

Demo Abstract:

SicsthSense - Dispersing the Cloud

Liam McNamara, Beshr Al Nahas, Simon Duquenooy, Joakim Eriksson and Thiemo Voigt

SICS Swedish ICT AB, Sweden

Email: ljm@sics.se, beshr@sics.se, simondqu@sics.se, joakime@sics.se and thiemo@sics.se

Abstract—This demo presents SicsthSense, our open cloud platform for the Internet of Things. SicsthSense enables low power devices such as sensor nodes and smartphones to easily store their generated data streams in the cloud. This allows the data streams, and their history, to be made permanently available to users for visualisation, processing and sharing. Moving sensor data computation and monitoring into the cloud is a promising avenue to enable centralisation of control and redistribution of collected data.

We showcase SicsthSense running with real sensor nodes collecting environmental data and posting it to our datastore. This live data is then visualised and made available for sharing between users of the platform. Our Android App will also be distributed to enable participants to stream their phone sensors into the system, demonstrating how simple it can be to start machine-to-machine interactions with SicsthSense.

I. MOTIVATION

The “Internet of Things” (IoT) is a unifying term for sensor networks, pervasive computing/connectivity and cloud computing. It represents a large set of modern computing challenges, stemming from the huge scale of the vision. Incorporating billions of devices with a huge variety of heterogeneity in the hardware and applications being run. This idea will involve needs that we cannot even foresee at the moment. Therefore developing solutions with the maximum standards interoperability and flexible extensibility is crucial.

SicsthSense is an open cloud platform for the Internet of Things’ data, designed to ease the interconnection of the billions of sensors and actuators that we expect in the near future. With SicsthSense, devices are registered, discovered, configured, and programmed from the cloud, either through a Web interface or an open machine-to-machine API. This makes it possible to connect heterogeneous devices ranging from smartphones to low-cost embedded sensor and actuators.

We believe the following features are important for promoting easy development of applications in IoT, allowing developers to concentrate on their domain specific problems rather than the “nuts and bolts” of making use of sensed data.

- Easily connect devices to the cloud;
- Store/retrieve sensor data in the cloud;
- Make decisions in the cloud;
- Visualise data;
- Actuate from the cloud.

Solving these problems led to the creation of web-based “resource” repository, where users can have arbitrary de-

vices/websites/URLs periodically polled and parsed to be turned into data streams that are then easy to use in further processing. Alternatively, data can be pushed into the system using our API. All data streams are stored indefinitely, giving users complete historical access. There are also social aspects, whereby users can share reading/writing access to their streams, and subsequently revoke such access.

We aim to facilitate research on the Internet of Things, but also on Big Data analysis, distributed data storage, and interaction design etc. Furthermore, we expect researchers from other fields to benefit from SicsthSense, as users of the platform. SicsthSense supports the HTTP and CoAP protocols, through poll and push interaction schemes, as the main application layer protocols. The system is structured as a true RESTful interface, simplifying the design of client applications. Particular consideration is given for low-power IP devices, such as Contiki nodes using 6LoWPAN and RPL. Figure ?? presents the architecture of SicsthSense.

Long running interactions are enabled by Websockets, so that devices with very high update frequencies can be efficiently supported. Such persistent connections are also used to enable publish/subscribe interactions between devices, so data points can be immediately sent to interested parties, avoiding the need for wasteful polling. Beyond simply storing and retrieving data, SicsthSense also provides features for basic actuation, allowing the system to affect the real world when certain conditions are met by the incoming data streams. For instance, when the average temperature of a user’s house goes below 15C, SicsthSense can signal the heater to switch on.

There are similar commercial offerings in this field, offering a combined data store, visualisation and limited processing engine for IoT applications. The main offering is Xively (formerly COSM), with Open.sen.se and ThingSpeak.com acting as competition. All these commercial products either charge for data access, or require other forms of revenue. SicsthSense not only provides data access free of charge, but allows users to fork the code and run their own instance, giving them complete control of their data and its usage. Other researchers have realised the value of an open academic solution to the problems we have identified [3].

II. ARCHITECTURE

The main SicsthSense platform is a Java-based server that was explicitly designed to be able to run on a wide

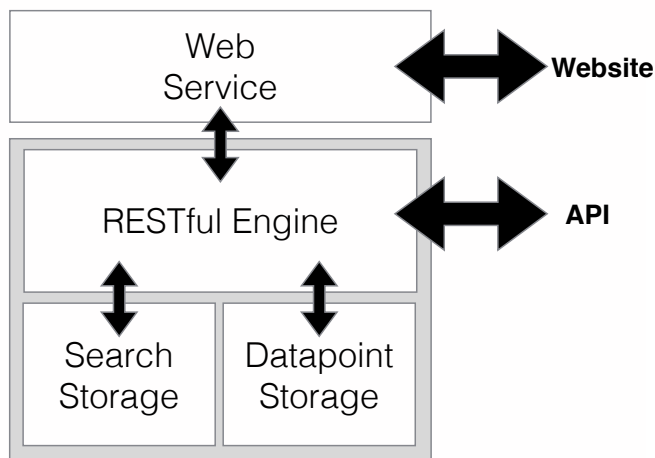


Fig. 1. System Architecture. Showing the main components in the grey box, which provides the machine-to-machine interface, and the optional web interface for a nicer user experience through a web browser.

range of devices, from powerful cloud-based servers to small home gateway machines. The server is distributed as a single .jar file, to enable easy packaging and operation. Storage is provided by two subsystems, first a normal RDBMS that is specifically designed to enable easy searching for relevant feeds. Secondly, a highly scalable asynchronous masterless replication database system is used for storing data points. All the logic, processing and authentication is implemented in the engine sub-system which provides a RESTful HTTP interface for machine-to-machine interactions with SicsthSense. An optional web server is also provided which will provide a user friendly interface to the system. This richer interface is particularly useful non-technical users and allows immediate visualisation of collected data in the SicsthSense system.

Communication with the API is performed using JSON to represent entities. So when users need to request attributes of some part of our system (e.g. user definition or a data point), they simply perform an HTTP GET on a well defined URL and receive a JSON representation of that entity. HTTP POSTs to the same URL can then reconfigure the system.

III. DEMO

The demo will consist of a live demonstration of datastreams being pulled and pushed into the SicsthSense platform. This data will then be visualised, processed and distributed out to endpoints. We will operate a set of sensor nodes running our SicsthSense image of Contiki, each having data posted and/or polled into the SicsthSense system. We will show how the system can be distributed into many federated silos, allowing a local copy of SicsthSense to be run in one's own network, providing low latency and high bandwidth connections. This local silo will then communicate with the wider SicsthSense server, whenever connectivity allows. Our demo will have some sensor nodes communicating with a Raspberry Pi acting as this gateway on a user's own network, before synchronising with the cloud server. The demo setup is illustrated in Figure 2.

Local processing and actuation can also be performed by these local silos, simplifying connectivity and security issues that present themselves when attempting to control devices

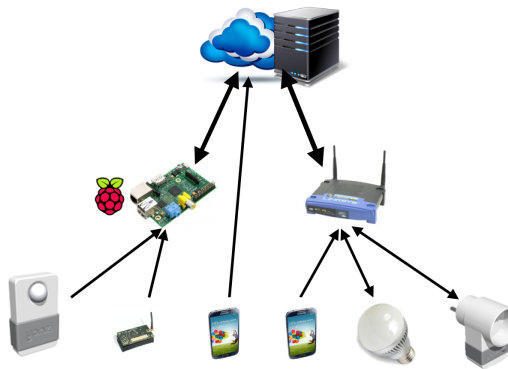


Fig. 2. Demo Setup. With many sensing devices (sensor nodes, IoT devices and phones) all storing their data in the cloud server. Some devices are directly connected, while others connect through a gateway. Some gateway connected devices have their updates aggregated before being transmitted to the cloud.

directly from the cloud. This will be shown through the Raspberry Pi controlling LED lamps that will be present at the demo dependant on the incoming data streams. All the data collected throughout the demonstration will be publically available via the cloud service.

Attendees will be invited to install a SicsthSense App for their Android smartphone to post data themselves. This App will send the phone's current battery life, connectivity strength and other features, allowing their data to be visualised and shared with other attendees at the conference. The simplicity of interaction with our platform will be demonstrated through these interactions, showing how little code and complexity is required to achieve the desired outcomes.

IV. CODE RELEASE

We provide a live running instance of SicsthSense for free (<http://sense.sics.se>) [2] so users can interact with the cloud data platform without having to run their own server. We also release its source code under a BSD open source license [1]. This is of particular use where, for example, a company needs to deploy its own, private, instance of SicsthSense due to sensitivity of their data, or lack of connectivity. Meaning that most importantly the code is open source and free. This repository also contain code for the Contiki image and the android client. It is freely forkable and has already been used as the starting point for other projects. Documentation for creating clients and consumers of the data is also provided on the project Wiki page.

ACKNOWLEDGMENTS

This project was supported by VINNOVA, Energimyn-digheten (the Swedish Energy Agency), and EIT ICT Labs.

REFERENCES

- [1] Available on github here <https://github.com/sics-iot/sicsthsense>.
- [2] Live instance of the platform running at <http://sense.sics.se> and an early feature testing site at <http://presense.sics.se>. 2013.
- [3] Michael Blackstock and Rodger Lea. Iot mashups with the wotkit. In *Internet of Things (IOT), 2012 3rd International Conference on the*, pages 159–166. IEEE, 2012.