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 first Open Call for new project partners to submit proposal for experimentation on the project's test facility.

Open Call Summary

The SmartSantander project is offering up to 200k EUR funding contribution 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 public. The funding will be allocated through a series of open calls. The first open call opens 1st October 2011 and will close on 16th November 2011, targeting the Internet of Things and Smart City communities.

The SmartSantander is a unique experimental facility as it is deployed in a real city, with citizens using 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-1-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 2011
Call deadline:	The call closes on **16th November 2011 at 17h00** (Brussels time)
Expected duration of participation:	January 2012 to December 2012
Maximum funding per experiment:	Up to 200,000 EUR
Maximum funding for call:	600k Euros
Number of experiments:	3-5
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.

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 the deploying such a unique infrastructure which allows true field experiments. Researcher will be allowed to reserve the needed 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 Santander and Guildford facilities which will be available for the first open call. Additional information about the operational facilities is available in the annex of this document as well as in [D1.1].

Santander summary

The Santander testbed is composed of around 2000 IEEE 802.15.4 devices deployed in a 3-tiered architecture:

- IoT node: Responsible for sensing the corresponding parameter (temperature, CO, noise, light, car presence,...).
- 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.
- Gateway: Devices that gather all the information retrieved by IoT nodes and repeaters, acting as intermediate nodes between the sensor networks and the SmartSantander backbone.

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 has been adopted for this first deployment. In this sense, all nodes (IoT nodes, repeaters and gateways) will be provided of two IEEE 802.15.4 modules, one of them performing network management as well as service provision, and the other one in charge of traffic associated to experimentation.

- Service Provision: Both outdoor parking area control and monitoring of environmental parameters (temperature, CO, noise, light), are the first services developed over the deployed infrastructure.
- Experimentation: Nodes can be flashed over the air in either a one-hop way (OTAP) or a multihop fashion (MOTAP) with different programs, thus allowing the researchers to test different experiments on the deployed network.
- Network management: In order to manage both experimentation and service provision, communication between IoT Nodes/repeaters and gateway nodes, is performed through the Testbed Runtime (TR). In order to manage these new wireless devices, it is added a mux/demux functionality as well as the corresponding device drivers to the TR. Furthermore, in order to fulfill experiments support, platform management and service provision in a joint way at the node level, it is needed to load 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 to manage the deployed network, i.e. sending commands or flashing nodes.

Guildford summary

The first phase of the Guildford deployment is the creation of a Smart environment based on an indoor deployment of the testbed in the Centre for Communication Systems Research. It will serve as initial core and experimental micro-cosmos of the envisioned Smart Campus facility.

The IoT node tier will consist 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 will consist of 200 TelosB based platforms and about 50 SunSpots. The deployment of the nodes currently stretching over two floors of the building. Figure 1 shows the current deployment snapshot as an example of the deployment topology. As the Phase 1 deployment is not yet finalized the final layout will vary.

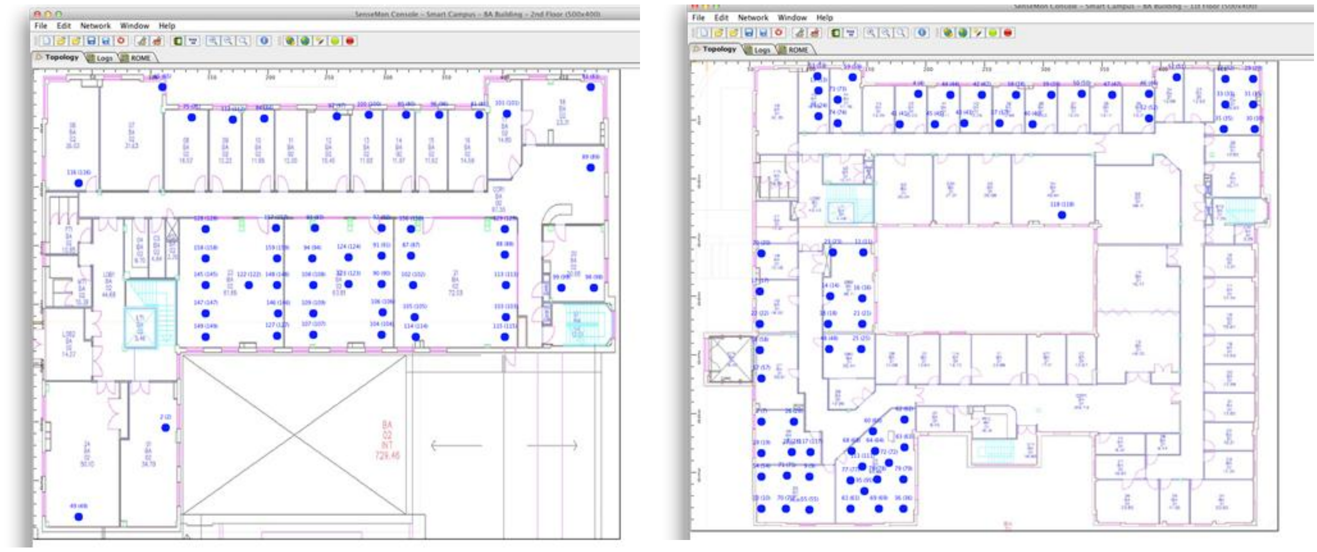
SenseMan Console – Smart Campus – BA Building – 2nd Floor (500x400)
File Edit Network Window Help
Topology Logs ROME
File Edit Network Window Help
Topology Logs ROME

Figure 1: 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, are deployed collocated to the sensor node for their management while offering the possibility to act at the same time as data GW. A server cloud hosts the testbed management servers and allows the on-demand creation of other application servers and data management tools.

Figure 2 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 back bone network were application servers and testbed management servers reside. The data plane of the testbed is realised 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, and Ethernet of from the GW towards the servers in the backbone network. An out-of band testbed management and control plane is realised 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.

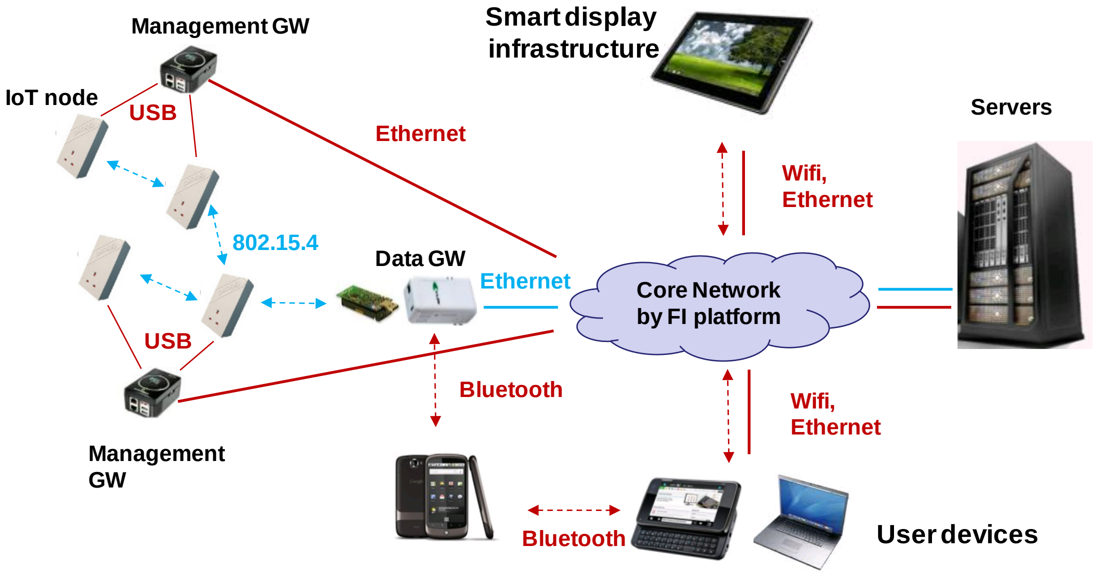

Figure 2: High level network diagram for Guildford Phase 1 deployment

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.

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 3. 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.

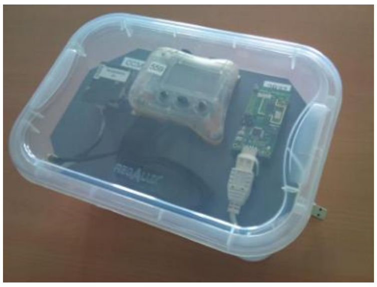

Figure 3: UZL cluster with an iSense, Pacemate and TelosB mote

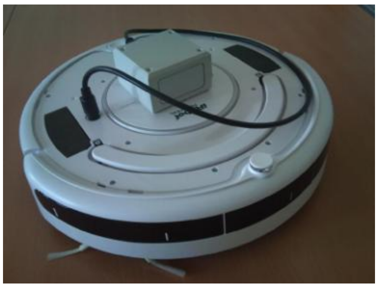

Figure 4: Roomba with iSense node

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 4) with attached iSense sensor nodes and with Lego Mindstorms).

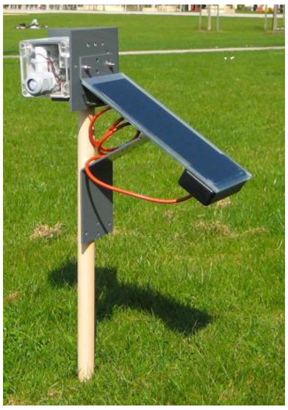

Figure 5: An outdoor node

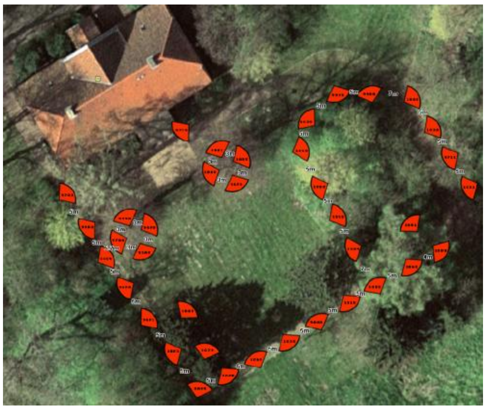

Figure 6: Outline of the outdor testbed

In addition to the indoor and mobile nodes, UZL has deployed a number of outdoor, solar-powered iSense sensor nodes (see Figure 5). It features 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 6).

Belgrade summary

The EkoBus system deployed in the cities of Belgrade and Pancevo is made available for experimentation on IoT data level. The system utilizes public transportation vehicles in the city of Belgrade and 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.

An overview of the EkoBus architecture

Every IoT node (sensor) in the system is described by its set of capabilities (characteristics, parameters, availability...), which are published and stored in the Resource Directory (RD). 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 or USSD and to receive that information via SMS. The SMS module is responsible for this feature.

Analysis of the stored data is used for various traffic calculation and prediction. Accordingly, additional information is available from the MYSQL database: **Static data**: geo locations and names of the stations, geo locations of curves and semaphores on the bus route, bus timetables, IMEI of the GPRS modules which are mounted on the buses, 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* – 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 Application is calling methods exposed by Data control centre (bus information web service). 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/USSD* – end-user can query the system using USSD or 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 Belgrade and 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 utilizing simple REST interface that allows extraction of the measurements for a given period.

- Direct access to IoT nodes is not available. It is also not possible to change the code or configuration of the IoT nodes.
- In Belgrade and Pancevo, 5 and 60 devices are deployed respectively.

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 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 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. Experiments 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.

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.

[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/

- 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 December 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].

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

Authentication, Authorization and Accounting (AAA): Access control 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 (ESS): It 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 (MSS): It 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 (ASS): It 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, SENSEI and TELCO 2.0) in order to fulfil the described requirements. A graphical representation of the provided integration and the involved components is shown in *Figure 7*.

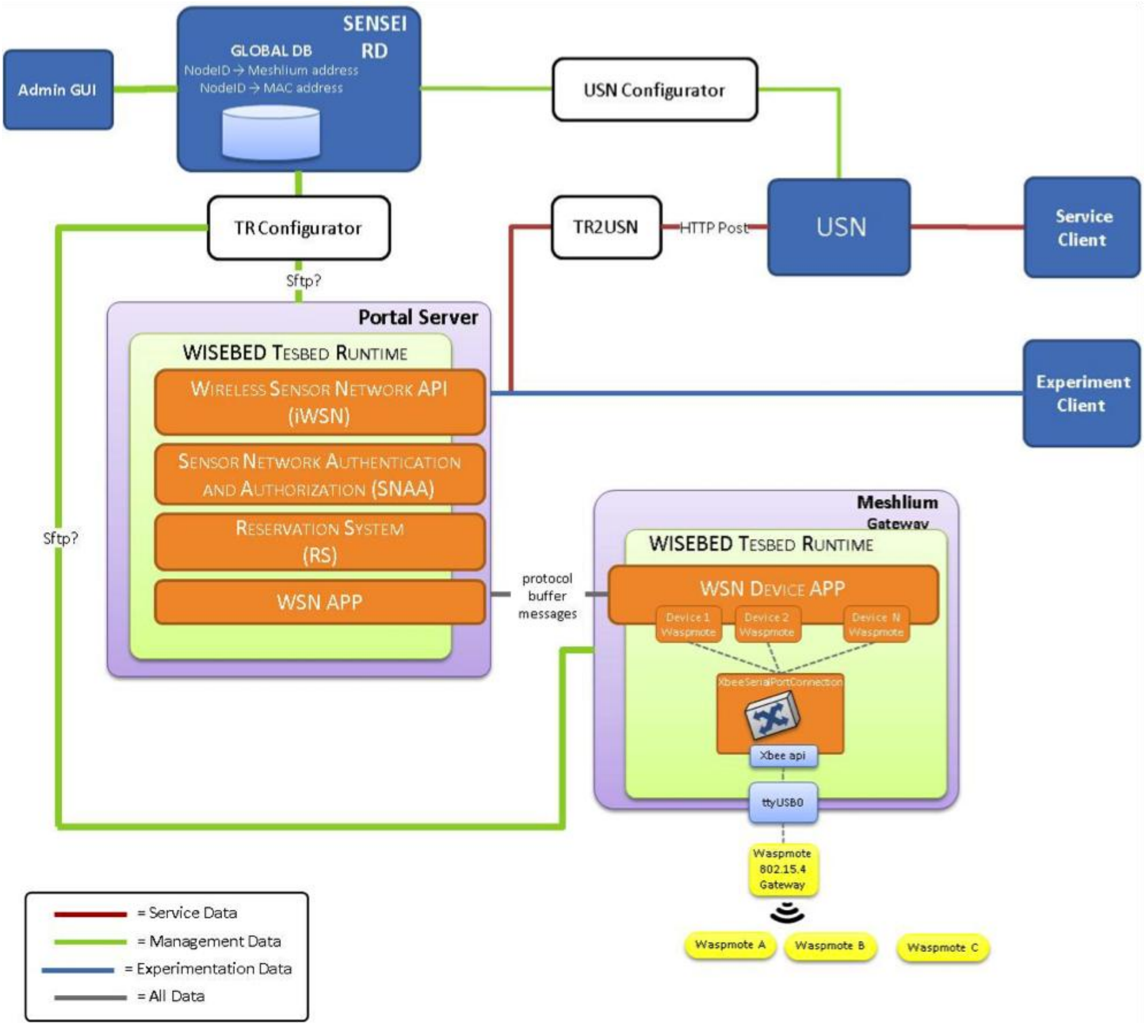

Figure 7: Components Integration

The three main domains composing the SmartSantander architecture (Portal Server, GW node and Waspmote IoT nodes) can be identified in *Figure 7*. 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 GW/IoT nodes. The communication is achieved by exploiting the protocol buffer message scheme. The iWSN API counterpart on the GW 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.

In order to run experiments, by means of an Experiment Client, the experimenters is 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 it can perform experiments configuring them through the iWSN API.

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. 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. The SENSEI Resource Directory (RD) has been chosen for storing the required configuration parameters. Through the provided Resource Publish Interface (RPI) the provided configuration is pushed and stored into the RD. Two new components, namely 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 WISEBED Tesbed Runtime and the USN. This distribution is achieved by means of a secure ftp connection.

Finally, in order to allow the user to access services provided data, a Service 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 SmartSantander facility can be then available to the USN system and then to the user, by means of new a communication channel implemented by the TR2USN component. Exploiting the Controller implementation provided by the iWSN API that realizes a communication channel towards each service nodes, the data generated by the IoT nodes can be forwarded to the portal server, exploiting the protocol buffer messages, and then pushed to the USN module, using an HTTP post method.

An overview of the architecture

Aiming at achieving a massive deployment of an Internet of Things infrastructure in a city scenario, requires careful planning in order to cope with the requirements imposed by the different FIRE stakeholders as well as to reduce the impact on the city services and the citizens. Hence, SmartSantander has conceived a 3-tiered architecture, as defined next:

1. **IoT node**: Responsible for sensing the corresponding parameter (temperature, CO, noise, light, car presence,...). 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 within Phase 1 of the Santander testbed deployment. This solution, provided by the Spanish company Libelium, consists of nodes implementing two different physical interfaces, as shown in the figure.

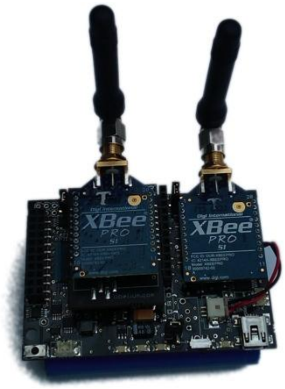

Figure 8: IoT Node deployed in Santander Phase 1 testbed

The node depicted in *Figure 8* 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.
- **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 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 9 illustrates this mux/demux functionality deployed within the TR to support communication with Waspmote devices. The hatched boxes in the figure, namely CommServer, Multiplexed Connector and Waspmote Driver, implement the mux/demux functionality in order to support the wireless Waspmote devices. The first two modules are generically-applicable and IoT device-independent, whilst the WaspMote driver is device-specific i.e. a new driver needs to be implemented for each new device type.

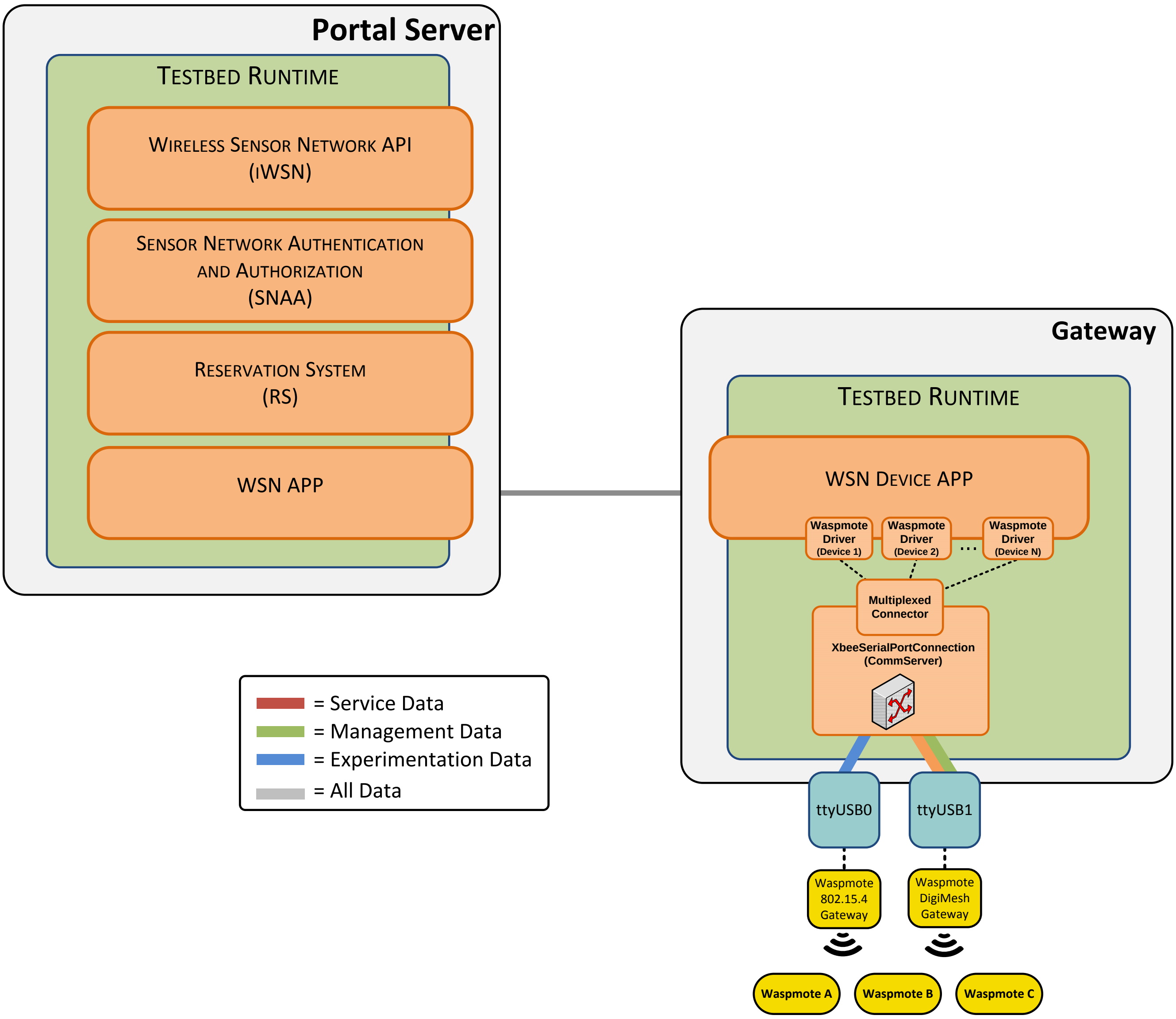
Portal Server
Testbed Runtime
Wireless Sensor Network API (iWSN)
Sensor Network Authentication and Authorization (SNAA)
Reservation System (RS)
WSN APP
Gateway
Testbed Runtime
WSN Device APP
Waspmote Driver (Device 1)
Waspmote Driver (Device 2)
…
Waspmote Driver (Device N)
Multiplexed Connector
XbeeSerialPortConnection (CommServer)
🟥 = Service Data
🟩 = Management Data
🟦 = Experimentation Data
⬜ = All Data
ttyUSB0
ttyUSB1
Waspmote 802.15.4 Gateway
Waspmote DigiMesh Gateway
Waspmote A
Waspmote B
Waspmote C

Figure 9: High-level architecture for integration of Waspmote-based WSN

- **CommServer**: The CommServer module is responsible for directly interacting with the physical radio interfaces in charge of wirelessly interacting with the WSN and of providing a unique interface for the access to/from the rest of the system.
- **Multiplexed Connector**: Since TR assumes that there is one connector per IoT Node managed in the testbed, it is necessary to implement a module that takes the unique interface offered by the CommServer and create one connector associated to each of the IoT Nodes. These connectors (WaspMote Driver in *Figure 9*) are the ones that TR actually uses for interacting with the IoT Nodes. In this sense, when a command or message is to be transmitted, TR simply sends it through the connector associated with the IoT Node that should be receiving it. The Multiplexed Connector will manage the message and forward it to the CommServer which will subsequently send it through

the appropriate physical interface. When a message is received from a node, the Multiplexed Connector will forward it to the TR through the connector associated with the sender IoT Node.

- **Waspmote Driver**: Functionalities and message formatting that are specific for the Waspmote devices are implemented in this module. As TR assumes one connector per IoT Node in the WSN, there is one of these modules per device in the WSN.

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.

First implemented Use case

As it has been stood out, a key feature of the SmartSantander system is that it is not only an experimental facility but also a real-world deployment that has to be leveraged for the provision of advanced services that showcase the benefits provided by Future Internet technologies. For Phase 1 deployment, the service to be provided is an outdoor parking area control, as well as the provision of real time information related to environment conditions (e.g. temperature, CO, noise, light sensors). In Figure 10, it is presented a diagram which shows the interactions between the three constituent blocks (IoT nodes, repeaters and gateways), in order to provide the aforementioned use case.

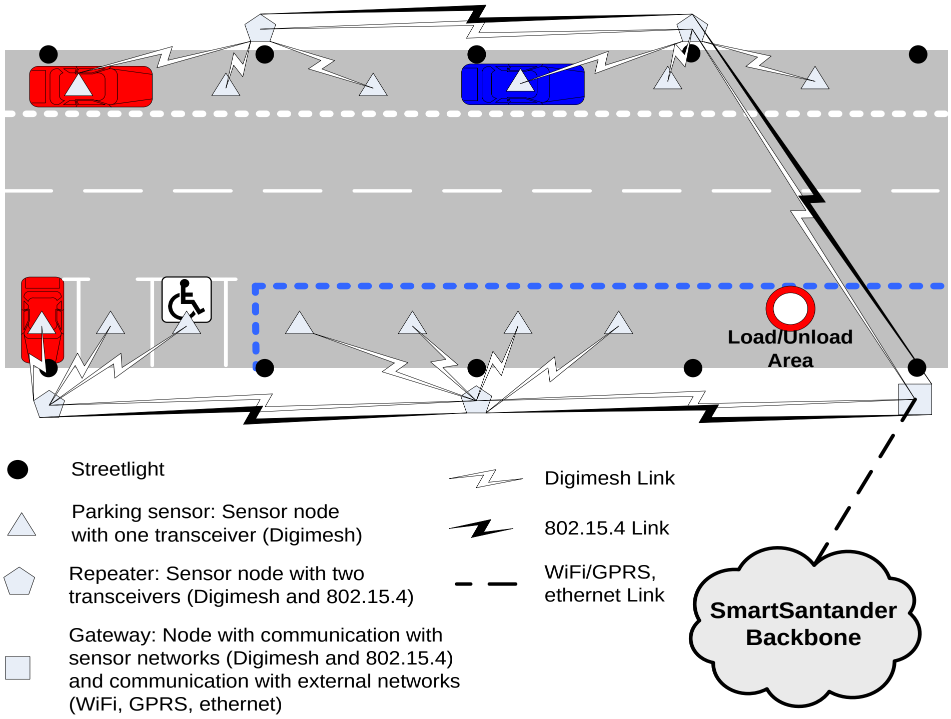
Load/Unload Area
Streetlight
Parking sensor: Sensor node with one transceiver (Digimesh)
Repeater: Sensor node with two transceivers (Digimesh and 802.15.4)
Gateway: Node with communication with sensor networks (Digimesh and 802.15.4) and communication with external networks (WiFi, GPRS, ethernet)
Digimesh Link
802.15.4 Link
WiFi/GPRS, ethernet Link
SmartSantander Backbone

Figure 10: Architecture instantiation for Phase 1 in the Santander deployment

Figure 10 shows the way the limited parking management use case is performed 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. At the same time, all these repeaters are equipped with temperature, CO, noise and light 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.

Regarding to the data associated to experimentation, this is transmitted through the 802.15.4 native interface that works in an independent way from the Digimesh interface, 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 10 shows the logical architecture corresponding to the deployment whilst Figure 11 depicts node deployment at downtown.

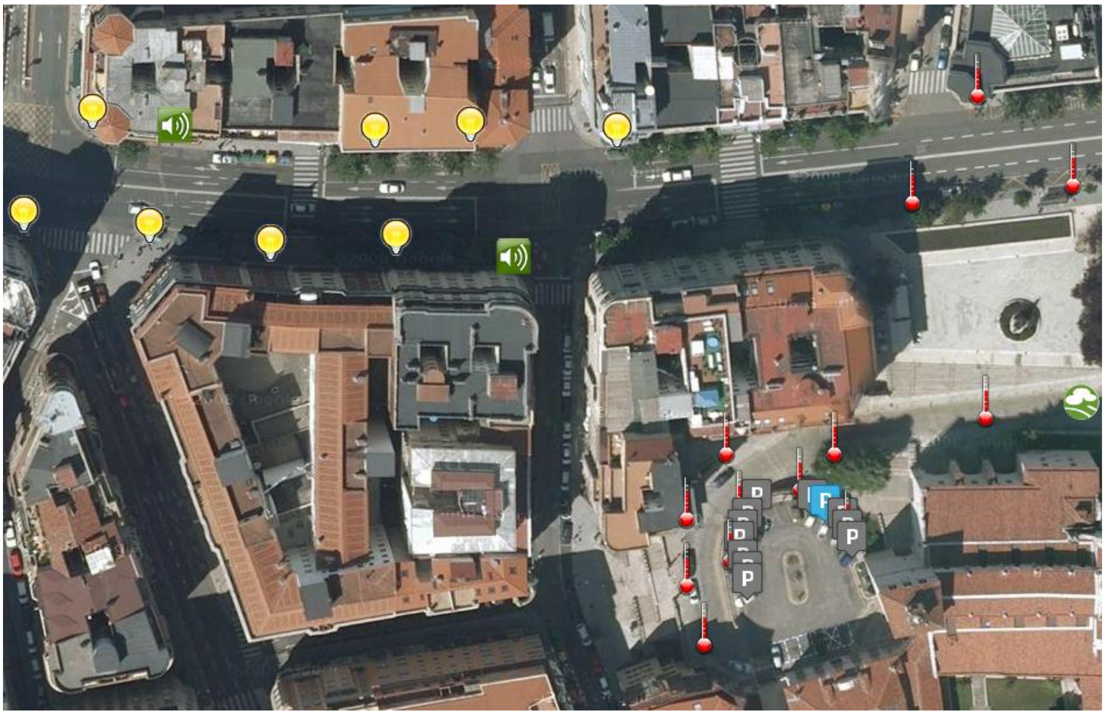

Figure 11: Cluster deployed in Santander city centre

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.

References

[D1.1] First Cycle Architecture Specification

[D.5.3] Regulations for use of experimental facility

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