



HAL
open science

A Survey on Facilities for Experimental Internet of Things Research

Alexander Gluhak, Srdjan Krco, Michele Nati, Dennis Pfisterer, Nathalie Mitton, Tahiry Razafindralambo

► **To cite this version:**

Alexander Gluhak, Srdjan Krco, Michele Nati, Dennis Pfisterer, Nathalie Mitton, et al.. A Survey on Facilities for Experimental Internet of Things Research. IEEE Communications Magazine, 2011, 49 (11), pp.58-67. 10.1109/MCOM.2011.6069710 . inria-00630092

HAL Id: inria-00630092

<https://inria.hal.science/inria-00630092>

Submitted on 17 Feb 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Survey on Facilities for Experimental Internet of Things Research

Alexander Gluhak^{*}, Srdjan Krco[†], Michele Nati[‡], Dennis Pfisterer[§], Nathalie Mitton[¶]
and Tahiry Razafindralambo[¶]

Abstract

The initial vision of the Internet of Things (IoT) was of a world in which all physical objects are tagged and uniquely identified by RFID transponders. However, the concept has grown into multiple dimensions, encompassing sensor networks able to provide real-world intelligence and goal-oriented collaboration of distributed smart objects via local networks or global interconnections such as the Internet. Despite significant technological advances, difficulties associated with the evaluation of IoT solutions under realistic conditions, in real world experimental deployments still hamper their maturation and significant roll out. In this article we identify requirements for the next generation of the IoT experimental facilities. While providing a taxonomy, we also survey currently available research testbeds, identify existing gaps and suggest new directions based on experience from recent efforts in this field.

Index Terms

Internet of Things, Experimental environment, Testbed

1 INTRODUCTION

The IoT is a multidisciplinary domain that covers a large number of topics from purely technical issues (e.g. routing protocols, semantic queries), to a mix of technical and societal issues (security, privacy, usability), as well as social and business themes. IoT applications, both existing and potential, are equally diverse. If we are to summarize all of them into one, then its enabling the machine perception of the real world and seamless interactions with it. Environmental and personal health monitoring, monitoring and control of industrial processes including agriculture, smart spaces and smart cities are just some of the examples of IoT applications.

Designing a system that can efficiently support such a large range of applications and be compliant with a plethora of often contradicting requirements as well as integrating all required components and technologies is a complex task. Simulations, as an important phase during this process are useful for developing further understanding of a system. However, they suffer from several imperfections [1] as they make artificial assumptions on radio propagation, traffic, failure patterns, and topologies. What makes it particularly difficult is the strong dependency of IoT systems on real-world processes that are often a result of complex systems-of-systems interactions and extremely difficult to model accurately.

To design robust applications, developers need appropriate tools and methods for testing and managing their applications on real hardware in large-scale deployments. In the early days of IoT research, the availability of smart devices was limited and only recent advances in technology increased their availability at lower costs. Although experiments were mainly small-scale and

^{*} UNIS - University of Surrey

[†] Ericsson

[‡] CCSR - Centre for Communication and System Research

[§] Institute of Medical Informatics

[¶] Inria Lille - Nord Europe

conducted in research labs, they allowed for an improvement in understanding the impact and limitations of real hardware on performance of protocols and design choices. However, the daunting logistical challenge of experimenting with thousands of small battery-powered nodes is the key factor that has greatly limited the development of this field.

Overcoming the technical challenges and socio-economic barriers of a wide-scale IoT deployment in our daily lives requires a thorough, practical evaluation of IoT solutions using interdisciplinary, multi-technology, large-scale, and realistic testbeds. The ability to test IoT solutions on a larger scale and outside of research labs, i.e., in real environments and with real end users has been considered only recently [2]. These new testbeds aim to design and deploy experimentation environments that will allow for *i)* the technical evaluation of IoT solutions under realistic conditions, *ii)* the assessment of the social acceptance of new IoT solutions, and *iii)* the quantification of service usability and performance with end users in the loop.

This article identifies the main requirements for a next generation of experimental research facilities for the IoT (Section 2) and provides a taxonomy and survey of existing testbeds that are currently available (Section 3) throughout the world. We identify the main features and capabilities of these testbeds, analyze how they fulfill the identified experimentation needs and challenges and finally suggest research directions based on experience from recent efforts in this field (Section 4).

2 REQUIREMENTS AND CHALLENGES

Testbeds for wireless sensor network (WSN) research have provided a solid basis for experimentation with solutions to overcome the problems and challenges pertinent to the research domain. However moving from islands of WSNs to a global networked infrastructure – as envisioned by the Internet of Things – opens up new challenges that demand new capabilities and features from suitable testbeds. In the following we will briefly highlight the key challenges when moving from WSNs towards IoT. Based on those we motivate requirements for adequate facilities for research and experimentation with future IoT-based technologies.

2.1 Challenges for IoT experimentation

WSN research has primarily focused on advances within WSN islands, providing optimized solutions for resource-constraint devices of which they are composed. In contrast, a major goal of IoT research is to integrate these island technologies into a globally interconnected infrastructure, moving from currently existing *Intra-net* to a real *Inter-net* of Things [3]. A resulting challenge is an increase in *scale* that the developed IoT solutions have to cope with. Another challenge is an increasing *heterogeneity* of devices and device technologies. Both require new approaches to support interoperability at different layers of the communication stack for resource-constraint devices, which have to be adequately reflected in IoT experimentation environments. Increased heterogeneity and scale, however, also make *repeatability* of experiments on a testbed or across different testbeds more difficult. Finally, as the IoT is considered part of a larger Future Internet, *federation* of IoT testbeds with other Future Internet testbeds becomes more important for end-to-end experimentation and is a further challenge to be tackled.

Other key differences are the horizontalization of infrastructure and the pervasiveness of the IoT environment. While WSN research mainly focused on providing solutions that are optimized for a particular problem domain and application area, IoT research envisions supporting multiple application domains, possibly at the same time. A challenge represents the increased *concurrency* of access to the infrastructure, which has to be reflected by IoT testbeds as well. The experimentation space is not limited to interoperability at the communications layer, but also requires data-layer solutions. As more real world applications and services will rely on

IoT infrastructures, the *realism of the environment for experimentation* becomes more important. With it comes the support for realistic *mobility*. Finally, the pervasiveness of the infrastructure and the potential social impact of IoT technologies call for human users to be included into the experimentation loop. A challenge is to design mechanism that support adequate *user involvement* during experimentation.

2.2 Requirements

The above identified challenges motivate requirements that adequate facilities have to support. In the following we provide a brief description of those.

Scale: As the IoT will become a globally interconnected infrastructure, supporting experimentation at adequate scale is an important aspect contributing towards the realism of IoT experimentation. While smaller-scale testbeds in the range of 10s up to hundreds of nodes were sufficient for most WSN experiments, many IoT experiments will demand an order of magnitude larger scale. In order to make experimentation with 1000s of IoT nodes possible techniques are required that minimize human intervention, maximizes plug-and-play configuration, and automatic fault-management. As experimenters can easily lose an overview when experimenting with such large numbers of testbed resources, support mechanisms that ease the selection of adequate testbed resources and the composition of experiments are necessary.

Heterogeneity: The expected heterogeneity of devices and underlying solutions demands that experimental testbeds reflect this heterogeneity in terms of the hardware substrate and the nature of experiments supported on top of them. Furthermore, gateway devices which have been previously considered only as sinks in WSN testbeds, should be an active part of the experimentation infrastructure. This requires tools allowing for an effective configuration and execution of experiments across heterogeneous testbed resources and the corresponding management of devices. A further requirement is providing support to ease programmability of heterogeneous devices, which often come with diverse execution environments.

Repeatability: There is an increasing trend in the scientific community to repeat experiments within and across different testbeds. Replayability goes even further by providing an adequate packaging of experimentation specifications, traces, and results so they can be easily re-executed and compared across different testbeds. While contributing to scientific rigor, it will allow others to quickly build on other work thus accelerating the scientific progress of the community. The heterogeneity and the wireless nature of IoT testbeds combined with ever changing ambient conditions make both features hard to implement. While a total replication of real-world conditions may be impossible, solutions supporting the monitoring of the radio environment during experimentation and benchmarking of different testbeds could partially overcome the associated issues by providing hints for the interpretation of experimental data. Replayability requires agreement on standards for the specification of experiments, collection of traces, and the packaging of experimental results across a variety of testbeds.

Federation: Federation with other testbeds is necessary to achieve scale or to add capabilities for experimentation, which are not locally available. Reasons for limited scale and diversity are hardware cost, space constraints, lack of expertise, or required management resources. Federation is a viable solution to create larger and more heterogeneous facilities out of specialized, smaller-scale ones. From the management perspective, requirements on federation are a common framework for authentication, authorization, accounting, reservation, and experiment scheduling.

Concurrency: With an increased investment in infrastructure, supporting multiple concurrent users and experiments is a necessity for an economically viable operation and larger-scale testbeds must support multiplexing of concurrent experiments. Unlike existing testbeds for Internet experimentation (e.g, PlanetLab), IoT devices are substantially resource-constrained, making virtualization at the hardware-layer very difficult. Such "virtualization" is more feasible at the

testbed-level than on the node-level and will support the multiplexing of experiments in space and time. IoT testbeds require advanced mechanism that allow selection of testbed resources to minimize interference of concurrently executing experiments (e.g., caused by collocated wireless transmissions, access to same hardware resources, etc.), while satisfying the requirements of the experiment.

Experimental environment: IoT technologies heavily depend on ambient environmental conditions in which they are deployed, so does the service logic of the diverse IoT applications. Moving testbed deployments out of the lab and into the *wild* requires more robust techniques for realizing out-of-band management and control planes, as many of the currently wired solutions will have to be replaced by wireless ones. Increased robustness is also important due to the increased overhead of the maintenance of testbed equipment in such deployments, as ease of access is often traded off against the threat of physical tempering, deliberate damage, or theft.

Mobility: A key role of the IoT is to provide information on real-world entities and events or the interaction capabilities that influence them. Many such entities may move around in a real-world environment, thus making the IoT devices attached to them mobile. Handling such mobility and the associated system dynamics is thus a key requirement for future IoT solutions. Therefore mechanisms to control and exploit realistic mobility of both IoT devices and real world entities during experimentation are necessary.

User involvement and impact: Many IoT applications are centered on human users or require their active participation, which make experimentation more difficult to control and may invalidate results. Experimental facilities that involve users must offer mechanisms allowing for evaluating social impact and acceptance of IoT solutions and applications. This includes automated detection of situations when user behavior influences the validity of collected data, but also the provision of efficient multi-modal mechanisms for user feedback.

3 SURVEY OF EXISTING TESTBEDS

In the following section we provide a survey of the currently available testbeds for IoT experimentation and develop a taxonomy for evaluating the capabilities and features they offer. We concentrate our discussion on four main dimensions: the testbeds' primary scope, underlying architecture and organization, hardware capabilities and offered testbed services. We further discuss the extent to which the available testbeds fulfill the requirements for future IoT testbeds introduced in the previous section. For this article we restrict ourselves to testbeds publicly accessible and with notable use (Table 1). The table also contains references to other IoT-related testbeds that are not publicly available or are a work in progress, but still provide interesting features.

3.1 Testbed scopes

When building an IoT testbed, one key consideration is its scope (Figure 1). In terms of the technological scope, a testbed can be *single-domain* if it addresses only a particular type of IoT technology, e.g. wireless sensor network or RFID devices. Nearly all of the surveyed testbeds belong to this category ([14]). A *multi-domain* testbed is one that combines different IoT technologies into a common experimental facility. The latter one is increasingly important in addressing the IoT heterogeneity requirements ([10]). Testbeds can be realized either *indoor* (Motelab) or *outdoor* (CitySense, Oulu smart city, SmartSantander), thereby influencing the choice of the underlying hardware architecture and software mechanisms. Indoor installations often provide easier access to power and cabling for the testbed control and management, making them easier to manage and more available (Section 3.4). In contrast, outdoor testbeds often rely on wireless connections, requiring additional mechanisms in order to increase reliability. They also have to be protected

| Testbed | Scope | Nodes & Locality | Noteworthy |
|-----------------------|-----------|---|--|
| MoteLab [4] | WSN | 190 TelosB motes, indoor, spread across 3 floors of a lab building | One of the first and longest lasting testbeds. In-situ power measurements on some nodes. MoteLab testbed service framework used as a basis for various other testbeds, e.g. CCNY-CWSNET and INDRIYA |
| NetEye [5] | WSN | 130 TelosB motes, indoor, 1 room, 15 wooden benches 1 feet apart | FCFS scheduling approach, static 3db attenuator to each mote antenna for realising multi-hop network and different power levels, to be integrated as part of the Kansai-Geni testbed |
| TutorNet [6] | WSN | 104 motes (91 TelosB, 13 MicaZ), indoor, 1 room | The user indicates the start and end time for its reservation, as well as the list of motes it would like to reserve. The reservation will fail when attempting to exceed the allowed nodehours quota, where a nodehour represents a reservation unit corresponding to reserving one mote for one hour |
| MIRAGE Intel [7] | / WSN | 100 MicaZ motes(currently only 58 available), indoor, lab environment | The reservation is based on an abstract resource specification language for resource discovery (only per-node attribute available) and by using the MIRAGE bidding language in order to reserve the discovered resources |
| VineLab [8] | WSN | 48 TelosB motes , indoor, 1 floor inside a lab | unlike other testbeds provides only very basic experimental user support via Python scripts, utilisable for research on smart in-door environments |
| Kansei [9] | Mesh, WSN | 210 XSM motes, 210 Stargate gateways, 50 Trio motes, indoor | One of the most advanced surveyed testbeds with various testbed service functions, co-simulation support, mobility support using mobile robots, event injection possible both at GW and mote level |
| WISEBED [10] | WSN | 750 motes (200 iSense, 143 TelosB, 108 G-Node, 100 MSB-A2, 44 SunSPOT, 60 pacemate, 24 Tnode), in- and outdoor, 9 different locations in Europe | Federation architecture, co-simulation support, topology virtualization, in-situ power measurements on some nodes, mobility support using 40 mobile robots |
| FRONTS [11] | WSN | 21 iSense motes, indoor, lab environment | In-situ power measurements on some nodes, purely wireless in-band management |
| DES-Testbed [12] | Mesh, WSN | 95 MSB-A2, 95 Linux nodes, in- and outdoor | Mobility planned, currently one prototype robot available |
| w-iLab.t Testbed [13] | Mesh, WSN | 200 TMoteSky motes, indoor, lab environment | Measurement and battery emulation, repeatable mobility support |
| Senslab [14] | WSN | 1024 WSN430 (521 with 802.15.4 MAC, 512 with free MAC layer), indoor, 4 different sites in France | Energy measurement supported for every node, repeatable mobility via electric toy trains |
| KanseiGeni [15] | Mesh, WSN | 576 motes (96 XSM, 384 TelosB, 96 iMote2) attached to Stargate gateways, indoor, lab environment | Based on Kansei and Neteye, federation with other GENI envisioned, heterogeneous mote platforms |
| TWIST [16] | WSN | 204 motes (102 TelosB, 102 eyesIFX) | Forms the basis for a variety of other testbeds such as WUSTL |

| Testbed | Scope | Notes |
|-------------------------------|-----------|--|
| CitySense [17] | Mesh | City-scale deployment on light poles and private or public buildings. Nodes comprise of an embedded PC, 802.11a/b/g interface, and various sensors for monitoring weather conditions and air pollutants. Envisioned scale is around 100 nodes. Two features make this testbed particularly interesting: its realism and domain specificity provided by a permanent out door installation in an urban environment and the realization of the control and management plane based solely on wireless links. |
| Oulu Smart City [18] | Smart WSN | Smart City project to study urban computing, or the interaction between urban spaces, human, information technology and information. IEEE 802.15.4 technology on the 868 MHz band |
| Friedrichshafen [19] | Multiple | Smart City project (comprising PDAs, Sensors, Smartphones, etc.) to evaluate prototypical services such as e-Learning, Smart Metering, e-Health, e-Tourism, etc. |
| Motescope [20] | WSN | WSN testbed for energy metering at UCB. Large smart metering deployment that will comprise more than 500 heterogeneous sensor nodes with the aim of providing building energy consumption profiling and a platform for protocol experimentation |
| FlockLab [21] | WSN | Wireless sensor network deployment based on a new board (observer) providing advanced features such as energy consumption monitoring of the mote and emulation of radio behavior. Using particular adapter, the board results compliant with different off-the-shelf motes. |
| METRO real- Future Store [22] | RFID | RFID Innovation Center, Test lab area of 2000 sqm., Covering various areas of RFID application in retail such as supply commissining, stock management, warehouse, instore and consumer household. |
| SAP Future Retail Centre [23] | RFID, WSN | The main purpose is the experimentation with IoT technology and services to improve the customer experience while optimizing in-store processes by enabling complete visibility and traceability of handling units along the supply chain. The centre covers three experimentation scenarios: retail, retail management and logistics. |
| CITC testbed | RFID | The main purpose is to experiment within a typical environment combined with specific frequencies in order to study the RFID effectiveness in a given environment (metal, liquids, etc). More information available at http://www.citc-earfid.com . |
| RFIT Lab | RFID | The testbed allows experimentation with RFID devices in a given environment and an evaluation of electromagnetic compatibility. More information available at http://rftlab.grenoble-inp.fr . |
| VizBee | RFID | The main purpose of this testbed is to integrate RFID receivers, tags, excitors and repeaters. More information available at http://www.vizbee-rfid.com/technology.asp?ID=2 . |

TABLE 1

Active and publicly available testbeds (top) and testbeds with interesting features (bottom)

against malicious physical access and other threats that do not occur as frequently in controlled environments. Furthermore, testbeds can be realized either as *permanent* installations that are constantly available or *portable* ones, usually temporarily deployed for measurement campaigns.

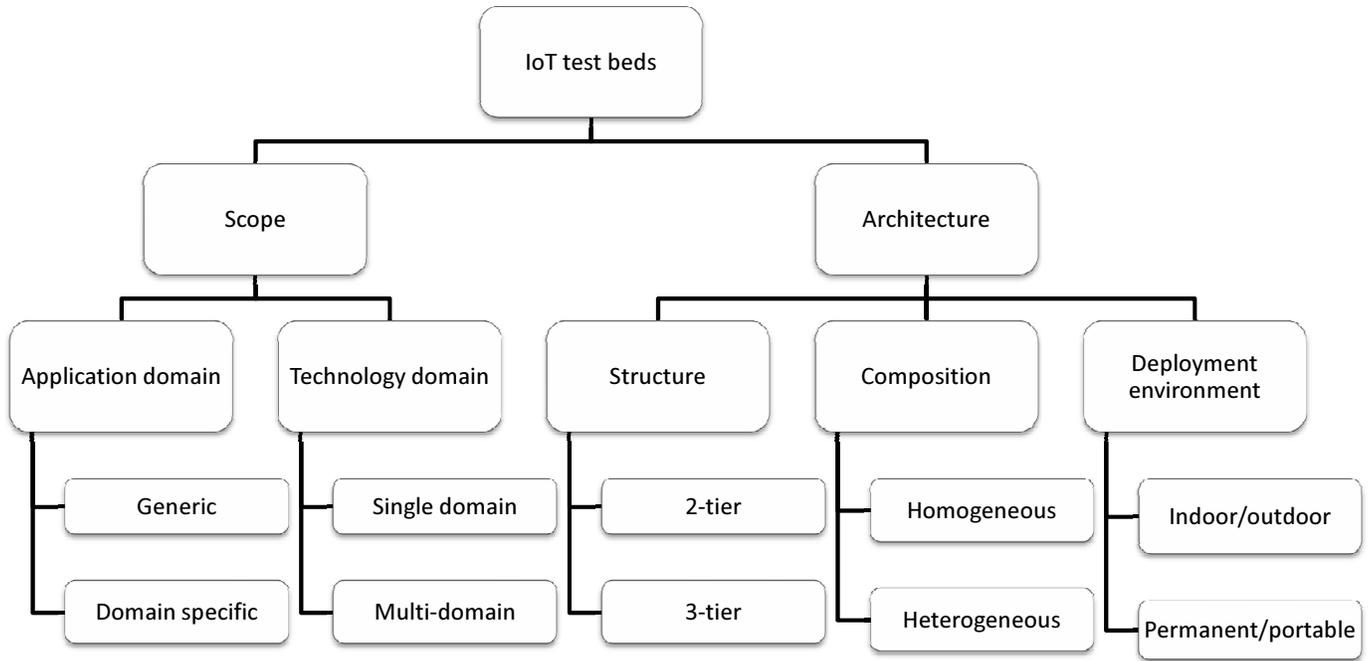


Fig. 1. Scope and architecture of testbeds.

Testbeds can differ based on the *application domain* they are designed to investigate. *Generic testbeds* such as MoteLab, WISBED or SenseLab enable experimentation research of communication protocols at different levels of the protocol stack or application level algorithms that are mostly independent of a particular application or service. More recently domain-specific testbeds have emerged that allow the evaluation of services and application in a real environment and from the end-user perspective. Examples of these domains are smart city services (CitySense, Oulu Smart City, Friedrichshafen) or supply commissioning and stock management (METRO real, SAP Future Retail Centre).

3.2 Architecture

The architecture is another key dimension of IoT testbeds, as it inherently determines their properties. We distinguish between three important features as shown in Figure 1. The structure of a testbed refers to the organization of the testbed hardware components. Most testbeds either follow a *2-tier* structure with a server tier and directly attached IoT device tier or a *3-tier* structure that introduces an intermediate GateWay (GW) device tier (see Figure 2).

Smaller testbeds (hundred of nodes) often follow a 2-tier structure, in which IoT devices are organized in a flat tier and directly connected or attached to a server tier. These interconnections can be wired (e.g. via USB or Ethernet) or wireless (e.g. via WiFi). The initial realization of *Motelab*, one of the first publicly accessible testbeds, was based on a 2-tier architecture, in which sensor motes with an attached Ethernet interface were connected via a LAN to a testbed server. Examples of other 2-tier testbeds with a wired backbone are *MIRAGE* or *Vinelab*, the latter using USB cables for interconnections to a server. Many RFID-based testbed environments follow a 2-tier architecture, where RFID readers are attached via wired backbones to middleware servers [22]. *CitySense* and *FRONTS* are 2-tier testbeds that have connections to the server via wireless links. *FRONTS* implements the testbed management in-band over wireless links. In contrast, *CitySense* nodes make use of two separate WiFi radio interfaces, in order to separate the experimentation and testbed management planes, which are interconnected via multi-hop

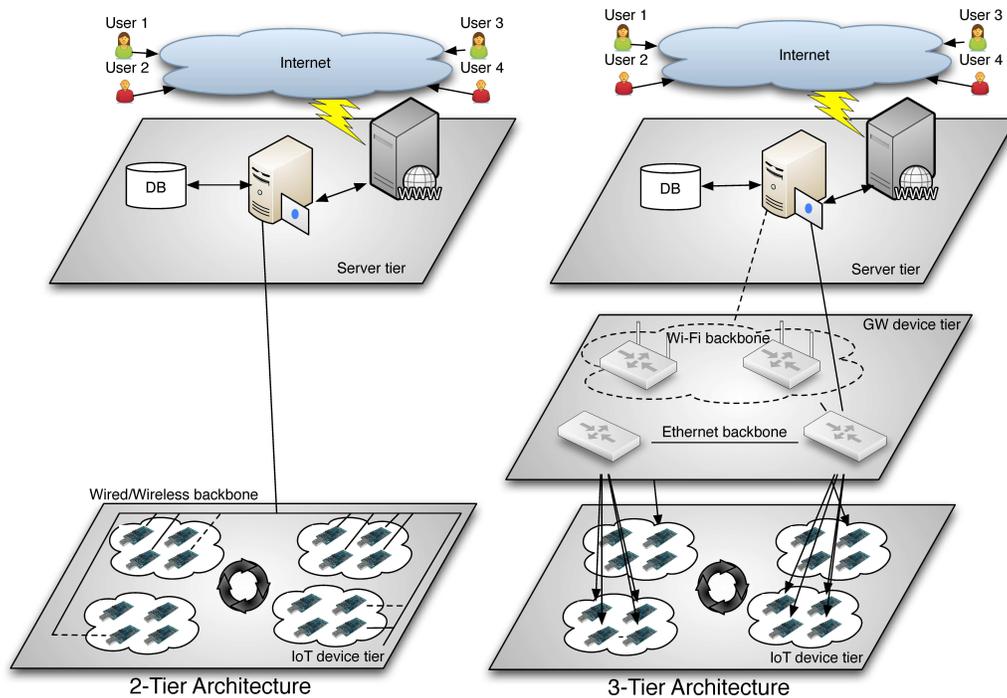


Fig. 2. Typical IoT testbed architectures.

links. While 2-tier testbeds are easy to setup, they become more complicated to manage once they start growing in size. Furthermore, experimentation capabilities are limited to the IoT device tier only, thereby constraining experimentation with tiered network architectures and protocols for internetworking with other networks.

Larger testbeds typically exhibit a 3-tier architecture, in which one or up to dozens of IoT devices are connected via wired connections to an intermediate GW device. This reduces the need for each individual IoT device to provide a possibly costly network interface card, in order to enable communications with remotely located testbed servers. The *TWIST* testbed as well the latest version of *Motelab* and *INDRIYA* make use of low-cost USB-to-Ethernet devices as intermediate aggregation nodes to which various motes are attached via cheap USB cables and hubs, implicitly providing power to the motes. GW nodes are realized by different types of hardware such as laptop computers (*NetEye*), embedded Linux servers (*Kansei*, *KanseiGeni*) or custom-built hardware (*FlockLab*). Most of the surveyed 3-tier testbed in this paper utilize the GW tier as an utility, providing only connectivity aggregation and passive experimentation support. Only few testbeds such as *Kansei/KanseiGeni* allow the GW devices to be programmed with experimentation code to become an active participant in experimentation. This allows the realization of more complex IoT experimentation scenarios with protocols for a larger-scale interconnection of IoT islands, which are crucial when moving from Intra- to the Inter-net of Things technologies [3]. An example for this is end-to-end 6LowPAN experimentation, which require not only the deployment of 6LowPAN stacks on IoT nodes, but also IPV6 routers on the GW tier to support interaction with applications on the server tier.

Within each tier, many testbeds comprise the same type of testbed hardware, e.g. *Motelab* or *Neteye*. We consider the composition of such testbed *homogeneous*. Some of the recent testbeds such as *KanseiGeni* and *WISEBED* are *heterogeneous* within a tier and are realized by different device platforms, thereby providing an experimenter with more choices and increased realism, at the cost of the increased complexity in handling experimentation and testbed management.

Finally, the physical deployment architecture or, *testbed topology*, has a strong influence on the

possibilities of direct radio links between the nodes and determines what types of experimental scenarios can be realized within the testbed. Most of the testbeds are permanent and indoor, spanning one or more floors of a given building. Examples of 3D testbeds are *INDRIYA* and *Motelab*, where in both cases, three different floors are covered by sensors. The *Kansei* testbed provides a large grid-like structure of motes evenly distributed on tables within a warehouse. All of the testbeds usually span an area large enough to obtain a natural multi-hop network topology. Topology control for experimentation is mainly done by selecting a set of nodes for experiments based on their geographic properties (e.g. distances) and by adequate configuration of transmission power levels. In order to constrain the physical topology in dense indoor deployments, attenuators are often attached to the antennas of the IoT nodes, e.g. *NetEye*. Topology changes during experimentation can be achieved by turning on/off motes. This is supported in *TWIST* and *INDRIYA* by toggling the power on ports of the USB hubs to which the motes are attached. *WISEBED* provides increased flexibility for topology configuration even across remote IoT islands using *virtual links* [24].

3.3 Hardware features

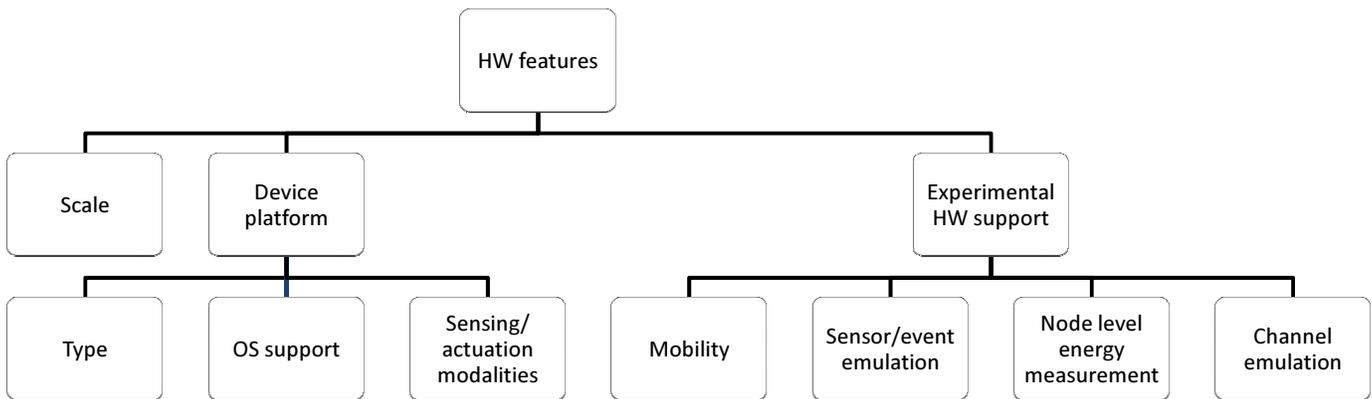


Fig. 3. Classification of hardware features.

This section examines the hardware features provided by different testbeds. As shown in (cf. Figure 3), many existing testbeds vary in the type of underlying *device platforms*. At the IoT tier, most publicly available testbeds focus on WSNs, which utilize off-the-shelf mote platforms, although in some cases custom-built solutions can be found (Senslab). RFID based testbeds realize this tier with typical off-the-shelf readers and tags. The GW tier is mostly based on more powerful devices, such as embedded servers.

For WSN testbeds, Operating System (OS) support depends largely on the mote platforms used in the IoT tier. Commonly found OS are TinyOS, Contiki, Sunspot, iSense or MoteRunner. While in principle most mote platforms support one or more OS's, a few testbeds are limited to a particular OS (Motelab, TWIST, Vinelab). A notable example on the other end of the spectrum is the WISEBED testbed, which offers seven different hardware platforms and thus a variety of OS choices. Currently, the majority of mote platforms use a standard MAC layer (IEEE 802.15.4), which inhibits research on custom MAC layer protocols and only a few testbeds (Senslab, DES-Testbed) provide the flexibility of replacing the MAC layer. Some testbeds (Kansei, DES-testbed, w-iLab.t, FlockLab) provide additional, more powerful hardware such as embedded Linux computers.

Sensor nodes (TelosB/TmoteSky testbeds) are equipped with basic temperature, humidity and light sensors. More sophisticated testbeds (MicaZ and iSense platforms) provide a selection of

other sensors such as magnetometers, microphones and pressure sensors. Apart from LEDs, actuators can rarely be found.

Another important metric is the *scale* of testbeds. The majority of single location testbeds feature a limited number of nodes (a few dozen up to 200 nodes, except KanseiGeni which has over 600 nodes) due to cost and space constraints. A promising approach is the federation of individual testbeds interconnected over the Internet as in the cases of Senslab and WISEBED. Here, an important aspect is to allow the distributed testbeds to appear as a single one at the application layer by using approach like *virtual links* [24].

Some testbeds allow realization of more complex experimentation scenarios, thereby increasing the flexibility and control possibilities. An example is the introduction of node *mobility*. Devices are attached to movable objects with controllable trajectory like robots (WISEBED) or electronic toy trains (Senslab), allowing repeatable mobility patterns during experimentation.

Another important feature is the *emulation of sensor events* by attaching D/A converters to the IoT platforms, from which arbitrary sensor events can be generated during experimentation. For example, the w-iLab.t testbed provides *environmental emulators*, which trigger sensor readings on the individual sensor nodes that correspond to emulated environmental events. This also allows replaying of sensor events obtained from a real-world environment on the testbed. Some testbeds (FlockLab, Senslab) are able to *emulate channel characteristics* between IoT nodes. Lastly, a useful feature is the ability to measure energy consumption of nodes by means of dedicated on board HW (iSense) or connected GW (FlockLab, Senslab), or perhaps with the use of battery emulators (w-iLab.t, TWIST).

3.4 Testbed services

Testbed services enable efficient execution of experiments throughout the complete life-cycle of an experiment (cf. Figure 4).

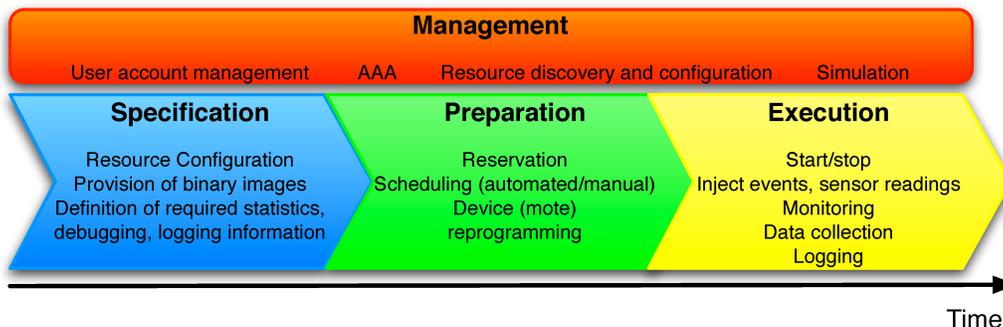


Fig. 4. Experiment execution workflow and corresponding testbed services.

In most testbeds (Motelab, Senslab, TWIST, NetEye, Kansei, WISEBED), *experiment specification* is done via web interfaces where users define the number and the type of resources needed, the programs to upload and the metrics to be collected.¹ Other platforms, such as TUTORNET, Vinelab and Motescope, provide script-based tools only. WISEBED defines a set of Web Service APIs that compatible testbeds must implement. Clients (e.g., a web interface or a script) invoke these services to run experiments and to interact with the testbed. WISEBED also provides a generic XML-based language (WiseML) for experiment description, configuration, and the storage of results.

1. A similar framework for assisting users in the complete life-cycle of an experiment with wireless/wired devices is represented by the OMF tools [25], developed within the OneLab project. An Experiment Description Language assisting users in deploying experiments is provided along with a cOntrol Measurement Library (OML), useful for instrumenting the experiment in order to produce logging and analysis information.

Testbeds like NetEye (in its basic implementation) use a priority-based approach for *reservation* and *scheduling* in which only experiments requiring available resources at any given time take place and the resources allocation is prioritized based on the start time of each submitted experiment. Motelab allows for the reservation of specific radio channels. Other approaches enable resource reservation using calendar applications. Examples are Motelab (using a custom calendar application), FRONTS (Google Calendar), and WISEBED (extensible, database backed Google calendar, or in-memory storage). To avoid long-term blocking of resources, some testbeds have daily user quotas (e.g., TUTORNET). Others calculate optimal schedules based on reservation requests (e.g., Senslab’s OAR server or MIRAGES micro-economic processes).

Before an experiment is executed, the devices are reprogrammed. Typically, *reprogramming* uses a wired infrastructure, while only a few testbeds use wireless reprogramming techniques (e.g., FRONTS). *Execution control* allows influencing of the experimentation flow, e.g. by injecting events or sensor readings (w-iLab.t and FlockLab). Some testbeds offer access to ongoing experiments through web interfaces, via SSH (CitySense), cURL tool (TWIST), or Web Services (WISEBED) to adjust parameters, including aspects of *monitoring and data collection*. The monitoring and collection of statistics during experimentation allows users to debug experiments, to follow the progress of the execution or to store performance indicators/statistics for later analysis. However such processes can interfere with the experiment. Motelab and similar testbeds let users to download such data only after an experiment. Other testbeds use out-of-band mechanisms (see Section 3.2) or efficient logging schemes [26] that collect statistics and debug information during periods of node inactivity, in order to allow real-time access to messages and metrics (Senslab, WISEBED, and FRONTS) and the transmission of control messages to nodes (WISEBED).

Implementations of *management services* are mostly proprietary (FRONTS, TWIST), although some testbeds use standards such as Shibboleth (WISEBED) to realize a subset of functionality (e.g. access control). *Resource discovery and configuration services* ensure that the connected testbed resources are discovered and properly configured to operate as expected for experimentation. With increasing scale autonomous operation of these services becomes more important. Most of the surveyed testbeds require manual configuration of resources or at least a manually initiation process. One exception is Kansei, which features tools for node configuration and plug-and-play registration of resources. The same applies to *fault management* tools, responsible for hardware and service recovery after failures. In current deployments, this is manually initiated by testbed administrators, when *performance monitoring* tools report misbehaving elements. This is often realized by pinging the nodes to obtain their status (NetEye) or by using controllable cameras to provide ground truth during experimentation, especially in the presence of node mobility (Kansei and Senslab).

Co-simulation tools, if offered, permit the scaling of experiments beyond the available testbed hardware using simulated nodes or allow injecting of real data into a simulator to make simulations more realistic (Kansei and WISEBED). With the same aim of extending existing testbed capabilities and to introduce heterogeneity, a recent trend in testbeds is *federation*, which allows experiments to span multiple physical testbeds (WISEBED and KanseiGeni).

4 TOWARDS A NEW GENERATION OF IOT EXPERIMENTATION FACILITIES

We have analyzed a range of publicly available IoT testbeds, evaluated their scopes, architectures and features and identified the requirements for future IoT experimentation. While many of these testbeds provide interesting and useful features, this survey has demonstrated that there are still gaps in fully addressing the requirements as illustrated in Figure Figure 5) for the most promising testbeds.

expressed user requirements ². This will allow a more optimised testbed resource allocation and contribute towards concurrent execution of experiments. Furthermore testbed traces and results will be made available through machine readable and inter-linkable formats, making a comparison of experiments across different testbeds easier.

A service platform ³ will complement the above described infrastructure, with the ability to develop and experiment with smart urban services and applications on top of the infrastructure. Primary users of these services will be the city authorities and its citizens. User feedback about usefulness of services or after IoT intervention will be collected using participatory sensing mechanisms provided by the testbed through the mobile phones of the users and installed terminal devices in the city.

We hope that this new generation facility for experimental IoT research will accelerate IoT research and global deployment by providing an adequate experimentation environment that fulfills the needs not only of IoT researchers, but also service providers and ultimately us citizens who will be the primary beneficiary of this technology.

REFERENCES

- [1] E. Egea-López, J. Vales-Alonso, A. Martínez-Sala, P. Pavón-Mario, and J. García-Haro, "Simulation tools for wireless sensor networks," in *SPECTS*, 2005.
- [2] Smartsatander, "Smartsantander," 2010, <http://www.smartsantander.eu/>.
- [3] M. Zorzi, A. Gluhak, A. Bassi, and S. Lange, "From Today's INTRANet of Things to a Future INTERNet of Things: a Wireless and Mobility-related View," *IEEE Wireless Communications*, vol. 17, no. 6, pp. 44–51, 2010.
- [4] G. Werner-Allen, P. Swieskowski, and M. Welsh, "Motelab: a wireless sensor network testbed," in *IPSN*, 2005.
- [5] D. Sakamuri and H. Zhang, "Elements of sensor network testbed design," in *Handbook of Sensor Networks*, H. C. Yang Xiao and F. H. Li, Eds. World Scientific Publishing Co, ch. 35, pp. 1–36, 2009.
- [6] "Tutornet: A tiered wireless sensor network testbed," pp. 19–28, 2009, <http://enl.usc.edu/projects/tutornet/>.
- [7] B. N. Chun, P. Buonadonna, A. AuYoung, C. Ng, D. C. Parkes, J. Shneidman, A. C. Snoeren, and A. Vahdat, "Mirage: a microeconomic resource allocation system for sensor network testbeds," in *Proceedings of the 2nd IEEE workshop on Embedded Networked Sensors*, pp. 19–28, 2005.
- [8] The University of Virginia, "VineLab wireless testbed," 2009, <http://www.cs.virginia.edu/~whitehouse/research/testbed/>.
- [9] A. Arora, E. Ertin, R. Ramnath, M. Nesterenko, and W. Leal, "Kansei: A high-fidelity sensing testbed," *IEEE Internet Computing*, vol. 10, pp. 35–47, 2006.
- [10] I. Chatzigiannakis, S. Fischer, C. Koninis, G. Mylonas, and D. Pfisterer, "WISEBED: An open large-scale wireless sensor network testbed," in *Sensor Applications, Experimentation, and Logistics*, vol. 29, pp. 68–87, 2010.
- [11] T. Baumgartner, I. Chatzigiannakis, S. P. Fekete, S. Fischer, C. Koninis, A. Kröllner, D. Krüger, G. Mylonas, and D. Pfisterer, "Distributed algorithm engineering for networks of tiny artifacts," *Computer Science Review*, vol. 5, no. 1, 2011.
- [12] *DES-Testbed A Wireless Multi-Hop Network Testbed for future mobile networks*, 2010.
- [13] S. Bouckaert, W. Vandenberghe, B. Jooris, I. Moerman, and P. Demeester, "The w-iLab.t testbed," in *TRIDENTCOM*, pp. 145–154, 2010.
- [14] sensLAB, "Very Large Scale Open Wireless Sensor Network Testbed." 2010, <http://www.senslab.info/>.
- [15] M. Sridharan, W. Zeng, W. Leal, X. Ju, R. Ramnath, H. Zhang, and A. Arora, "Kanseigenie: Software infrastructure for resource management and programmability of wireless sensor network fabrics," in *Next Generation Internet Architectures and Protocols*, 2010.
- [16] V. Handziski, A. Köpke, A. Willig, and A. Wolisz, "Twist: a scalable and reconfigurable testbed for wireless indoor experiments with sensor networks," in *REALMAN*, pp. 63–70, 2006.
- [17] J. Bers, A. Gosain, I. Rose, and M. Welsh, "Citysense: The design and performance of an urban wireless sensor network testbed."
- [18] T. Ojala, "Open Urban Testbed for Ubiquitous Computing," pp. 442–447, 2010.
- [19] German Telekom and City of Friedrichshafen, "Friedrichshafen Smart City," 2010, <http://www.telekom.com/dtag/cms/content/dt/en/395380>.
- [20] UC Berkeley, "WSN Test beds at UC Berkeley," 2010, <http://www.millennium.berkeley.edu/sensornets/>.
- [21] J. Beutel, R. Lim, A. Meier, L. Thiele, C. Walser, M. Woehrle, and M. Yuecel, "The flocklab testbed architecture," in *SenSys*, pp. 415–416, 2009.
- [22] METRO AG, "real- future store," 2010, <http://www.future-store.org/fsi-internet>.
- [23] SAP Research Living Labs, "Future Retail Center," 2010, <http://www.sap.com/about/company/research/livinglabs/futureretail>.

2. This functionality exploits parts of the rendezvous infrastructure (resource directory, semantic query resolver) developed by the SENSEI project, <http://www.sensei.eu>

3. USN platform provided by Telefonica I&D

- [24] T. Baumgartner, I. Chatzigiannakis, M. Danckwardt, C. Koninis, A. Kröller, G. Mylonas, D. Pfisterer, and B. Porter, "Virtualising testbeds to support large-scale reconfigurable experimental facilities," in *EWSN 2010*, pp. 210–223, 2010
- [25] T. Rakotoarivelo, M. Ott, G. Jourjon, and I. Seskar, "Omf: a control and management framework for networking testbeds," *Operating Systems Review*, vol. 43, no. 4, pp. 54–59, 2009.
- [26] M. Mastrogiovanni, A. Modesti, and C. Petrioli, "JAMES: Java test-bed management system." in *VTC Spring'09*, 2009, pp. 1–6.