Open Call to select experiments for the FP7 SmartSantander project

Experimenting with the Internet of Things in the context of the city

The SmartSantander project, currently active in the Seventh Framework Programme of the European Community for research and technological development, announces the second Open Call for new project partners to submit proposal for experimentation on the project's test facilities.

Open Call Summary

The SmartSantander project is offering up to 100k EUR funding contribution (per proposal and a maximum of 4-6 proposals) for innovative applications and services, middleware developments as well as protocols and technologies that use the SmartSantander experimental facilities.

The aim is to stimulate, demand and establish a methodology of experimentally driven research as well as expand the service, protocol and technology offering of the platform towards experimentation, but also the general public. The second open call opens 1st October 2012 and will close on 14th November 2012, targeting the Internet of Things and Smart City communities.

The SmartSantander platform is a unique experimental facility as it is deployed in a real city, with citizens using every day the services offered by the platform. It also strives to be the largest public Internet of Things test bed with a deployment of over 12,000 actuators, sensors and tags by the year 2013, with additional sites in Guildford, Belgrade and Lübeck adding another 8,000 sensors.

Call identifier:	SmartSantander-2-Open-Call
Contact email:	opencalls@smartsantander.eu
Call website:	www.smartsantander.eu/opencalls
Call open:	The call will be open for submissions from 1st October 2012
Call deadline:	The call closes on **14th November 2012 at 17h00** (Brussels time)
Expected duration of participation:	January 2013 to June 2013
Maximum funding per experiment:	Up to 100,000 EUR
Maximum budget for call:	785 K€ (EC contribution up to 432,279€)
Number of experiments:	4-6
Number of partners per experiment	1-2 partners (typically)
Proposal submission language:	English
Call objective:	**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**.

We welcome submissions targeting:

- Innovative applications/services in the framework of the smart city supported by IoT technology.
- Middleware developments bridging applications and technologies, allowing a plug and play approach.
- Protocols/technologies for maximising efficiency & sustainability of IoT deployments in the smart city.

Once more the city and its partners welcome you to experiment on it!

Platform Summary

Introduction

The **SmartSantander** project aims at the creation of an experimental test facility for the research and experimentation of architectures, key enabling technologies, services and applications for the Internet of Things in the context of a city (the city of Santander located in the north of Spain). The envisioned facility is conceived as an essential instrument to achieve the European leadership on key enabling technologies for IoT, and to provide the European research community with a one-and-only platform of its characteristics, suitable for large scale experimentation and evaluation of IoT concepts under real-life conditions.

SmartSantander project provides a twofold exploitation opportunity. On the one hand, the research community gets benefit from deploying such a unique infrastructure which allows true field experiments. Researcher will be allowed to reserve the required resources within the whole network and for a determined time period in order to run their experiments. On the other hand, different services fitting citizens' requirements will be deployed. Different from the experiment applications, it will be either the authorities or the service manager/responsible, the ones in charge of determining the cluster of nodes running each service, as well as, the time duration of the aforementioned service.

Facilities Description

The project considers the deployment of 20,000 sensors in Belgrade, Guildford, Lübeck and Santander (12,000). Bellow we provide the details of the facilities which will be available for this second open call. Additional information about the operational facilities is available in the annex of this document as well as in [D1.1] and [D1.2].

Santander summary

The Santander testbed is composed of around 3000 IEEE 802.15.4 devices, 200 GPRS modules and 2000 joint RFID tag/QR code labels deployed both at static locations (streetlights, facades, bus stops) as well as on-board of mobile vehicles (buses, taxis). Over the deployed testbed, several use cases have been implemented:

- Environmental Monitoring. Around 2000 IoT devices installed (mainly at the city centre), at streetlights, facades provide measurements on different environmental parameters, such as temperature, CO, noise, light and car presence).
- Outdoor parking area management. Almost 400 parking sensors (based on ferromagnetic technology), buried under the asphalt have been installed at the main parking areas of the city centre, in order to detect parking sites availability in these zones.
  Deployment for environmental monitoring and outdoor parking area management is shown in the next figure:

Figure 1: Outdoor parking and Environmental Monitoring deployed architecture

Figure 1 shows a screenshot of the deployment in Santander city centre, where parking, temperature, luminosity, CO and noise sensors can be observed. Whole deployment can be accessed in [MAP_SDR].

- Mobile Environmental Monitoring: In order to extend the aforementioned environmental monitoring use case, apart from measuring parameters at static points, devices located at vehicles retrieve environmental parameters associated to determined parts of the city. Sensors are installed in 150 public vehicles, including buses, taxis and police cars.
- Traffic Intensity Monitoring: Around 60 devices located at the main entrances of the city of Santander have been deployed to measure main traffic parameters, such as traffic volumes, road occupancy, vehicle speed or queue length.
- Guidance to free parking lots: Taking information retrieved by the deployed parking sensors, 10 panels located at the main streets' intersections have been installed in order to guide drivers towards the available free parking lots.
- Parks and gardens irrigation: Around 50 devices have been deployed in two green zones of the city, to monitor irrigation-related parameters, such as moisture temperature and humidity, pluviometer, anemometer, in order to make irrigation as efficient as possible.

Description of hardware deployed, software architecture and integration within SmartSantander platform are discussed in detail in the ANNEX I.

- Augmented Reality: Around 2000 RFID tag/QR code labels have been deployed, offering the possibility of "tagging" points of interest in the city, for instance a touristic point of interest, shops and public places such as parks, squares, etc. In a small scale, the service provides the opportunity to distribute information in the urban environment as location based information.
- Participatory Sensing: In this scenario users utilize their mobile phones to send physical sensing information, e.g. GPS coordinates, compass, environmental data such as noise, temperature, etc. This information feeds the SmartSantander platform. Users can also subscribe to services such as

"the pace of the city", where they can get alerts for specific types of events currently occurring in the city. Users can themselves also report the occurrence of such events, which will subsequently be propagated to other users that are subscribed to the respective type of events, etc.

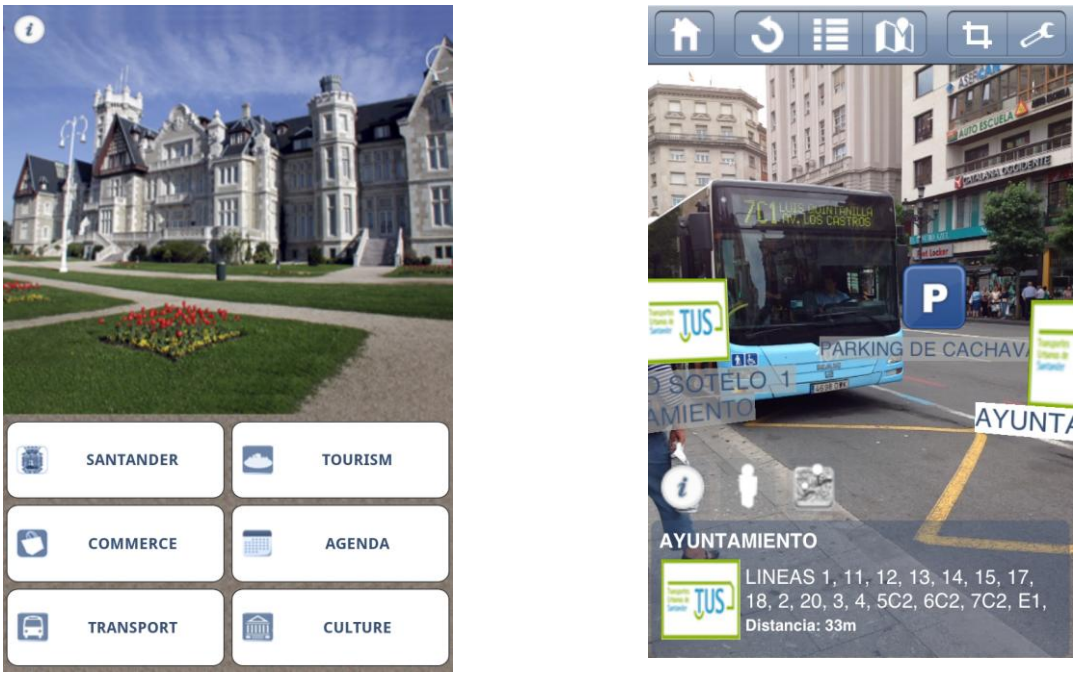

Figure 2: Detail of Augmented Reality application

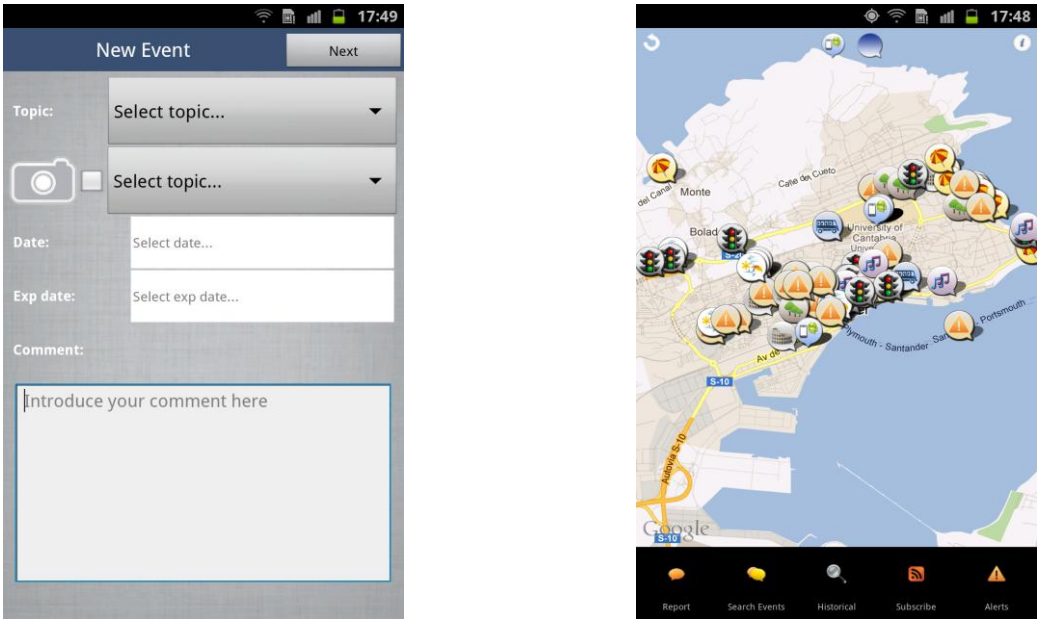

Figure 3: Detail of Participatory Sensing application

Figure 2 and *Figure 3* show some screenshots of the applications developed on Android and IOS for Augmented Reality and Participatory Sensing use cases, respectively.

Apart from service offered by the aforementioned use cases, and in order to cover the twofold approach, experimentation and service provision, pursued by the project; within these use cases researchers are also provided with the possibility of experimenting with the deployed nodes. In this sense, two types of experimentation can be carried out over the facility:

- Native experimentation: Most of the deployed IoT Nodes (those with fewer constraints in terms of battery) can be flashed, as many times as required with different experiments, through OTAP (over-the-air programming) or MOTAP (Multihop OTAP), for nodes more than one hop away from the gateway. In this sense, researchers can test their own experiments, such as routing protocols, data mining techniques or network coding schemes. This experimentation is made available by using an additional IEEE 802.15.4 transceiver, thus isolating data traffic associated to experimentation from the generated by the service provision. *Figure 4* shows a screenshot about the experimentation skills offered by the platform.

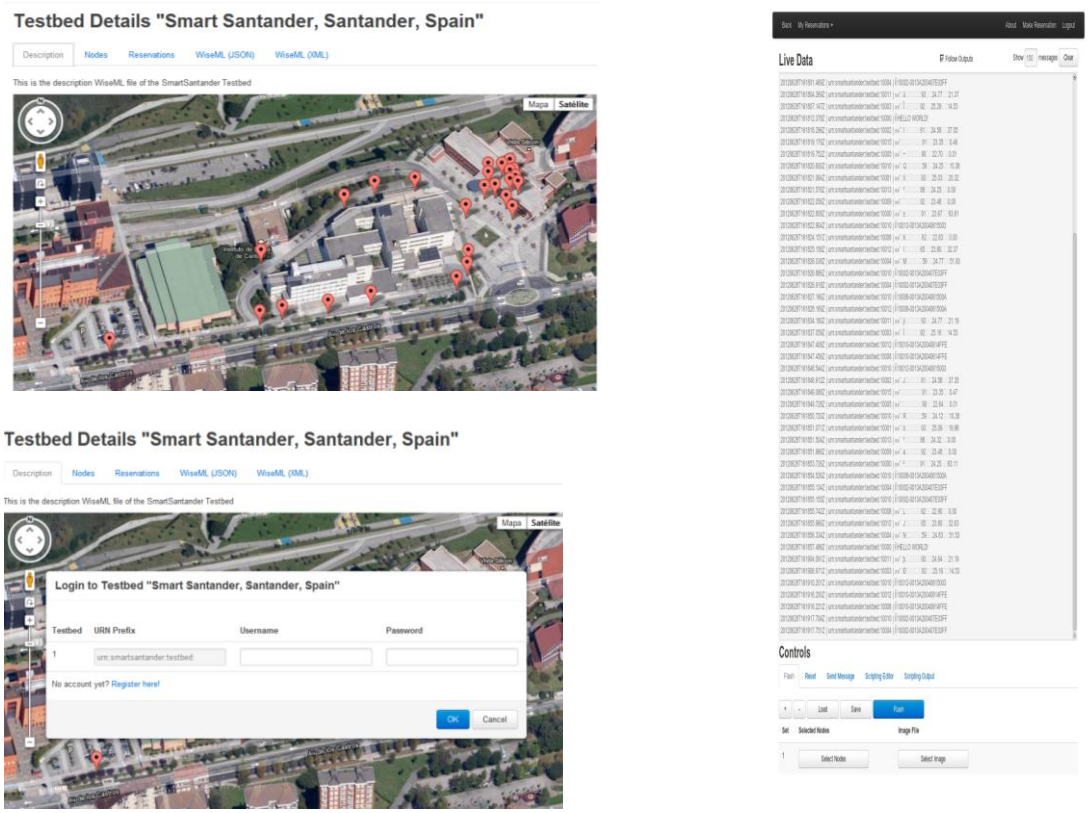

Figure 4: Native experimentation skills

- Experimentation at service level: Some of the deployed nodes, due to battery constraint or memory capacity issues, are not provided with an additional 802.15.4 transceiver and cannot be

flashed over the air to load different experiments. For these cases, data generated by the different services (described in next section) provided by the project, is offered to the researchers for developing new services on top of it. In this sense, development of new added value services, as well as, correlation between information retrieved by different services, could be examples of this type of experimentation.

Apart from experimentation and service provision, network management is needed in order to access deployed nodes to send/receive commands from them, update the firmware running on them, or load experiments if allowed (OTAP). All these functionalities are performed through the Testbed Runtime (TR) module, which in order to manage these new wireless devices, implements a mux/demux functionality as well as the corresponding device drivers. Furthermore, in order to fulfill experiments support, platform management and service provision in a joint way at the node level, it is needed to flash node with a default program (called *golden image*), at network start-up.

Considering the aforementioned hardware and software architecture, Santander testbed allows the service provision and the experimentation in a simultaneous way, as well as network management, i.e. sending commands or flashing nodes. Detailed information about Santander testbed architecture at physical and logical level is included in ANNEX I.

Guildford summary

Currently the Guildford testbed provides a Smart environment, based on an indoor sensor nodes deployment located in the Centre for Communication Systems Research (CCSR). It serves as initial core and experimental micro-cosmos for the envisioned Smart Campus facility.

The IoT node tier consists of 250 freely programmable sensor nodes deployed across all offices of CCSR with various sensing modalities (temperature, light, noise, motion, electricity consumption of attached devices, vibration). The availability of these sensing modalities may vary across some of the nodes. The IoT nodes consist of 200 TelosB based platforms and 50 SunSpots. Other sensor node platforms will be deployed soon in order to achieve additional hardware heterogeneity in the testbed. The nodes' deployment currently stretches over three floors of the building. Figure 5 provides an example of the final sensors deployment at floor 1 and 2 of the CCSR building. Ground floor deployment is still ongoing and the final layout could vary, however the aim to cover all the desks used by CCSR employees will be maintained. Due to the reduced number of spaces assigned to CCSR the ground floor deployment will result as the sparsest among the three floors.

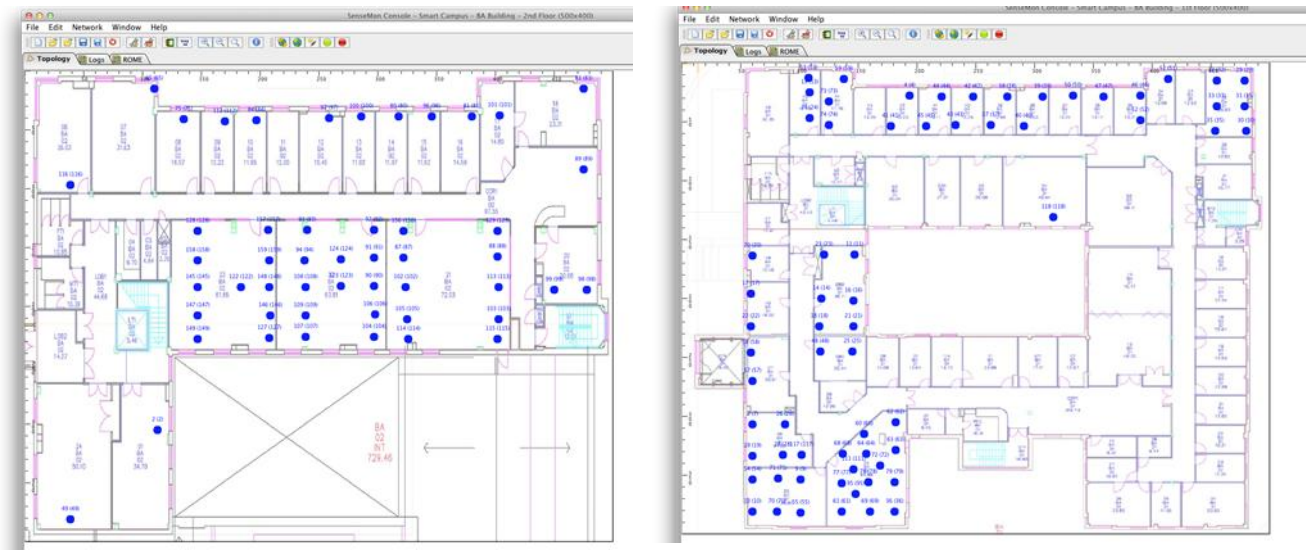

Figure 5: An overview of the current IoT node deployment snapshot on two floors of CCSR.

100 embedded Linux servers (GuruPlug Servers), directly connected to an Ethernet backbone, have been deployed and connected to the sensor nodes for their management. By carefully selecting and configuring sensor nodes to act as sinks at experimentation time, the deployed GWs can offer also the possibility to act at the same time as data GWs realizing a data plane for interconnection of the testbed to the Internet. A server cloud hosts the testbed management servers and allows the on-demand creation of other application servers and data management tools.

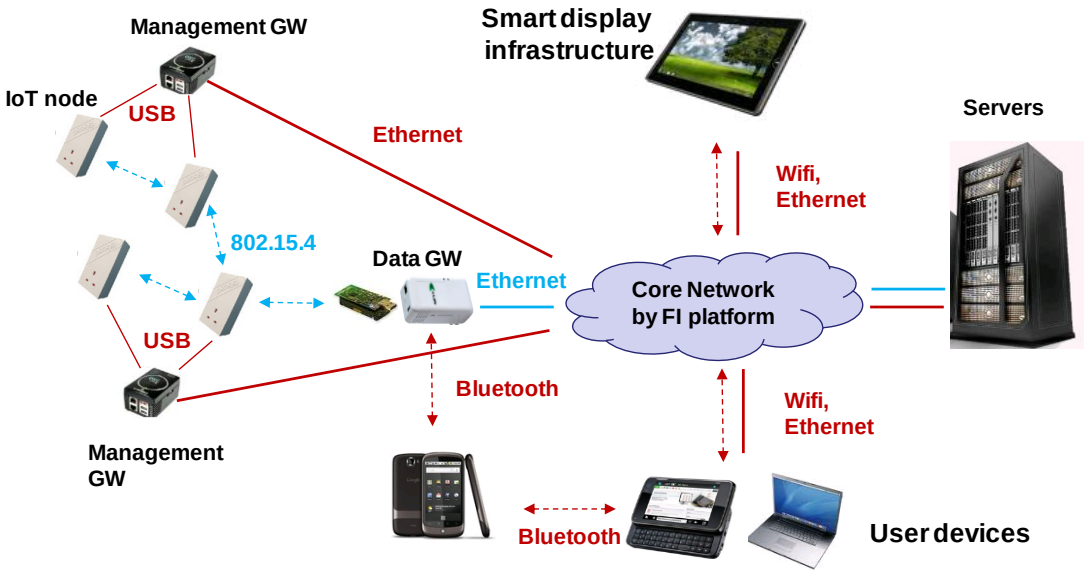

Figure 6: High level network diagram for Guildford Phase 1 deployment

Figure 6 provides an overview of the network architecture of the Phase 1 deployment. All devices of the IoT tier are connected through a gateway tier to a backbone network were application servers and testbed

management servers reside. The data plane of the testbed is realized via wireless links (highlighted in Blue) based on 802.15.4 which can be single/multi-hop between the IoT nodes towards the GW devices. The GWs can act as data GWs, relaying data back to the server using the Ethernet connection provided by the backbone network. An out-of band testbed management and control plane is also realized via USB infrastructure from the IoT nodes to the GW devices, which in turn are connected through an Ethernet backbone towards the testbed management servers. In addition the testbed allows the connection of Smart Displays and end user terminals (laptops, desktops or mobiles) via WiFi and Ethernet towards the internal network, or directly via Bluetooth to the GW devices.

The ratio of GW nodes to IoT nodes is between 1:1 to 1:4, depending on the number of IoT nodes that are deployed in a room and availability of Ethernet ports in the office space for the connection of GWs.

In order to access the testbed and configure and run an experiment a set of tools fully integrated with the Guilford testbed are provided. Examples of these tools are:

- A REST server for accessing the last readings from the different sensors when running some specific collection application;
- A JAVA based GUI called TMON (Testbed MONitor) for exploring the topology by browsing nodes using a semantic description of them and visualizing relation between them such as links and presence of sources of interference (i.e., nearby WiFI access point). After selecting the adequate resource for an experiment, through the TMON GUI the user can also reserve nodes, configure them and automatically run experiments and collect results by providing a first analysis/visualization of them or by storing them in an Experiment Repository;
- An Experiment Repository for experiment results storage/load accessible through a REST interface and working in both modalities as standalone component accessible from custom application or as component already integrated within the TMON GUI.

Lübeck Facility

Lübeck offers a number of different testbeds, all accessible through the already mentioned WISEBED experimental facility features such as the testbed runtime, portal servers, etc.

UZL's major testbed consists of three sensor node hardware types: iSense, TelosB, and Pacemate. The nodes are arranged in clusters. Each cluster has one sensor node of each node type as shown in Figure 7. This testbed consists of roughly 300 stationary sensor nodes organized into 100 clusters. There are two different cluster layouts which differ in the sensor module connected to the iSense node. Half of the iSense sensor nodes are equipped with temperature and light sensors while the remaining nodes are equipped with a passive infrared sensor and accelerometer sensors.

Figure 7: UZL cluster with an iSense, Pacemate and TelosB mote

Figure 8: Roomba with iSense node

All clusters are connected to a total of 35 Acer Aspire One netbooks forming the backbone of the testbed connected to the Internet. The sensor nodes are connected to the netbooks via USB. The netbooks are connected to the Internet over 802.11g Wi-Fi using a testbed-private ESSID and enterprise WPA2 encryption. This backbone enables the user to program or reset the sensor nodes without the need of an additional OTAP (Over-the-Air-Programming) protocol. The fixed network can be extended by mobile nodes (Roomba cleaner robots (see Figure 8) with attached iSense sensor nodes and with Lego Mindstorms).

Figure 9: An outdoor node

Figure 10: Outline of the outdor testbed

In addition to the indoor and mobile nodes, UZL has deployed a number of outdoor, solar-powered iSense sensor nodes (see Figure 9). They feature a 6000mAh rechargeable battery pack, an iSense Solar Power Harvesting System which recharges the battery during sunlight times and provides energy from the battery otherwise. It also features an infrared sensor capable of detecting movement. Currently, approximately 35 nodes are deployed in the garden of the University of Lübeck's manor house (see Figure 10). Please note that, due to the wireless and solar-powered nature of the outdoor testbed, sustaining operation at all times is very time-consuming and expensive. Therefore, the testbed is not constantly available for experimentation.

Pancevo summary

The EkoBus system deployed in the city of Pancevo is made available for experimentation on IoT data level. The system utilizes public transportation vehicles in the city of Pancevo to monitor a set of environmental parameters (CO, CO2, NO2, temperature, humidity) over a large area as well as to provide additional

information for the end-user like the location of the buses and estimated arrival times to bus stops. As the system is in commercial use, a replica of the system is made available for experimentation. The replica is connected to the live system and is being updated continuously. All data generated by the live system is immediately made available to the experimenters.

An overview of the EkoBus architecture

Every IoT node (sensor) in the system is described by its set of capabilities (characteristics, parameters, availability, etc), which are published and stored in the Resource Directory (RD). The Resource Directory stores dynamic information about all available resources in the system at a given time, so that they are available to the end users (applications). Resources make measurements and periodically send the results to the server application for further analysis and database storage. Web and Android application collect information from the resources and perform their visualization (location of the vehicles and atmospheric measurements). It is also possible to request information about the arrival time of the next bus on a certain line to a certain bus stop via SMS.

Analysis of the stored data is used for various traffic calculation and prediction. Accordingly, additional information is available: **Static data**: geo locations and names of the stations, geo locations of curves and semaphores on the bus route, bus timetables, average time that bus spends at the specific station, initial average time of bus travel between two consecutive stations; **Dynamic data**: calculated average time of bus travel between two consecutive stations for the different part of day and week.

Therefore methods and tools for experimentations are:

- *Web interface* – A web application is responsible for displaying bus locations, bus arrival time information and data received from environment and gas sensors in a web browser, and can be extended in order to provide public survey i.e. feedback from the end user; or additional services. The application interacts with the backend system using a set of web services. The data is obtained in XML format.
- *Mobile application* is similar to the web application, providing the visualization of the current location of all vehicles in the system as well as the measurements for the end-user with mobile phone.
- *SMS* – end-user can query the system using SMS
- *Database and offline analyzer* – database data that can be used for new offline traffic analysis procedures. Module for offline data analysis is responsible for updating the data in database with statistics obtained during previous measurement period. This information is used by the Traffic management agency of the City of Pancevo to optimize the public transport system.

The following is offered to the experimenters:

- Access to historical data stored in the database. It is made available via dedicated Resource End Points representing individual nodes installed on the buses, utilizing a simple REST interface that

allows extraction of the measurements for a given period. All IoT nodes are accessible via SmartSantander framework.

- Direct access to IoT nodes is not available. It is also not possible to change the code or configuration of the IoT nodes.
- 60 devices are deployed.

Additional information

Details about the test bed Architecture and the Regulations for the use of the experimental facility can be found in D.1.1 "First Cycle Architecture Specification" and D.5.3 "Regulations for use of experimental facility" available for download from the SmartSantander website: http://smartsantander.eu/index.php/deliverables.

Target Outcomes

The Open Call aims to attract exciting experiments and high impact scientific evaluations that make use of the unique features provided by the SmartSantander facility. In this context, the proposals should address at least one of the following three areas of experimentation:

1) Innovative applications and services for smart cities and built environment

The project is seeking innovative applications and services running in the framework of the smart city paradigm supported by Internet of Things technology. In Santander as well as at other sites, a significant number of multi-modal sensor nodes are now available to provide a large variety of real world data streams. These data streams can be exploited by novel smart services and applications with the goal to provide added value towards citizens and the city authorities.

The services and application must demonstrate a clear benefit towards the stakeholders of the facility (e.g city, university or citizens), which goes well beyond the benefit of the proposing parties. Therefore proposed application and services should not only strive to demonstrate the feasibility of a service or application, but also evaluate the end user acceptance thereof as part of the experimentation and its commercial viability.

In its current form the infrastructure natively supports application domains such as transportation, energy and environment. However extensions to the infrastructure can be proposed as a part of the work, in order to make it suitable to realize experimentation for other high impact application domains. Extensions could also include the provision of advanced mechanism for (semi-) automatic capturing of end user feedback or quality of experience during experimentation.

2) Internet of Things communication protocols and technologies

The first phase of the SmartSantander facility development provides the ability to test Internet of Things related protocols and technologies on a larger scale in realistic deployment environments. Proposed experiments should go beyond traditional island associated to specific wireless sensor network research (e.g. Intranet of Things) and address the evaluation of key solutions and protocol building blocks that contribute towards the realization of a globally networked Internet of Things. In particular this includes the experimental evaluation of one or more of the following aspects:

- New approaches and architectural paradigms that support interoperability of resource constraint Internet of Things devices, taking into consideration not only different layers of the communication stack but also the data layer of an Internet of Things well. This could include the evaluation of evolutionary protocol stacks supporting the RESTful interactions of the existing Internet paradigm, but also more disruptive approaches that explore data centricity and information centric interactions on larger scale across a diverse set of IoT deployments.
- Studies that provide a more detailed understanding of the properties and particularities of large scale Internet of Things deployments, leading to new insights in the form of design principles and guidelines that can be applied to a variety of Internet of Things protocols and solutions.
- Mechanisms and techniques that allow the exploitation of opportunistic availability of (mobile) Internet of Things devices for computing and communication tasks.
- Key enabling building blocks of an Internet of Things such as resolution infrastructures, supporting the scalable lookup and discovery of heterogeneous Internet of Things resources and their relationships with real word entities.
- Mechanisms for more efficient and reliable data dissemination in larger scale resource constraint environments. Examples are novel rateless coding schemes, such as Luby Transform codes [LT codes], to reduce the overhead when reprogramming over-the-air several clusters of nodes. In the same direction, the testbed offers a unique opportunity for experimenting with network coding techniques aiming at reaching the multicast throughput capacity on top of the IoT available infrastructure.

In its current form the infrastructure provides experimentation support with mainly static nodes, offering no real physical mobility of IoT experimentation nodes. Experimenters are welcome to propose extensions to the existing infrastructure to enable experimentation with mobile nodes in the facility as part of their experiment.

3) Internet of Things middleware solutions

In order to make (large volumes of) Internet of Things generated data easily accessible for services across multiple application domains efficient middleware solutions are required. Such middleware solutions should contribute to an efficient management and processing of IoT generated data, in order to allow an easy integration of these IoT endpoints into the service layer of the Internet and algorithms contribute towards and increased real world awareness of software based systems. Proposals for experiments should address at least one of the following aspects:

- Platforms and mechanisms that allow a 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, extending concepts of the semantic web, such as linked data to the resource constraint devices of the IoT.
- Algorithms for real world awareness, contributing to an increased machine understanding of complex processes and system behavior 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 Impact

Project submission to topic 1) need to demonstrate a clear benefit and value to the targeted service end users, such as city and citizens or the university and its employees/students.

Project submission to topic 2) and 3) must have the potential to lead to high quality scientific outcomes. Proposers must demonstrate an excellent scientific/technical track-record in the proposed research or application area. Supporting evidence of how to achieve the above expected impact, e.g. adequate dissemination plan should be provided.

In addition proposal submission should demonstrate at least one of the below listed expected impacts:

- Improving / extending the existing capabilities of the SmartSantander experimental test facility, by bringing in complementary expertise to the consortium by providing one of the above outlined extensions to the facility. This includes expertise in large scale experimentation on mobile sensing platforms or participatory sensing or tools and methodologies for evaluations of end-user acceptance and quality of experience.
- Stress-testing the capabilities of the current facility by challenging experimentation requirements in order to improve and mature the existing experimentation environment.

Who can participate

The profile of organisations includes both public and private R&D organisations with expertise in the fields of "Smart City" and "Internet of Things" that need to run experiments to further test, consolidate or optimise developments and research on Internet of Things and Smart City technologies.

The rules of participation are the same as for other FP7 project. In summary:

- Any legal entity established in a Member State or an FP7 Associated country[1] (including the European Commission's Joint Research Centre), or created under Community law (e.g. a European Economic Interest Grouping),
- Any international European interest organisation,
- Any legal entity established in an FP7 International Cooperation Partner Country (ICPC). A complete list of these countries is given in annex 1 of the ICT Workprogramme[2], but in principle it includes the developing countries of Africa, Asia and Latin America, as well as those European countries which are not already Member states or associated countries.
- Organisations from certain other countries may also receive a Community financial contribution, as defined in the Rules of Participation in FP7.

Existing SmartSantander partners can not apply for the SmartSantander Open call.

[1] The FP7 Associated countries are Albania, Bosnia and Herzegovina, Croatia, FYR Macedonia, Iceland, Israel, Liechtenstein, Montenegro, Norway, Serbia, Switzerland, Turkey and Faraoe Island.

[2] Obtainable at http://cordis.europa.eu/fp7/ict/

Full details of the Commission's funding arrangements can be found in "Guide to Financial Issues" at: http://cordis.europa.eu/fp7/find-doc_en.html

We foresee to have typically one or two participant organisations per experiment. The activities to be carried out in the experiment related to this call are the following:

- Design the experiment and explain the motivation.
- Plan and deploy the concrete tests of the overall experiment.
- Define the metrics and evaluation process of the experiment.
- Prepare a show case of the experiment that can be use for dissemination purposes.
- Report the necessary effort and costs according to FP7 rules and management practices requested by the Coordinator.

The duration of a proposed experiment should is set between January 2012 to June 2012.

The SmartSantander "Guide for Applicants", http://www.smartsantander.eu/opencalls , contains more detail about:

- Funding of Participation
- How to prepare and submit the proposal
- Proposal evaluation and selection
- Support for proposers including help desk, national contact points and intellectual property rights

Smart Santander Background Information

Project contract number:	257992
Project acronym:	SmartSantander
Instrument type:	Integrated Project
Challenge 1:	Pervasive and Trustworthy Network and Service Infrastructures
Thematic priority:	Future Internet experimental facility and experimentally driven research
Objective and Call ID:	ICT-2009.1.6 / FP7-ICT-2009-5
Project Coordinator:	José Manuel Hernández-Muñoz, Telefonica I+D
Project website:	http://www.smartsantander.eu/

ANNEX I: SmartSantander Experimental Test Facilities

Detailed information about the architecture can be found in [D1.1] and [D1.2].

The SmartSantander testbed is split in to four subsystems:

- Authentication, Authorization and Accounting (AAA) subsystem
- Testbed management subsystem
- Experimental support subsystem
- Application support subsystem

These testbed functions will operate across a set of different devices providing different characteristics and capabilities. In particular the involved devices are:

- IoT nodes
- Gateway nodes
- Testbed server nodes

The interaction among these subsystems and devices is shown in *Figure 11*.

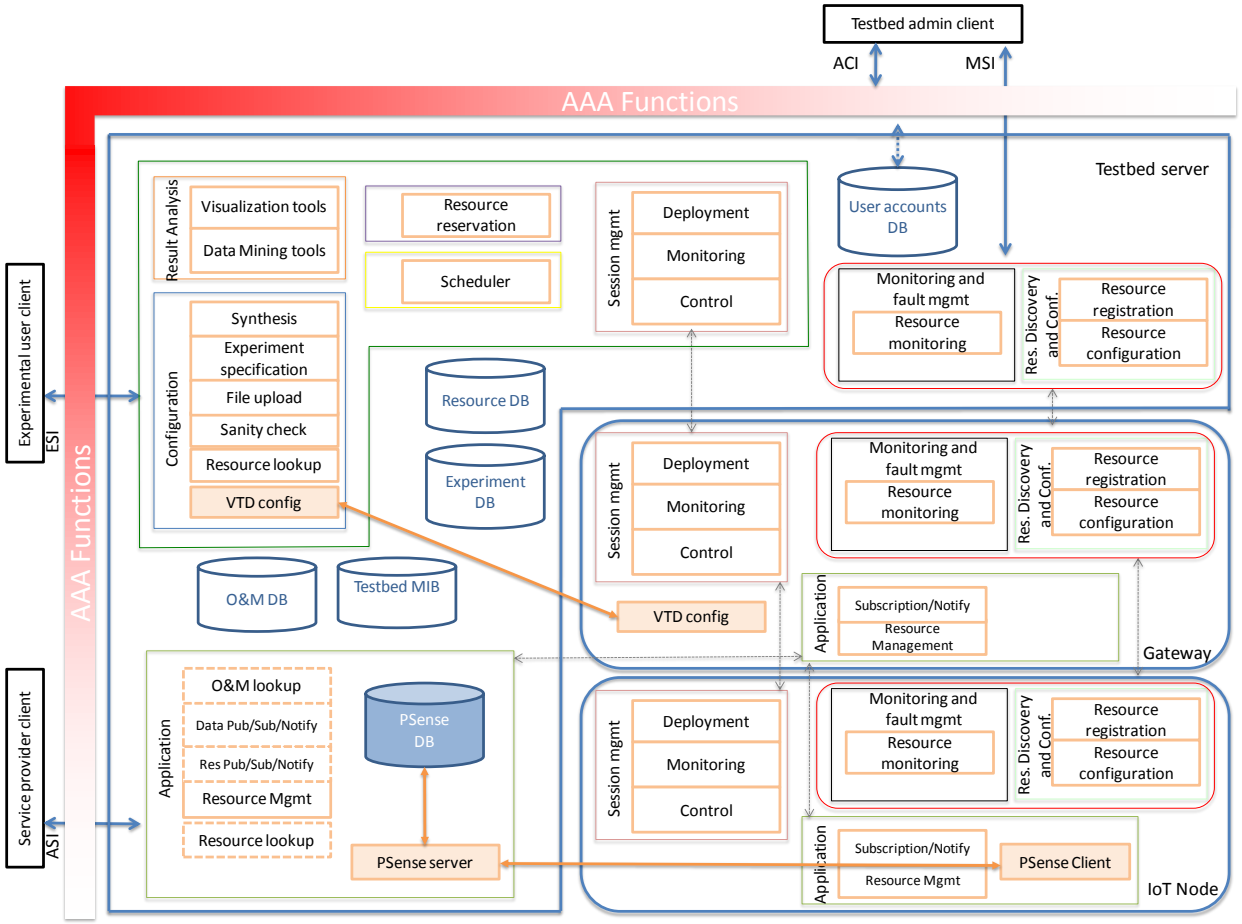

Figure 11: Overview of SmartSantander architecture

The role of each considered subsystem and interface is briefly summarized in the following:

Access control interface (ACI): Authentication, Authorization and Accounting (AAA) subsystem is meant to ensure that only authorized actions are performed on WSN testbeds. Individuals accessing the testbed must be identified and authenticated, and their role must be identified. Authentication should be possible also in case of federated testbeds, where an experimenter/user belonging to a different research organization will have the possibility to carry out experiments in any testbed belonging to the federation.

Experimental support interface (ESI): Experimental support subsystem (ESS) provides the required functions for reserving nodes, configuring and deploying experiments, running them and collects and analyzes the produced results. The Configuration Management module addresses the issue of configuring resources and experiments. In order to reserve one or multiple resources, the Resource Reservation is in charge of updating the state of each resource from free to busy in a proper database, namely the Resource DB. At the same time, the Scheduler provides the way for interacting with the Experiment DB in order to consult or change experiments scheduling. Data Mining and Visualization tools are also envisioned to provide functionalities concerning analysis and visualization of the network status or of the data, measurements of conducted experiments either in real time or after execution has ended. Finally, a Session Management module keeps track of the interactions between the testbed user and the SmartSantander facility, providing the necessary control endpoint through which the user may manipulate its experiment session.

Management support interface (MSI): Management support subsystem (MSS) provides the needed functionalities for adding/removing and configuring the resources composing the testbed and monitoring their status. The resource and testbed monitoring modules are also in charge of monitoring resources availability and type.

Application support interface (ASI): Application support subsystem (ASS) is intended to provide the basic functionalities that can facilitate the development of services either for experimentation or final service provisioning. The Application subsystem should also to provide the possibility for lookup for specific resource or observation and measurement sorted in the corresponding databases (such as Resource DB and O&M DB) maintained by this subsystem.

Santander facility

Based on the logical testbed architecture summarized above, one of the first achievements of the SmartSantander project has been to provide a careful integration of components coming from different existing projects (namely WISEBED [WISEBED], SENSEI [SENSEI] and TELCO 2.0 [TELCO]) in order to fulfil the described requirements. A graphical representation of the provided integration and the involved components for all the aforementioned use cases is shown in *Figure 12*.

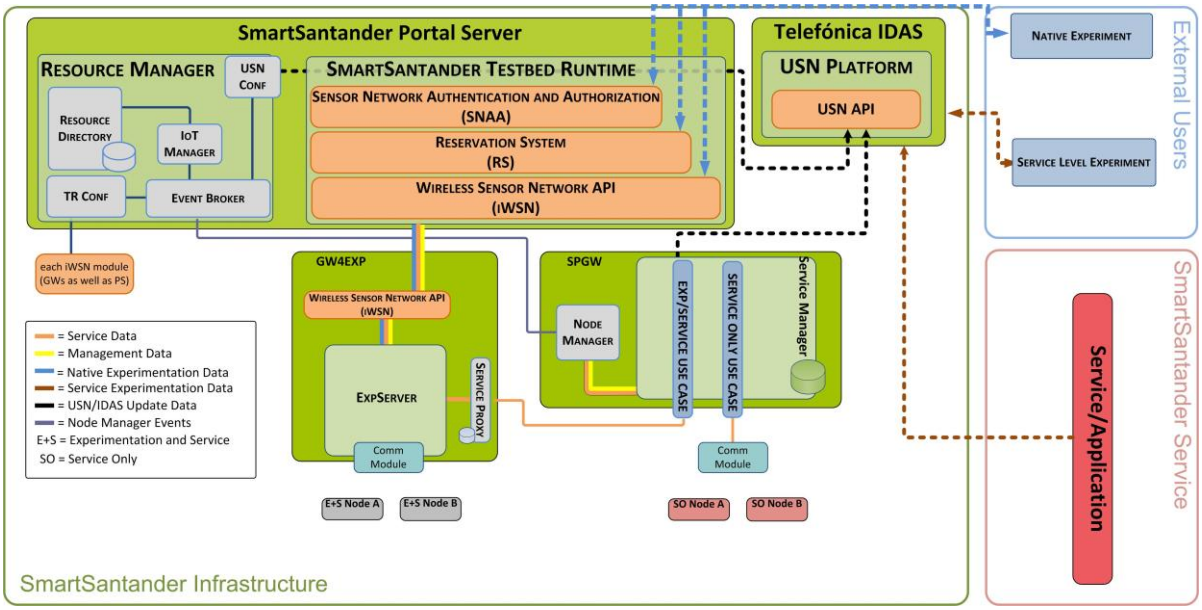

Figure 12: SmartSantander logical architecture

In *Figure 12*, they can be identified the main components composing the SmartSantander architecture: Portal Server, Service Provision GW (SPGW), GW for Experimentation (GW4EXP) and IoT nodes.

The Portal Server represents the access point to the SmartSantander facility for Administrators, Services and Experiment Users. It hosts an adapted version of the WISEBED Testbed Runtime components, including Sensor Network Authentication and Authorization (SNAA), Reservation System (RS) and iWSN API/WSN APP implementation. The SNAA component offers the basic functions for access control through Shibboleth-based authentication and authorization. Therefore, at this stage, no account-based access control and other accounting functions are provided. The Resource Reservation system (RS API from Testbed Runtime) is instead used for making, querying and editing reservations of IoT nodes and supports a number of solutions for the persistence of these reservations, such as in-memory persistence, Google Calendar persistence and database persistence. The iWSN API represents the back-end implementation of the set of functionalities required for interacting with the IoT nodes with commands such as reset, reprogramming, checking if a node is alive, adding/removing virtual links and many others. The iWSN API provides also the implementation of a channel for exchanging debug and control message between the Portal Server and the corresponding GW for experimentation (GW4EXP), as well as with IoT nodes. This type of gateway is only needed for use cases and IoT nodes with capacity for running native experimentation. The communication is achieved by exploiting the protocol buffer message scheme. The iWSN API counterpart on the GW4EXP is indeed extended with a new component, named WSN Device App, that allows for a [1:N] message exchanges between the IoT node directly connected to the GW (by means of a wired USB/serial connection) and purely wireless nodes, as per in the SmartSantander architecture. This new module permits to overcome the limit of the previous iWSN API implementation that assumed a 1:1 [sensor device, serial port] pairing.

Overcoming a previous limitation of the initial Testbed Runtime implementation, relying on statically maintained configuration files (to define the network topology and to correctly forward messages to and from nodes), that need to be manually updated and distributed to all nodes where the Testbed Runtime runs, the new architecture provides also features for automating this process within the Resource Manager module. In this sense, information sent from the different SPGW (Service Provision GW) through the corresponding NodeManager module is received by the EventBroker, which interacts with IoT Manager, TR Configurator and USN Configurator. IoT Manager is in charge of updating the Resource Directory (inherited from SENSEI project), chosen for storing the required configuration parameters. The TRConfigurator and USNConfigurator, have then been developed in order to automatically complete the reconfiguration process of the new resources. An adaptation of the RD has been realized in order to provide a notification mechanism to the components that subscribe it for, when a new resource is added. In this way, the TRConfigurator and the USNConfigurator can receive notifications when respectively; a new experimental or service node is added to the RD. Upon receiving a new notification and after fetching the required configuration information, the TRConfigurator and the USNConfigurator can build the needed configuration files and distribute them to the respectively controlled subsystem, namely the SmartSantander Tesbed Runtime (inherited from WISEBED project) and the USN (data platform provided by Telefonica IDAS). This distribution is achieved by means of a secure ftp connection. By means of an Admin GUI, an administrator can configure the addition of a new resource (either a new experimental or service node), before physically connecting it to the testbed and powering it up.

In order to run an experiment, by means of a Native Experiment Client, the experimenters are allowed to access to the Portal Server, using an authentication and authorization mechanism (provided by the SNAA feature), and to reserve resources (using the Reservation System) on which they can perform experiments configuring them through the iWSN API, thus accessing to the peer iWSN API module within the corresponding Experimentation GW. It must be taken into consideration that not all developed use cases allow native experimentation under their deployed nodes.

In order to allow the user to access services provided data, a Service Level Experiment Client has been developed. Through it, the user can access to the USN component providing a number of useful functions for the development of IoT applications and services ranging from sensor discovery, observation storage, publish-subscribe-notify to a trigger mechanism for the remote execution of tasks on IoT nodes and actuators. The service data generated by all use cases running under the SmartSantander facility can be then available to the USN system pushing them through an HTTP post method and then to the corresponding user.

Finally, within the SmartSantander project, applications have been also developed using information provided by deployed nodes (through the USN) and generating the corresponding service for the user.

Use Cases

In this section, the eight use cases developed within the city of Santander will be described in a detailed way, indicating how they fulfill the service-experimentation duality prosecuted by the project,at the same time as correct network management is also assured.

Environmental Monitoring, Outdoor parking area management, Parks and gardens irrigation

These three use cases present the same HW technology for the deployment and implement an specific SW architecture in order to manage service-experimentation duality as well as network management in a simultaneous way.

An overview of the architecture

These three use cases share the same architecture, conceived as a 3-tiered approach and defined next:

1. **IoT node**: Responsible for sensing the corresponding parameter (temperature, CO, noise, light, car presence, soil temperature, soil humidity). The majority of them are integrated in the repeaters, whilst the others stand alone communicating wirelessly with the corresponding repeaters (it is the case for the parking sensor buried under the asphalt). For these devices, due to the impossibility of powering them with electricity, they must be fed with batteries.
2. **Repeaters**: These nodes are high-rise placed in street lights, semaphores, information panels, etc, in order to behave as forwarding nodes to transmit all the information associated to the different measured parameters. The communication between repeaters and IoT nodes performs through 802.15.4 protocol.
3. **Gateways**: Both IoT nodes and repeaters, are configured to send all the information (through 802.15.4 protocol), experiment-driven as well as service provision and network management to the gateway. Once information is received by this node, it can either store it in a database which can be placed in a web server to be directly accessed from internet, or send it to another machine (central server), through the different interfaces provided by it (WiFi, GPRS/UMTS or ethernet).

Hardware deployment

Taking into account the twofold approach, experimentation and service provision, prosecuted by the project, it is needed to define an infrastructure that allows executing both experimentation and user-addressed services in a joint manner, thus providing flexibility for researchers to try their applications on the testbed, at the same time that a service addressed to ease and fulfill citizens' requirements is running. To handle this execution concurrency in an efficient way, a solution based on hardware independence is posed. This solution, provided by the Spanish company Libelium, consists of nodes implementing two different physical interfaces, as shown in the *Figure13*.

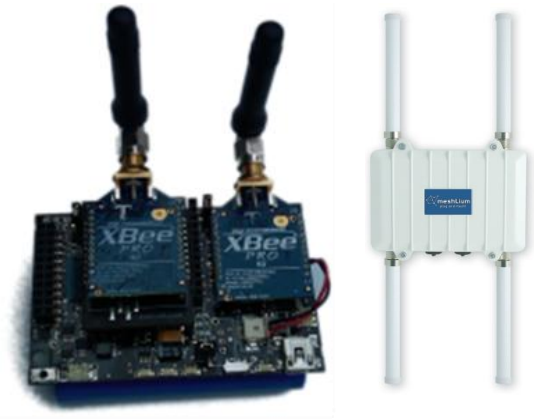

Figure 13: Deployed IoT node and gateway

The node depicted in *Figure 13* is composed of the following parts:

- **Main board**: This board (called Waspmote) is in charge of processing and memory issues, providing a set of interfaces for attaching different types of sensors (both analogue and digital), as well as to plug several radio modules to communicate with other nodes. The Waspmote comes with with a ATmega1281 microcontroller, and several types of memory, 8KB SRAM, 4KB EEPROM, 128KB FLASH and an extra storing SD memory with 2GB capacity. On the other hand, 7 analogue and 8 digital interfaces are available for external sensor connection, as well as 1 PWM, 2UART, 1 I2C and 1 USB interfaces for attaching different communication modules. All the development tools (libraries, API's, etc.) provided by Libelium are based on a pseudo-wiring solution which aims to promote the simplicity of the functioning of the micro-processor based on events and loops. Attached to the main board, they are placed the sensor boards with the corresponding sensing capabilities, such as temperature, luminosity, noise, parking, CO, soil temperature.
- **Two XBee-PRO radio modules**: Both modules manufactured by Digi company, run over 2.4 GHz frequency. One of the modules implements 802.15.4 protocol in a native way, and the other one runs 802.15.4 protocol modified with a proprietary routing protocol called Digimesh. This is a proprietary peer-to-peer networking topology protocol for use in wireless end-point connectivity solutions, allowing addressing in a simple way.

The three components composing the infrastructure: IoT nodes, repeaters and gateways, are equipped of the aforementioned component in order to guarantee the service provision as well as the experimentation over the same node in a simultaneous and independent way. In the case of the gateway node, as it is intended to gather and facilitate all the information taken from the WSN, either to external networks (internet) or application level services, it needs to implement high memory/processor capacity and added communication skills. To fulfill all these requirements, another device (called Meshlium), also manufactured by Libelium, with higher capacity in terms of processor (500MHz) and memory (256MB RAM and up to 32GB hard disk) is utilized. Regarding to its communication skills, apart from the two Xbee radio modules for communication with the deployed nodes, also WiFI, GPRS, Bluetooth and Ethernet interfaces are provided.

Network management, service provision and experimentation support

Considering the size of the deployed network, it is of utmost importance to be able to continuously monitor and manage such a large infrastructure in the most efficient way. For this purpose, and taking into account the aforementioned experiment-service duality and the two radio modules availability, the way IoT Nodes interact with the rest of the SmartSantander system is as follows:

1. SmartSantander experimental facility needs to be managed in a wireless way which basically involves wireless transmission/reception of commands to/from all nodes and node reflashing over the air. For this purpose, the Digimesh radio module provides the routing protocol for communication between nodes and gateway. In this sense, it will be possible to manage the IoT Nodes from the gateway by sending the appropriate commands and receiving the corresponding responses as they are issued by the IoT Nodes. On the other hand, IoT Nodes will be flashed also from the gateway as many times as required, through OTAP (over-the-air programming) or MOTAP (Multihop OTAP), for nodes more than one hop away from the gateway.
2. All the information derived from both service provider and city service use cases is retrieved by the deployed nodes to the gateway, which is the entrance towards the SmartSantander system, through WiFI, GPRS, Ethernet. In order to send all this data in a reliable and transparent way, the Digimesh-enabled network is used.
3. Regarding to experimentation use cases, researchers will flash the nodes with the corresponding programs, through (M)OTAP using the Digimesh interface. However, once the code is loaded in the node, all the data regarding to the experiment will be transmitted and received through the 802.15.4 native module.

In this respect, it is guaranteed that both management and service traffics are transmitted in a physically independent way from the experimentation information, thus obtaining interesting results:

- The provided service will never be interfered nor interrupted by experiments, thus avoiding the disruption of this service because of a misuse of the network by some experiment.
- The results retrieved from the experiment might be assumed as if the testbed were only for experimentation purposes, as there is no interfering traffic within the network, but only the one associated to the corresponding experiment.
- The management of the network is more reliable as all traffic running on the Digimesh interface is predictable (all services are installed at start-up); so no external traffic will affect the communications. In this sense, nodes will be provided with a default program (called “golden image”), which will carry out the functionality associated to the corresponding service, as well as all the management issues needed for the correct network operation. This image will be loaded in the nodes at the network start-up, and re-flashed when a node is restored to its default state.

Logical Architecture

Once defined the hardware deployment, as well as the service/experimentation duality prosecuted by the project, it is needed to fit this infrastructure within the logical architecture provided by the project.

In this sense, and as it was described in the introductory section, communication with IoT Nodes and repeaters from the gateway nodes, is performed through the Testbed Runtime (TR). The TR creates an overlay network for easy node addressing and message exchange with locally attached nodes independent from the actual underlying network connections. It performs message forwarding and offers communication primitives that are used for the control and management of experiments and the WSN itself. The TR design defines the Connection Services which handle the messages exchanged with the IoT Nodes. The architecture of the TR implies that there is one connection per IoT Node in the testbed which is accessed through an exclusive connector.

This approach is flawed in a wireless context. If wireless nodes are used, all the IoT Nodes are connected with the GW (Meshlium) running the TR through the same network interface (either through 802.15.4 or Digimesh interface). Hence, the isolation of the connection with each of the IoT Nodes has to be provided through appropriate multiplexing and demultiplexing of the communication within the Connection Service developed for the IoT devices.

Figure 14 (particularization of *Figure 12*) illustrates the data flow between Portal Server/USN platform and IoT nodes, though the Meshlium (GW4EXP in this particular case) and Service Provision Gateway (SPGW). As previously commented, for these three use cases, both native experimentation as well as experimentation at service level can be carried out in most of the nodes, except parking ones that do not allow it because of battery constraints.

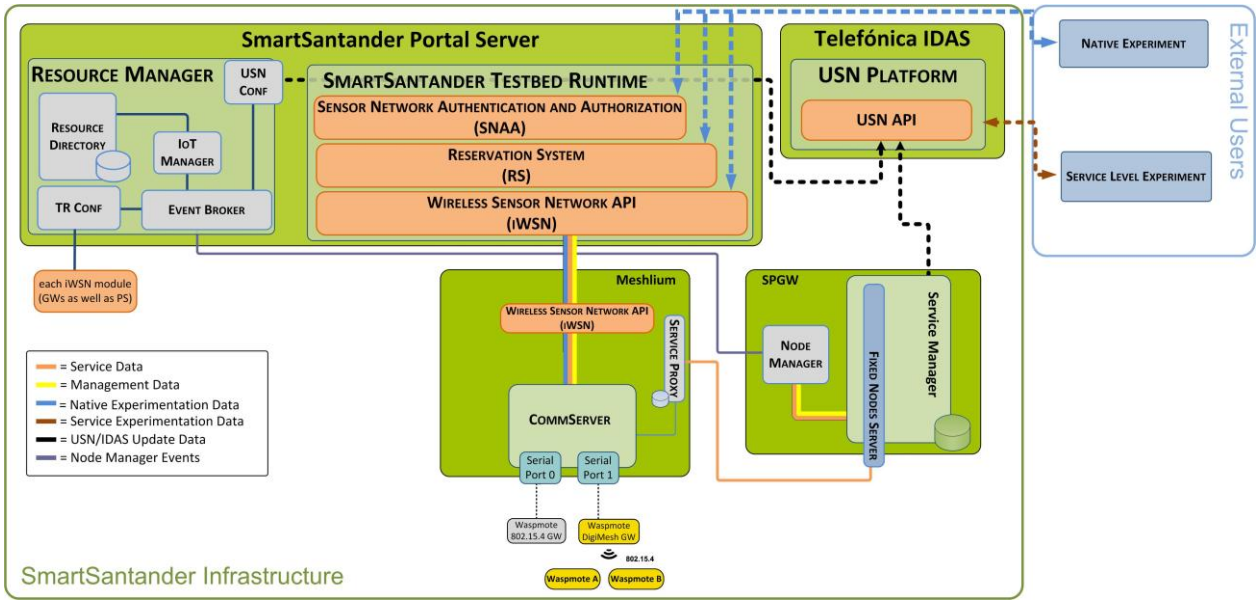

Figure 14: Environmental Monitoring, Parking and Irrigation logical architecture

Considering that Portal Server associated functionalities, as well as both native and at service level experimentation have been already described, next it is described the functionality of Meshlium and SPGW.

Meshlium module is in charge of gathering and processing the information retrieved by the deployed nodes, interacting accordingly with the Portal Server and the USN platform. CommServer module, through a single serial port connection (Serial Port0 and 1), receives the information retrieved by deployed nodes, associated to management and service (Digimesh interface), as well as experimentation (native 802.15.4 interface). Once information is received, this is sent to the Service Proxy and the Wireless Sensor Network API.

Service Proxy takes the information, process it and sends it to the corresponding process (in this case the associated to Fixed Nodes Service) running within the SPGW. This service provision information is stored in the Service Manager and sent to the USN Platform. Furthermore, SO (Service Only) Node Manager module uses this information to feedback Resource Manger module within the Portal Server.

iWSN is in charge of interaction between Meshlium and Portal Server through the TR provided functionalities in order to carry out experimentation in a native way. Some configuration commands, as well as OTAP procedures are carried out through this module.

Finally, it is not enough to implement the aforementioned components to fully support the IoT Nodes that have been deployed in the SmartSantander facility. There are three main functional features, namely experiments support, platform management and service provision, that have to be supported, at the same time, at the IoT Node level. These three functionalities have to coexist on the IoT Nodes in such a way that all of them are supported and they do not affect each other significantly. Basically, in order to fulfill this requirement, it is needed to implement these functionalities within the programs to be loaded on the IoT Nodes. In this sense, it is also necessary to implement these mandatory features and guarantee that they are always present on the IoT Nodes. Hence, the last module that has to be implemented will run on the IoT Nodes and will be on the one hand the responsible for supporting the provision of services and on the other hand for handling the commands and messages coming from the TR pertaining to Experimentation or Management Support Systems. As commented in previous section, this code (called golden image), has been loaded in the nodes at network start-up allowing the network management, without service provision disruption and loading different experiments on the deployed nodes.

Deployment

Environmental Monitoring and Outdoor Parking Management

Outdoor parking management and environmental monitoring use cases share the same infrastructure, where several parking sensors (IoT nodes) provided with one transceiver (running the Digimesh protocol) send their parking state (free or occupied), to the corresponding gateway through the repeaters placed at the streetlights. In addition to this, all these repeaters are equipped with temperature, CO, noise and luminosity sensors, thus sending this information to the gateway. The received information is stored and

processed in the gateway, in order to be used by different applications running over it, both in a local way or accessing from Internet through the SmartSantander backbone.

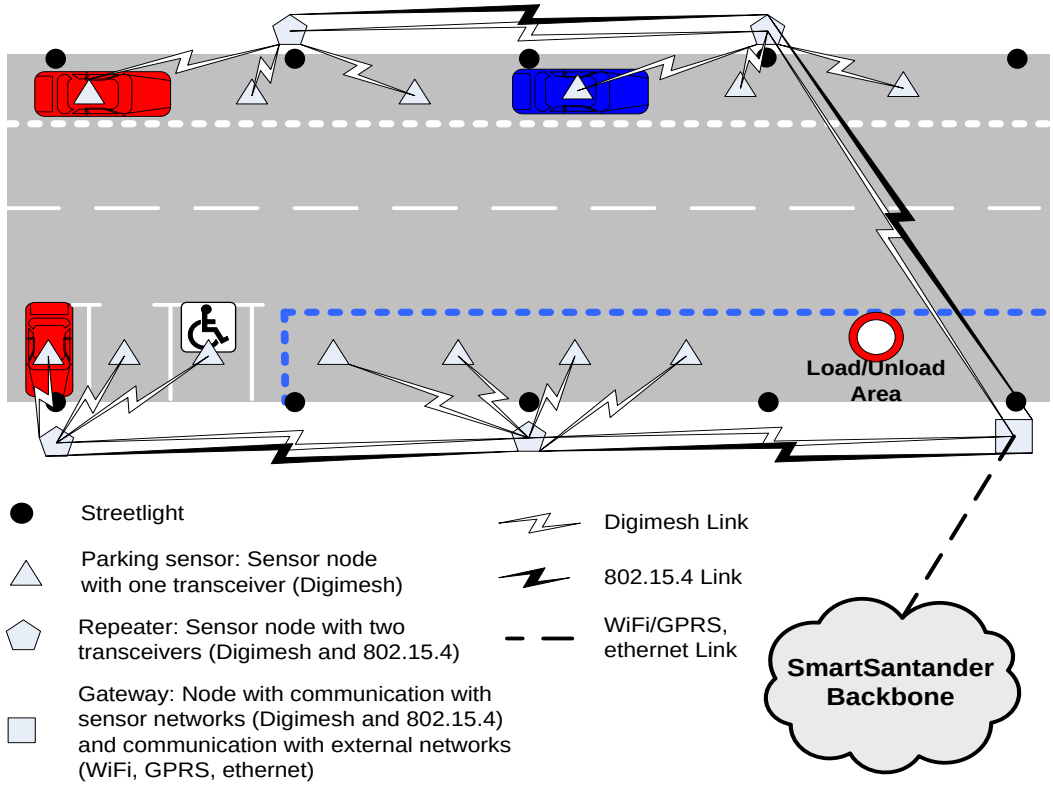

Figure 15: Architecture instantiation for Phase 1 in the Santander deployment

Regarding to the data associated to experimentation, as it can be observed in *Figure 15*, this is transmitted through the 802.15.4 native interface that works in an independent way from the Digimesh one, thus assuring no disturbance between experimentation and service provision/network management data.

Santander testbed composes of several clusters, being a cluster the set of IoT nodes and repeaters that are associated to a determined gateway. *Figure 16* shows in detail one of these clusters.

As it can be depicted from the figure different environmental sensors (temperature, CO, noise, light), as well as parking sensor nodes have been deployed in the city centre. All these nodes are programmed to send all the retrieved information, to the corresponding meshlium, and they are also able to be flashed with different experiments to be run on top of them.

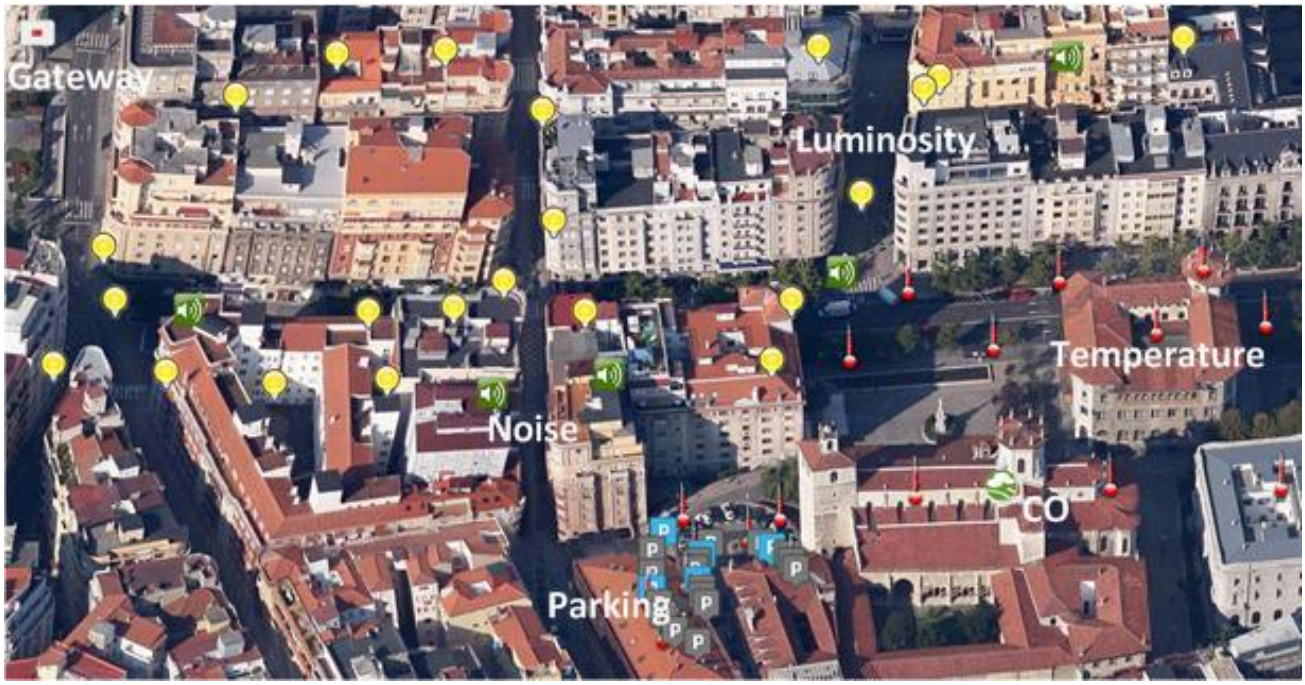

Figure 16: Detail of a cluster in the city centre

Parks and gardens irrigation

In order to control and make it more efficient the irrigation in certain parks and gardens within the city of Santander, several type of sensors will be installed in order to firstly monitor and evaluate the current situation and then to act over the irrigation systems making it more efficient the use of water. For this purpose, the following types of sensors are deployed:

- Weather station: Anemometer, pluviometer.
- Atmospheric pressure, solar radiation, air humidity and temperature sensors.
- Soil temperature and humidity sensors.
- Evaluation of water consumption sensor.

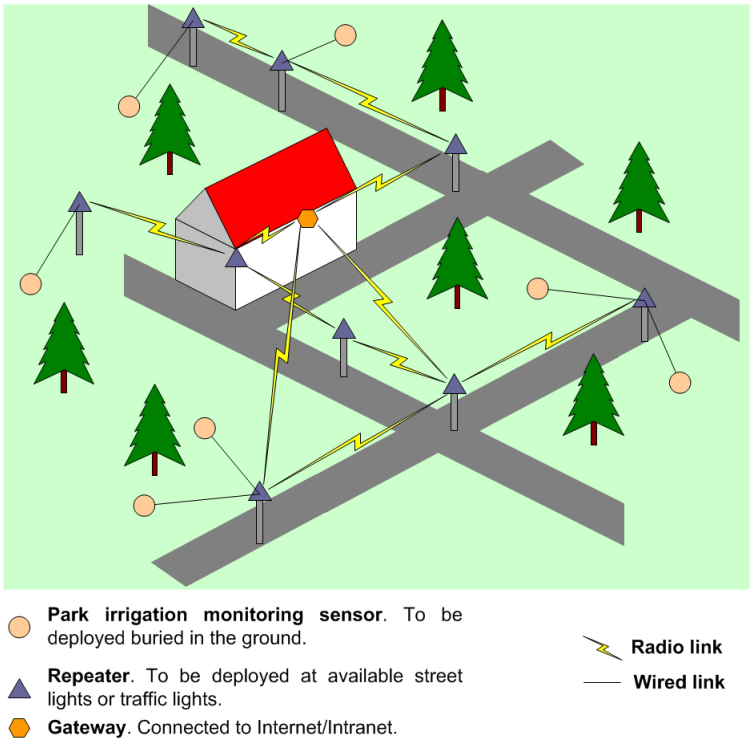

Figure 17: Detail of parks and gardens irrigation use case

Figure 17 shows the different sensors deployed in a park/garden, thus retrieving main parameters associated to soil and air status.

Mobile Environmental Monitoring

As previously indicated, within the SmartSantander project more than 2,000 environmental monitoring sensors have been already deployed. These sensors are monitoring CO index, temperature, noise level and light intensity. Although this deployment is very representative as it is located at the Santander city centre, it is necessary to extend the Environmental Monitoring service to other areas of the city. Hence, instead of continuing with fixed deployments all over the city, the hardware to be supplied will be deployed on municipal public buses, police cars and other municipal service vehicles. This way we will be able to cover a much wider area on a much more efficient way.

An overview of the architecture

The architecture deployed divides in several parts:

1. **Sensors Board**: Responsible for sensing the corresponding environmental parameters (temperature, humidity, CO, PM10, O3, NO2), sending these values through a serial port to a local processing unit.
2. **CANBUS module**: This module is in charge of measuring main parameters associated to the vehicle (GPS position, altitude, speed, course and odometer), sending them to the local processing unit.

3. **Waspmote board**: For this use case, nodes similar to those already deployed (waspmote nodes) have been installed. For this case, waspmotes are only provided with the 802.15.4 interface for experimentation issues, as service-related information and network management data is transmitted by the local processing unit.
4. **Local processing unit (LPU)**: This unit, called CLV, is in charge of managing the service provision (data retrieved from sensor board and CANBUS module), experimentation (sending the log messages generated by 802.15.4 interface) and network management (OTAP procedures, transmission/reception of commands).
5. **Gateway for mobile nodes (GWM)**: Equipment with high capacity in terms of computational power and memory which is in charge of gathering and processing all the information sent from the LPUs, just storing the information in the SmartSantander backbone.

All the buses will be provided with a sensor board, a CANBUS module, an IoT device and an LPU, whilst in each taxi and police car only a sensor board and an LPU will be installed (no native experimentation allowed at taxis and police cars).

Hardware deployment

From the hardware point of view, in *Figure 18* they are shown the LPU and the waspmote, the sensor and waspmote boards.

Figure 18: Detail of LPU, sensors and waspmote boards

Hardware modules shown in *Figure 18* have the following characteristics:

- **LPU (called as CLV)**: 32-bit RISC processor with up to 60 MIPS ARM7 at 70 MHz or higher under a Linux operating system and with a 8 MB Flash memory for user applications and 16MB of RAM. Regarding to connectivity issues, it is provided with RS232/485 and CAN interfaces for devices, 7 digital inputs, 5 digital outputs and 2 analog inputs. In terms of communication, it provides a GPRS interface.
- **Sensor board**: Measures several environmental parameters, such as CO, NO2, O3, PM10, temperature and humidity, including a box adapted for air circulation, as well as a basic RISC Micro-Controller at 8MHz for simple operations. Power and data transmission/reception is carried out through an RJ45 specific connector.

- **Waspmote board**: It is provided with an 802.15.4 native interface with an antenna of 5dBi, used for experimentation issues within coverage areas of the static 802.15.4 static network already deployed. On the other hand, a serial communication (through a RJ45 connection) is enabled between the waspmote board and the LPU.

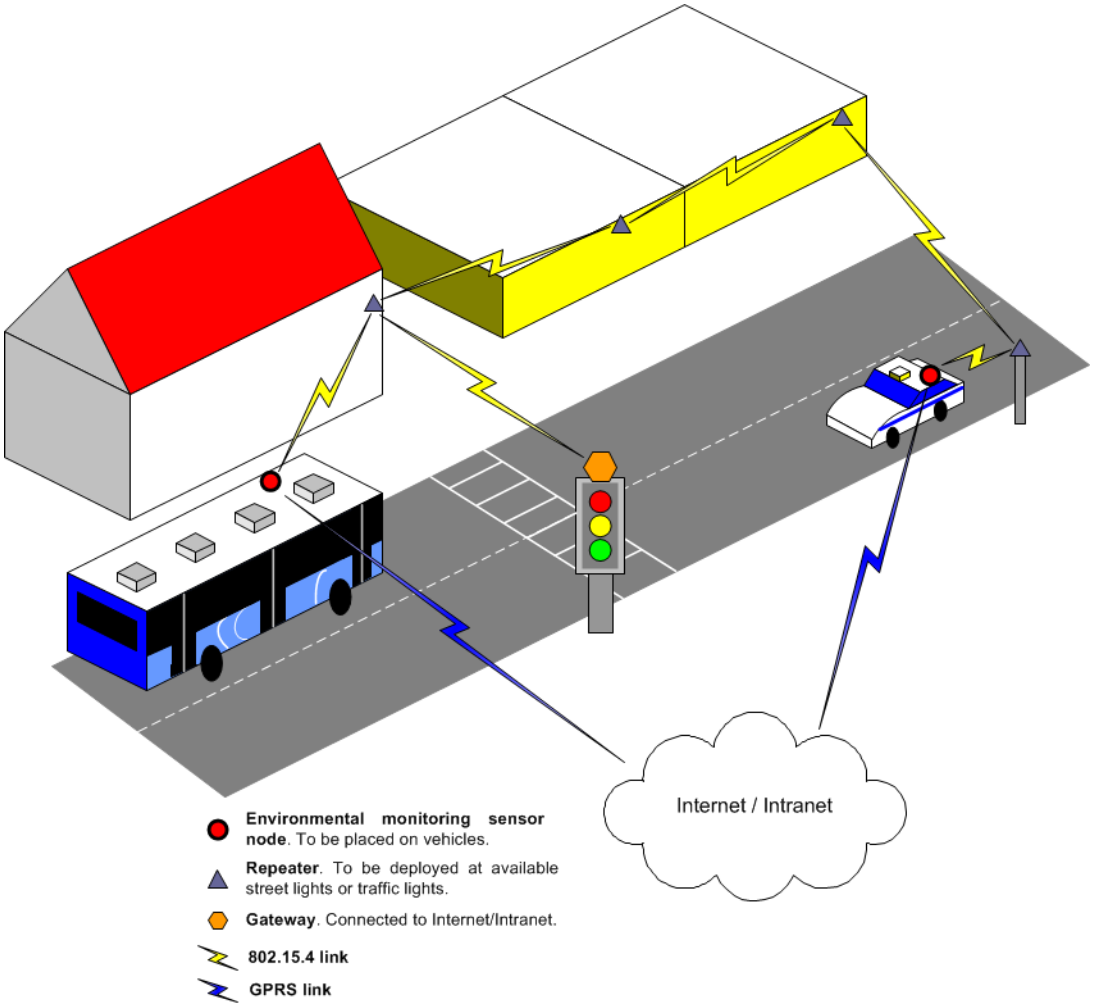

Figure 19: Intended mobile environmental monitoring scenario

As it can be derived from *Figure 19*, mobile nodes send the retrieved information (through GPRS interface) to the internet/intranet, and also, interact with the corresponding static nodes (through 802.15.4 interface) placed at streetlamps and facades, already deployed.

Network management, service provision and experimentation support

In order to perform network management, at the same time the service is running as well as an eventual experiment could be executed over the platform, the operation would be as follows:

1. Network management: Considering that LPU is provided with a GPRS interface which offers global coverage, both commands as well as MOTAP packets will be transmitted to this interface. Once these messages have been received in the LPU, this will process and forward them to either sensor board or GWM, accordingly.
2. Service provision: Measurements carried out by sensors board and CANBUS module (where installed), will send periodic measurements to the LPU which forwards these values to the GWM, thus providing service data to the SmartSantander backbone.
3. Experimentation: For the flashing of the corresponding experiments, the image to be loaded is sent from the GWM to the LPU through the GPRS interface. From the LPU, the image code is transmitted to the waspmote board through the corresponding serial interface.

In this respect, it is guaranteed that both management and service traffics are transmitted in a physically independent way from the experimentation information, thus obtaining interesting results:

- The provided service will never be interfered nor interrupted by experiments, thus avoiding the disruption of this service because of a misuse of the network by some experiment.
- The results retrieved from the experiment might be assumed as if the testbed were only for experimentation purposes, as there is no interfering traffic within the network, but only the one associated to the corresponding experiment.
- The management of the network is more reliable as all traffic running on the GPRS interface is predictable (all services are installed at start-up); so no external traffic will affect the communications. In this sense, nodes will be provided with a default program (called “golden image”), which will carry out the functionality associated to the corresponding service, as well as all the management issues needed for the correct network operation. This image will be loaded in the nodes at the network start-up, and re-flashed when a node is restored to its default state.

Logical Architecture

Once described the hardware architecture as well as the intended deployment, next it is shown the logical architecture for this use case.

The architecture shown in *Figure 20* is very similar to the one used for fixed nodes except from, instead of Meshlium, the GW module for managing communications with mobile IoT devices is the Gateway for Moble Nodes (GW4MN), in charge of gathering and processing the communication between the IoT devices and the Portal Server and USN platform. In this sense, the FleetServer is in charge of receiving the information transmitted by the deployed mobile nodes through a GPRS interface. Both experimentation at a native way and at service level are allowed over this use case.

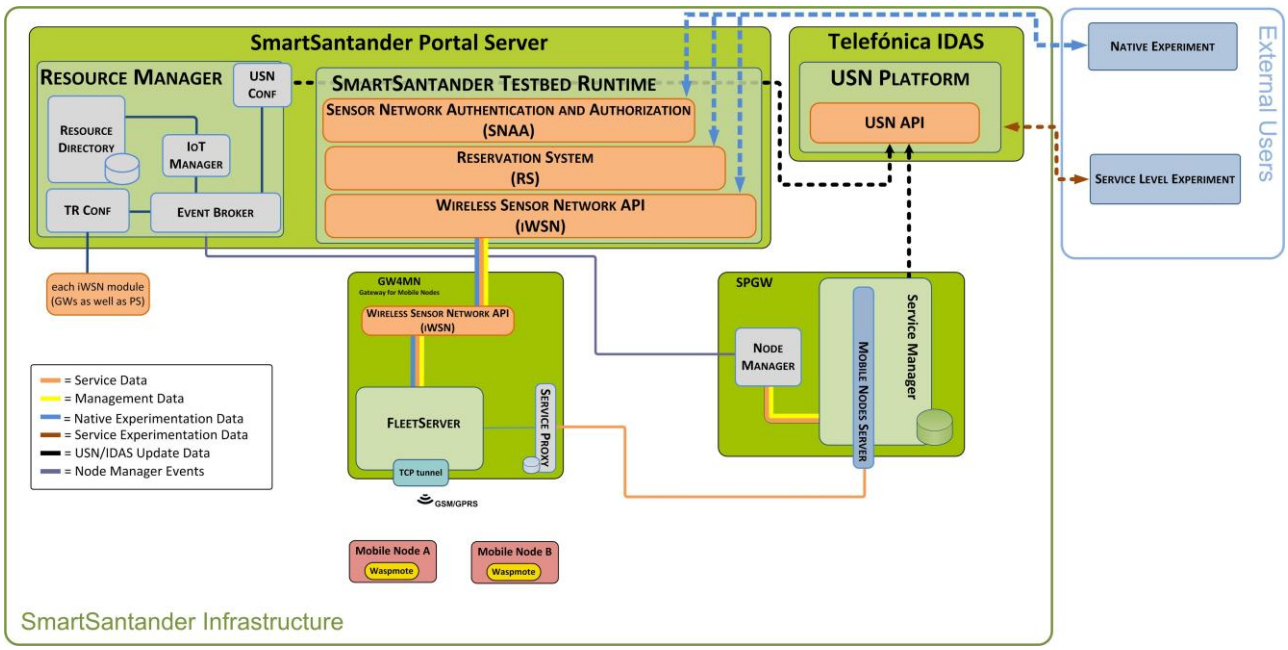

Figure 20: Mobile Environmental Monitoring logical architecture

In order the IoT Nodes to support experiments, platform management and service provision, it is also necessary to implement these mandatory features and guarantee that they are always present on the IoT Nodes. Hence, the IoT Nodes will run a specific module in charge of both supporting the provision of services and handling the commands and messages coming from the TR pertaining to Experimentation or Management Support Systems. As commented in previous section, this code (called golden image), has been loaded in the nodes at network start-up allowing the network management, without service provision disruption and loading different experiments on the deployed nodes.

Traffic Intensity Monitoring

Nowadays, the measure and classification of vehicles in road traffic is accomplished by inductive loops placed under the pavement. These inductive loops allow monitoring vehicle passing by means of different configurations which provide us a number of data in order to control several parameters of the traffic (vehicle speed, traffic congestion, traffic accidents).

However, these systems have several problems and disadvantages like their deployment, maintenance, high cost, and put into gear, among others. In this sense, within the SmartSantander project a solution based on sensors buried under the asphalt has been deployed to control traffic parameters at main entrances of the city of Santander.

An overview of the architecture

The architecture deployed divides in several parts:

1. **Traffic Sensor**: Buried under the asphalt, they are accountable for sensing the corresponding traffic parameters (traffic volumes, road occupancy, vehicle speed, queue length), sending these values through an 802.15.4 interface to the indicated repeaters.
2. **Repeater**: This module is in charge of forwarding the measurements retrieved from the traffic nodes, making it available to the corresponding access point.
3. **Access point**: Both traffic sensors and repeaters, are configured to send all the information to the access point. Once information is received by this node, it can either store it in a database which can be placed in a web server to be directly accessed from internet, or send it to another machine (central server), through the different interfaces provided by it (GPRS/UMTS or ethernet).

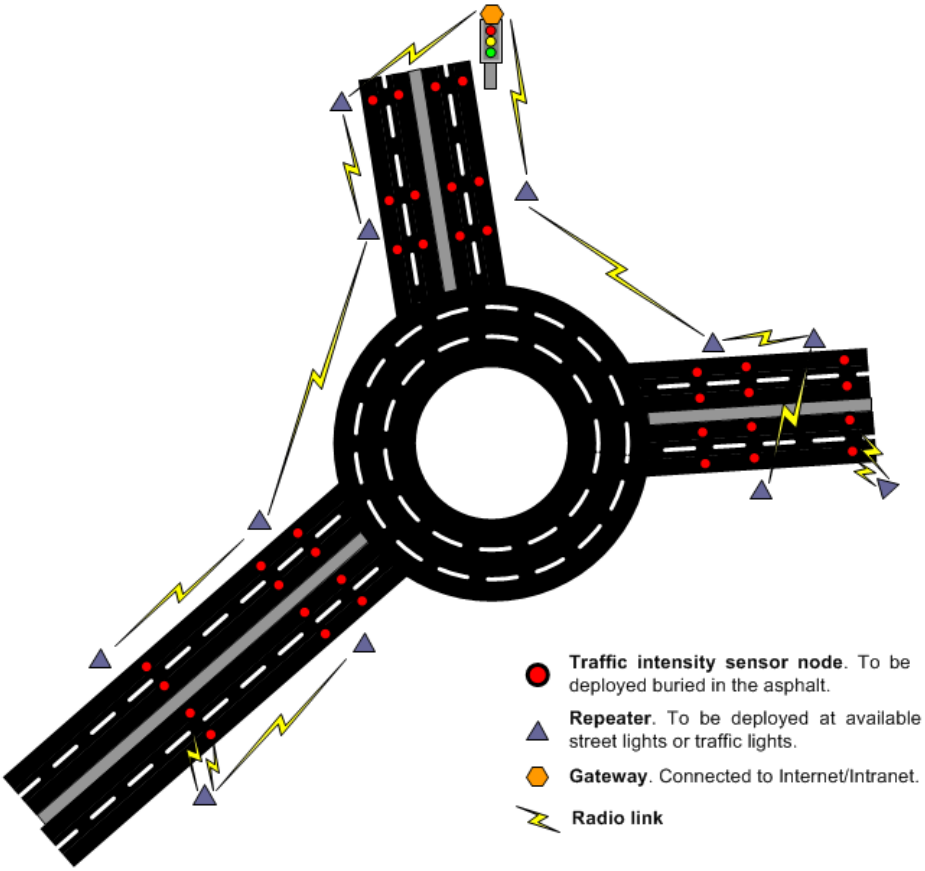

Figure 21: Intended traffic intensity monitoring scenario

In *Figure 21*, some traffic monitoring sensors are deployed in different road lanes, thus retrieving information on vehicle speed, flow and queue length.

Hardware deployment

From the hardware point of view, in *Figure 22* they are shown the traffic sensor, the repeater and the access point.

Figure 22: Detail of traffic sensor, repeater and access point

Main characteristics of deployed hardware are indicated next:

- **Traffic Sensor**: It uses magneto-resistive sensors to detect the presence and movement of vehicles, working at a frequency of 2.4 GHz and a data rate of 250Kbps. Furthermore, it is provided with an IP68 ingress protection.
- **Repeater**: It extends the range and coverage of an installation's access point. Both access point and repeater antennas provide a 120°, so for increasing coverage area a repeater can be mounted on the same pole or mast as the access point, but pointed in the opposite direction. Repeaters are battery powered (battery replacement every two years), as they present a low power consumption.
- **Access Point**: Intelligent device operating under the Linux operating system that maintains two-way wireless links to sensors and repeaters, as well as it uses a wired or wireless connection to relay the sensor detection to the corresponding SmartSantander remote server. For this purpose, access points are provided with an 802.15.4 interface and an Ethernet or GPRS/UMTS interface, respectively.

Network management, service provision and experimentation support

For this use case, native experimentation is not supported, but only addressed experimentation at service level. In this sense, deployed platform must only support service provision and network management, as indicated next:

1. **Network management**: For this purpose, from Portal Server commands and firmware updates are sent to the Access Points, which forwards them to the corresponding repeaters and, also, to the buried nodes.
2. **Service provision**: For service provision, data retrieved from repeaters/sensors can be processed, either in the Access Points and then sent them to the Portal Server or directly transmit raw data and then process them in the Portal Server.

In this sense, main service parameters can be varied in order to adapt service to the corresponding requirements.

Logical Architecture

Figure 23 shows the logical architecture associated to this use case.

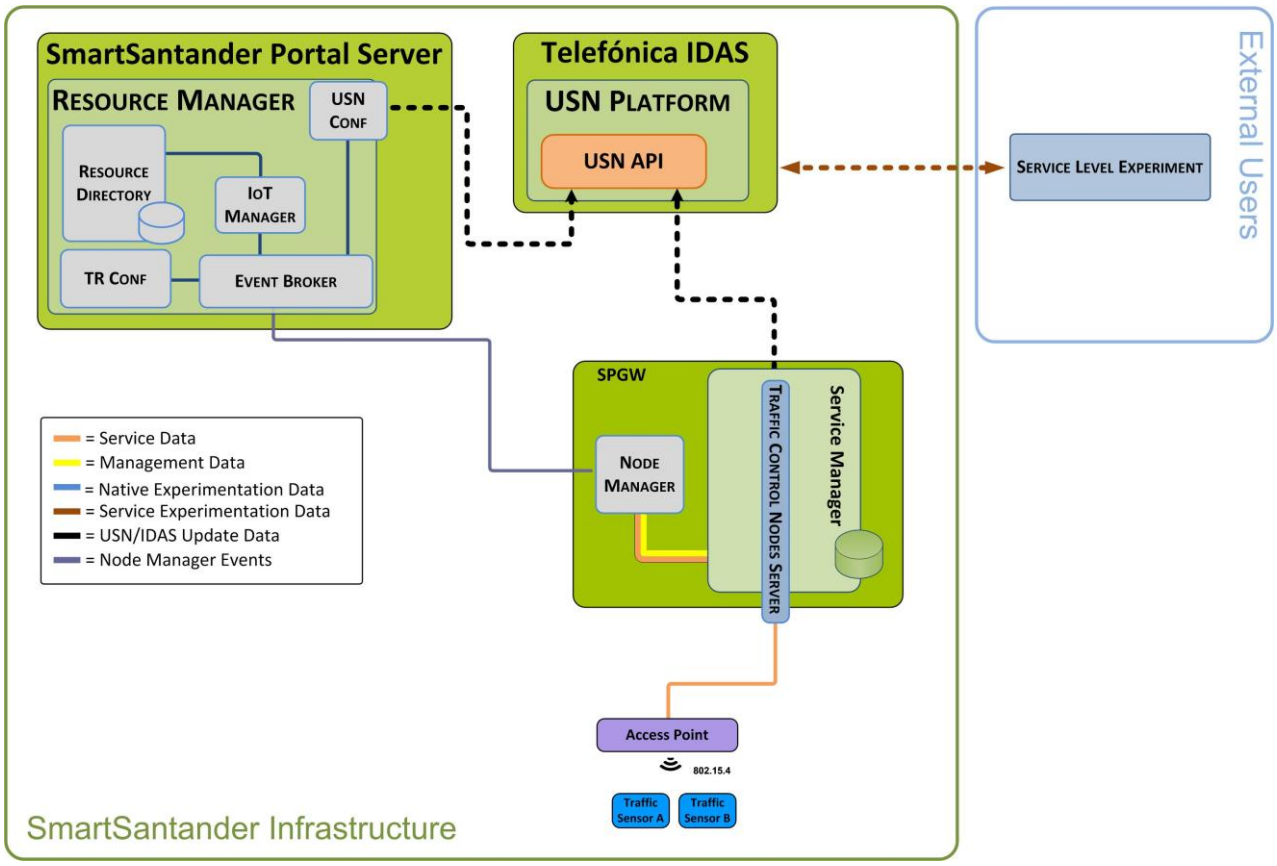

Figure 23: Traffic intensity monitoring logical architecture

Unlike the aforementioned use cases, as this one does not allow native experimentation over the deployed nodes, it is not needed a specific GW for experimentation issues and neither functionalities associated to experimentation within TR are implemented. In this sense, deployed traffic nodes send the corresponding information to the access point, which gather all the retrieved information sending it through an stream socket to the corresponding service (Traffic Control Nodes Server) within the Service Manager module.

Guidance to free parking lots

During the first phase deployment of the SmartSantander project, around 400 parking sensors have been installed in order to detect the occupancy degree of determined parking lots. Once the nodes are providing information on the occupancy status of the different parking places, the next step is to guide the drivers towards available free lots through the use of several panels, mainly placed at the streets' intersections.

An overview of the architecture

The architecture deployed divides in two parts:

1. **Panel**: According to the information received from the Central Station, the panel will show the number of places available in a determined parking zone, displaying it in different colors depending on the occupancy degree. The panel can show other information and messages that are not necessarily related to parking (e.g. 'street closed for renovation').
2. **Central Station**: It receives, from the Portal Server, all data retrieved by the sensors already deployed to detect parking lots availability, thus processing it accordingly and transmitting the suitable information to the corresponding panel. In addition to the reception of information from SmartSantander architecture, Central Station also sends incidences and other data related to the gathered information.

Figure 24 shows an overview of the installed infrastructure.

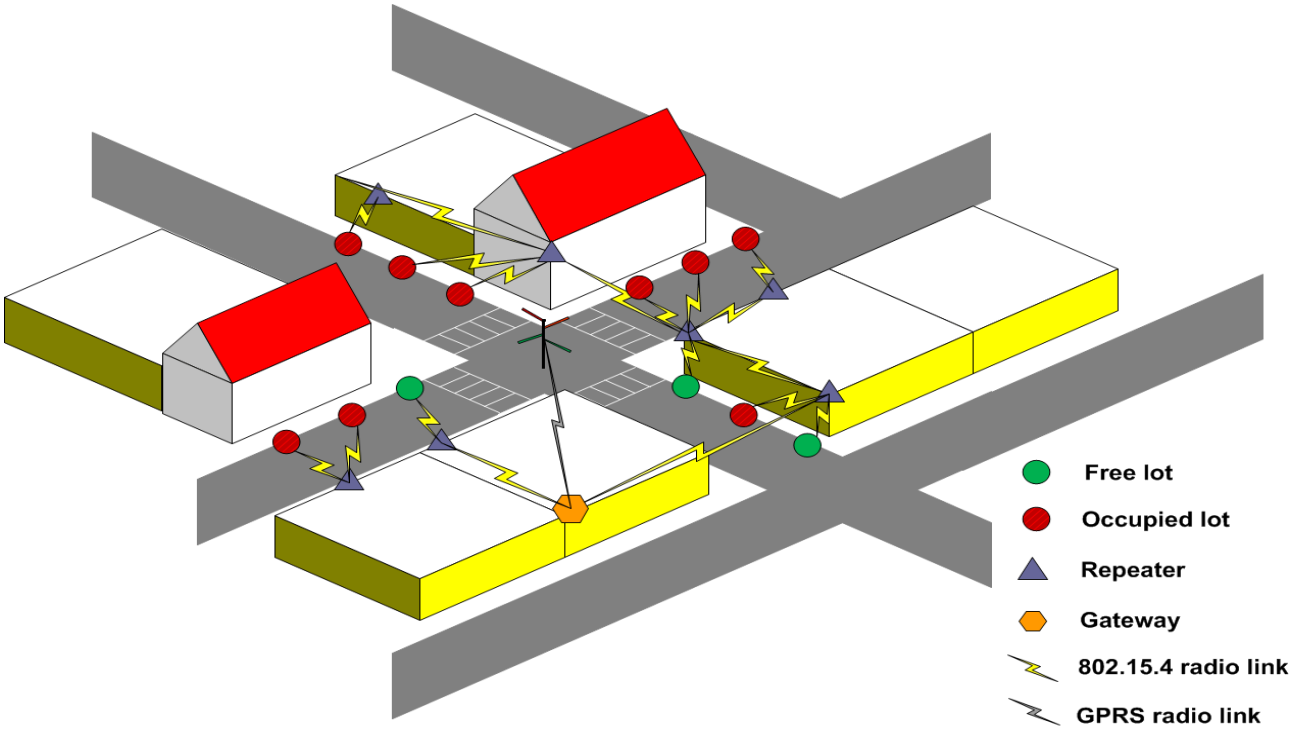

Figure 24: Guidance to free parking lots use case

As it can be derived from *Figure 24*, several panels are in charge of showing the number of free lots in different streets, green colour for streets with lots available, whilst red one for streets with no free parking lots.

Hardware deployment

From the hardware point of view, in *Figure 25* they are shown some examples of the panels to be installed.

Figure 25: Detail of some of the panels to be installed

Main characteristics of the panels are indicated next:

- **Panel**: Panel is provided with a multicolor LED display (green/red/amber) and a 120 mm digit height alphanumeric, assuring a reading distance up to 50 meters. Panels must have uninterruptible power supply (220 V), although they are also equipped with a UPS (Uninterruptible Power Supply) to protect against small outages (30 minutes). Furthermore, panel is provided with an IP67 protection and a GPRS connection for transmitting/receiving information.

Network management, service provision and experimentation support

For this use case, native experimentation is not supported, but only experimentation at service level is provided. In this sense, deployed platform must only support service provision and network management, as indicated next:

1. **Network management**: Panels can be accessed from the Central Station in order to modify several parameters, such as refresh period of available parking lots. Panels can also send incidences regarding to communication problems, operation failures. Direct communication between panels is not available.
2. **Service provision**: For service provision, data retrieved from Portal Server (provided by the static parking sensors buried under the asphalt) is sent to the Central Station that will process it, transmitting to each panel the information associated to the zone covered by this panel. Apart from the traffic associated to this service, Central Station also process the information received, sending statistical information to the Portal Server.

In this sense, main service parameters can be varied in order to adapt service to the corresponding requirements.

Logical Architecture

The logical architecture associated to panels is shown in *Figure 26*:

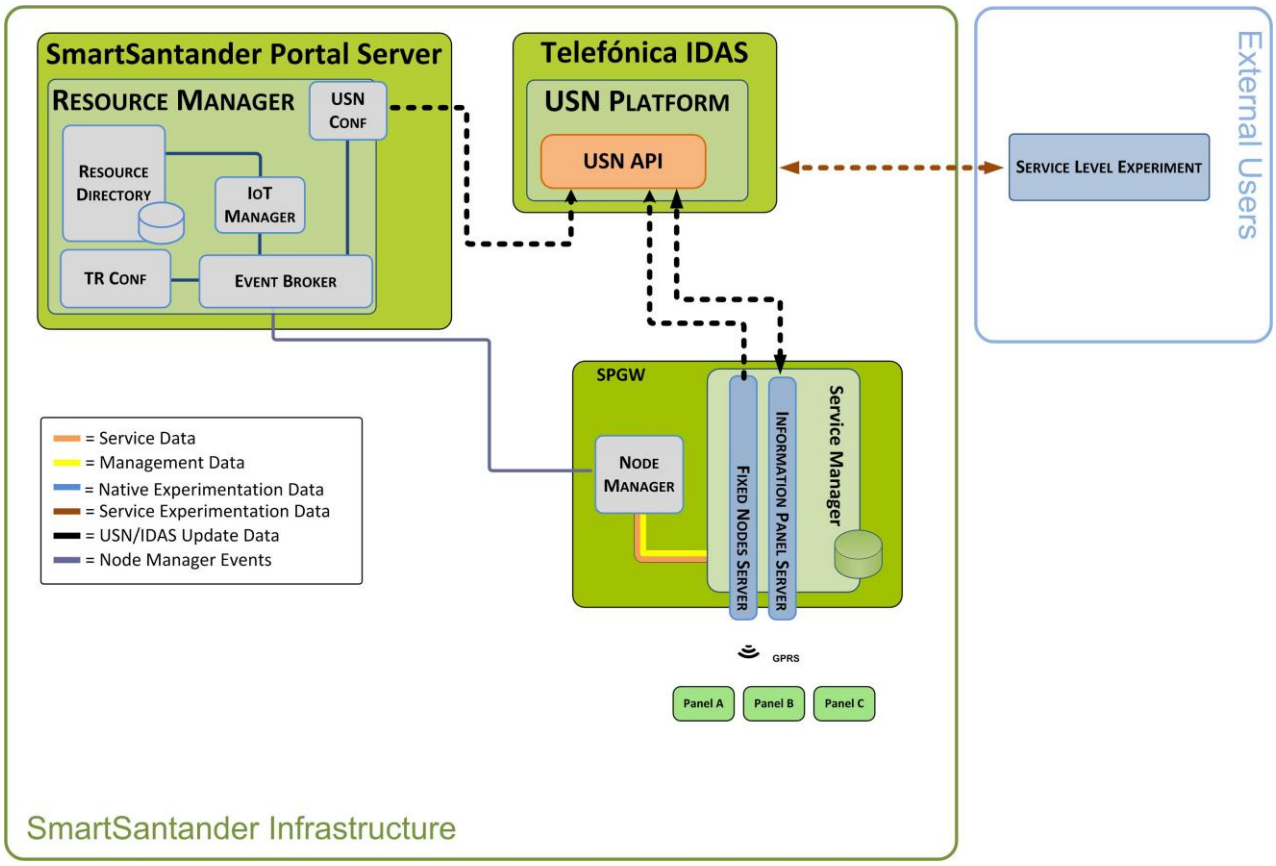

Figure 26: Guidance to free parking lots logical architecture

This use case is very similar to the traffic one as native experimentation is not supported. In addition, this use case has the particularity that the Information Panel Service, not only sends service related information to the USN, but also it takes information regarding to available parking lots (stored in the USN by Fixed Nodes Server), processing it and sending that to the corresponding panel.

Augmented Reality and Participatory sensing

Unlike to the aforementioned use cases, these ones are intended as applications and services developed within the SmarSantander project over the deployed infrastructure. Nevertheless, both ones allow experimentation at service level with the information provided by anonymous users through their smartphones, tablets (participatory sensing); as well as the information inferred by the access to different links associated to the reading of a determined RFID tag or a QR code label (augmented reality).

Figure 27: Design of the RFID/QR sticker to be placed at shops

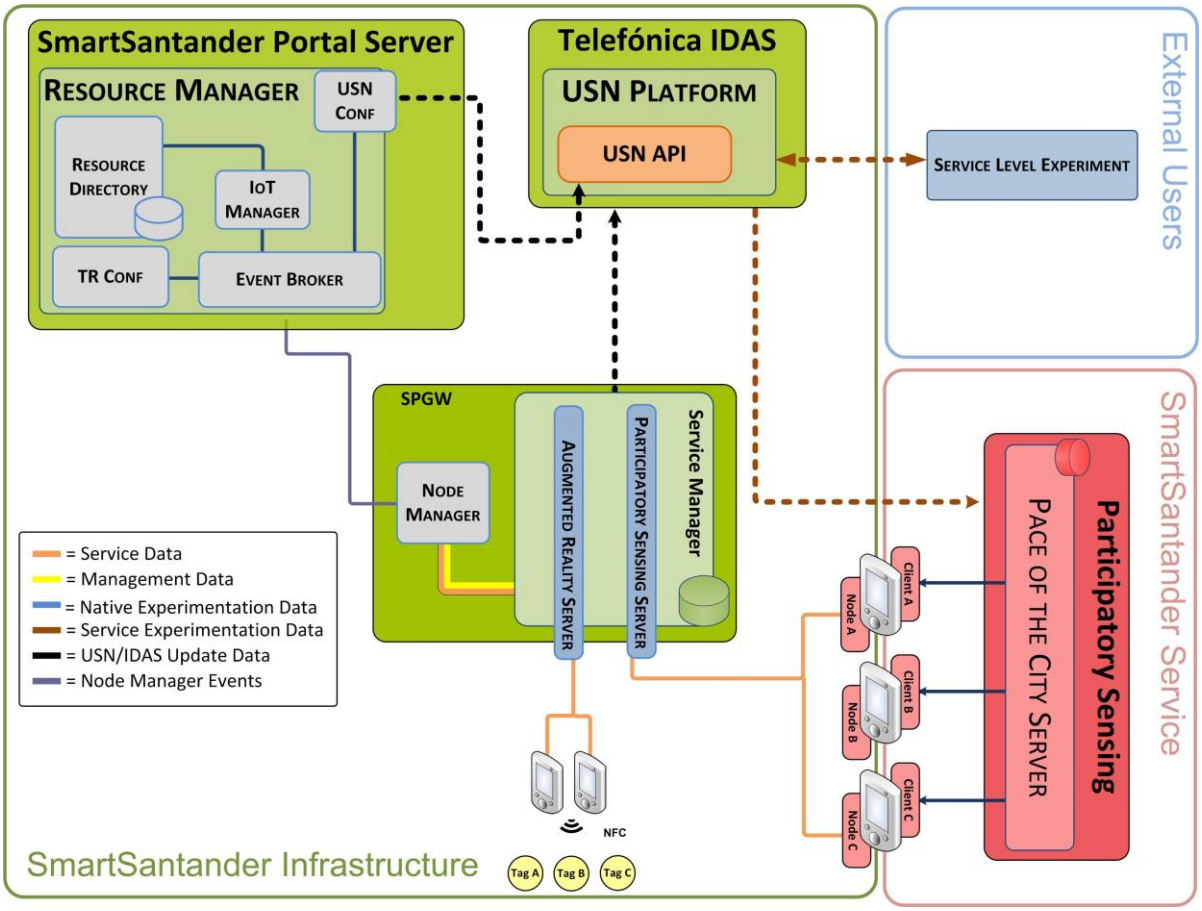

Figure 28: Augmented Reality and Participatory Sensing logical architecture

Figure 27 shows one example of the stickers to be installed at shops, where it can be observed that both NFC and QR access is available for the user.

In *Figure 28*, logical architecture is depicted, where it can be observed that augmented reality application takes the corresponding information from the tag and process it in the corresponding server within the Service Manager. For the Participatory sensing use case, smartphones and tablets behave both as information producers and consumers (prosumers), so information sent from the Participatory Sensing Server to the USN is taken by the Pace of the City Server, processing it accordingly and offering in a suitable way to the end users.

The Pace of the City application could be considered as an example of experimentation at service level, as information provided by SmartSantander facility is used by an application for offering a determined service.

References

[D1.1] First Cycle Architecture Specification

[D1.2] Second Cycle Architecture Specification

[D.5.3] Regulations for use of experimental facility

[MAP_SDR] http://www.smartsantander.eu/map/

[WISEBED] WISEBED - Wireless Sensor Network Testbeds. http://www.wisebed.eu

[SENSEI] FP7 Project SENSEI. www.sensei-project.eu

[TELCO] Bernat, J.; Pérez, S.; González, A.; Sorribas, R.; Villarrubia, L.; Hernández, L.,"Ubiquitous Sensor Networks in IMS: an Ambient Intelligence Telco Platform," in ICT Mobile Summit, Stokholm, Sweeden, 2008.

[LT codes] M. Luby, “LT codes,” in Proc. 43rd Annu. IEEE Symp. Foundations of Computer Science, Vancouver, BC, Canada, Nov. 2002, pp. 271–280.