [
          { "value":"on", "timestamp":"2013-01-10 08:12:34" },
          { "value":"off", "timestamp":"2013-01-10 09:05:57" },
          { "value":"on", "timestamp":"2013-01-10 13:40:03" },
          { "value":"off", "timestamp":"2013-01-10 17:33:11" }
       ]
    }
    ```
- *Reducing the need for storage*
- *Only interesting data are stored*
    ```

Implementation - Technologies



Implementation - Architecture



Evaluation - Performance

- **Realized on the LESO building of the EPFL, CH**
  - 265 devices
  - 795 groups

Measure type	Result
ETS archive processing time	30 [min]
Maximum HTTP requests per second	45
Maximum simultaneous HTTP requests	620
Average event reaction time	33 [ms]

- ***Suitable results for most BMS***
- ***Raspberry Pi is adequate for such applications***

Evaluation – User experiences

- **Limitation of the DNS approach**
  - Security issues – write access to main DNS not allowed
  - Need for a dedicated server
- **No security**
  - Authentication
  - Encryption
- **Positive feedback from developers**
  - Short integration time
  - Open technologies

Conclusion and future works

Conclusion

- WoT approach is suited for BMS
  - Simplicity
  - Open standards
  - Naturally fits with sensor networks
- Fast integration of the gateway in existing installations
- Raspberry Pi is an alternative to classical PCs

Future works

- Adding a security layer
- Gateway for EnOcean (under development)
- Distribution of rules generated by proactive BMS

Questions