SEVENTH FRAMEWORK PROGRAMME

SMART SANTANDER

Information on 1st Open Call for Experiments

Dr. Alex Gluhak, University of Surrey
Dr. Luis Sanchez, University of Cantabria

Open call information day
Brussels, 14th of September 2011

Outline

- First call overview (AG)
  - Call procedure and guidelines for proposals
  - Detailed technical call objectives
  - Expected impact of proposals
- SmartSantander infrastructure (AG, LS)
  - Testbed architecture
  - Deployment description at the 4 testbed sites
- Experimentation on top of SmartSantander (AG,LS)
  - Available experimentation tools
  - Example experiments
- Q&A

Overview of First Call for Experiments

Open call information

- Web: http://www.smartsantander.eu/index.php/open-calls
- Contact e-mail: opencalls@smartsantander.eu

Key documents

- Detailed open call text
  - Provides an overview of the technical call objectives and expected impact
  - Overview of the facility
- Guide for applicants
  - Template adapted from EC how to prepare a proposal
  - Follows FP7 regulations
- Regulations for the use of the facility
  - 7 articles outlining expectations and guidelines for experimentation

Key call facts

Call identifier	SmartSantander-1-Open-Call
Call opening	1st October 2011
Call closing	16. November 2011, 17.00 Brussels time (last received version prior deadline counts)
Experimentation Timeframe	Jan – December 2012 (length may vary based on nature of experiment – typical 9 month)
Max funding per experiment	200k Euros
Maximum funding for call	Up to 600k Euros
Number of expected experiments	3+
Number of partners per experiment	1-2 partners
Proposal language	English

Proposal submission

- E-mail: opencalls@smartsantander.eu
- Only PDF version of proposal acceptable in English language
- Resubmission possible, last version prior deadline considered valid
- Time in our email system counts
- Acknowledgement of receipt of proposal will be email to you soon after closing date

Who is eligible to participate

- Any entity that can participate in FP7
- Funding is cost-shared basis and follows usual FP7 funding rules
- Current partners of SmartSantander are not eligible for funding through open calls

Proposal text structure (1/2)

- Front page, abstract and table of contents
- Cost and funding breakdown

- Section 1: Scientific and/or technical quality, relevant to the topics addressed by the call
  - 1.1 Concept and objective
  - 1.2 S/T methodology and associated work plan
  - 2 mandatory WPs
    - Experimentation framework with 3 deliverables providing experimentation plan (M1), initial results (M6) and final report (M9)
    - Promoting SmartSantander experimentation platform, 1 deliverable showing the dissemination activities

Proposal text structure (2/2)

- Section 2- Implementation
  - 2.2 Participants
  - 2.4 Resources to be committed
- Section 3 - Impact
  - 3.1 Expected impact
  - 3.2 Dissemination and/or exploitation of results, and management of intellectual property
- Section 4 Ethical issues

Evaluation criteria

- Three key criteria
  - 1. Scientific and/or technological excellence (relevant to the topics addressed by the call)
  - 2. Quality and efficiency of the implementation and the management
  - 3. Potential impact through the development, dissemination and use of project results

- Minimum score to pass evaluation threshold
  - At least 3/5 in each and 10/15 in total

Review and selection process

- Independent experts will be appointed after call closure
- Each proposal will be reviewed by at least two independent experts
- Physical consensus meeting will determine final score
- Proposal with the highest score will be selected within the funding limit - but
  - Highest ranked proposal may not be selected, e.g. based on objective grounds
  - No proposal may be funded if consortium finds highest scoring proposals to be of inadequate quality – call may be reopened at a later stage

Further help with submission

- Call help desk
  - Name: José Manuel Hernández-Muñoz
  - Email: jmhm@tid.es
  - Tel: +34 91 483 26 74
- Your NCPs
- IPR help desk

Detailed call objectives

- Main goal:
  - To expand the project's service, protocol and technology offering towards future IoT experimentation as well as the public in the context of the Smart City.

- Objectives should target at least one of 3 areas
  1) Innovative applications and services for smart cities and built environment
  2) Internet of Things communication protocols and technologies
  3) Internet of Things middleware solutions

1) Smart-city application & services

- Innovative IoT based services and application for city and built environment
- Services must demonstrate clear benefit to stakeholders, e.g. city and citizens
- Evaluation of end user acceptance & commercial viability desired
- Current infrastructure targets transportation, energy and environment however extensions of infrastructure to new application domains can be proposed
- Other extensions could be mechanisms to evaluate end user feedback and quality of experience

2) IoT communication technologies

- Evaluation of new approaches and architectural paradigms for an **Inter-net** of Things (IoT)
- Studies that provide a more detailed understanding of the properties and particularities of large scale IoT deployments leading to new design principles
- Mechanisms and techniques that allow the exploitation of opportunistic availability of (mobile) IoT devices for computing and communication tasks
- Key enabling building blocks of an Internet of Things such as resolution infrastructures
- Mechanisms for more efficient and reliable data dissemination in large scale resource constraint environments

3) IoT middleware solutions

- Platforms and mechanisms for large scale distributed processing and querying of real world events and event streams
- Mechanisms and techniques that contribute towards increased data interoperability on an emerging global Web of Things
- Algorithms for real world awareness, contributing to an increased machine understanding of complex processes and system behaviour in a city and built environments
- Visual analytics tools for the efficient analysis of real world events and complex relationship between real world generated data

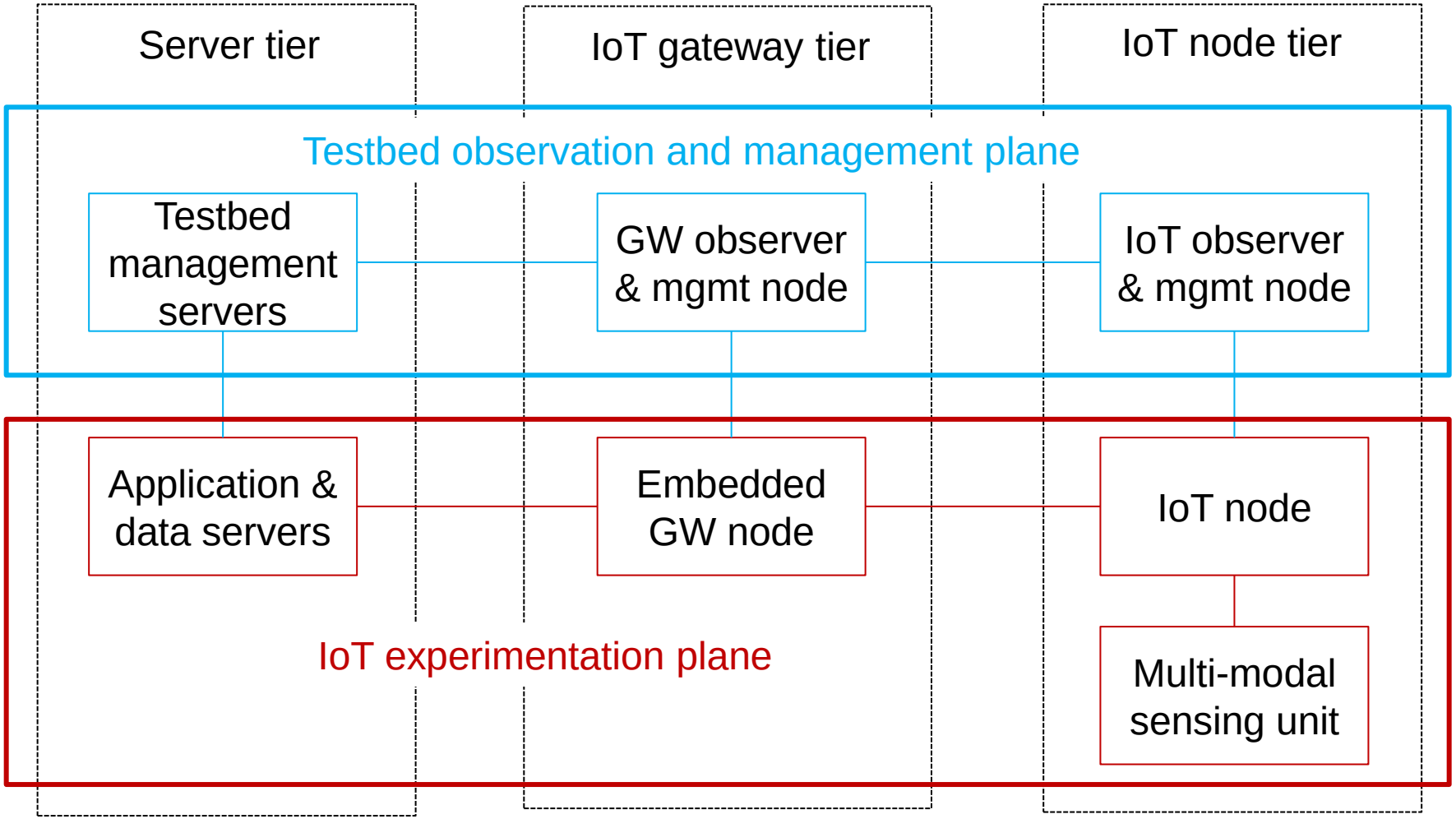
Expected impacts

- Submission to area 1)
  - Must demonstrate a clear benefit and value to the targeted service end users, such as city and citizens or the university and its employees/students
- Submissions to area 2) and 3)
  - Must have the potential to lead to high quality scientific outcomes
  - Supported by evidence (expertise, dissemination strategy)

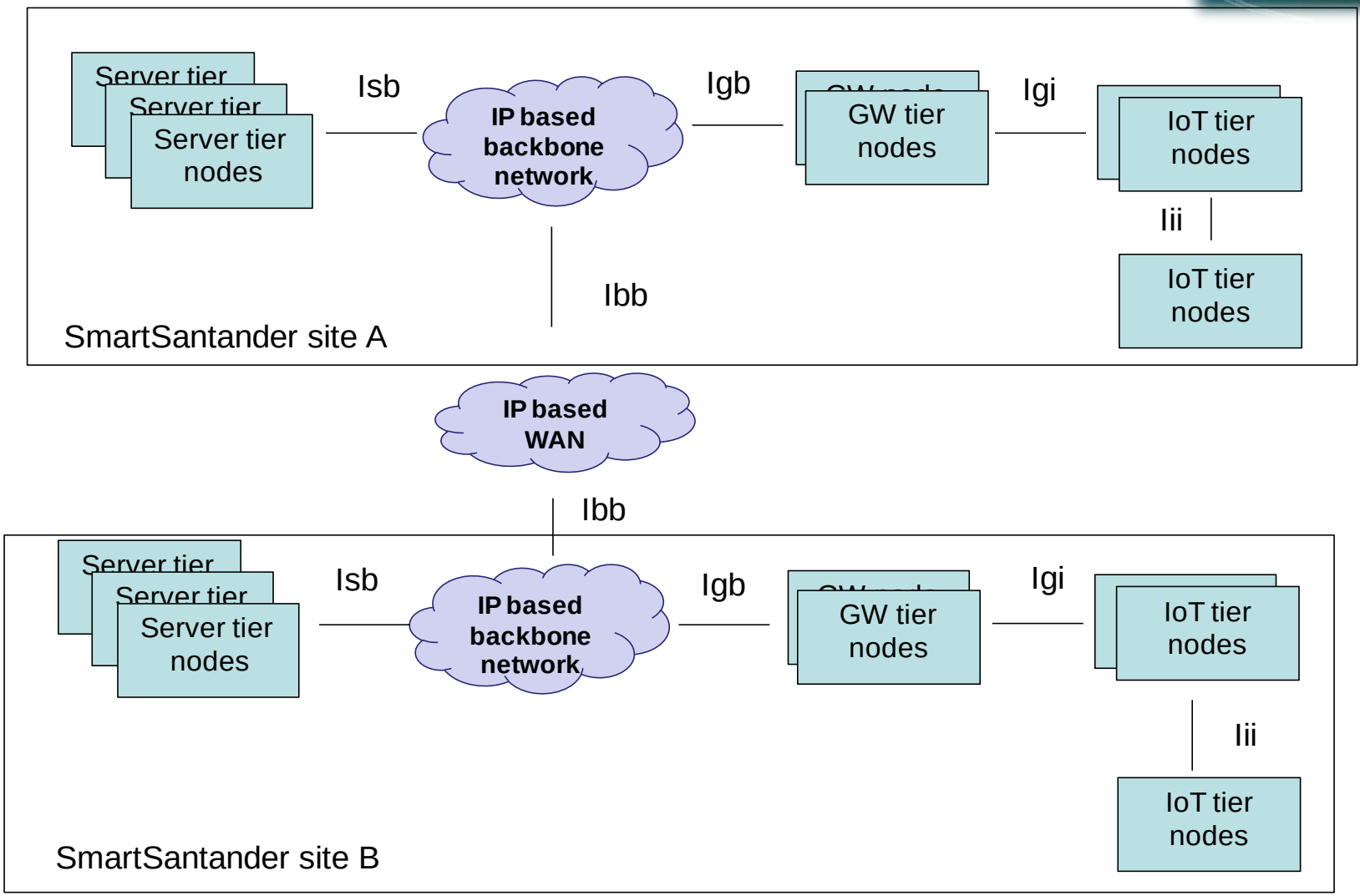
- Additional impacts (at least one of the following):
  - Improve / extend the existing capabilities of the SmartSantander experimental test facility
  - Stress-test the capabilities of the current facility

SmartSantander infrastructure

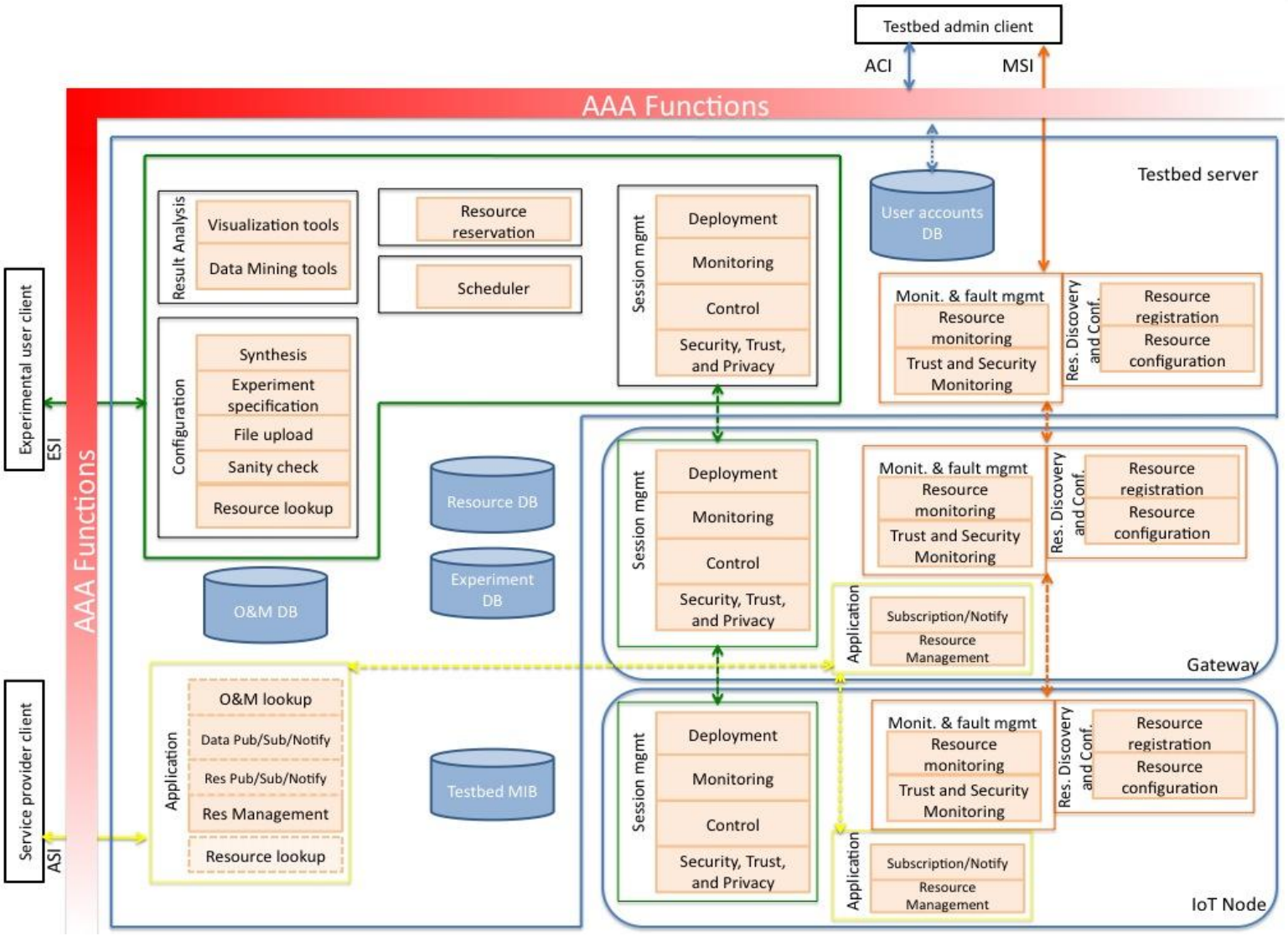
HW architecture



Network architecture



SW architecture



Santander testbed site

IoT experimentation plane

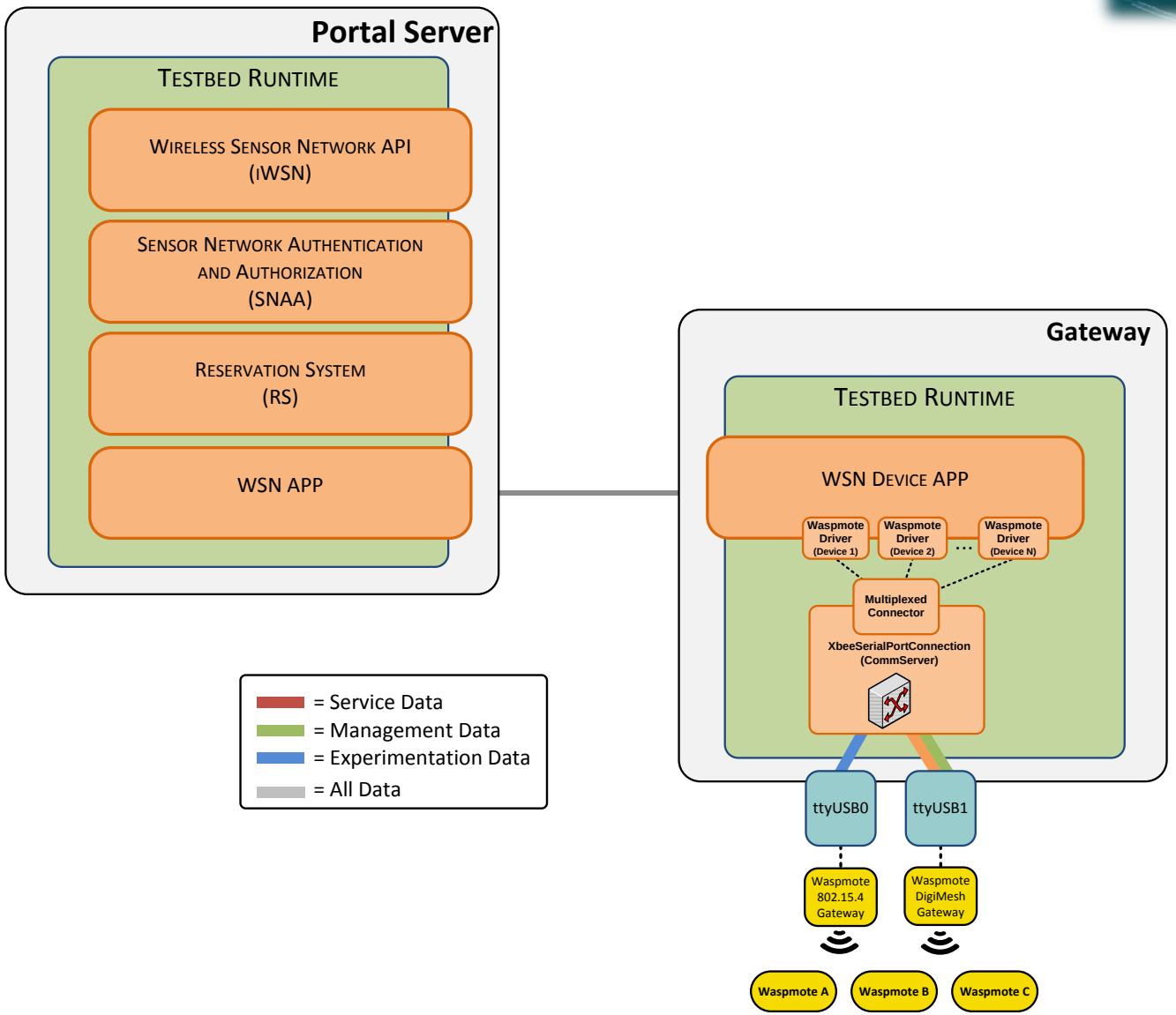
- **WASP motes**
  - AtMega1281L (8 MHz)
  - 1-2 802.15.4 radios
  - Flash + SD card

- **Servers cluster**
  - USN service platform

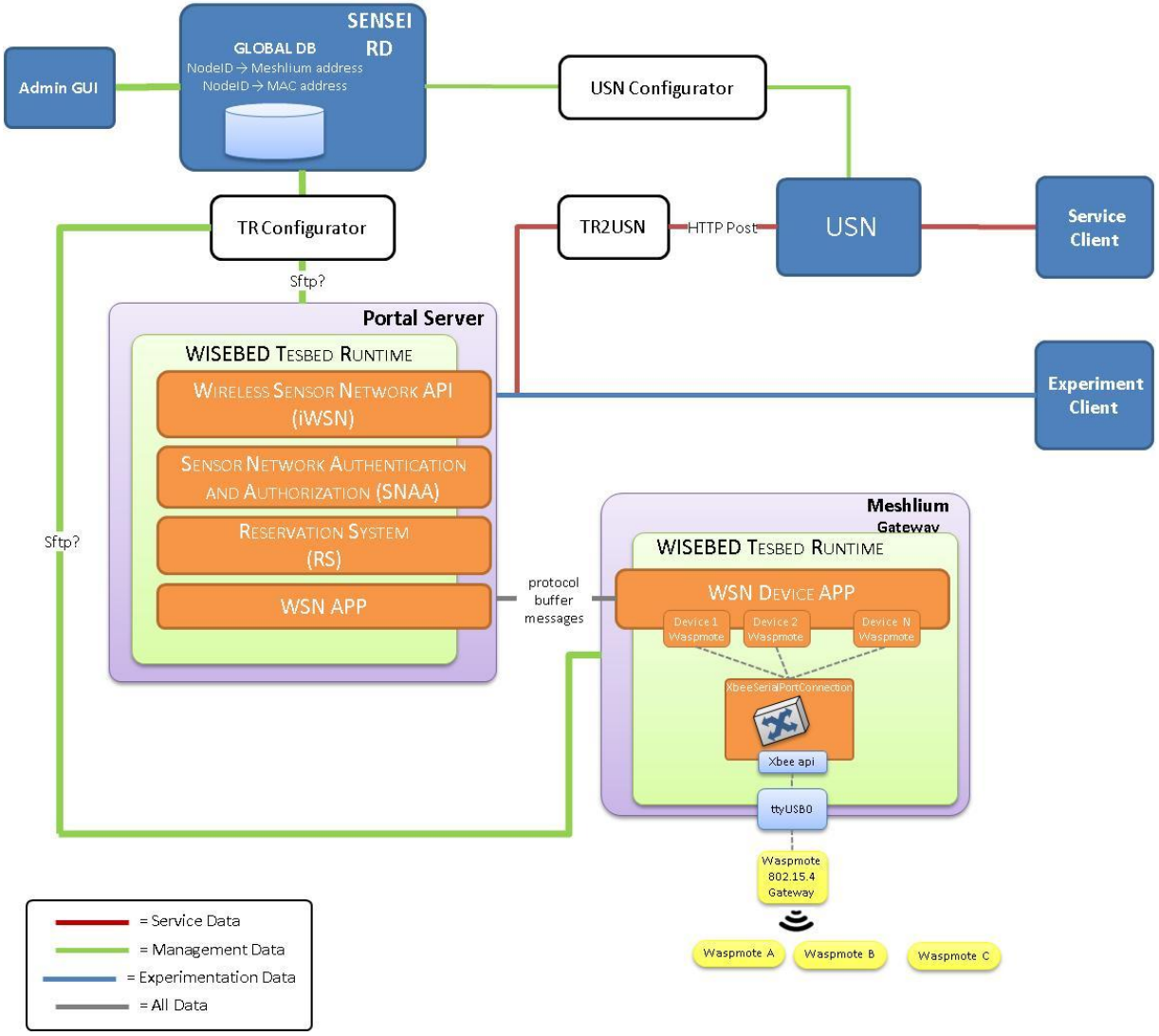
- **Meshlium Extreme**
  - x86 based, 500Mhz
  - Wifi,802.15.4
  - 8GB storage

- Ferromagnetic sensors for car park detection
- Noise and CO2 sensors

IoT experimentation plane

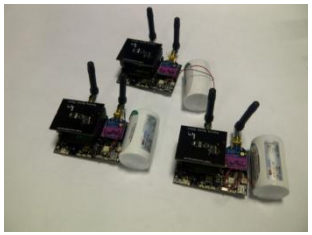


IoT experimentation plane

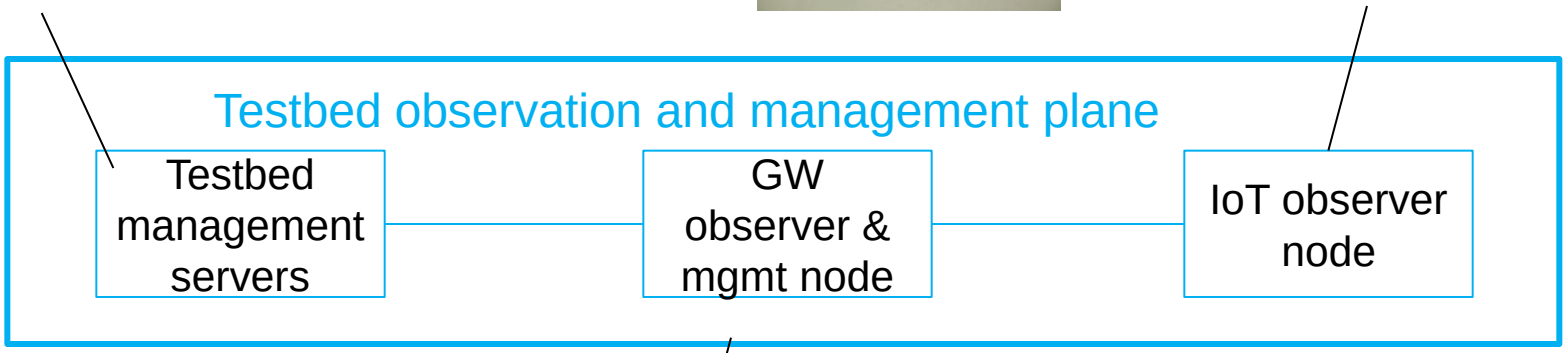


Observation & management plane

- **Linux based server cluster hosting**
  - Testbed management services
  - Databases for user management and AAA, testbed resources, experimental results collection



- **WASP Mote**
  - Same as experimentation node
  - Dual transceiver for control plane and service data



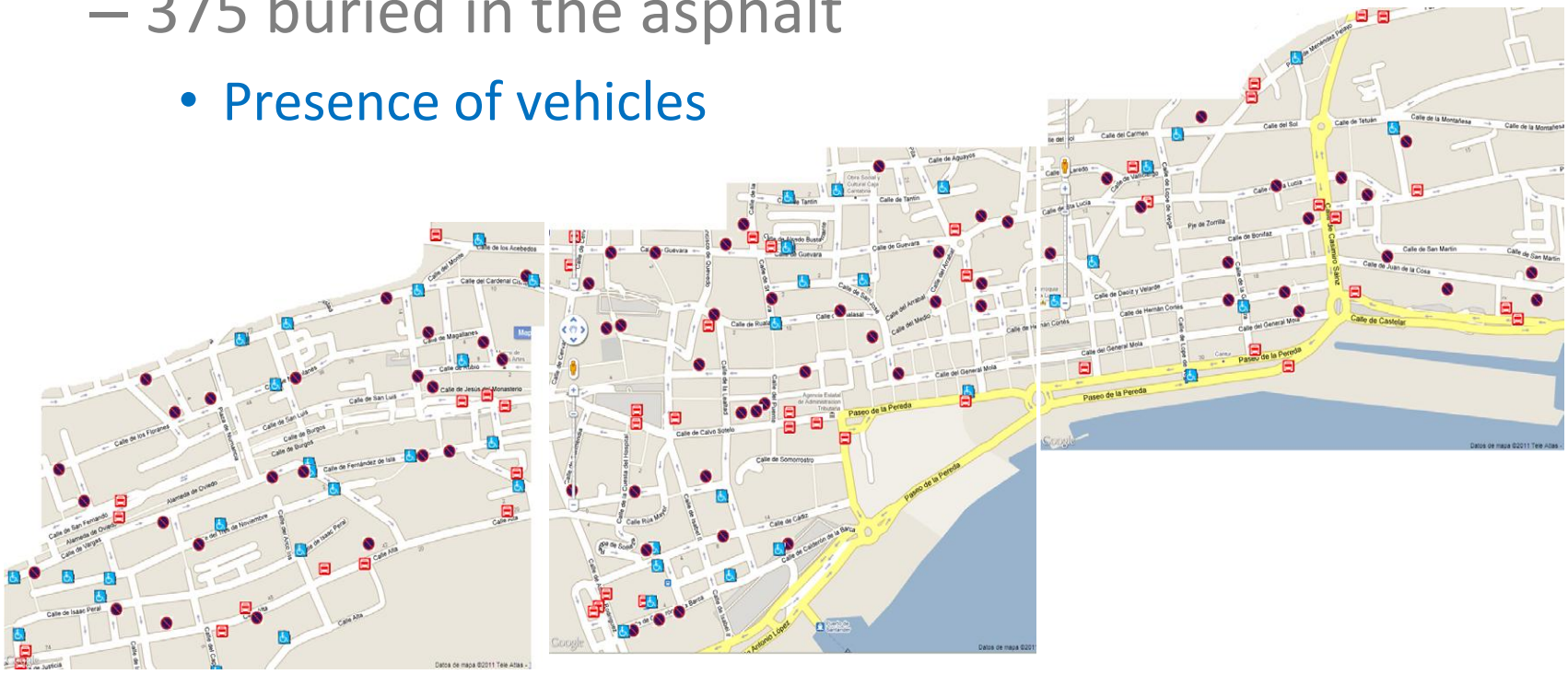
- **Meshlium Extreme**
  - Same as experimentation node

Horizontality support

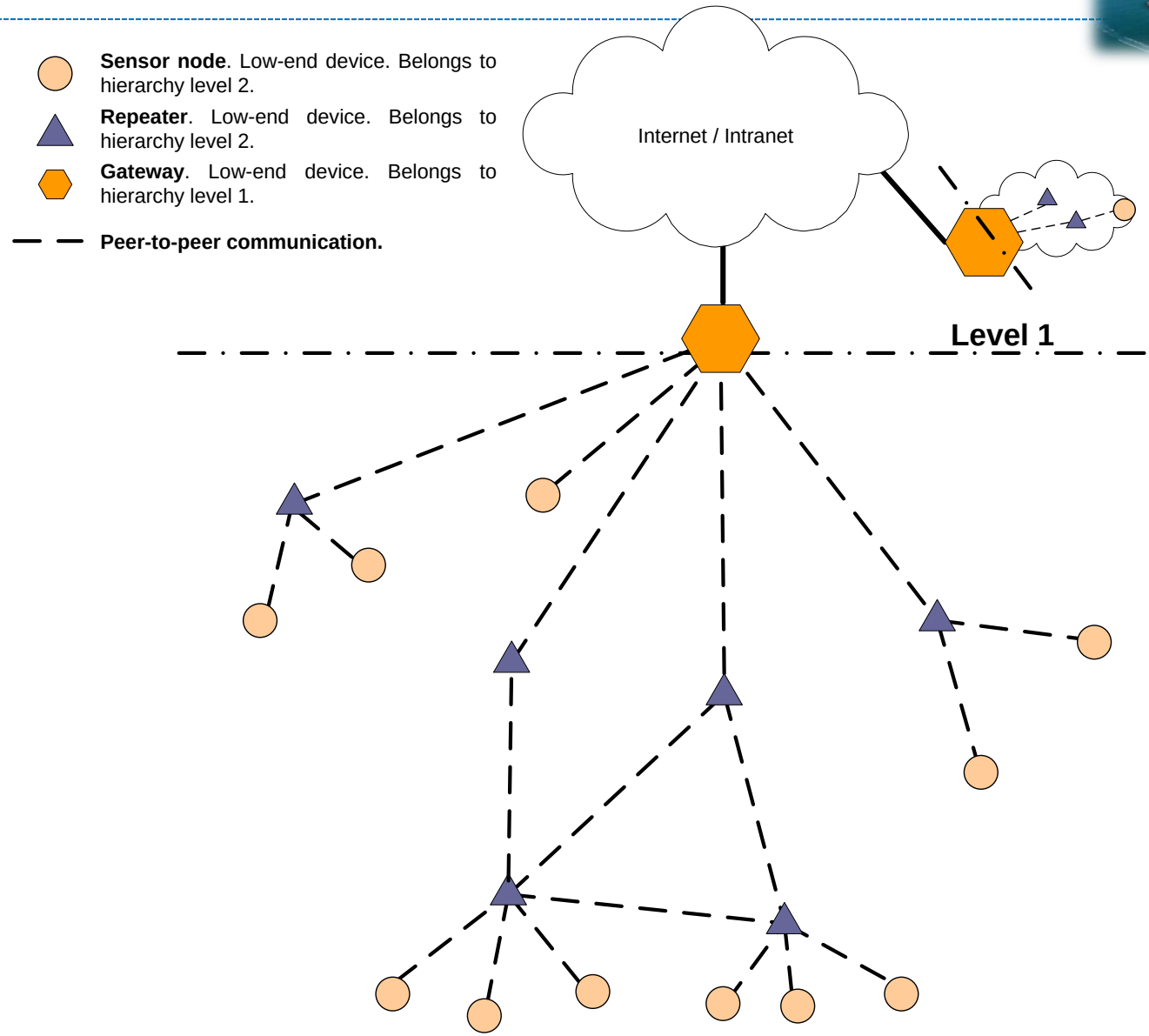
- IoT experimentation in parallel with service provision
  - Binary images to be loaded on IoT nodes support experiment and Smart City service
  - Dedicated libraries containing service and testbed management functionalities

Phase 1 deployment

- Santander - Phase 1 target deployment
  - 1400 installed on lamp posts
    - Temperature, Relative Humidity, Noise Levels
  - 375 buried in the asphalt
    - Presence of vehicles

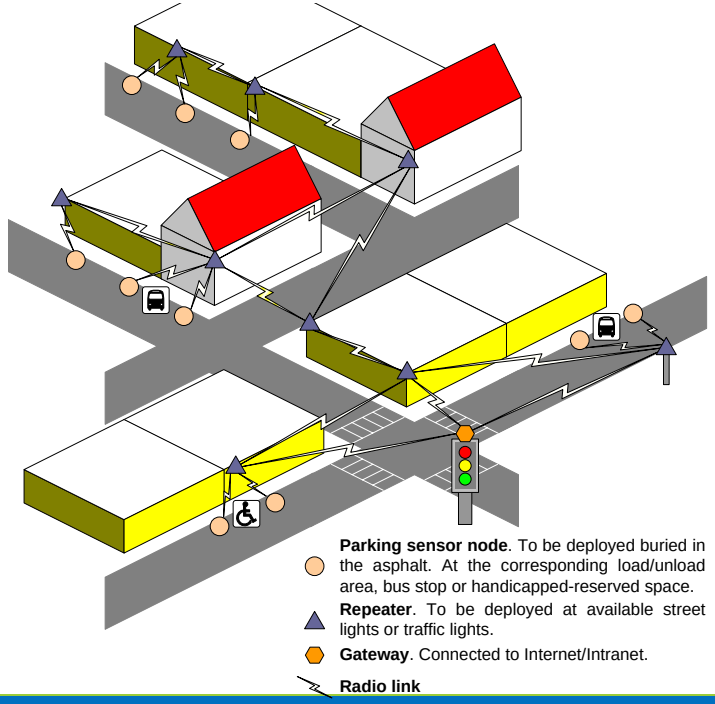


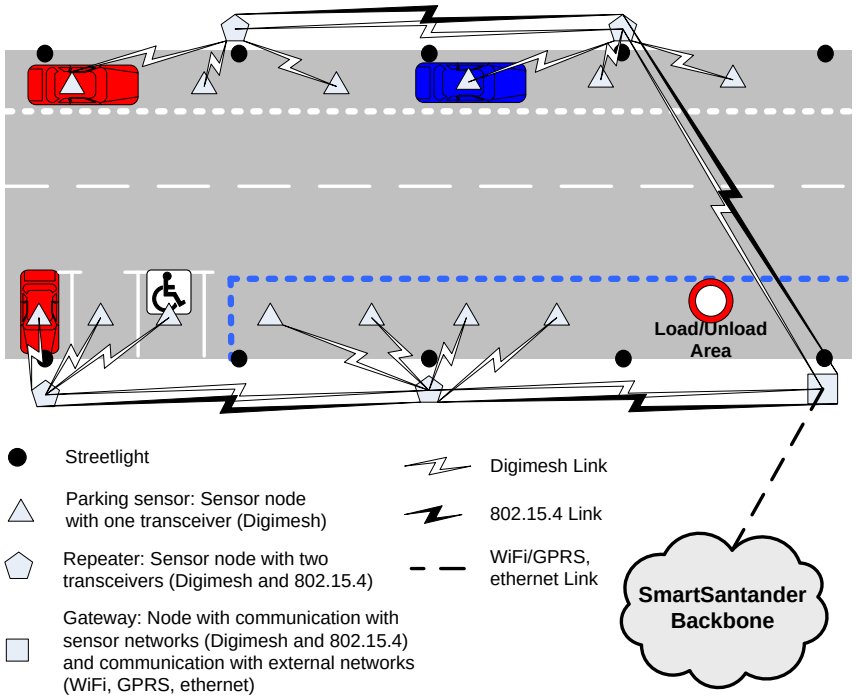
Network architecture



Deployment architecture

- Full-meshed network architecture
  - Topology controlled by experimenter
  - Reprogrammable over-the-air
  - Dual transceiver architecture, one dedicated for control plane and one for experimentation





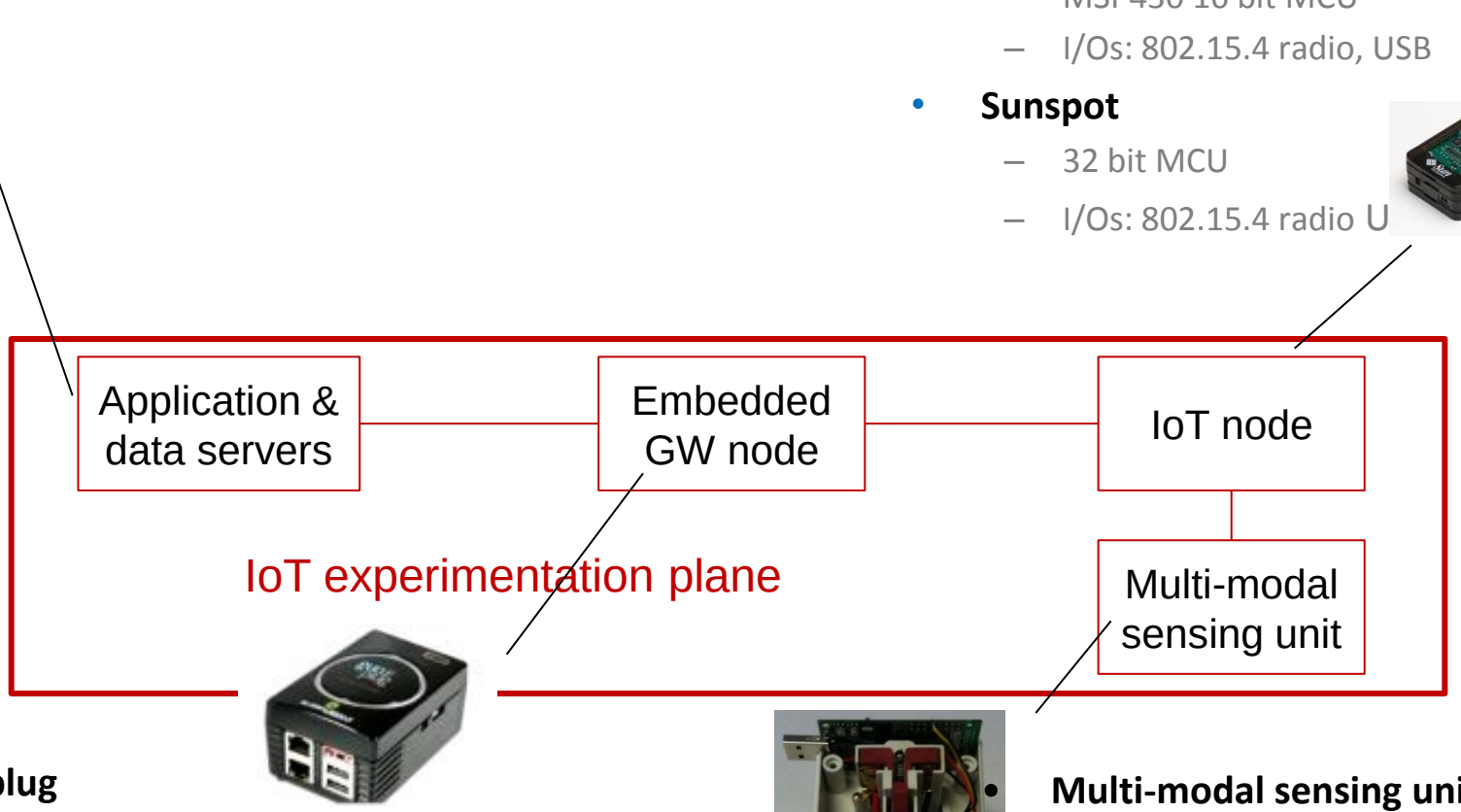
From the lab into the city

Guildford testbed site

IoT experimentation plane

- **Linux server cluster hosting applications**

- **X-Bow TelosB**
  - MSP430 16 bit MCU
  - I/Os: 802.15.4 radio, USB
- **Sunspot**
  - 32 bit MCU
  - I/Os: 802.15.4 radio U



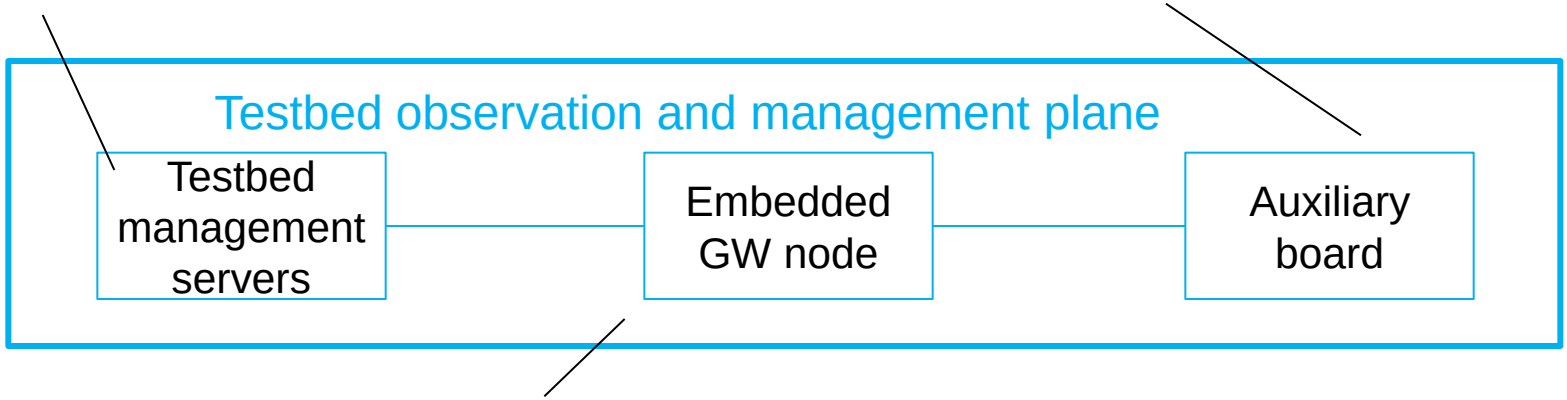
- **Guru plug**
  - ARM5 based embedded Linux server
  - I/Os: Ethernet, Wifi, Bluetooth, 2 USB, 802.15.4

- **Multi-modal sensing unit**
  - Energy meter, Light, Temp, PIR, Vibration, Sounds

Observation & management plane

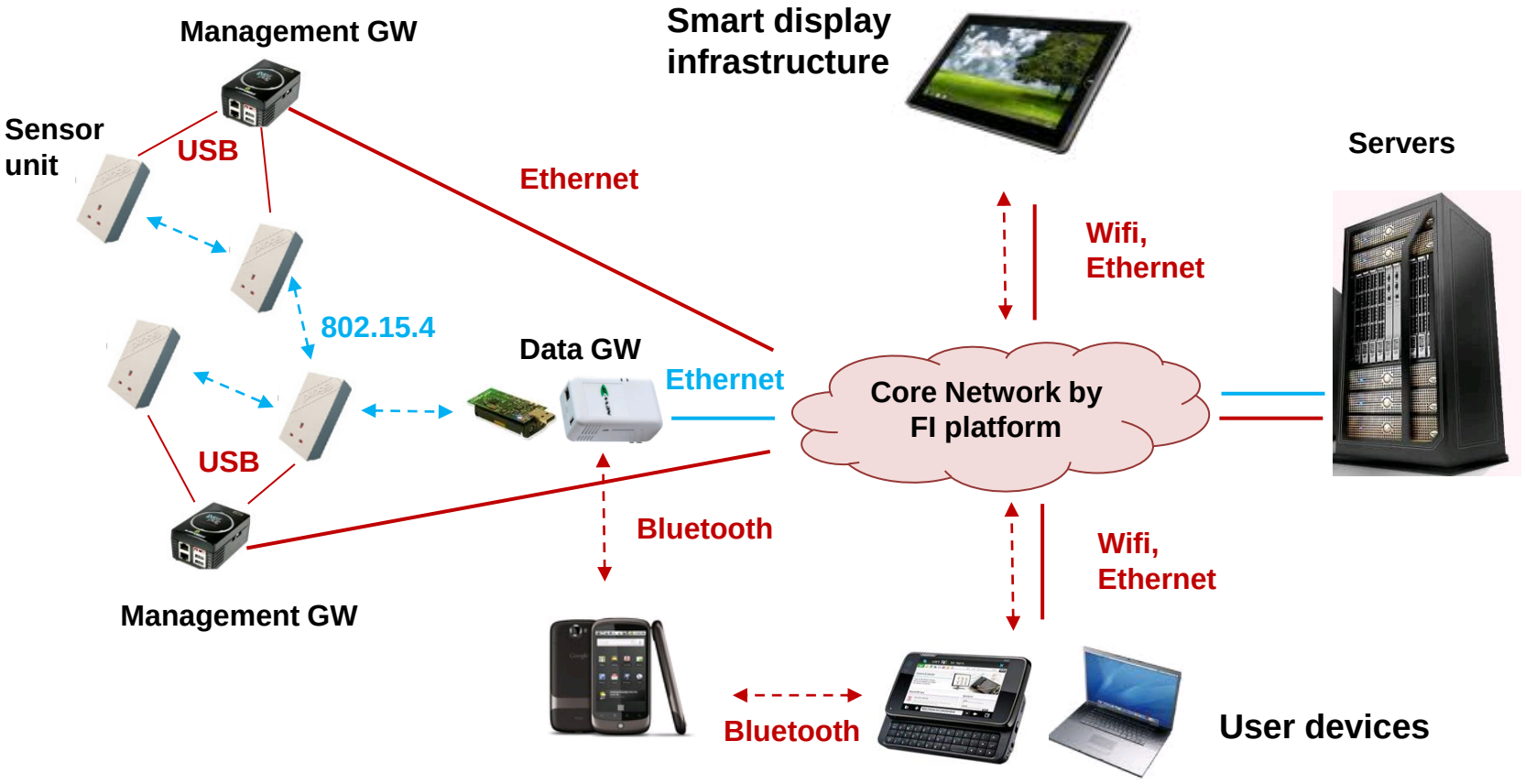
- **Linux based server cluster hosting**
  - Testbed management services
  - Databases for user management and AAA, testbed resources, experimental results collection

- **Auxiliary board**
  - PIC18 based board with integrated USB hub
  - Measurement of IoT node energy consumption
  - Sensor event/phenomena emulation using DACs

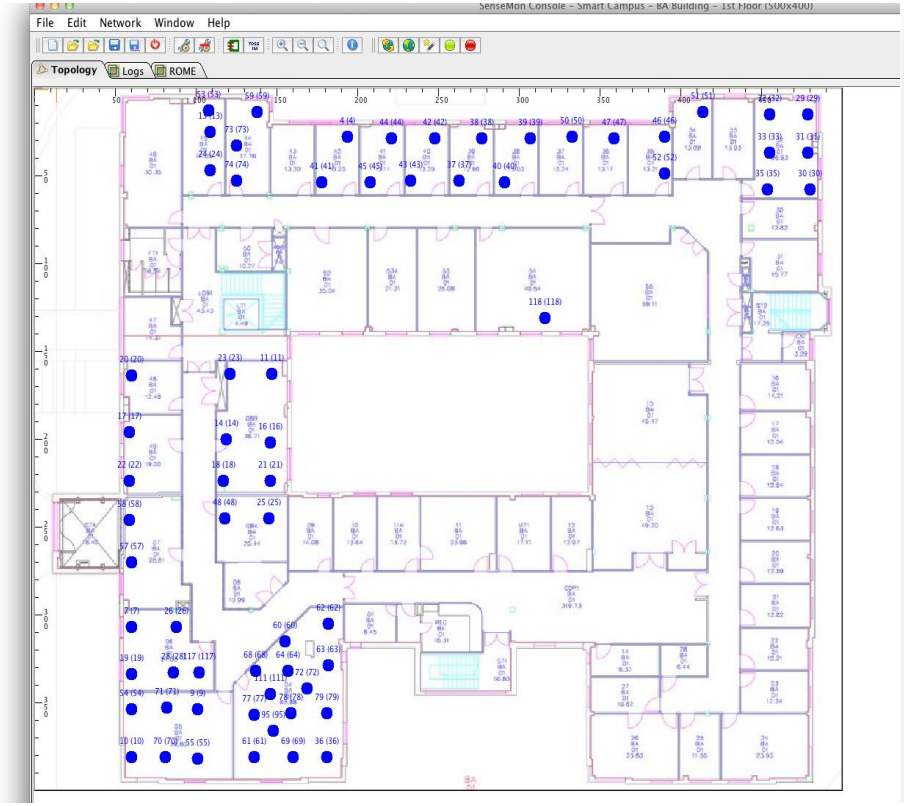


- **Management GW node: Guruplug**
  - ARM5 based embedded linux server
  - USB based out of band interaction with IoT node (Reset, reprogramming, statistics collections, debugging)
  - Packet sniffer

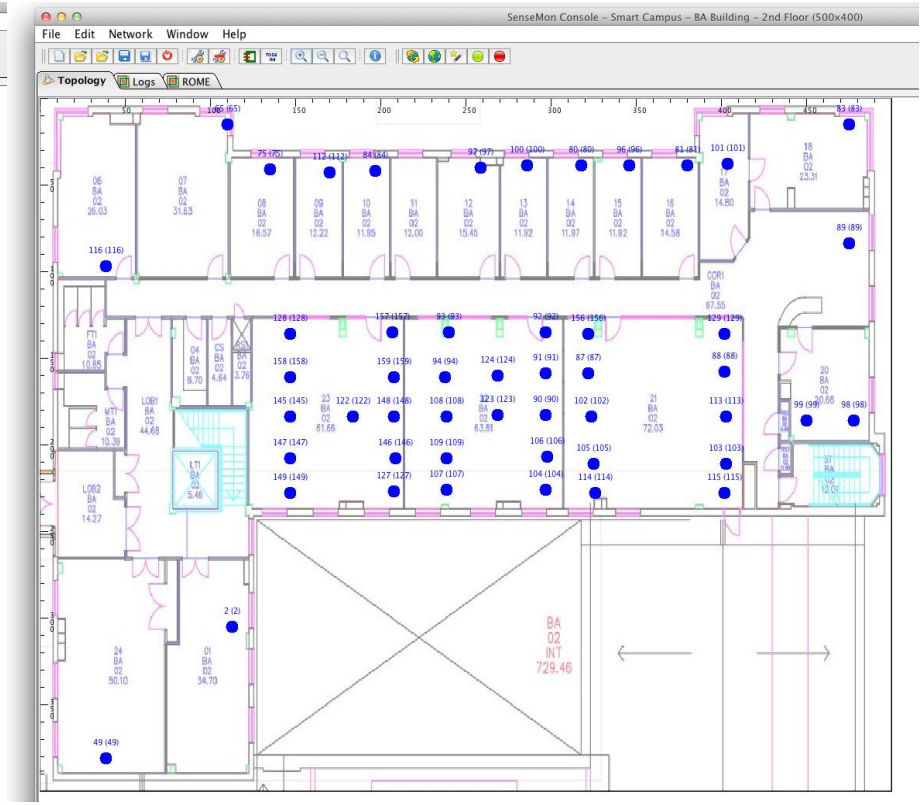
Network architecture



Deployment snapshot



1st floor



2nd floor

IoT node

PEN Unit

Multi-modal sensing unit
- Light, noise, PIR, temperature

Power Consumption Monitoring Unit
- Power, reactive power, current, phase, Voltage/RMS,Time operational since connection
- Max frequency every 2 seconds

Sensor emulation board

Sensor node with MCU, 802.15.4 radio and USB

Luebeck testbed site

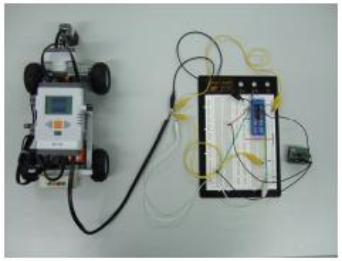
Lübeck Site & WISEBED

- The testbed site at Lübeck University is part of the bigger WISEBED experimental facility and offers possibilities to experiment with sensor networks.

- Advantages of the WISEBED connection:
  - The testbed can basically be used the same way as in the Santander site
  - Apart from the roughly 350 nodes in Lübeck, the whole WISEBED experimental facility can be used (1500 nodes when everything is up and running)

Hardware

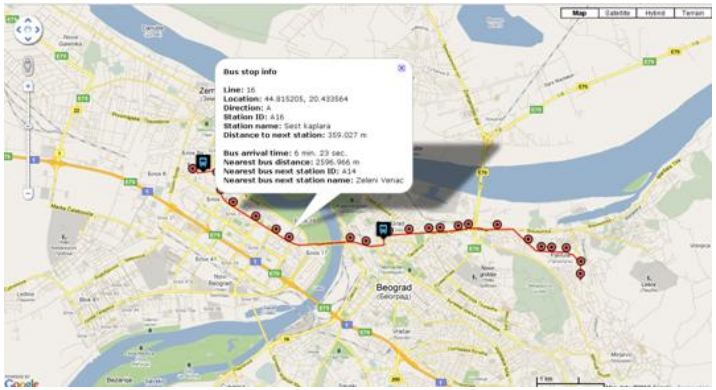
- Stationary sensor node deployment
  - 100 iSense, Pacemate, and TelosB each
  - Temperature, humidity, light, Passive Infrared (PIR), and accelerometer
- Mobile sensor node deployment
  - 10 iSense and Roomba robot
  - 1 Lego Mindstrom NXT robot
  - Touch, cliff, and dirt
- Outdoor sensor node deplyoment
  - 35 iSensor nodes with solar recharge
- Gateways
  - 35 Acer Aspire One Netbooks



Belgrade testbed site

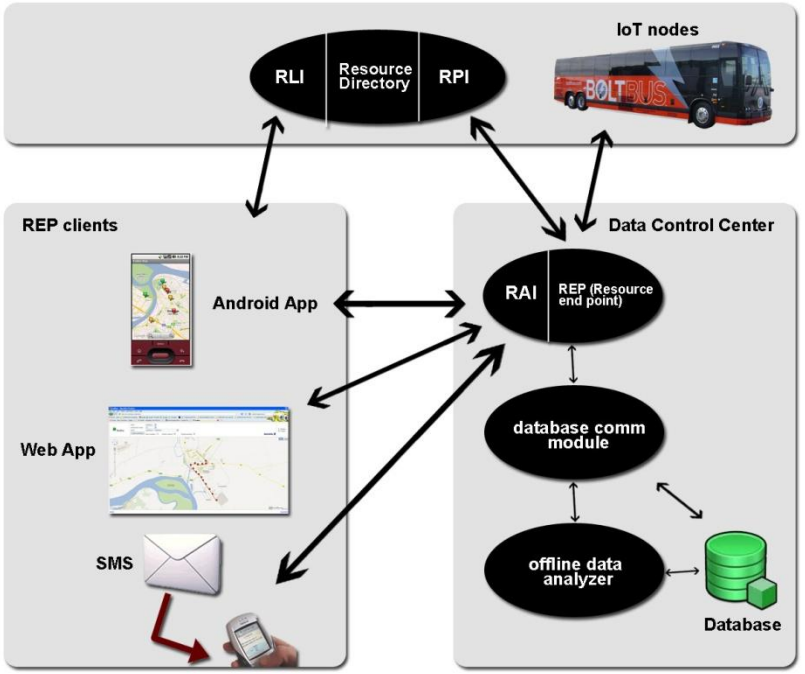
Overview of EkoBus platform

- EcoBus system provides realism and possibilities for experimentation in the real-world environment offering evaluation and engagement of the end-user
- Platform utilizes public transportation vehicles in the city of Belgrade and the city of Pancevo to monitor environmental conditions and track vehicles



Platform architecture

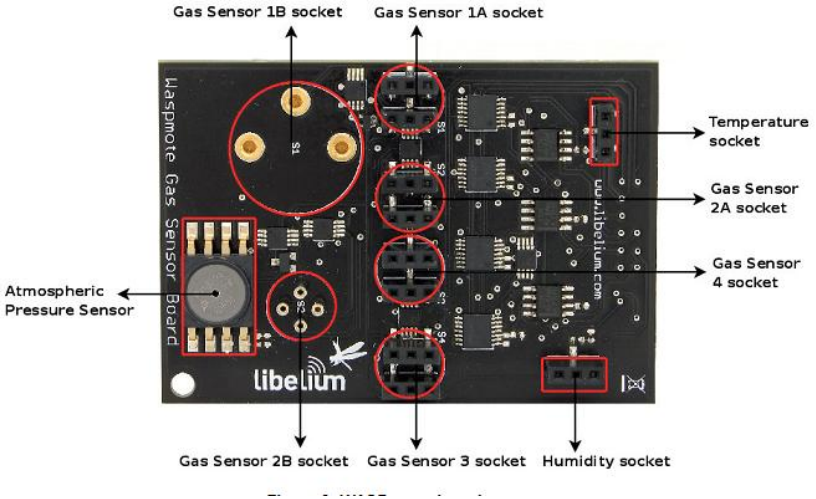
- Overall 60 IoT nodes mounted on busses
  - 45 nodes with GPS Location and speed
  - 15 nodes with environmental sensors
    - Carbon Monoxide Sensor (CO) sensor
    - Nitrogen Dioxide Sensor (NO2) sensor
    - Carbon Dioxide Sensor (CO2) sensor
    - Temperature Sensor sensor
    - Atmospheric Humidity Sensor sensor
- Nodes are not configurable but can be queried for data through Smartsantander platform



Resource overview

- All interaction in the system is REST based
- Power consumption is not an issue as both resources are connected to a vehicle power source (backbone)

- WASP mote
  - C programming
  - Deployment (top of the vehicle)
  - Interfaces: GPS, GPRS



- TELIT module
  - Python programming
  - Deployment (inside the vehicle)
  - Interfaces: GPS, GPRS

Summary (1/2)

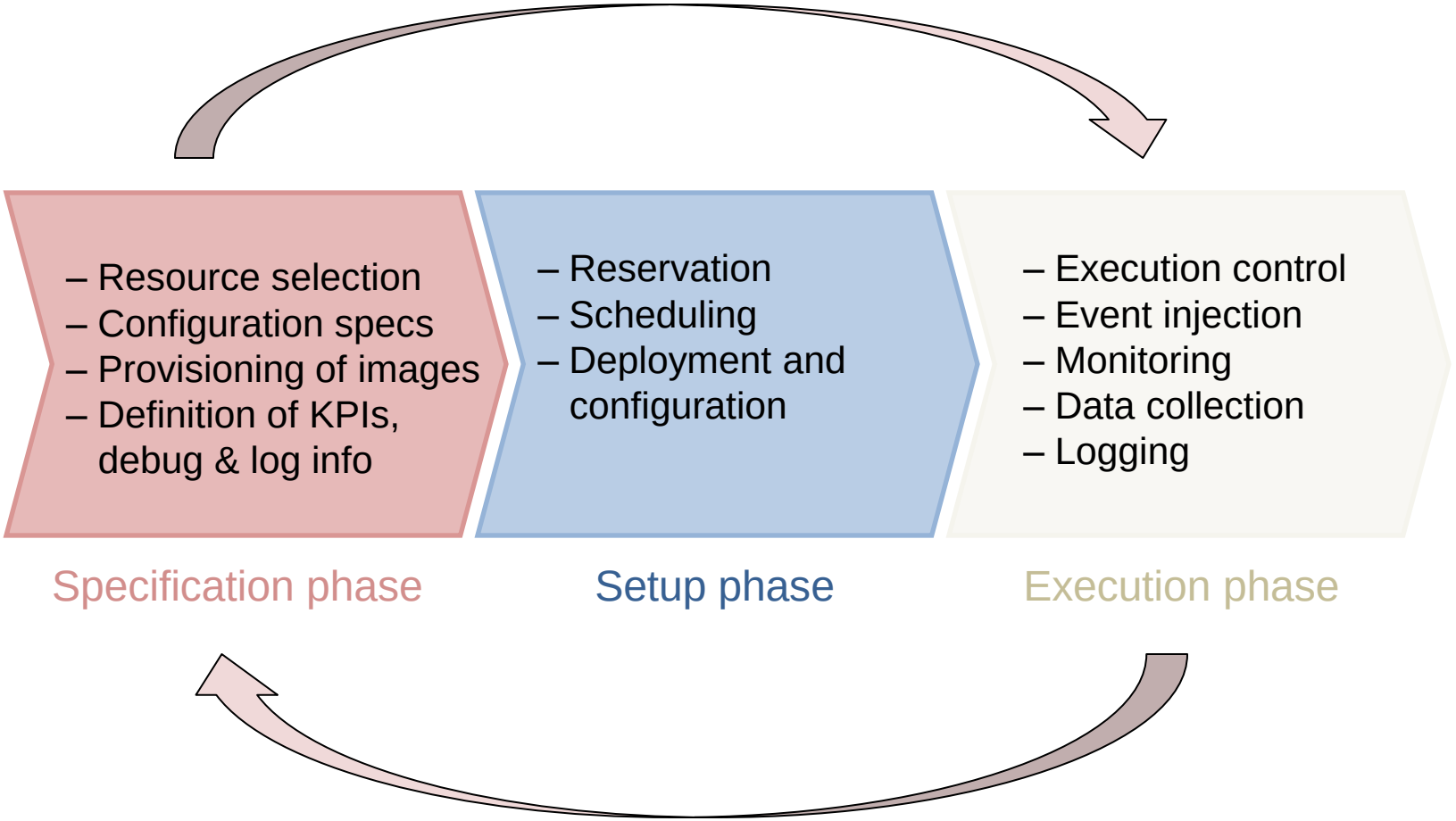
	Santander (Phase 1)	Guildford (Phase 1)
Environment	Outdoor	Indoor
IoT nodes and GW devices	1400 Libellium motes on lamp posts (programmable) 375 Libellium in parking bays ~25 Meshlium GWs	200 TelosB based IoT nodes 50 Sunspots (all freely programmable) 100 embedded GWs
OS and programming support	Embedded C on Libellium motes	TinyOS, Contiki or C on TelosB Java on Sunspots Linux on GWs, C/C++/Java/Python
Sensing modalities	Ferromagnetic field (cars), temperature, noise levels, light intensity and CO	Presence, light, temp, noise, energy and load switching (TelosB) Light, Temp, 3D accel (SunSpot)
Special features	MOTAP, 2nd radio on repeater motes for out-of-band control plane and continuous service provision	Node energy monitoring, injection of sensing events on auxiliary board

Summary (2/2)

	Luebeck (Phase 1)	Belgrade (Phase 1)
Environment	Indoor and Outdoor	Outdoor mobile
IoT nodes and GW devices	300 indoor IoT nodes (100 iSense, Pacemate, and TelosB), 35 oudoor iSense 11 mobile nodes 35 Acer One netbooks as GWs	60 based WASP motes mounted on buses, all with GPS Location and speed, 15 nodes with additional environmental sensors GPRS for connectivity
OS and programming support	iSense OS and WiseLib on all platforms (all motes programmable)	Embedded C on Libellium motes (non programmable, data only)
Sensing modalities	Temperature, humidity, light, Passive Infrared (PIR), and accelerometer	GPS, Speed CO, NO2, CO2, Temp, Humidity
Special features	WISEBED federation, mobile robots, WiseLib	Realistic mobility

Experimentation on top of SmartSantander

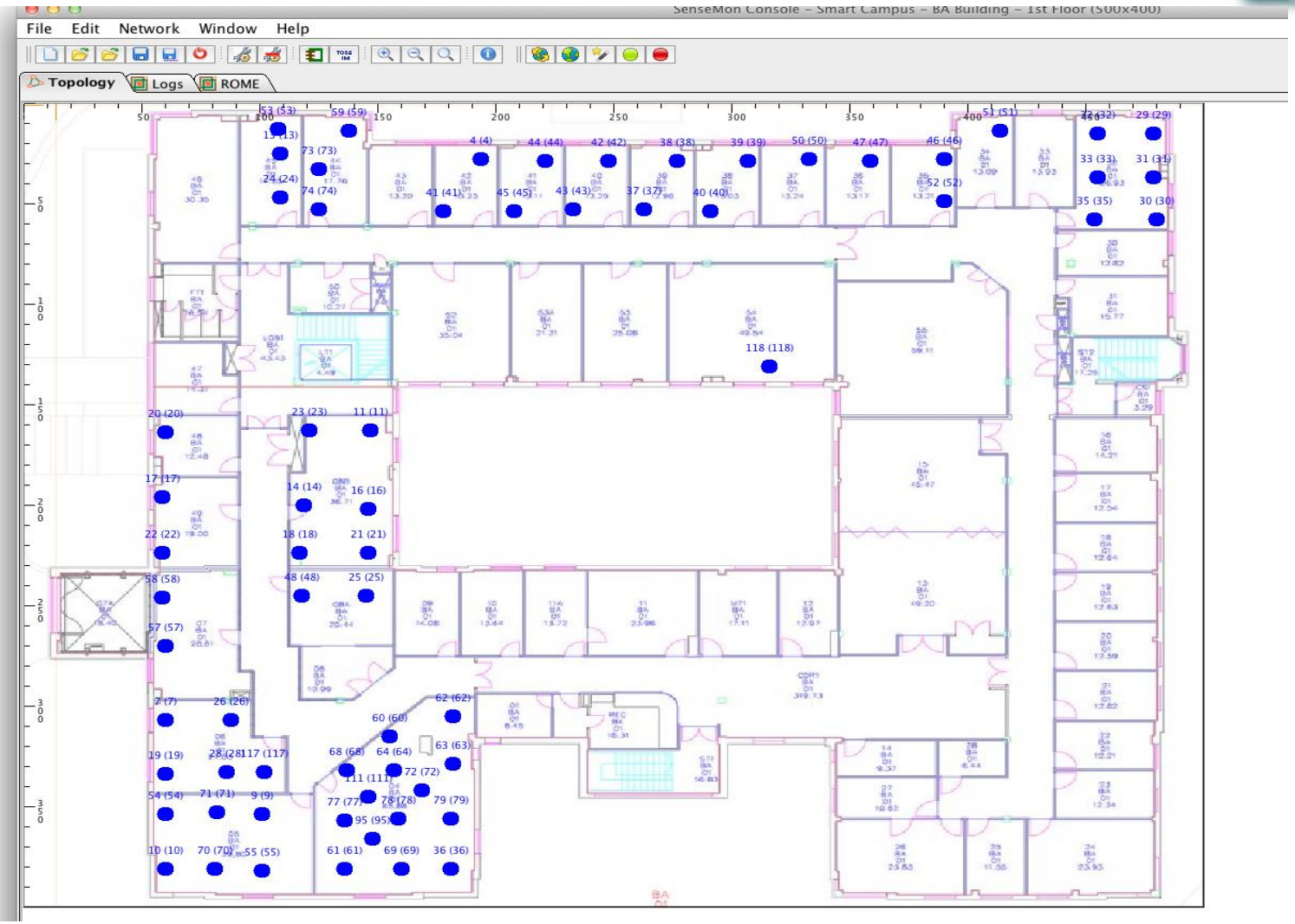
Experimentation life-cycle



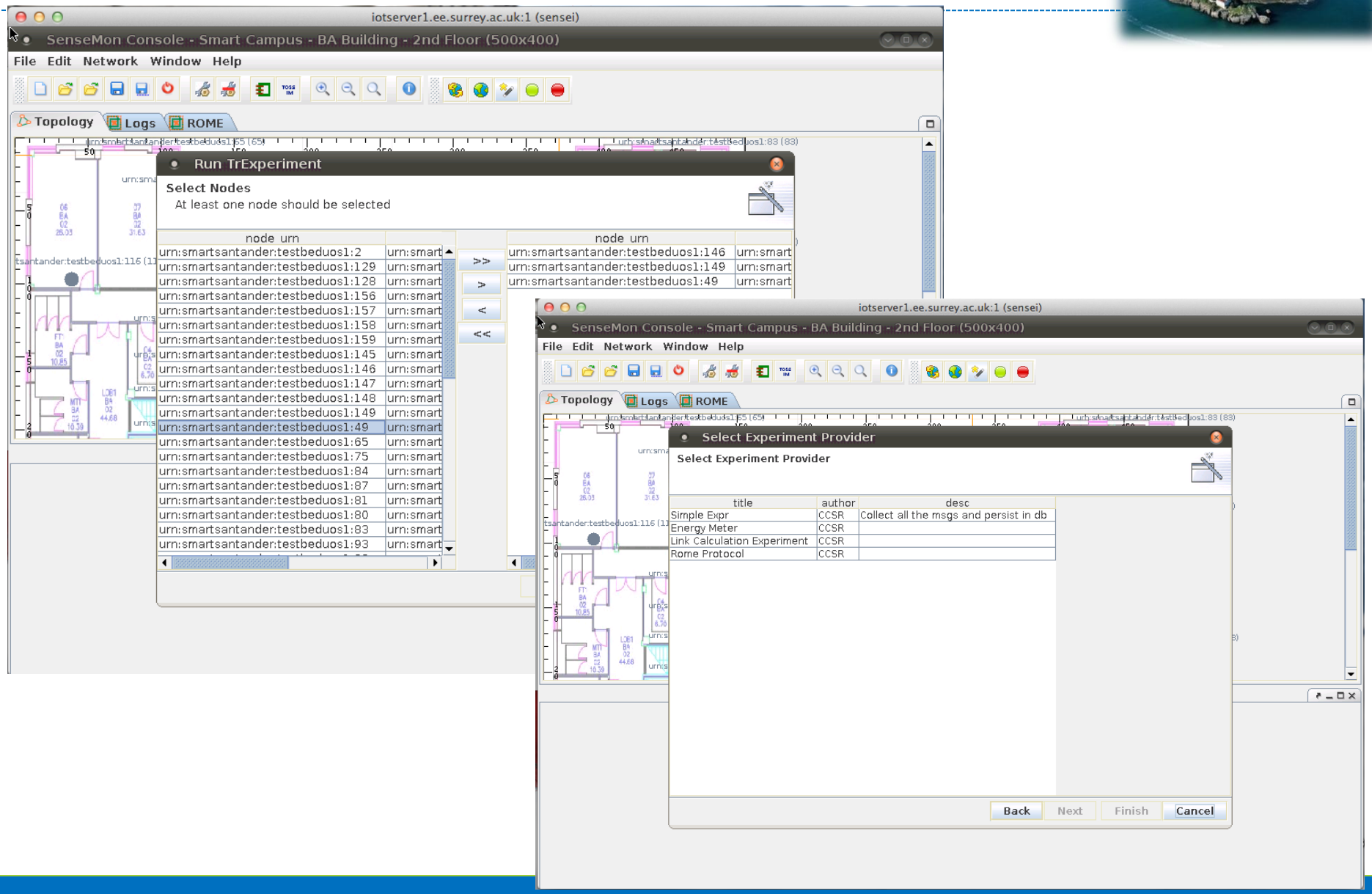
Specification phase

- Identify clear experimentation objectives
  - What is the goal of your experiment?
  - What are the scenarios you want to evaluate and the assumptions you take?
  - What are the KPIs you want to collect or debugging parameters?
- Select adequate experimentation resources
  - Derive from above requirements for environment, scale, node capabilities, timing and topology
  - Know the existing testbed topology and characteristics
  - Consider availability of existing resources
- Configuring your experiment
  - Choose carefully how you want to log statistics and debug output and how to derive them
  - Instrument experimentation code adequately
  - Determine number of runs and duration of experiment interactively based on errors and desired confidence intervals

Testbed resource browser



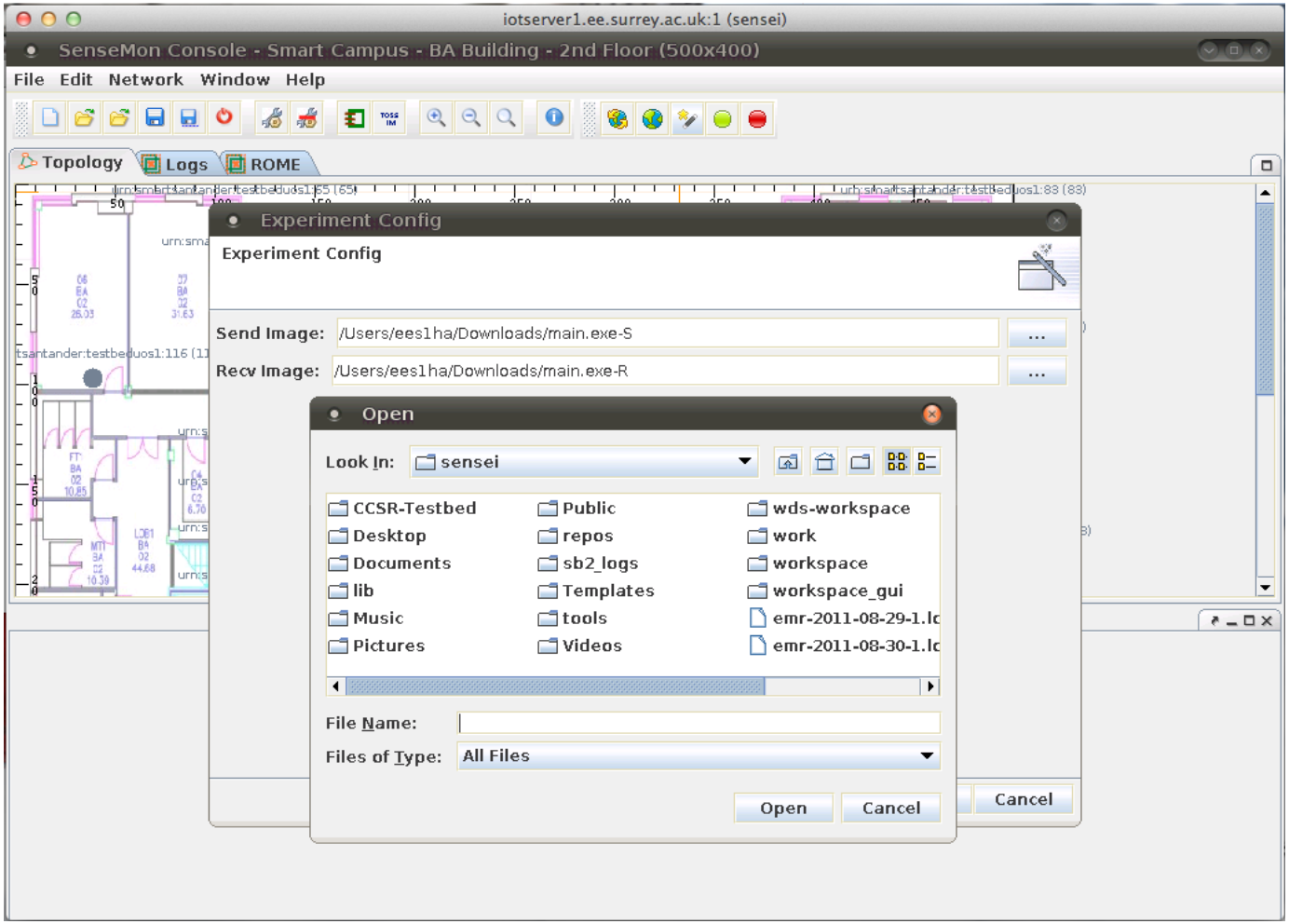
Resource and experiment selection



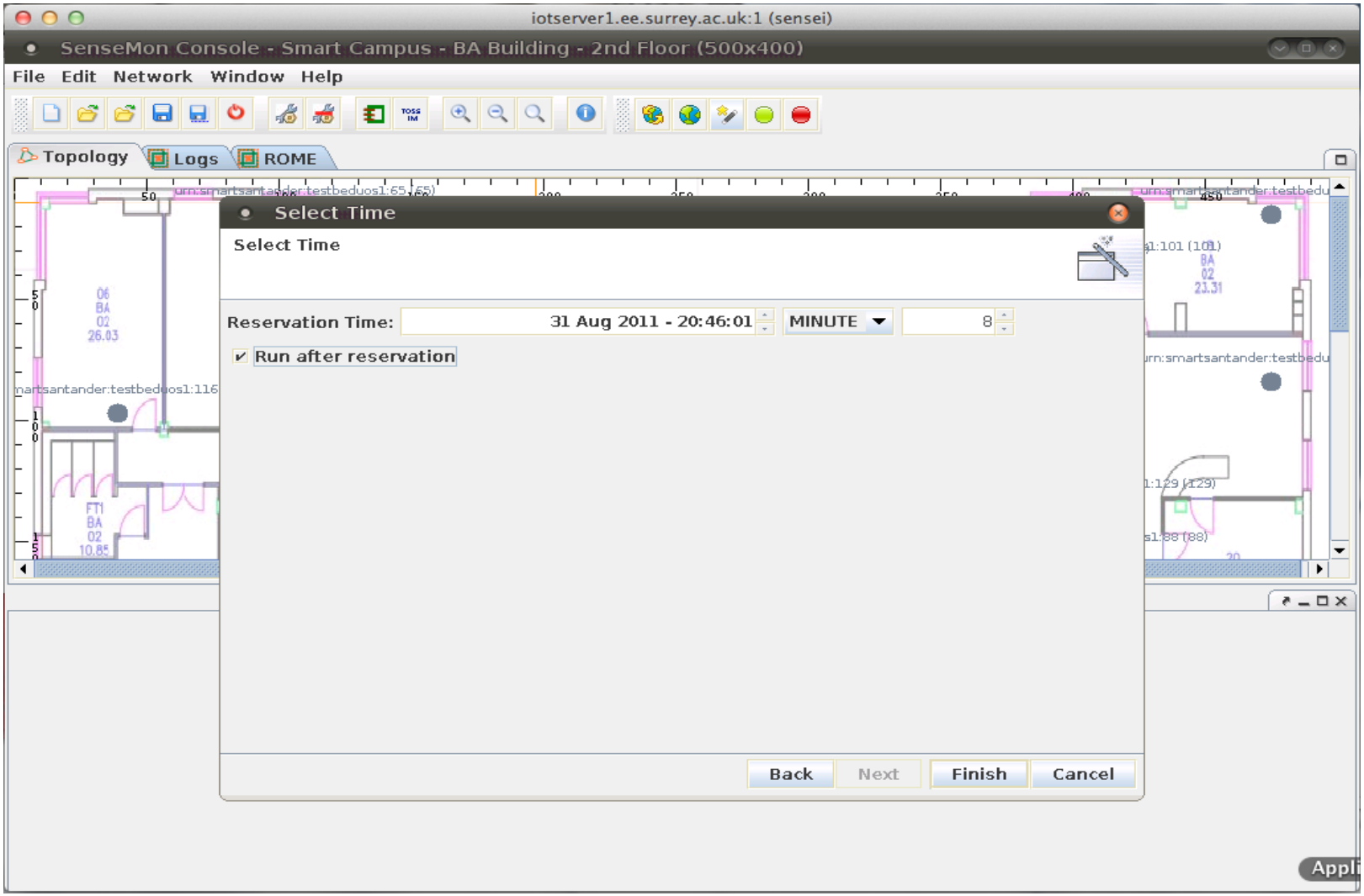
Setup phase

- Reserve selected resources for experiment
  - 3 methods of reservation supported, hidden behind the reservation API: GoogleCalendar, database or persistent memory
- Provide finalised images for upload to selected resources
  - An images will be mapped to each selected experimentation resource
- Initiate upload of experiment
  - Manual or automatic scheduling

Experiment configuration



Experiment reservation



Execution control

- Experiments can be initiated and controlled during reserved time periods
  - Testbed environment provides functions to flash nodes, reset nodes, check if nodes are alive
  - A controller instance is required per experiment to interact with nodes (relay message back and forth)
  - Per default a controller sends data to you
  - Testbed provides means to send commands to a set of nodes through the controller

- Your logic in the experimentation code defines how your experiment can be controlled
  - Define handlers for start and stop commands, change of parameters, injection of specific events such as node failure or environmental events
  - Auxiliary board can provide energy measurement information or generation of external sensor stimuli

Handling traces

- You can decide freely the message payload of the trace messages generated by your experimentation nodes
  - Our implementation uses a TinyOS packet as payload
- Per default messages are sent live to your experimentation client via the controller and stored into an SQL experimentation data base
  - Store the binary format of the payload plus meta information (time stamp, node id, experiment id, etc.)
- You can see traces live or replay later in the testbed UI from the data base
- Beware of synchronisation of time stamps between nodes
  - GWs are synched via NTP and can be used to provide reference to attached nodes

Message log browser

iotserver1.ee.surrey.ac.uk:1 (sensei)

SenseMon Console - Smart Campus - experiment (500x400)

File Edit Network Window Help

Topology | Logs | ROME

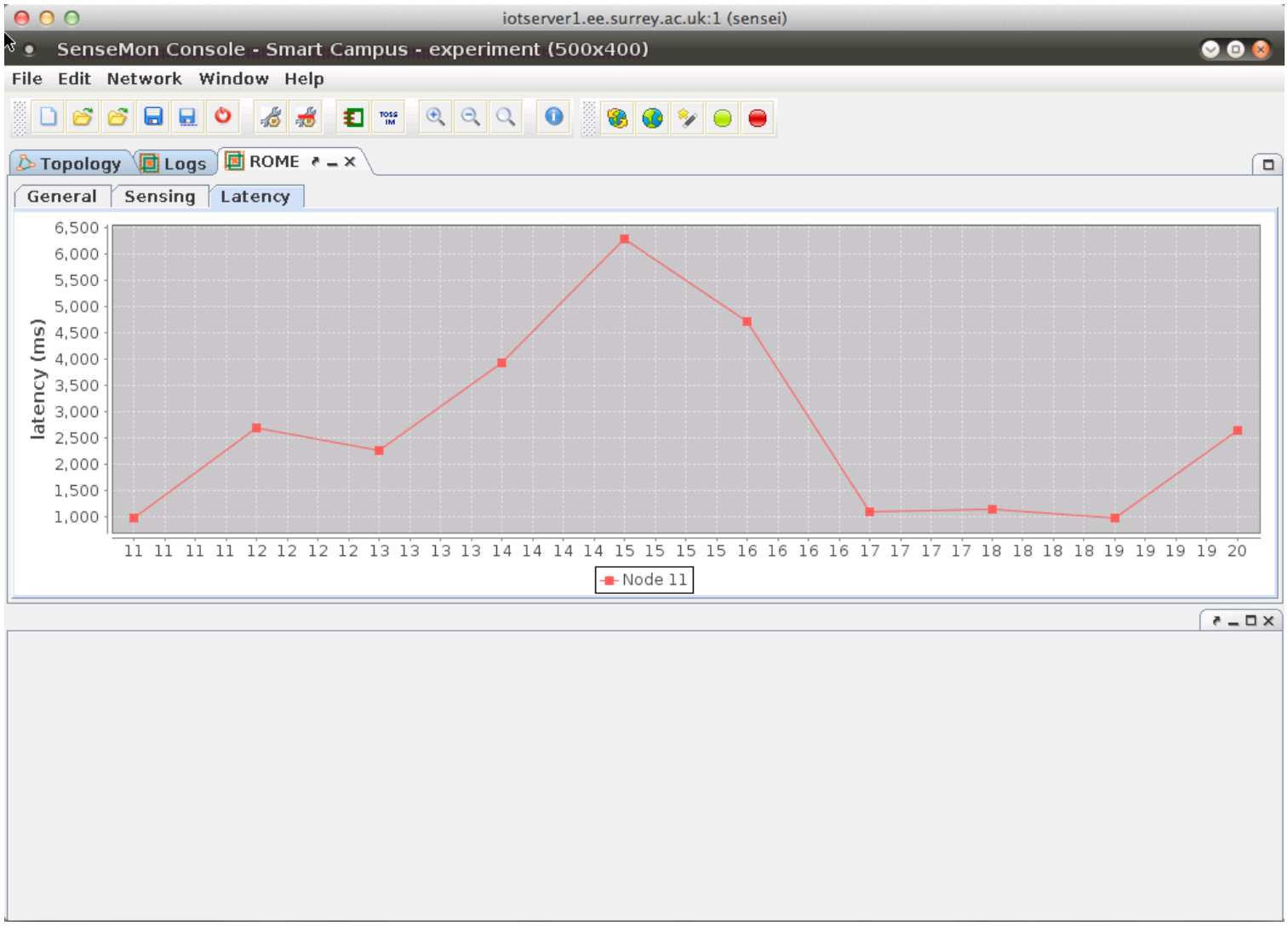
pnlLog | Sent Messages | Received Messages

#	when	srcNode	dstNode	amt...	len	body
110	2011-08-31 13:23:51:0	11	1000000	21	40	ROMEMSG
111	2011-08-31 13:23:52:0	11	1000000	21	40	ROMEMSG
112	2011-08-31 13:23:53:0	11	1000000	21	40	ROMEMSG
113	2011-08-31 13:23:56:0	11	1000000	21	40	ROMEMSG
114	2011-08-31 13:23:57:0	11	1000000	21	40	ROMEMSG
115	2011-08-31 13:23:59:0	11	1000000	21	40	ROMEMSG
116	2011-08-31 13:23:59:0	11	1000000	21	40	ROMEMSG
117	2011-08-31 13:24:00:0	11	1000000	21	40	ROMEMSG
118	2011-08-31 13:24:01:0	11	1000000	21	40	ROMEMSG
119	2011-08-31 13:24:01:0	11	1000000	21	40	ROMEMSG
120	2011-08-31 13:24:03:0	11	1000000	21	40	ROMEMSG
121	2011-08-31 13:24:04:0	23	1000000	0	31	46 6c 61 73 68 20 6e 6f 64 65 20 6f 70 65 72 61 74 69 6f 6e...
122	2011-08-31 13:24:04:0	11	1000000	21	40	ROMEMSG
123	2011-08-31 13:24:05:0	11	1000000	21	40	ROMEMSG
124	2011-08-31 13:24:06:0	11	1000000	21	40	ROMEMSG
125	2011-08-31 13:24:07:0	11	1000000	21	40	ROMEMSG
126	2011-08-31 13:24:08:0	11	1000000	21	40	ROMEMSG
127	2011-08-31 13:24:08:0	11	1000000	21	40	ROMEMSG

Analysing traces

- Plug-in based architecture to provide different views to trace data from the database
  - Raw traces from messages sent and received by nodes
  - Packet flow in topology view
  - Display metrics in line charts, e.g. end-to-end latency
  - Sensed values
- Traces can be exported to text file and processed outside the environment by analytical tools

Visualise statistics and readings

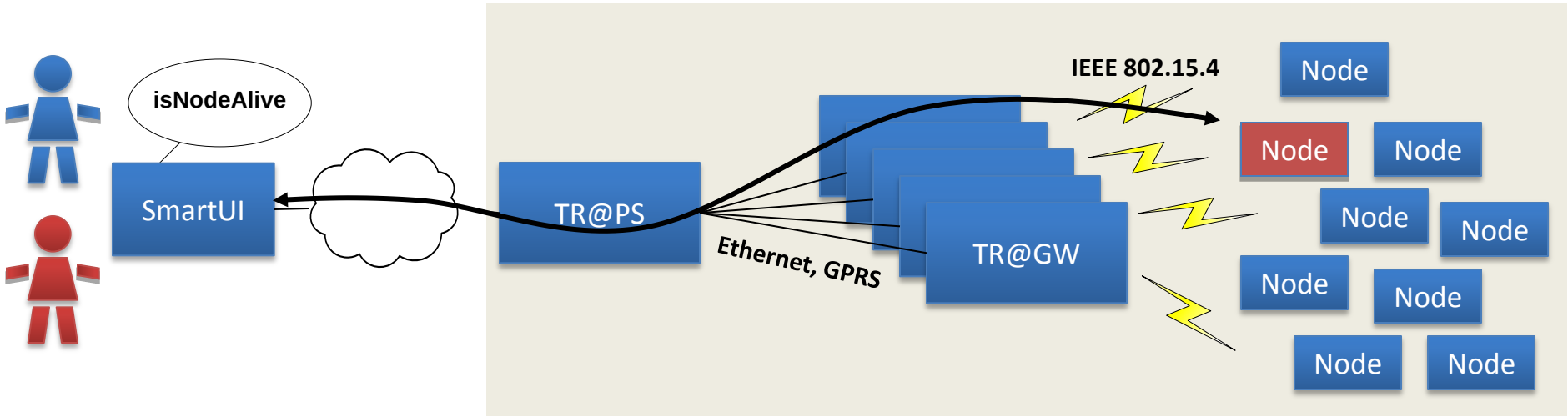


Experimenter client

- GUI or script based
- Tools supporting of the entire experimentation cycle
  - Open APIs with appropriate documentation
  - Transparent and seamless
  - Secure

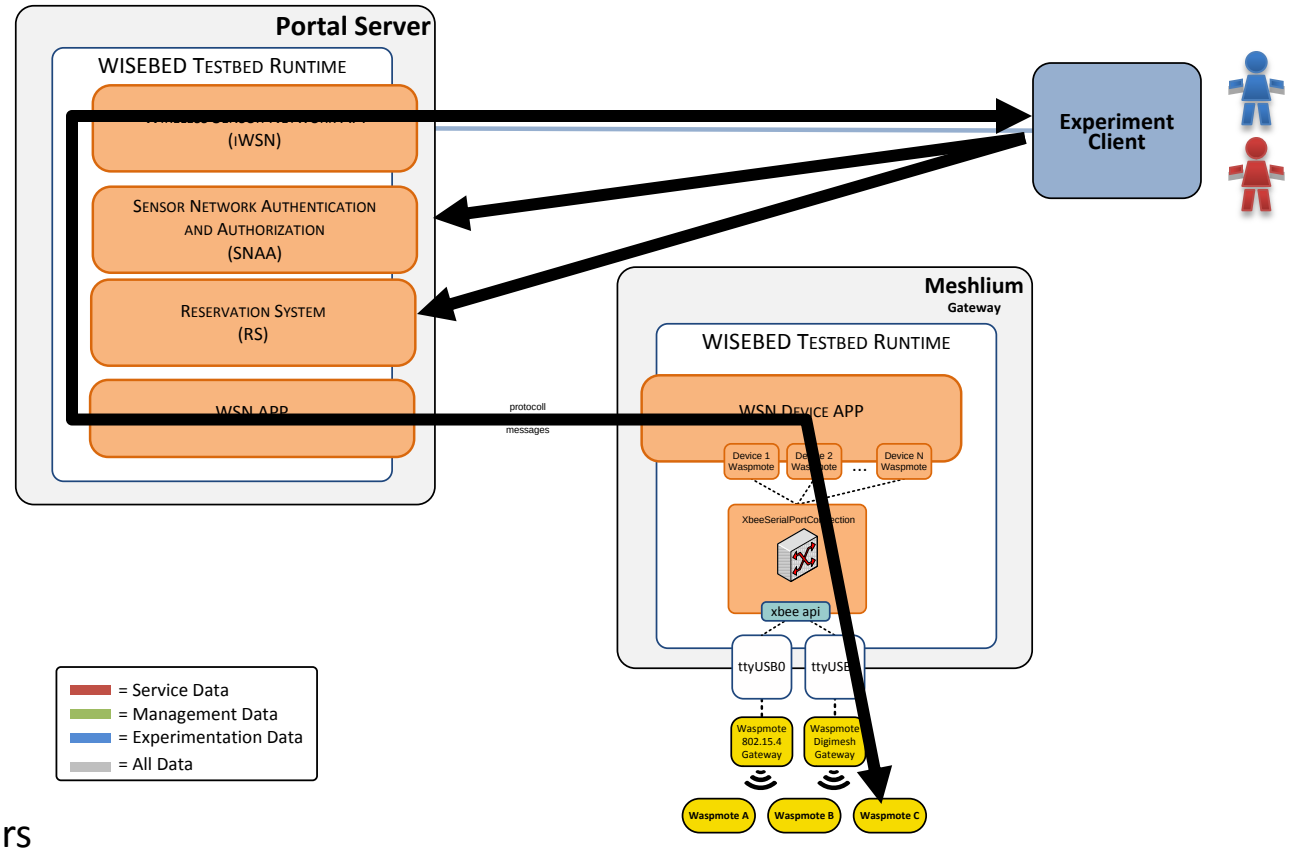
Experimentation script

- Management and experimentation support functionalities
  - isNodeAlive()



Experimentation script

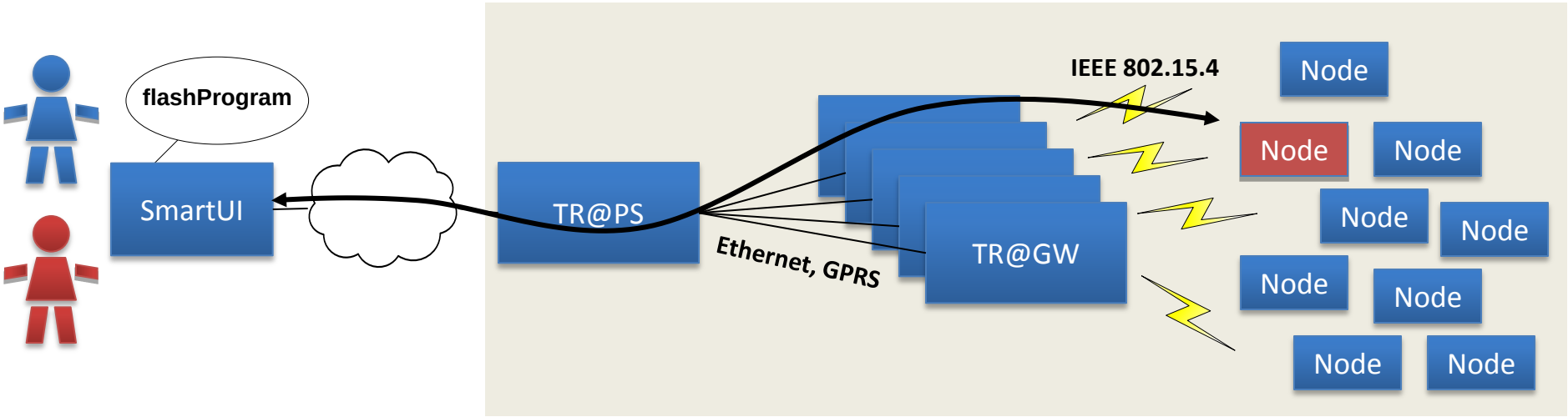
- isNodeAlive()



Administrators
Experimenters

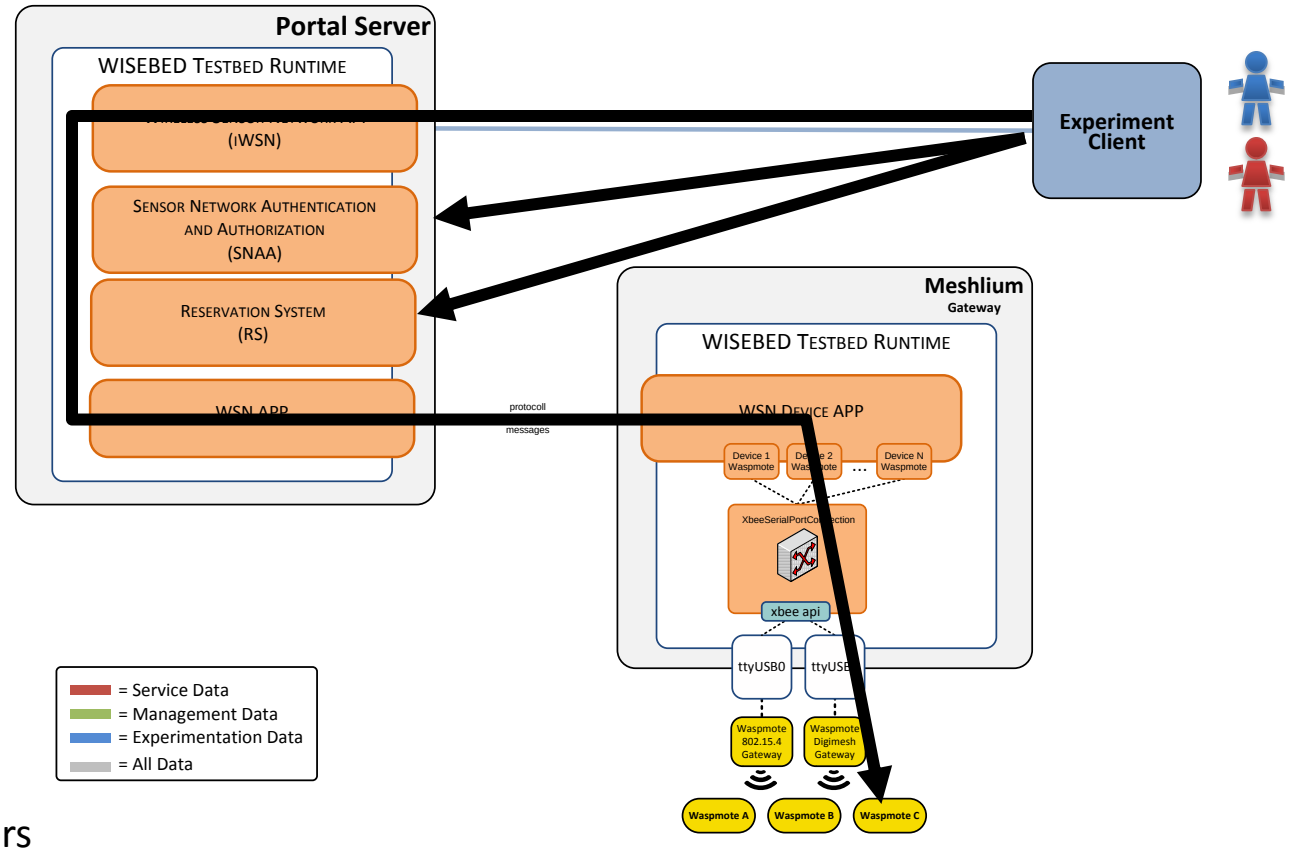
Experimentation script

- Wireless deployment of experiments and updates
  - flashProgram()



Experimentation script

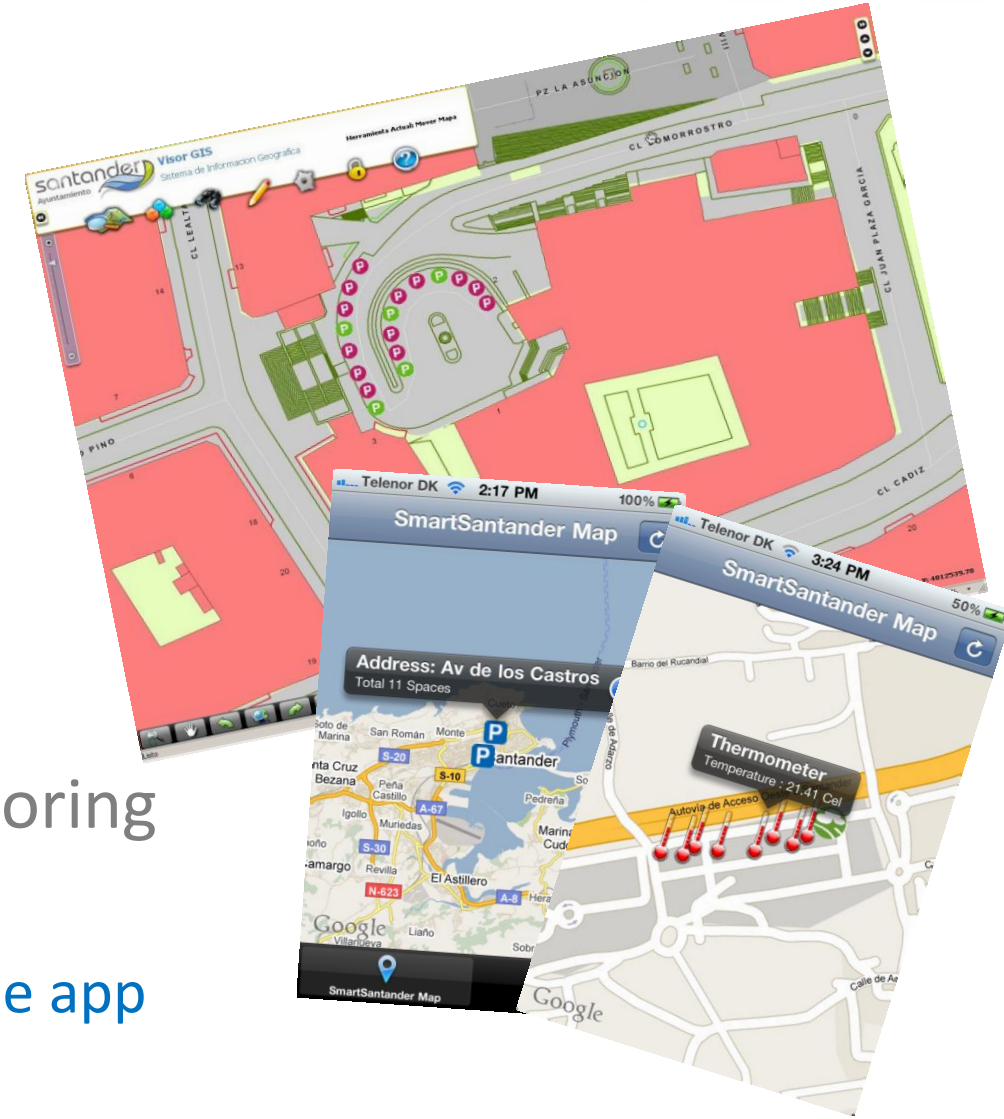
- flashProgram()



Administrators
Experimenters

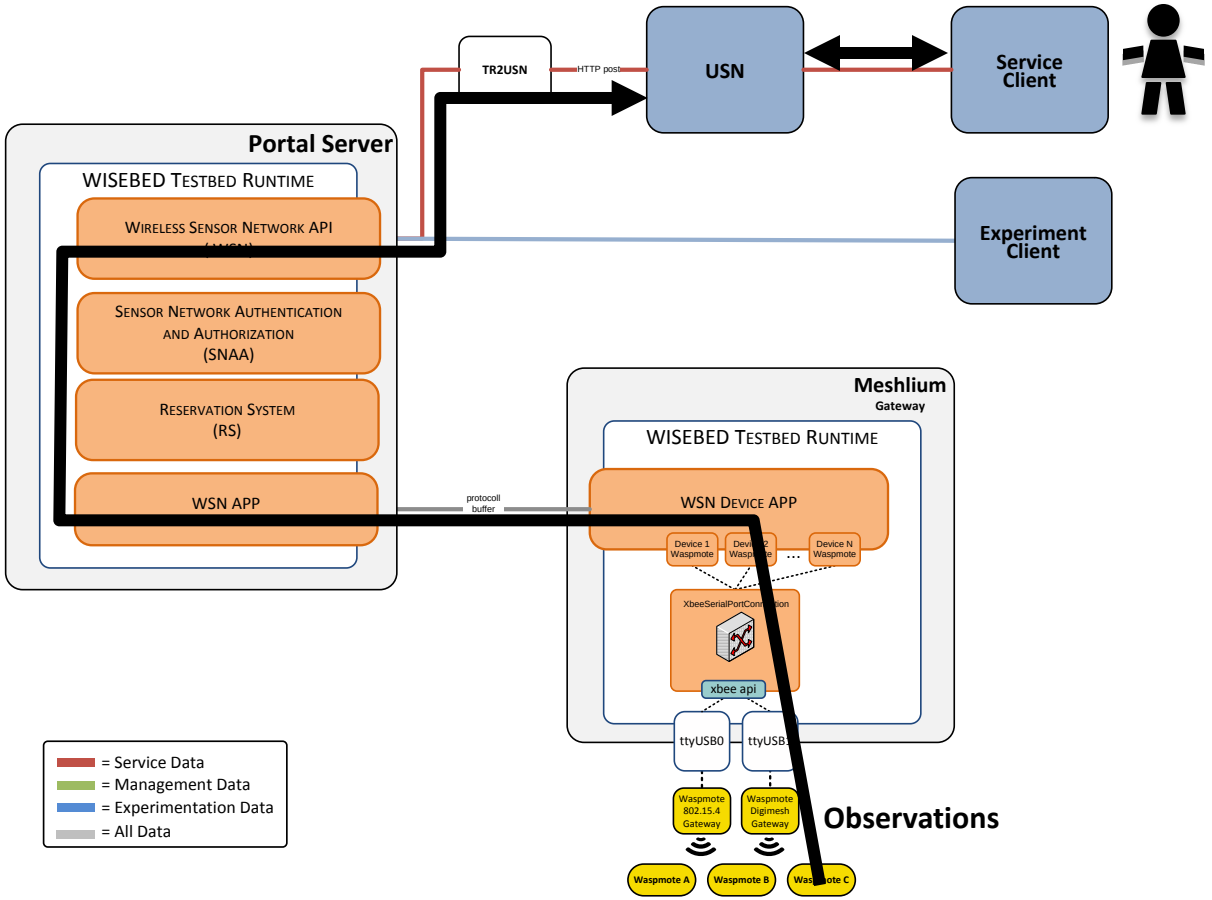
Experimentation script

- Service provision
  - Parking Service
    - Parking iPhone app
    - GIS desktop app
    - Parking Panels
  - Environmental Monitoring Service
    - Env Monitoring iPhone app
    - GIS desktop app



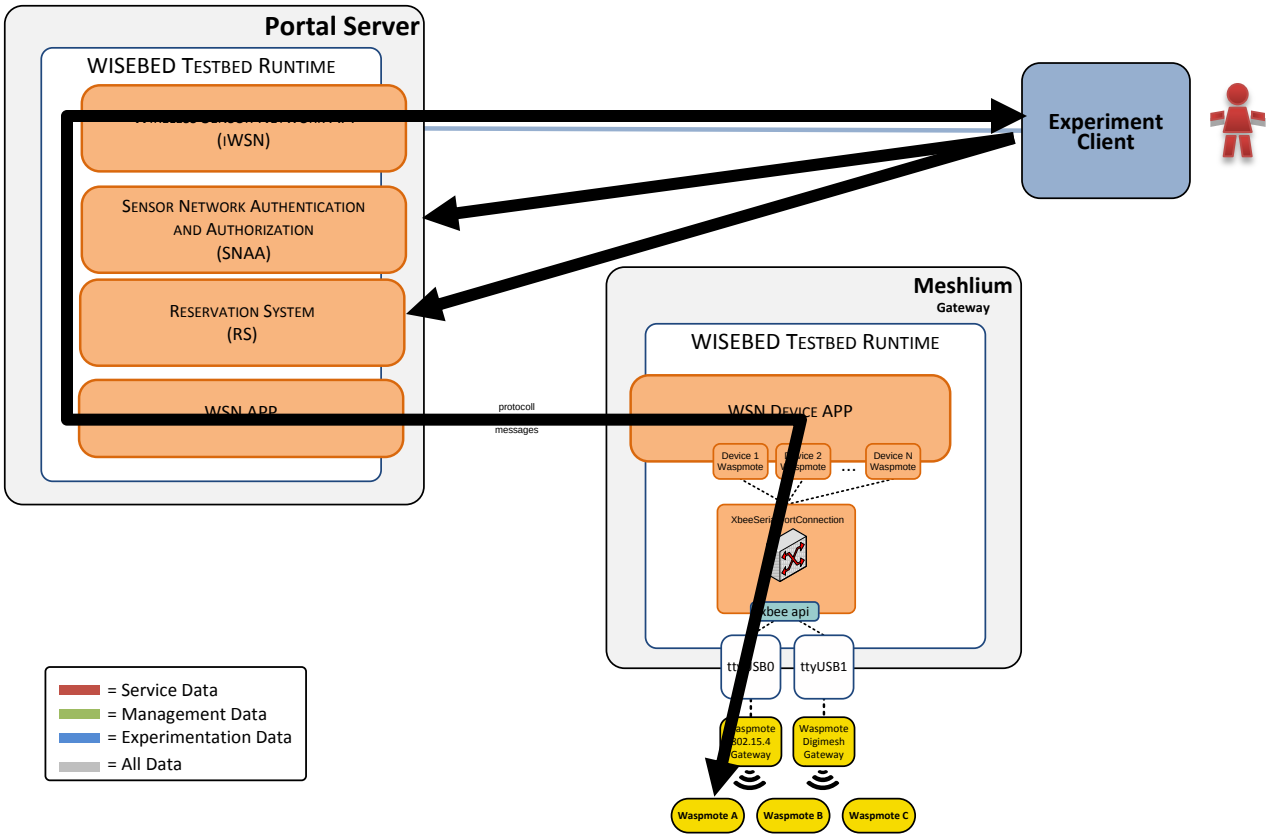
Experimentation script

- Parking and Environmental monitoring observations and measurements



Experimentation script

- Experimentation



Experimenters

1st Open-Call for Experiments

We are looking forward to your feedback and your experiments!