Ad Hoc Networks xxx (2015) xxx–xxx

ELSEVIER

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

How to choose an experimentation platform for wireless sensor networks? A survey on static and mobile wireless sensor network experimentation facilities

Anne-Sophie Tonneau, Nathalie Mitton, Julien Vandaele *

Inria Lille – Nord Europe, France

A R T I C L E I N F O

Article history:
Received 5 September 2014
Received in revised form 16 February 2015
Accepted 2 March 2015
Available online xxxx

Keywords:
Internet of Things
Wireless sensor networks
Testbed
Experimentation platform
Mobility

A B S T R A C T

With the development of new technologies, these last years have witnessed the emergence of a new paradigm: the Internet of Things (IoT) and of the physical world. We are now able to communicate and interact with our surrounding environment through the use of multiple tiny sensors, RFID technologies or small wireless robots. This allows a set of new applications and usages to be envisioned ranging from logistic and traceability purposes to emergency and rescue operations going through the monitoring of volcanos or forest fires. However, all this comes with several technical and scientific issues like how to ensure the reliability of wireless communications in disturbed environments, how to manage efficiently the low resources (energy, memory, etc.) or how to set a safe and sustainable (both hardware and software) platform maintenance. All these issues are addressed by researchers all around the world but solutions designed for IoT need to face real experimentations to be validated. To ease such experimentations for IoT, several experimental testbeds have been deployed offering diverse and heterogeneous services and tools. In this article, we study the different requirements and features such facilities should offer. We survey the different experimental facilities currently available for the community, describe their characteristics. In particular, we detail the different hardware used for sensor networks and robot platforms and the scope of services the different facilities offer with a specific focus on testbeds which enable experimentations with mobility. We expect this survey assist a potential user to easily choose the one to use regarding his own needs. Finally, we identify existing gaps and difficulties and investigate new directions for such facilities.

1. Introduction

The Internet of Things (IoT) results from the combination of technological progresses and the new habits humans have developed facing it. By means of recent technological advances in the wireless, Internet and micro-electromechanical fields, we are now able to communicate and interact with our surrounding environment through the use of multiple tiny sensors, RFID technologies or small wireless robots. This allows a set of new applications and usages to be envisioned ranging from logistic and traceability purposes to emergency and rescue operations going through the monitoring of volcanos or forest fires. The communication establishment between (wireless) heterogeneous objects, without requiring any human-to-human nor human-to-device interaction, is a key aspect of the Internet of Things concept.

* Corresponding author.
E-mail addresses: Anne-Sophie.Tonneau@inria.fr (A.-S. Tonneau), Nathalie.Mitton@inria.fr (N. Mitton), Julien.Vandaele@inria.fr (J. Vandaele).

http://dx.doi.org/10.1016/j.adhoc.2015.03.002
1570-8705/

Nowadays, technologies have improved, becoming more complex and more efficient, and new technological challenges have emerged. The applications developed on top of these technologies need to be tested and improved before being exposed to the reality. Efficient simulation tools are useful to help in the design of IoT applications, since they offer a quick and flexible way to experiment the behaviour of an application, a protocol, and in a repeatable manner. But simulation leads to assumptions on several parameters of the environment, that is a cause of uncertainty. IoT applications and wireless propagation are seriously influenced by unpredictable events and physical characteristics that are very difficult to simulate.

There is a strong need to deploy applications in a real-life like context, therefore conducting experiments on real hardware, at large-scale, and to benefit from appropriate tools for experimentation management. Indeed, the recent technological advances have driven to cost reduction and increased availability of the hardware needed for such experimentation, thus allowing the deployment of large-scale testbeds. But experimenting on large scale requires a lot of hardware and is a fastidious and time-consuming task. Therefore, several testbed platforms have been deployed all around the world to allow faster experimentations, with various sizes, hardware, topologies, and degrees of flexibility. Some facilities focus on large-scale deployment, others on mobility. Some are quite specialised, others more flexible, allowing experimentation of purely technical issues as well as higher level applications.

This paper first defines the basic requirements a facility should address and the challenges faced up for such deployments. Then, it browses the existing available wireless sensor network testbeds, describing them with regards to those requirements. Of course, we do not pretend to be exhaustive since the number of testbed initiatives in the world is huge. We choose to focus on the currently most meaningful and active testbeds. Likewise, some additional functionalities of the mentioned testbeds may have been omitted for the sake of clarity and coherency. The interested reader is invited to refer to the cited papers for further information. This paper especially highlights wireless sensor network testbeds, with a focus on the mobility in those testbeds, and brings an up-to-date overview of the state-of-the-art. We expect this survey constitute a tool to assist an experimenter to find the adequate facility that better matches its specific needs.

The remainder of the paper is organised as follows. Section 2 defines the different requirements for an IoT experimental testbed and sets the terminology used later on in the paper. Section 3 describes the existing facilities and discusses their main purposes and functionalities. Some comparisons and highlights on strengths and weaknesses of each platform are provided, highlighting what they enable or not. Finally, Section 4 broaches the expectations from a user point of view of the next generation of testbeds and the attempts performed to fulfill the future requirements.

2. Functionalities and related challenges

Because of the diversity of wireless networking issues and applications, there exists a large variety of wireless sensor network (WSN) testbeds, that can be either specialised, or more flexible, supporting various network topologies and network layer protocol options. In any case, facilities must enable the design of as much realistic IoT experimentations as possible, in terms of scale, behaviour, functionalities, environment and constraints and offer a set of specific tools.

We have identified several services and functionalities a testbed platform is likely to offer depending on the targeted applications and protocols. They are gathered into five main categories summed up in Fig. 1: (1) Experimentation tools, (2) hardware features, (3) maintenance, (4) mobility enabling and (5) extra features, and described in more details in the following.

2.1. Experimentation

From the testbed user perspective, it is essential to benefit from assisting tools all along the lifetime of an experimentation, before, during and after running it, from the design to the result analysis. The services and tools offered to design and interact with the experiment should be easy and intuitive to take in hand.

2.1.1. Before an experimentation

Simulation: When designing a wireless sensor network application, emulation and simulation are essential steps ahead of experimentation to eliminate design issues. Some efficient simulation and emulation tools exist and are widely used, like WSNet,[1] NS-2/3,[2] and Wsim.[3]

However, simulation tools suffer from a lack of accuracy in capturing realistic environmental conditions, like radio propagation. Some wireless characteristics cannot be modelled with precision. Therefore, most of the wireless sensor network testbeds focus on real-world experimentation, since there is a vital need of facing up to the reality in that field. Some testbeds include simulation tools to alleviate the design of experiments, and to verify the consistency of a protocol or algorithm, before putting it into practice by using the testbed hardware. However, an interesting approach, detailed later in Section 3.1.3, is to combine simulation, emulation and physical elements together into a single testbed, in order to gain flexibility on the scale and the offered configurations and to lower the trade-off between repeatability, reliability and scalability.

Experimentation specification: Specification is the first step for conducting an experiment, *e.g.* the selection of the adequate resources in terms of number, type or other properties, but also the specification of the programs to upload, and the data to be collected. The way to set up an experimentation and validate the configuration is an important feature of an IoT testbed.

1 http://wsnet.gforge.inria.fr.
2 http://www.nsnam.org.
3 http://wsim.gforge.inria.fr.

Please cite this article in press as: A.-S. Tonneau et al., How to choose an experimentation platform for wireless sensor networks? A survey on static and mobile wireless sensor network experimentation facilities, Ad Hoc Netw. (2015), http://dx.doi.org/10.1016/j.adhoc.2015.03.002

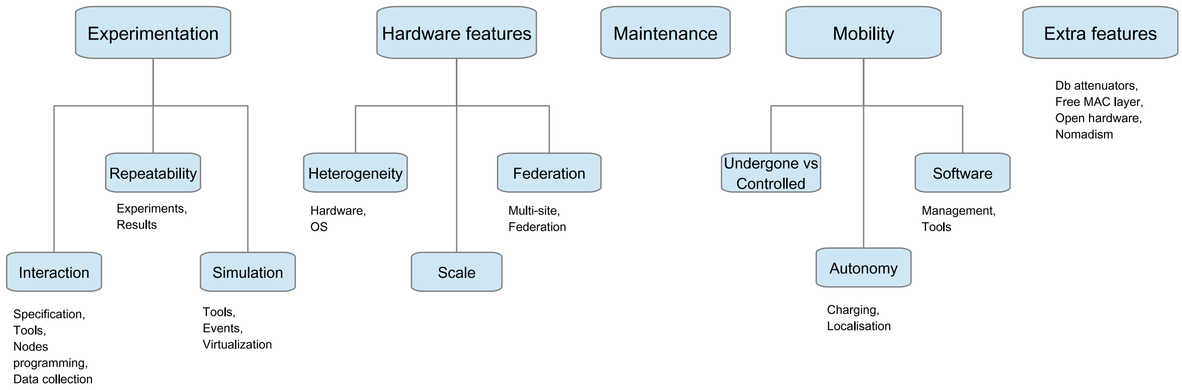

Fig. 1. Summary of the considered features for the survey.

During the specification step, and all along the experiment lifetime, the user should be provided with tools for interacting with the nodes, and its requested job. Device programming step is likely to become a tricky task when the application has to be deployed remotely on possible heterogeneous devices, with diverse execution environments. Users are offered the possibility to develop their own client applications on top of the web services API, using any programming language. Nevertheless security and availability questions have to be taken into account because of the exposure of the testbed resources over the Internet.

2.1.2. During an experimentation

Testbeds provide various interfaces to interact with the devices, and with the ongoing experiment in order to follow its progress, adjust parameters or debug. While most of the existing testbeds are accessible via web interfaces, only a few provide ssh front-end to access resources. Some testbeds also expose their resources and functionalities as web services.

During an experiment, it is also necessary to facilitate the access to sensors and to network-related metrics, such as the end-to-end delay, throughput or overhead. Some platforms offer additional tools like the possibility to monitor their energy consumption.

2.1.3. After an experimentation

Each testbed proposes different ways to collect, store and analyse data. We will see the different visualisation means and the different metrics every testbed offers to interact with the code and device resources on line.

Repeatability: There is a need, in order to validate results, to be able to repeat experiments within and across different testbeds. For instance, to analyse the influence of one specific parameter in an IoT application, an experimenter needs to run several experiments with this specific parameter varying. Then he/she is able to analyse the consequences on the results, to compare and draw conclusions. As for simulation, users have results available, consisting of data measurements and traces. Analysing these results a posteriori and drawing a conclusion from it cannot be reliably performed if the experiment had been run only once. It is essential to be able to reproduce it in the same conditions, to dispose of more representative results. Even so, it is quite challenging to make an experiment entirely repeatable, especially in wireless environments. Some real-world constraints are not fully controllable, such as local radio interference due to infrastructure, human activity, and possibly other experiments.

At the experiment level, repeatability can be achieved by standardising the experiment specification and recording it, as well as the firmwares to upload on the nodes. Even if the total real-world conditions replication is not possible, this repeatability requirement can be partially overcome by keeping the experimenter updated about the environmental conditions, and collecting traces, to help him/her to contextualise his/her results. For instance, radio interference data (possibly representing another running experiment) can be provided.

2.2. Hardware features

The hardware constitutes one of the main concerns of the user, since the goal of an experimental platform is to facilitate access and testing on real equipment, under realistic conditions and constraints. The hardware used has to match as much as possible the needs of targeted applications. This section describes the different features that can qualify the hardware offered by every platform in terms of scale and device.

2.2.1. Heterogeneity

Although experimentation of sensing applications (for monitoring for instance) do not necessarily need different kinds of devices, in general the Internet of Things concept relies on the key feature of heterogeneity, meaning that devices are made from different technologies with various sensing ranges, sensor types, different communication and computation capabilities. Making these heterogeneous devices communicate together is a user and application requirement in many experimentation designs. In response, some testbeds address this heterogeneity criteria, some others not. The diversity in devices, coming with diverse drivers, toolchains or operating systems, drives to the need of easy programming and configuration

Please cite this article in press as: A.-S. Tonneau et al., How to choose an experimentation platform for wireless sensor networks? A survey on static and mobile wireless sensor network experimentation facilities, Ad Hoc Netw. (2015), http://dx.doi.org/10.1016/j.adhoc.2015.03.002

of these heterogeneous devices. Consequently, means for programming the devices that fill the gap between the user and the remote hardware, like drivers, communication libraries, or operating systems porting, may be provided.

2.2.2. Scale

Another prominent property of an IoT testbed is the scale, *i.e.* the number of devices available for experimentation. IoT systems and technologies apply on much more than a few dozen of devices. Indeed, users should have the opportunity to run some applications on real hardware at very large-scale, as some technological challenges can only be addressed that way *e.g.* applications designed for smart cities that imply several hundred of nodes. Building such facilities is now made possible since the hardware cost has decreased. Nevertheless, it comes with constraints like lowering human intervention as much as possible and implementing automated fault management mechanisms. Furthermore, scalability is desirable to easily expand and update hardware, and the possibility to include more recent devices, in a plug-and-play manner.

2.2.3. Federation

In general, building up large-scale testbed platforms is made possible by multi-site deployments, or by the federation of existing testbeds. In this survey, we distinguish a multi-site testbed from a federation of several testbeds.

In a multi-site testbed, *i.e.* the resources are distributed over different locations, with a unique access point to all the nodes. It is most often the same type of devices deployed to reach a significant scale for experimentation.

The federation approach allows users to be provided with a large number of devices, to numerous capabilities, while sharing the required management of resources, and to overcome the constraints of space and hardware cost. It can also enlarge experimentation possibilities by federating testbeds in different fields. In a federation, several testbeds on different locations are independent and can work alone although there is an additional layer to offer a uniform access. The federation of existing testbeds requires a common framework, a layer built on top of all existing infrastructures, allowing the user to authenticate and reserve resources on every interconnected testbed simultaneously.

2.3. Maintenance

The testbed maintenance occurs at several frequencies. A daily maintenance needs to be considered to ensure the good functioning of the testbed and experimentation. A more general maintenance is also needed to verify that the hardware is operational and that the software architecture still offers appropriated services.

To alleviate the maintenance, the testbed design needs to consider to lower as much as possible the human intervention for the testbed management, in order to limit the resulting cost of extra human resources involved to prevent hardware malfunctioning or even damages and take care of the health of the testbed. In parallel, a constant monitoring of the software is important to prevent from crash and constantly update firmwares and services.

2.4. Mobility

IoT applications, mainly in the smart cities field, involve mobile devices that collect information on the environment, or cooperate with each other, which leads to the design and implementation of robotics and automation systems consisting of networked vehicles, sensors and actuators. Regarding the need for experimenting IoT applications on real-world entities, the mobility of the devices becomes an essential feature of IoT testbeds. This topic has received important attention recently. For instance, to experiment applications for area coverage, or protocols allowing to find better routes for packets, to enhance data reliability, users need to access and control mobile devices in a testbed environment.

2.4.1. Undergone vs controlled mobility

Several types of mobility exist: undergone or controlled mobility. We refer to undergone mobility for a device which is embedded on an object/person that cannot be controlled by the device itself. Such undergone mobility can either be non predictive (or partially predictive), in the case for instance of human or animal carriers, or predictive, for example if the devices are carried by public transport, like a bus, which has a predefined and known journey.

While mobility offers larger possibilities in terms of applications, handling such a feature and providing adequate support for controlling an experiment and exploiting results is a real challenge. Some testbeds have introduced undergone mobility, predictive or not [1,2]. We will also see that some testbeds provide controlled mobility [3], which implies additional constraints in terms of localisation and charging, and also because a dedicated space could be needed, to avoid sudden obstacles like a person walking around.

2.4.2. Autonomous charging & localisation

Because of the mobility, potential collisions may happen causing damages to the hardware and interrupting a running experiment prematurely. Therefore, the testbed should autonomously run, with a remote access provided to users that allow them to perform experimentation without compromising the safety of the hardware thus the continuity of the experiment. This leads to several material constraints: the robot should be self-rechargeable (and empowers the embedded device at the same time), and able to locate and reach the charging system. Furthermore, mobility requires autonomous localisation and path planning with obstacles avoidance. Therefore, an accurate positioning mechanism is needed to overcome these issues. Section 3.3 describes some existing localisation solutions.

2.4.3. Software management & tools

Another important aspect is the design of an experimentation that uses mobile devices, which can become a tedious task especially for users who are not robotic experts and who want to integrate basic moving functionalities in their experimentations. Therefore, an essential requirement for IoT testbeds that feature mobility is to ease the implementation of scenarios, and the interactions with the hardware. For robot control, there are several drivers and frameworks that provide services, hardware abstraction, low-level device control, and so on. Some testbeds use these control interfaces, some others have developed their own middleware. Testbeds should provide various mobility models that are ready-made and possibly customizable. Finally, it is paramount, when conducting experiments including mobile devices, to dispose of visualisation tools that display the running experiment state, but also past experiment configurations, like the paths the mobile devices had actually followed, and to ensure repeatability.

In that survey, we browse the existing mobile sensor network platforms which offer either undergone mobility, or full control on devices movement. We will see how existing testbeds comply with these requirements.

2.5. Extra features

There also exist some other valuable features in the surveyed testbeds that deserve to be mentioned. We will browse some of them like the possibility to monitor the physical medium and hardware layer. We will also see that some facilities are nomadic.

3. Survey of existing testbeds

In this section, we discuss how today's testbeds feature the requirements and address the challenges detailed above. We will compare them in terms of capabilities and features they offer, in accordance with the categories of needs previously described. We survey wireless sensor network testbeds that are technologically well-advanced or widely used, with a focus on mobility for some of them. However, we did not restrict the survey on the publicly and remotely accessible testbeds, since some testbeds provide very interesting features without offering remote access to their resources, but the work of making them available to remote users is still in progress, or leads to issues that have not been overcome yet. Tables 1 and 2 sum up the hardware and key features of the testbed platforms detailed below.

3.1. Experimentation

3.1.1. Interaction

Nodes selection, configuration storage: Most of the existing testbeds (FIT IoT-LAB, TWIST, Kansei, NetEye, SmartSantander or WISEBED) provide web interfaces for job scheduling, to specify the resources needed, set up and program the nodes to define their behaviour and the data to be collected. FIT IoT-LAB (IoT-LAB in short in the following) benefits from a well designed system for nodes selection: the nodes can either be physically selected, or user can specify needed properties (location, radio chip, whether nodes are mobile or not, etc.). To store the experiment configuration, testbeds generally use XML or JSON files (FlockLab, ISRobotNet, IoT-LAB). WISEBED goes further with its own generic XML-based language called WiseML, used for experiment and testbed description, configuration, and results storage.

Scheduling system: Most of the testbed platforms are multi-users, thus the possible concurrent reservation requests from several users are in general handled by a first-come-first-serve approach, meaning that the first user submitting an experiment on available resources gets the access (NetEye, IoT-LAB, WISEBED). For instance, IoT-LAB's scheduling system relies on the open-source OAR tool[4] (resource manager & batch scheduler). However, other testbeds remain single-user, like CONET-IT, which is the only mobility-focused testbed that is remotely accessible together with FIT IoT-LAB. Thus, it does not benefit from any scheduling system, and the user has to book the entire testbed for an experiment two weeks in advance.

Nodes configuration & programming: To configure and program the selected nodes for the experimentation, a user may need to log on the testbed's server. To this purpose TWIST or IoT-LAB provide ssh access to start, stop, reset, update the nodes, and read or write on the serial links. CONET-IT offers a virtual private network to interact with the Integrated Testbed (IT) during the experiment. SmartSantander and WISEBED have developed experimentation scripts that consist of a set of command-line scripts to execute basic operations, control and interact with the experiment, automate, repeat experiments, and even programmatically analyse, convert and process output from the nodes. In SmartSantander, some nodes called "service nodes" (the most battery-constrained nodes), only produce data and can be configured by the administrators of the testbed, but are not open to be reprogrammed by users. However, users are able to develop new services on top of the data generated by these service nodes.

General interaction: For general interaction with the testbed, every testbed provides at least a command-line tool, and more commonly a web-based front-end. Although having been decommissioned, the service framework of the very popular WSN testbed MoteLab, (Harvard University) remains a basis for various other testbeds, *e.g.* INDRIYA. This testbed of the National University of Singapore is considered as an alternative to MoteLab. Using the web interface it has inherited from MoteLab, users can upload, monitor, and control their jobs remotely and in real-time. Regarding the testbeds focused on mobility, the need of an efficient and interactive visualisation tool is crucial. Some testbed platforms have developed their own tools, corresponding to their own needs: CONET-IT's Integrated Testbed GUI allows the visualisation of the experiment and data log (programming the WSN nodes and robots, graphically setting waypoints to the robots, accessing the camera and laser views, data logging, etc.). Mint-m's network/experiment management subsystem, called MOVIE (Mint-m cOntrol and Visualisation InterfacE) affords a user full interactive control over the testbed as well as real-time visualisation of the testbed activities. For further analysis of an experiment, WISEBED has created WeyesBED, an open-source visualisation tool for WiseML files, thus allowing interactive visual description, and representation of distributed algorithms and protocols. Key-events leading to problematic states or bugs

[4] http://oar.imag.fr.

Please cite this article in press as: A.-S. Tonneau et al., How to choose an experimentation platform for wireless sensor networks? A survey on static and mobile wireless sensor network experimentation facilities, Ad Hoc Netw. (2015), http://dx.doi.org/10.1016/j.adhoc.2015.03.002

Table 1
Surveyed WSN testbeds.

Testbed	Hardware summary	Notes
FIT IoT-LAB [2]	2728 heterogeneous, specifically developed motes located in 6 different sites across France	SensLAB follow-up, repeatable mobility via electric toy trains, energy consumption measurement, multi-site experiments, part of FIT federation
TWIST [4]	204 motes (102 TmoteSky + 102 eyesIFX) spread across 3 floors	Supports flat and hierarchical setups, emulation of dead nodes or nodes addition, cost-effective and open solution which can be reproduced by others
Kansei [5]	210 XSM motes: large grid-like structure of motes evenly distributed on tables within a warehouse	Supports various wireless platforms such as Extreme Scale Motes (XSMs), TelosB, Imote2 and Stargates, event injection possible both at GW and mote level
NetEye [6]	130 TelosB motes, indoor	Static 3db attenuators are attached to the mote antennas for multi-hop network and different power levels
SmartSantander [1]	20000 (fixed, mobile & smartphone) sensors located in 4 different countries, indoor and outdoor	Most advanced testbed in terms of hardware, scale, functionalities offered to the user, mobility via public buses Multi-site and real-life experiments
WISEBED [7]	750 motes, mainly iSense, MicaZ, and Pacemate, SunSPOT, and TelosB motes	Large federation that includes some SmartSantander testbeds, simulator engines that create virtual testbeds
DES-Testbed [8]	95 nodes: embedded PC board, equipped with up to 3 IEEE 802.11 network cards, and a wireless sensor node	Virtualizer running several virtual machines that recreate the testbed topology and its lossy links 1 mobile DES-Node node using a Roomba 530 Vacuum cleaner robot
FlockLab [9]	30 observers and 1 server spread across one level of a building at ETH Zurich and the surrounding rooftops	Sensor node pairing with dedicated hardware for monitoring and simulation
INDRIYA [10]	139 TelosB motes spread across 3 floors	Experiment prototyping with TOSSIM simulation environment, web-based interface designed based on Harvard's MoteLab interface, nodes replacement with Arduino motes
W-iLAB.t [11]	200 Tmote Sky + 60 more powerful nodes in 2 different locations	Different types of wireless nodes: sensor nodes, Wi-Fi based nodes, sensing platforms, and cognitive radio platforms, uses the Emulab software at is base

Table 2
Surveyed WSN testbeds focused on mobility.

Testbed	Sensors	Robot platforms	Localisation solution	Software	Remote access	Notes
FIT IoT-LAB [2]	3000 static nodes: WSN430, A8, M3	200 robots: wifibots, turtlebots	decentralised, kinect	ROS and Player Stage	yes, multi user	different pattern motions with obstacle avoidance or total control by user
CONET-IT [3]	WSN of 21 static nodes: TelosB, MicaZ, Mica2, Iris	5 Pioneer 3-AT[a] + Aspire One ZG5 netbook + Hokuyo 2D Laser + Kinect + Wireless a/b/g/n Bridge	Vision-based with AMCL, decentralised	Player Stage (ROS porting planned)	yes, single user	random motion with obstacle avoidance
RoombaNet [12]	WSN of 6 static nodes	6 Roomba[b] + mobile controller, extension up to 14 robots planned	Odometry and orientation sensor-based positioning mechanism	Wiselib ported to the sensor node platform	No, opening to remote users planned	2 mobility models: random & semi-random Wiselib extension to support
MOTEL [13]	TelosB, MicaZ, Scatterweb	22 e-puck[c] + Thymio II[d]	Vision-based with cameras, centralised	FLEXOR MuRobA	No	No communication sensor-robot, software for interactive control, not permanently installed
Mint-M [14]	Wireless network node supporting 802.11 interfaces	12 Roomba	Vision-based with cameras, centralised	MOVIE (Mint-m cOntrol and Visualization InterfacE)	No	Testbed can operate 24×7 without human intervention for weeks deployable in a limited physical space (radio signal attenuation), strong simulation tool
ISRobotNet [15]	?	4 Pioneer 3-AT + 1 ATRV-Jr[e]	Vision-based with cameras, centralised	YARP networking software	No	Fully decentralized use of the resources

[a] http://www.mobilerobots.com/ResearchRobots/P3AT.aspx.
[b] http://www.irobot.com/us/learn/home/roomba.aspx.
[c] http://www.e-puck.org.
[d] https://aseba.wikidot.com/en:thymio.
[e] http://www.ing.unibs.it/arl/projects/minerobots_archive/atrv_jr.pdf.

can potentially be detected thanks to that monitoring, which helps users to achieve an internal insight on the behaviour of their algorithms.

In addition to a web-based front-end, for more advanced and specific interactions with the testbed, some expose their services via a web services API (WISEBED,

Please cite this article in press as: A.-S. Tonneau et al., How to choose an experimentation platform for wireless sensor networks? A survey on static and mobile wireless sensor network experimentation facilities, Ad Hoc Netw. (2015), http://dx.doi.org/10.1016/j.adhoc.2015.03.002

SmartSantander, IoT-LAB). Users are able to develop their own tools, scripts in any programming language that is able to invoke web services. WISEBED even provides a selection of open-source Web- and Desktop Clients, that can be adapted by users for their specific needs.

Data collection & experiment analysis: Finally, data collection and experiment analysis are essential steps of the experimentation process. Most of the time, users can request the nodes during an experimentation, to get measurements when needed. But the testbed platforms also record the whole sensing data that users can access upon job completion, for processing and visualisation. IoT-LAB gathers resulting periodic measurements coming from sensors into CSV files. Databases are also used (SmartSantander, INDRIYA and W-iLAB.t), thus enabling persistent storage of observations and measurements, as well as live and historic information. SmartSantander also provides a tool called TMON, a Java-based experimentation environment, allowing the visualisation of traces and live results.

3.1.2. Repeatability

To help users repeat their experiments, most of the testbeds save the experiment description, and propose, like FIT IoT-LAB, to reload the configuration to run it again. SmartSantander and WISEBED made the automated repetition possible thanks to experimentation scripts.

However, full reproducibility of experimentation is not trivial because of the environmental conditions inconsistency. A solution (SmartSantander) is to provide relevant data for repeatability, such as statistics on observed radio interferences that have been gathered over the lifetime of an experiment, to assist users in understanding and contextualising their results. WISEBED proposes seamless traces replay of an experiment described in WiseML. Finally, Kansei features events injection, enabling the repetition of an experiment in the exact same conditions, with emulation of data read by sensors.

3.1.3. Simulation

Although simulation is not part of the experimentation process, it remains an essential step for prototyping an application. Testbed users are often advised to use a simulator ahead of experimentation (INDRIYA). Especially for testbeds that feature mobility (CONET-IT), simulation (on Gazebo[5] or Stage[6]) is mandatory before experimenting on the testbed.

Some testbeds are linked with simulator that permits to deploy the same code both in simulation and experimentation, like TWIST: the Cooja-TWIST plugin lets experimenters use the testbed directly from the Cooja[7] simulator, facilitating the code upload and the supervision of the experiment execution remotely. Mint-m's MOVIE tool also deserves to be mentioned, since it allows the control of hybrid ns-2 simulations run dynamically. It includes functionalities like pausing a simulation run at a user-specified breakpoint, deeply inspecting parameters of the system (internal states, network conditions), and even a rollback mechanism to go back to a previous state of the simulation.

Another approach is the emulation of events, characteristics, into the experiment in order to extend hardware capabilities, when some features are not physically supported on the actual nodes. Examples of emulated events (TWIST and FlockLab) are the death of nodes, by energy depletion, or the addition of new nodes, enabling the modification of the network topology. Kansei and FIT IoT-LAB testbeds make available the event injection, *e.g.* simulation of data read by the sensors at large-scale. This can be useful to close the possible hardware gap in a specific experimentation, or even recreate environmental conditions in order to repeat an experiment. SmartSantander also allows this virtualization of the sensing capabilities. WISEBED goes further, providing the functionality to create virtual links between nodes, thus dynamical experiment specification, composed of physical or virtual nodes and virtual links. These hybrid testbeds can be accessed and controlled by common tools, in a transparent way: the experimenter can access the nodes seamlessly, whether they are physical or simulated. With this aim, WISEBED exposes iWSN[8]: a web service interface that provides an uniform management access to the hybrid testbed created by the user. Physical, emulated and simulated elements of wireless sensor networks are then mixed together to offer a more flexible testbed for experimentation.

3.2. Hardware features

3.2.1. Heterogeneity

Facilities have deployed a variety of hardware, where motes can be off-the-shelf as well as custom-built for a specific need.

Fig. 2 draws the distribution of motes available on every testbed platform.

It is shown that only a few testbeds (NetEye, DES-Testbed) offer a single kind of motes. Most of the testbeds have deployed TelosB[9] motes, that integrate a TI MSP430 microcontroller, a Chipcon CC2420 radio, as well as the usual light, temperature and humidity sensors.

Other testbeds have designed their own motes accordingly to their specific needs. The core of the DES-Testbed is formed by the IEEE 802.11 network, while the wireless sensor nodes, custom MSB-A2 sensor nodes (developed at Freie Universitat Berlin), create a parallel testbed called DES-WSN. This custom sensor node is equipped with a Chipcon CC1100 transceiver as well as temperature and humidity sensors. FIT IoT-LAB has also deployed custom-built motes, based on TI MSP430, and more recently on ARM Cortex M3 and ARM Cortex A8 microcontrollers, more powerful and equipped with other sensors such as accelerometer, magnetometer and gyrometer.

Fig. 2 also depicts heterogeneity in most of the testbeds. SmartSantander and WISEBED offer a large variety of devices, since deployment is not only indoor, as for all previously described testbeds, but also outdoor and in vivo. Indeed, SmartSantander offers smart city services,

5 http://gazebosim.org.
6 http://playerstage.sourceforge.net/?src=stage.
7 https://github.com/contiki-os/contiki/wiki/An-Introduction-to-Cooja.
8 http://wisebed.eu/api/documentation/iwsn/current/iwsn-api.pdf.
9 http://telosbsensors.wordpress.com.

Please cite this article in press as: A.-S. Tonneau et al., How to choose an experimentation platform for wireless sensor networks? A survey on static and mobile wireless sensor network experimentation facilities, Ad Hoc Netw. (2015), http://dx.doi.org/10.1016/j.adhoc.2015.03.002

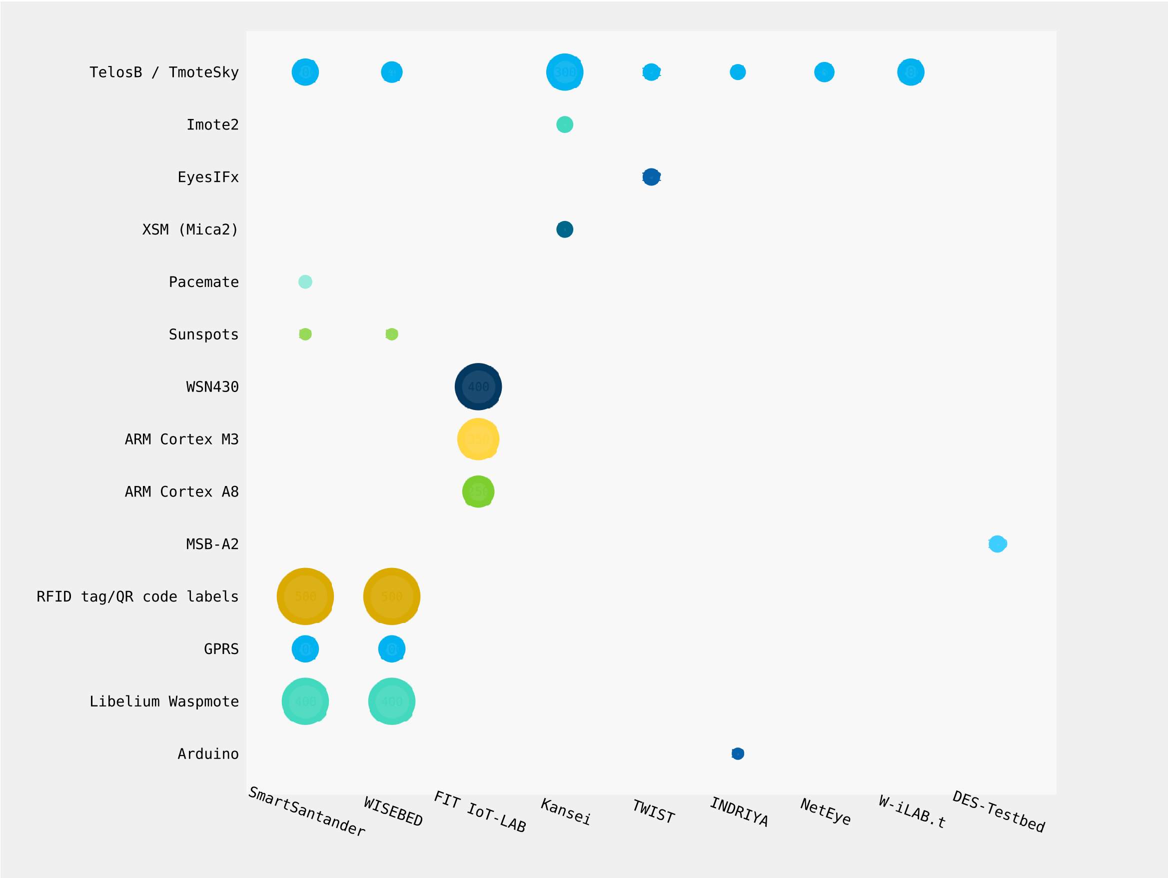

Fig. 2. Scale and composition of surveyed testbeds; size of circle is proportional to the number of devices.

involving the citizens into the experimentation loop. With this aim, IEEE 802.15.4 devices, GPRS modules, and joint RFID tag/QR code labels are deployed both at static locations (streetlights, façades, bus stops) as well as on-board on mobile vehicles (buses, taxis). WISEBED provides as many heterogeneous devices, and a wide range of sensor types, ranging from most commonly used temperature sensors to more sophisticated Anisotropic Magnetoresistance (AMR) sensors.

One of the concerns of a testbed user is the execution environment on the motes. Some testbeds impose an operating system: on TWIST, NetEye as well as INDRIYA, the nodes run TinyOS only. However, FIT IoT-LAB, SmartSantander and WISEBED offer to users more freedom in the development of their applications, with no mandatory operating system. Indeed, FIT IoT-LAB provides drivers, MAC layers, communication libraries, and OS porting for Contiki, FreeRTOS, TinyOS, and RIOT. We can finally mention Wiselib,[10] an algorithm library for sensor networks maintained by the WISEBED team. It contains various algorithms classes that can be compiled for several sensor platforms (iSense, Contiki) or the sensor network simulator Shawn, and helps users in the design of experimentations that involve motes amongst the seven different hardware platforms provided by WISEBED.

3.2.2. Scale

Fig. 2 also gives at a glance an idea on the scale of every testbed platform. Actually, the fact is that most of the single location testbeds (*e.g.* INDRIYA, TWIST) feature a limited number of nodes (up to 200 nodes), most likely due to cost and space constraints. Nevertheless, Kansei succeeded to deploy about 700 motes in a single location, an indoor grid-like structure of motes evenly distributed on tables within a warehouse. However, the largest testbeds are the distributed ones, such as WISEBED which count up to 750 motes, while FIT IoT-LAB and SmartSantander have deployed several thousands of nodes.

3.2.3. Federation

The FIT IoT-LAB testbed is multi-site, spread across 6 different locations in France and offering forward access to 2728 wireless sensor nodes. It is composed of previously named SensLAB testbeds,[11] plus latest deployments of relatively new wireless sensor devices. It is an actual distributed testbed which offers the functionality to experiment on

[10] www.wiselib.org.

[11] http://wiki.senslab.info.

several different locations and on different hardware at the same time. However, note that FIT IoT-LAB is also part of the FIT federation. The FIT (Future Internet of Things) Equipex is a federation of FIT IoT-LAB, OneLAB[12] and CorteXlab.[13]

As for WISEBED, this 3 year-old project is a joint effort of 9 academic and research institutes, and aims to provide a federation of large wireless sensor network testbeds across Europe. The federation comprises about 750 devices supporting a range of sensor modalities. On each of the 9 sites, a server exposes the testbed capabilities through an iWSN interface, providing a uniform access to every site. Each site differs in its choice of hardware, software, and physical layout, providing a very heterogeneous large-scale testbed.

Finally, the largest federation in terms of number of devices is SmartSantander. This project proposes a city-scale experimental research facility in order to support applications and services for smart cities. Indoor, outdoor as well as in vivo devices are deployed across 4 different cities in Europa: Belgrade (Serbia), Guildford (UK), Lübeck (Germany) and Santander (Spain). Some of the testbeds are accessible through the already mentioned WISEBED experimental facility thus are also part of this federation.

3.3. Mobility

In this section, we examine how mobility is provided by different testbeds. The majority of the facilities does not offer any type of mobility, some others have introduced mobile nodes into an existing grid of fixed nodes, or are currently dealing with that issue.

3.3.1. Undergone vs controlled mobility

Undergone mobility, as described in Section 2.4, is provided by some testbeds but for most of them, the experimenter has no control over it as for instance in SmartSantander where nodes are deployed in the cities of Belgrade and Pancevo, embedded into public transportation vehicles, like buses and taxis. These nodes are remotely accessible for experimentation and can be used to determine the location of the vehicles, estimate arrival time to bus stops, or make atmospheric measurements but the user has obviously no control on buses. But in FIT IoT-LAB, some nodes are embedded on 4 electrical toy trains maneuvring on 2 separated circuits and on more than 200 robots (Wifibots, Turtlebot2 and Roomba). The mobile nodes are part of grids of between 256 and 1000 nodes, enabling experimentation of scenarios involving both static and mobile nodes. In these specific cases, the user can choose the speed of the trains, and/or the mobility patterns of each robot.

Offering controlled mobility is an even more difficult issue. Some testbeds have introduced controlled mobility by means of robot platforms carrying the nodes. Whatever the kind of robots used, they can generally move at a speed oing up to 18 km/h (0.5 m/s, 18.5 in./s). But the testbeds do not offer remote access to this infrastructure except FIT IoT-LAB and CONET-IT. To date, most of testbeds have not deployed more than a few robots. While enlarging the scale is a goal for many of them, they usually adopt an incremental approach, solving technical problems. For instance, on the DES-Testbed, only one mobile node embedded on a iRobot Roomba vacuum-cleaner robot is running, still as a prototype, but it is planned to extend it with other robots and to provide different mobility patterns. The manufactured iRobot's Roomba vacuum-cleaner robot is the most widely used for introducing mobility in testbeds, since it is low-cost and self-operating, and has auto-recharging capabilities. WISEBED, MINT-m, FIT IoT-LAB and RoombaNet have a few of them available either occasionally (WISEBED) or permanently (RoombaNet plans to extend its fleet to 12 robots). Other testbeds have introduced mobility using different robot platforms, as summarised in Table 2. But for all these testbeds, scheduling and timeslices to the remote users remain to be clarified. This limitation is truly understandable since it is a very challenging issue to handle the sharing of these mobile resources between several users, in terms of space and infrastructure, with ensuring the safety and consistency of the testbed.

A more advanced testbed since it is remotely accessible to users, and works widely autonomously is the CONET Integrated Testbed, developed at the University of Sevilla (Spain). This open facility has deployed about a hundred heterogeneous wireless sensor motes, some of them piggy-backed on 5 Pioneer 3-AT mobile robots. Some extensions in terms of sensors and robot platforms are scheduled. However, it does not handle concurrency since one experimenter has to book the entire testbed.

Going further, FIT IoT-LAB also offers several nodes embedded on robots (up to 200 robots), either Roomba, Turtlebot2[14] or Wifibot,[15] depending on the site. These robots and their embedded nodes are included in the reservation and scheduling system and are managed as any other resource. They can be used to provide undergone mobility to their embedded nodes, using predefined mobility patterns. Users can also get full control of them, and design entirely their solution integrating the exact needed mobility, since the robots are able to interact with their embedded node.

3.3.2. Autonomous charging & localisation

Most of the testbeds have chosen to deploy off-the-shelf robots, depending on their needs. Some functionalities are already provided, however the customisation of these robots to adapt them to a wireless sensor network testbed remains often necessary, as well as software solutions to allow them to autonomously manoeuvre in a constrained space.

For instance, off-the-shelf robots are not provided with the appropriate connectors for plugging the carried wireless sensor node. It is also necessary to modify their auto-recharging circuitry to power up both the robot and the mote. Moreover, MINT-m has developed for Roomba robots a residual power algorithm that is able to estimate the amount of energy left, and a recharge scheduling

12 https://www.onelab.eu.
13 http://www.cortexlab.fr.
14 http://www.turtlebot.com.
15 http://www.wifibot.com.

algorithm, thus allowing testbed robots operations without human intervention for weeks, in a 24 fashion. More advanced robots, like the Turtlebot2 deployed in FIT IoT-LAB, are targeted for research and education purposes, and provide functionalities like auto-recharging. FIT IoT-LAB has also deployed Wifibot robots, for which a specific auto-recharging solution has been designed. In the other testbeds, human intervention is almost always required at one point or another to charge the robots.

For the robots to move autonomously in an area and to avoid obstacles, an accurate positioning mechanism is necessary. Therefore, several approaches exist, either centralised *i.e.* relying on a central server that computes the position of each robot, or distributed *i.e.* each robot is able to self-localise.

RoombaNet controls the directions and distances using the odometry data and an additional orientation sensor. Its developers are currently working on improvements on the control software to enhance the accuracy of the movements. Going further, other facilities (MINT-m, MOTEL, ISRobotNet) rely on more advanced systems for robots positioning like a vision-based system with cameras: overlooking cameras are mounted on the ceiling, and a central server computes images and identifies the positioning of each robot.

CONET-IT and FIT IoT-LAB have opted for a distributed approach for localisation: a robot is equipped with a laser or a kinect and uses the probabilistic system AMCL (Adaptive Monte-Carlo Localisation) to localise itself. The algorithm uses a particle filter captured by the laser as well as the odometry data to track the pose of the robot into the map of the environment. Therefore, each robot is able to autonomously localise itself, in a decentralised manner. The robots carry powerful enough netbooks for computing and control.

3.3.3. Software management & tools

Managing a mobile wireless sensor network testbed and offering the appropriate tools to allow the user to remotely design and control its experimentations is not trivial.

Some of the surveyed testbeds have developed their own tools. RoombaNet has ported Wiselib (mentioned in Section 3.2.1) on its sensor nodes platform and for the mobility control, they have developed a driver that encapsulates Roomba's basics movements. RoombaNet developers plan to extend Wiselib to support this control software for the robots. This will enable users to access the testbed remotely and to write WSN algorithms that can control the movements of the robots. MOTEL has developed so-called FLEXOR, a software architecture to control sensor nodes that run on top of TinyOS. FLEXOR is platform-independent and has extensive graphical support to programme and manage WSNs. The control of the robots is handled by a multi-robot architecture for coordinated mobility.

Other testbeds use open source software libraries and tools dedicated to robot and sensor applications: Player[16] (CONET-IT) allows multiple heterogeneous devices to present the same interface, and is language and platform-independent. Its module manages the WSN messages that should comply with the WSN-Player Driver interface. On the other hand, FIT IoT-LAB has deployed ROS[17] for the mobility control. This software benefits from an increasing community of developers, and offers hands on solutions and algorithms, thus expanding the possibilities in terms of experimentations, and the work sharing between users. Indeed, CONET-IT providers envision the migration to ROS software. Furthermore, on both testbeds, sensors can communicate with their carrier robots, enabling cooperative mobility where the robot follows sensor instructions (whereas for now on MOTEL, the robot is not able to communicate with the piggybacked sensor, it does not share energy nor other resources). Full control of the robot is given to the user, enabling experimentations both in the wireless sensor network area and in the robotic area.

To help users in the design of an experiment involving mobile sensors, mobility models are often provided. It is also useful to ensure a relative repeatability of the mobility. Mobility, as previously mentioned, can be provided either in an undergone or controlled way. Repeatability is easily feasible in the context of undergone mobility. For instance, using trains (FIT IoT-LAB), it is easy to repeat the exact same experiment since trains are always moving on the same circuit. The experimenter just has to set the same starting point and speed. However, when mobility is more complex, mobility patterns are necessary. Some testbeds provide predefined patterns. An example of mobility model (RoombaNet) is the random motion: a "semi-random" model allows the experimenter to modify the level of randomness in the pattern. Experimenters can also establish predefined paths, load them on the robots and reuse them. CONET-IT and FIT IoT-LAB allow the paths modification by interactively moving, adding or deleting waypoints.

As previously mentioned in the mobility requirements (Section 2.4), in such a context, it is essential to have visualisation tools available, to study the dynamics of an experimentation in a testbed in real time. Therefore, MWSN testbed providers have made a sizable effort to develop tools that enable, even remotely, a vision on the proceedings of an experiment. It is for instance possible to display real-time node position and physical environment throughout the experiment. MINT-m's MOVIE interface, mentioned in Section 3.1.1 also enables the collection of the nodes path traces, real-time display of network traffic load distribution as well as inter-node signal-to-noise ratios. Despite everything, there remains some degree of uncertainty when running a mobile WSN experiment. To overcome this, MOTEL provides a graphical post-experiment analysis tool, that allow the analysis of the actual proceedings of an experiment.

3.4. Extra features

In the following, we mention some interesting additional features offered by testbeds.

[16] http://playerstage.sourceforge.net/index.php?src=player.

[17] http://www.ros.org.

Please cite this article in press as: A.-S. Tonneau et al., How to choose an experimentation platform for wireless sensor networks? A survey on static and mobile wireless sensor network experimentation facilities, Ad Hoc Netw. (2015), http://dx.doi.org/10.1016/j.adhoc.2015.03.002

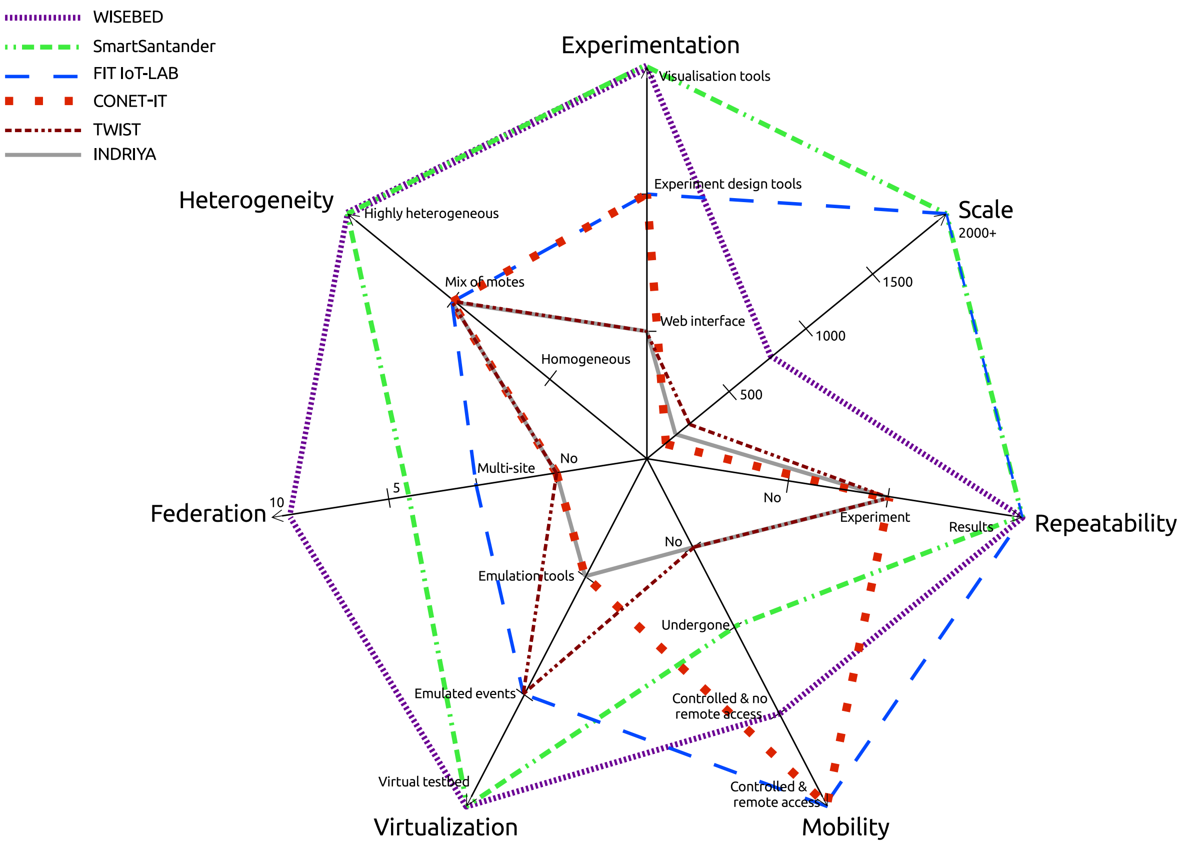

Fig. 3. Overview of the strengths and weaknesses of the main surveyed testbeds.

3.4.1. Radio signal attenuation

The goal of a wireless sensor network testbed is to offer a facility to experiment research protocols in a real-world like context. In order to get closer to the reality, most of the testbeds target large-scale deployments. In indoor environments and small spaces, the transmission ranges are higher than in outdoor environment in a real context. Therefore, some testbeds like NetEye and MINT-m have attached static db attenuators to each mote antenna to realise multi-hop network and different power levels. Radio signal attenuation allows the reduction of the signal coverage and thus of the physical space.

3.4.2. Free MAC layer

A way to offer more freedom to users in the design of their applications is to let them choose the MAC layer, like in FIT IoT-LAB or Kansei. IoT-LAB provides different MAC layer implementations, users are free to choose or to test their own, for example to observe its impact on the network lifetime or throughput.

3.4.3. Open hardware

Going further, to offer more freedom to experimenters in the choice of hardware, and to allow them to experiment with their own motes, some testbeds provide an open facility on which users can plug their own motes, assuming that they are compliant with the existing infrastructure. Users can then benefit from the hardware and software infrastructures provided by these facilities to manage their experiments: automatic deployment, experiment replay, interaction with the nodes...

FlockLab testbed consists of 30 observers organised in a mixed indoor/outdoor topology. Users are free to attach up to 4 arbitrary wireless sensor nodes over this generic hardware interface. Currently, there exist target adapters for Tmotes, TinyNodes, Opal nodes and MEMSIC Iris[18] nodes. FIT IoT-LAB proposes 96 free slots to plug any new hardware through USB interface.

3.4.4. Nomadic testbeds

Some testbeds are not necessarily permanently installed. MOTEL, for instance, is not a typical testbed. It is a robotic-assisted mobile wireless sensor network testbed which can be installed in any indoor environment in several hours for a particular experiment. This brings flexibility in the platform topology, since the number of robots is unlimited and the used robotic platforms can be exchanged easily. MOTEL provides both software frameworks to manage the WSN and the robot platform. When the testbed is not installed, the cameras and robots are used for other, non-WSN experiments and research.

[18] http://www.memsic.com/wireless-sensor-networks.

4. Concluding discussions

With the development of new technologies, these last years have witnessed the quick development of IoT and WSN. Many platform initiatives have thus pop up to help the design of protocols for IoT and WSN. But designing, deploying, setting up and then maintaining such platforms is a very difficult task as testifies the fact that large number of used testbeds have been decommissioned (MoteLab, Motescope, TrueMobile). This article describes some of the most currently active and meaningful testbeds for IoT related research.

To summarise, Fig. 3 features their different characteristics at a glance. It shows that most of the testbeds are heterogeneous and multi-purpose, offering an increasing set of services to answer a large range of user expectations, even though none of them manages to fulfill all requirements, especially in terms of maintenance, since testbeds need continuous updates to follow the technology advances. SmartSantander and WISEBED are the two best answering best on criteria we listed, especially in terms of scale, heterogeneity, and possibilities of experimentation. Even if FIT IoT-LAB is slightly set back, it is strongly oriented to scale and repeatability criteria, while answering at the same time to mobility concerns. Like CONET-IT, it features heterogeneity in terms of motes and mobile platforms, it is remotely accessible and the mobility can be fully controlled in a decentralised way. However, if the user focuses his work on mobility only, and needs to have full control of several mobile devices involved in his experimentation, CONET-IT could be the best choice regarding experimentation means. On the other hand, if a large scale and possibly mobile devices are not required for the user to experiment its applications in an indoor environment, INDRIYA and TWIST are valuable testbeds since they offer the basics functionalities of a WSN testbed, an interesting three-dimensional deployment of reliable and widely used motes.

Surviving testbeds are wonderful tools for the IoT solutions developments. Most of the ones discussed in this article follow the technological trends and have been continuously opening new services in order to integrate societal behaviour modifications and arising challenges. We can discuss potential future new directions and services they might consider. First, the inclusion of actuators could, not only allow the user to visualise an experiment (in the case of the LEDs) but also to act on the environment and change some of its characteristics by opening doors, turning on lights, etc. This could open the way to perform experimentations for house automation or personal health monitoring applications. Also, to address some new hot research topics like security and safety issues or energy harvesting integration in network decision, new hardware should feature some energy harvesting components (solar cell, piezo cells, etc.) and new tools should be designed to ease the tests for security and safety research purposes.

The purpose of the IoT is to make objects and environment able to communicate, to recognise themselves and to exchange data. Therefore, another direction to extend and improve testbeds could be the introduction the RFID technology into the experiments, thus making possible automatic identification and tags tracking.

To conclude, this survey shows the great advances made by different IoT experimental platforms in different directions, offering a large set of diversified services. They all present some strengths and lacks with regards to the user needs. We detailed some possible extensions that could be done to allow an even larger set of experimentations following the new research trends. Finally, we expect this survey assist a potential user to easily choose the one to use regarding his own needs.

References

[1] M. Nati, A. Gluhak, H. Abangar, W. Headley, Smartcampus: a user-centric testbed for internet of things experimentation, in: Proc. of International Symposium on Wireless Personal Multimedia Communications (WPMC), Atlantic City, NJ, USA, 2013.
[2] C.B. des Roziers, G. Chelius, T. Ducrocq, E. Fleury, A. Fraboulet, A. Gallais, N. Mitton, T. Noël, J. Vandaele, Using sensLAB as a first class scientific tool for large scale wireless sensor network experiments, in: Proc. of Networking, Valencia, Spain, 2011.
[3] A. Jimenez-Gonzalez, J.R.M.-D. Dios, A. Ollero, An integrated testbed for cooperative perception with heterogeneous mobile and static sensors, Sensors 12 (2011).
[4] V. Handziski, A. Köpke, A. Willig, A. Wolisz, TWIST: a scalable and reconfigurable testbed for wireless indoor experiments with sensor networks, in: Proc. Workshop on Multi-hop Ad Hoc Networks: From Theory to Reality, Florence, Italy, 2006.
[5] A. Arora, E. Ertin, R. Ramnath, M. Nesterenko, W. Leal, Kansei: a high-fidelity sensing testbed, IEEE Internet Comput. 10 (2006) 35–47.
[6] D. Sakamuri, H. Zhang, Elements of Sensornet Testbed Design, 2009.
[7] G. Coulson, B. Porter, I. Chatzigiannakis, C. Koninis, S. Fischer, D. Pfisterer, D. Bimschas, T. Braun, P. Hurni, M. Anwander, G. Wagenknecht, S. Fekete, A. Kröller, T. Baumgartner, Flexible experimentation in wireless sensor networks, ACM Commun. 55 (2012) 82–90.
[8] O. Hahm, M. Güneş, K. Schleiser, Des-testbed a wireless multi-hop network testbed for future mobile networks, in: Proc. of GI/ITG KuVS Workshop on Future Internet (KuVS), Hannover, Germany, 2010.
[9] R. Lim, F. Ferrari, M. Zimmerling, C. Walser, P. Sommer, J. Beutel, Flocklab: a testbed for distributed, synchronized tracing and profiling of wireless embedded systems, in: Proc. of International Conference on Information Processing in Sensor Networks (IPSN), Philadelphia, Pennsylvania, USA, 2013.
[10] M. Doddavenkatappa, M. Chan, A. Ananda, Indriya: a low-cost, 3d wireless sensor network testbed, in: Testbeds and Research Infrastructure. Development of Networks and Communities, Thessanoliki, Greece, 2012.
[11] S. Bouckaert, W. Vandenberghe, B. Jooris, I. Moerman, P. Demeester, The W-iLAB.t testbed, in: Testbeds and Research Infrastructures. Development of Networks and Communities, Shanghai, China, 2011.
[12] M.A. Hail, J. Pinkowski, T. Teubler, M. Danckwardt, D. Pfisterer, H. Hellbrück, Roombanet – testbed for mobile networks, Electron. Commun. EASST 37 (2011).
[13] E. Förster, A. Förster, T. Leidi, K. Garg, D. Puccinelli, F. Ducatelle, S. Giordano, L.M. Gambardella, Poster abstract: motel: towards flexible mobile wireless sensor network testbeds, in: Proc. of the 8th European Conference on Wireless Sensor Networks (EWSN), Bonn, Germany, 2011.
[14] P. De, A. Raniwala, R. Krishnan, K. Tatavarthi, J. Modi, N. Syed, S.Sharma, T. Chiueh, MiNT-m: an autonomous mobile wireless experimentation platform, in: Proc. of International Conference on Mobile Systems, Applications, and Services (Mobisys), Uppsala, Sweden, 2006.
[15] M. Barbosa, A. Bernardino, D. Figueira, J. Gaspar, N. Gonçalves, P.U. Lima, P. Moreno, A. Pahliani, J. Santos-Victor, M.T.J. Spaan, J. Sequeira, ISRobotNet: a testbed for sensor and robot network systems, in: Proc. of International Conference on Intelligent Robots and Systems (IROS), St. Louis, MO, USA, 2009.

Please cite this article in press as: A.-S. Tonneau et al., How to choose an experimentation platform for wireless sensor networks? A survey on static and mobile wireless sensor network experimentation facilities, Ad Hoc Netw. (2015), http://dx.doi.org/10.1016/j.adhoc.2015.03.002

Anne-Sophie Tonneau received the MSc in computer science in 2011 from Lille 1 University. She had a brief experience in an IT consulting company before joining the FUN research group in late 2011. Specialised in software development and project management, she is in charge of the deployment of the new FIT equipex IoT-LAB platform in Lille and the study of a mobility solution involving robots.

Nathalie Mitton received the MSc and PhD. degrees in Computer Science from INSA Lyon in 2003 and 2006 respectively. She received her Habilitation diriger des recherches (HDR) in 2011 from Lille 1 University. She is currently an Inria full researcher since 2006 and from 2012, she is the scientific head of the Inria FUN team which is focused on small computing devices like electronic tags and sensor networks. Her research interests are mainly focused on self-organisation, selfstabilisation, energy efficient routing and neighbour discovery algorithms for wireless sensor networks as well as RFID middlewares. She is involved in the set up of the SensLAB (www.senslab.info) and FIT platforms (http://fit-equipex.fr/) and in several program and organisation committees such as AdHocNets 2014, HPCC 2014, WiMob 2013, MASS 2012 & 2011, LoGASN 2012, WPMC2012, IST-AWSN 2012, iThings 2012, and Comnet-iot 2012.

Julien Vandaële received the MSc in computer science in 2005 from Lille 1 University. Since this date, he is Research Engineer at Inria Lille – Nord Europe. He is working on the development and implementation of the FIT IoT-Lab testbed since the start of the project in 2008, in the FUN research group led by Dr. Nathalie Mitton. Specialised in software development, ubiquitous and distributed systems and project management, he manages the Lille developer team, related to the other developers divided in different sites in France. In addition to this mission, as a member of the Experimentation and Development Department, he supervises the work of young engineers of the research center, advises them about methodology, technical choices and tools.

Please cite this article in press as: A.-S. Tonneau et al., How to choose an experimentation platform for wireless sensor networks? A survey on static and mobile wireless sensor network experimentation facilities, Ad Hoc Netw. (2015), http://dx.doi.org/10.1016/j.adhoc.2015.03.002