Smart Cities: An IoT centric approach

Rodger Lea
School of Computing and Communications
Lancaster University
Lancaster, UK
rodger@comp.ac.uk

Mike Blackstock
Human Communication Technologies Laboratory
University of British Columbia
Vancouver, Canada
mblackst@magic.ubc.ca

ABSTRACT
A number of recent Smart City testbeds and deployments have focused on the use of the Internet of Things (IoT) paradigm and technologies for improving the efficiency of city infrastructures. Building on this work, we have explored the use of IoT hubs as easy-to-use aggregators and focal points for access to emerging data infrastructures of smart cities. A hub can support not only access to infrastructure data, but also participatory sensing and crowd sourced data where city employees and citizens contribute directly to the data infrastructure of a city. In this way, smart cities can realize a variety of new applications created by local entrepreneurs and community groups without the need for ongoing coordination by governments. In this paper, we outline the growing interest in a hub-centric approach to the IoT and discuss our own experiences in building an IoT hub for two Smart City projects, one in the UK and the other in Canada.

Categories and Subject Descriptors
D.2.1 [**Software Engineering**]: Requirements/Specification; D.2.11 [**Software Engineering**]: Software Architectures; D.2.12 [Software Engineering]: Interoperability

General Terms
Smart Cities Software Architectures

Keywords
Smart City, Requirements, Architecture, Internet of Things

1. INTRODUCTION
With its potential to drive growth of local and global economies, the Smart Cities concept has long been recognized as an emerging market opportunity for the IoT. Significant research into the technologies needed to support Smart Cities has been carried out over the last decade, with a focus on using information and communications technologies to manage city infrastructures like transportation, traffic control, building management, energy monitoring, and pollution monitoring. Of particular interest has been the specification and development of platforms that have sought to exploit the Internet Of Things paradigm as the basis for Smart Cities. This has included work by partnerships between local public authorities and private companies such as IBM [13], Cisco [18], Living PlanIT [21], initiatives like the IoT-A [22], and large-scale urban testbeds e.g. [14,15,16].

While the intersection of the IoT and Smart Cities is still an evolving field, we feel that a promising approach is the use of IoT "hubs" as data aggregators and focal points to serve as an easy-to-use service access point to the emerging data infrastructure of a city. By providing a hub, urban data, including static assets and inventories, real time information sensed directly from city infrastructure, and data contributed by community groups and crowd-sourced from citizens, can be made available in an easy-to-use manner for local application developers. By doing this, communities can realize the variety of anticipated and new applications and services necessary to meet the needs of their constituents without requiring the continued involvement of governments and commercial partners.

In this paper, we outline the growing interest in a hub-centric approach to the IoT and discuss our own experiences in building an IoT hub for two Smart City projects, one in the UK and the other in Canada.

2. BACKGROUND
Clearly, the use of IoT solutions for Smart Cities is a broad topic, covering a variety of research ranging from sensor networks through to open-data portals. In our work we have focused on how to build and scale IoT middleware that can be used across a broad range of Smart City research and in particular, the emergence of IoT hubs as a central approach to building urban-scale IoT systems.

2.1 Technology for Smart Cities
IoT technologies in smart cities have included solutions for specific urban infrastructures such as transportation; critical infrastructure such as water, and energy management; urban-scale IoT Platforms; solutions for citizen engagement and integration of social networks into the IoT. Research addressing specific city infrastructures has included, for example, the use of traffic sensing technologies such as magnetic sensors and wifi scanners to assist traffic operators [11]. Experiments with large-scale sensor networks enable real-time monitoring of critical infrastructure such as the urban water supply [6]; defining key interfaces to buildings allows smart-grid managers to interactively manage energy use for the city [12].

While initiatives that focus on infrastructure are important, researchers also recognize that citizens themselves often provide both the needed data and intelligence to make a city 'smart'. Recent efforts have begun to explore platforms to engage citizens directly or by integrating social networks into the IoT [2,8] and using urban crowdsourcing to augment urban data infrastructures [1,19].

It is expected that both the variety and quality of data streams generated by city infrastructure and citizens will continue to increase as additional solutions come online to address efficiency in urban sub-systems. Understanding that it is not enough to create different sub-systems that don't 'talk' to each other, researchers have begun to address interoperability with unified urban-scale sensor networks and large-scale architectures toward unifying Smart City systems to create open innovation platforms [17].

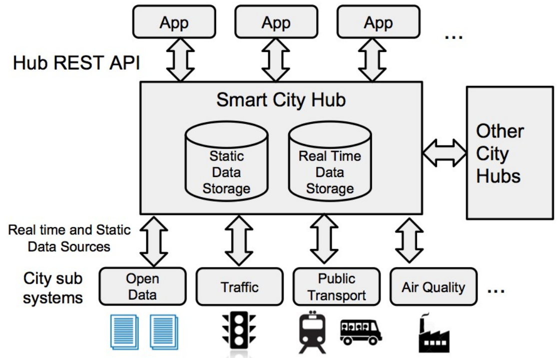

Figure 1. An IoT Hub acts as a portal for smart city infrastructures and a central access point for developers.

2.2 The Hub as an Architecture for the IoT

The use of IoT hubs as an architectural framework for smart cities is a promising one. Hubs address two of the key issues identified above. Firstly, they offer a consistent and easy-to-use interface for emerging IoT infrastructure of the city for systems integrators and application developer. Secondly, they offer a natural interoperability point whereby a common Hub API allows developers to access multiple hubs, each representing a subset of the Smart City infrastructure.

Smart City researchers have begun to recognize that a key enabler toward exploiting emerging Smart City IoT technologies is access to data streams from heterogeneous city sub-systems [5]. If these emerging systems, and existing IoT systems of a city are combined with an open data access policy, developers can begin to find the links between disparate data sources, build the next generation of Smart City applications, and identify opportunities to optimize city services.

In addition to infrastructure data, work has begun to explore how the integration of both static and real time urban data sets from government, community groups and participatory sensing systems. To manage and deliver these diverse data sets, hubs can act as a curated portal for end users and an easy-to-use service access point for developers. Applications accessing these hubs can use this data to adapt themselves to current or expected conditions, addressing needs in areas such as multi-modal transportation, environment waste management, and load management, driven by the needs of urban authorities, or by local entrepreneurs and citizen groups.

By aggregating many systems under a hub, efforts toward interoperability or federation of Smart City functionality can focus on hub integration, rather than the integration of individual city sub-systems. Through the use of interoperable data hubs, application developers can more easily create reusable applications that work in multiple cities.

2.3 Hub evolution

While Smart City hubs will offer the promise of a centralized and easy-to-use access point for Smart City data, we believe that they will offer different features and functionality until agreement is established on what key services a hub should offer and how they are exposed to application clients and other hubs. Until then, we expect the design and implementation of Smart City hubs to evolve over time, following a staged approach as follows [4]:

1. Leverage the Web of Things. Initially Smart City hubs are expected to expose things and associated data sets via Internet protocols, i.e. the IoT. However, going one step further and exploiting the web architecture and RESTful web services, i.e. a “web of things” offers the same opportunities for IoT data and services to be shared as other web-enabled resources [9].

2. Establish models and best practices. Once Smart City hubs are established, it should be possible to find agreement on basic approaches and models used in multiple hubs to facilitate the development of reusable Smart City applications and the federation of hubs.

3. Hub representations. Based on common approaches and models, hub developers can begin to standardize certain implementation issues such as concrete representations, URLs and schema for describing and querying catalogues and data from hubs. This will include support for security mechanisms, so that hubs can control access to hubs and offer some guarantees over who is providing ‘things’ and their data.

4. Hub semantics. Given the need for applications to find the same types of ‘things’ and datasets across hubs, it will be necessary to agree on the semantics of things and their associated data, for example that an air quality sensor in one city hub provides the same quality and value of as one in another hub. Essentially the taxonomy of ‘things’ and the ontological models that hubs support will need to be defined. By reaching agreement at this level, deeper application integration is possible allowing Smart City hubs and ‘things’ to link to and communicate directly with each other.

3. SMART CITY DATA HUBS

We have been exploring the use of IoT hubs as the basis for Smart City projects since early 2013. Our explorations have resulted in two significant deployments, one in the UK, focusing on road and highway infrastructure (Smart Streets[1]) and the other in Canada (Urban Opus[2]) focused more generally on Smart Cities.

3.1 Smart Streets

In early 2013, the UK’s Technology Strategy Board (TSB) invested in a project called the Internet of Things Ecosystem Demonstrator to stimulate the development of an open application and services ecosystem in the IoT. In this project, eight industry-led projects were funded to deliver IoT clusters in the spring of 2014. These projects all explored the use of an IoT hub to represent clusters of things from different aspects of smart cities and smart infrastructure. These clusters covered a range of areas including smart schools, urban transportation, airports, smart homes and critical infrastructure such as roads and highways. As developers of the Smart Streets IoT Hub, our focus was the Highways maintenance sector (a $6B sector in the UK), which gathered data from a variety of sources related to the UK’s national and regional road network. This effort was led by InTouch Ltd., a UK-based SME that provides data solutions for many companies that maintain the UK’s highways infrastructure. Partners included three large companies, Amey, Balfour Beatty, and Carillion, engineering companies that build and manage public infrastructure, Sense Tecnic Systems Inc. that provides IoT hubs, and the University of Lancaster as a research partner. Data included real-time traffic flows, incidents that affected traffic flows, road works, flood and rain data, all of which were made available via the Smart Streets Hub. A particular focus of the programme was establishing interoperability between the 8 hubs, which resulted in the specification of a lightweight interoperability protocol for IoT hubs, known as HyperCat [20]

3.2 Urban Opus

The Urban Opus Society is a non-profit corporation established in Vancouver, Canada, to foster the development of innovative Smart City applications, involving a mix of citizen, government, private sector, and infrastructure data. To support this effort, the Urban

[1] http://smartstreets.senstecnic.com
[2] http://www.urbanopus.net

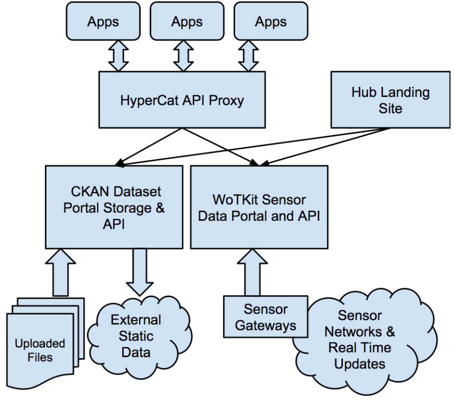

Figure 2. City data hub architecture unifies CKAN and WoTKit systems with a unified API.

Opus Hub provides data storage and federates existing data sources to provide a single on-line presence and point of access to these data sets. The system shares the same basic architecture as Smart Streets with support for both real time and static data, and an easy to use API for developers

4. HUB ARCHITECTURE

In both projects, a critical challenge we faced was the need to collect and manage a diverse set of existing data sources, ranging from real-time data on traffic flow or water levels in roadside drains, to soft real-time data, such as roadwork schedules, through to relatively static data, such as asset lists of highway signs, bridges, markings etc. in our IoT hub and provide a uniform APIs to this data.

Both hubs were built from two core components. Firstly, our own IoT platform called the Web of Things Tool Kit (WoTKit) [3] and secondly, an open source system called CKAN [10] designed to support static data and metadata storage as illustrated in Figure 2. We integrated the CKAN open-data portal as well as the WoTKit platform, allowing users to retrieve information about both static and real-time data sources using the same interface.

4.1 Managing Real-time Sensor Data

The WoTKit [3], under development since 2009, is a web-centric IoT toolkit, focused on managing 'things' that exhibit real-time behaviour. Running as a cloud service, its APIs offer developers a comprehensive set of IoT services making it easy to develop web applications and services for the IoT. Users can create 'sensors' with the UI or API that represent 'things', can receive data from those things, and can send control commands. Sensor data can include any mix of text and scalar values, and sensors can be grouped, tagged, and associated with metadata to facilitate search. The WoTKit includes a UI for viewing sensor data using customizable dashboards, managing alerts, and creating real-time sensor data mashups.

4.2 Managing Static Datasets

CKAN [10] is a data management system and portal that allows data publishers like governments, companies and other organizations to make their data available to others and is a critical part of the Open Data movement. It allows data publishers to easily upload and publish new datasets containing one or more data resources, providing versioning and support for multiple formats. Datasets can be associated with organizations for access control. CKAN provides an API allowing developers to search for, download and, in some cases, query for data within relevant datasets. In both hubs we used the CKAN system to store data sets that are static or do not change often (e.g. monthly or annually).

4.3 IoT Data and Access Management

In our data hub it was important to address two issues related to the management of data. Firstly, providing consistent access to IoT resources, specifically real-time and non real-time data. Secondly, integrating a variety of data available from other systems to make it available on the hub in a consistent and easy-to-use manner.

To address these, we created a tool called the *API Proxy* to provide a unified API to the catalogue of resources available on a Smart City hub, and the Harvester, a tool designed to aggregate data from diverse data sources and push them into the hub to make them available to IoT developers.

4.4 IoT Catalogue: Leveraging HyperCat

Both CKAN and WoTKit support API calls to view a 'catalogue' of resources, but the formats and APIs to access these catalogues are very different. To support interoperability between these systems, we needed to adapt the catalogue APIs for both the WoTKit and CKAN to a common API.

To achieve this, we leveraged the HyperCat specification [20], developed by the 8 consortia involved in the UK's IoT projects. HyperCat specifies a lightweight hypermedia catalogue for querying and representing catalogues of resources (URIs) on the web. Exposed resources are described by a list of RDF-like triple statements to provide information about the format and semantics of the URI. This enables applications to search for suitable resources and understand the data when they retrieve it. Because of its simplicity, developers can easily publish descriptions of the resources they expose; applications can easily query for the things they are interested in.

The *API Proxy* provided a unified HyperCat catalogue and query API to both CKAN and the WoTKit, allowing access to the real-time and non real-time parts of our hubs. This integration required that we address a number of technical issues:

Search engine implementation: To support flexible search, our initial HyperCat implementations leveraged the Apache Solr search platform to both store and search catalogues. Scripts were created to periodically import WoTKit and CKAN data into the search engine for access by the API. While this solution worked well for public sensors and data sets that did not change often, access control by the underlying system could not be supported without replicating access control logic of the underlying system. Moreover, if the catalogue changed, the catalogue exposed by the *API Proxy* was out of date until the next catalogue import. To address these issues, we extended the *API Proxy* to query the integrated systems' APIs directly. This involved translating the query format to an appropriate API call to the underlying system, and then converting the response from the API to the HyperCat format 'on the fly'. While the concept seemed straightforward, it raised a number of issues related to access control and security, query capability/semantics mismatch, and catalogue scale.

Unified access control: To ensure that users could only access datasets or sensors they were permitted to, we needed to unify the user accounts and access control mechanism used by both systems. CKAN supported the notion of a single key for use with the CKAN API for access control, while the WoTKit supports several authentication methods including OAuth2. To unify access control we decided to modify the WoTKit to support a CKAN authentication keys, associating this key with a WoTKit user. Another approach would have been to maintain a mapping to API keys or access tokens in the underlying systems. We ensured the user credentials in both systems were kept in sync on both systems using CKAN extensions and the administrative API on the WoTKit. With unified access control, only the sensors or CKAN data sets visible to the user associated with the API key will be queried and returned by the hub.

Query mismatch and filtering: The initial query semantics for HyperCat, called 'simple search' allows users of the API to query for certain catalogue items by specifying whether a metadata key exists, and/or is set to a certain value. To perform a simple search, clients provide a query string specifying the specific item URI, relationships and/or values of the items they are interested in. All items with metadata that matches the query parameters must be returned. While this sounds straightforward, because of limitations or the semantics of certain metadata in the underlying systems, we weren't always able to search by the existence or value of certain metadata, nor did it always make sense to do so. For example, we included WoTKit sensor location as metadata, but since the WoTKit API did not provide a mechanism to find all sensors with a certain latitude value, just geo queries in a range, we could not map the HyperCat simple search to the WoTKit search API. In the WoTKit, we provided a way to find sensors that contained certain text strings in the name or description, but not to find sensors that matched these metadata values exactly. To address this, we extended the *API Proxy* to filter generated HyperCat responses by query relationships and values. While this worked, it meant that we had to retrieve more data from the WoTKit than necessary to satisfy a query, reducing the performance of the system.

Large catalogues. Since the hub contained more than 40,000 gully sensors from a given region, we needed a way to partition this large catalogue into manageable 'chunks' to avoid overwhelming clients of the API. One approach we considered was to split the sensor catalogue into sub-catalogues corresponding to 'pages' of a larger catalog, but realized by doing so that this would not permit searching all sub catalogues. To address this, we requested a change to the specification to support catalogue paging; new query parameters were added to limit the number of items returned to a query on a large catalogue.

4.5 Data interoperability: The Harvester

A secondary issue we faced was integrating a variety of heterogeneous data from a set of disparate sources, ranging from real time air quality data to real-time traffic sign updates. To address this, we developed an additional tool called the Harvester, loosely based on the CKAN Harvester plug-in used for federating CKAN open data portals. Like data integration Extract Transform and Load (ETL) tools, the Harvester integrates "legacy" sources of data into a common 'web of things' hub. Rather than simply extracting data from databases, the Harvester extends that capability to also extract virtual sensor data buried in web pages, XML data feeds and other web formats. It normalizes this data and uploads it into the WoTKit for easy access by developers. Today the Harvester loads data into one or more instances of the WoTKit platform using its REST API. To accomplish this, the Harvester framework is responsible for managing and executing

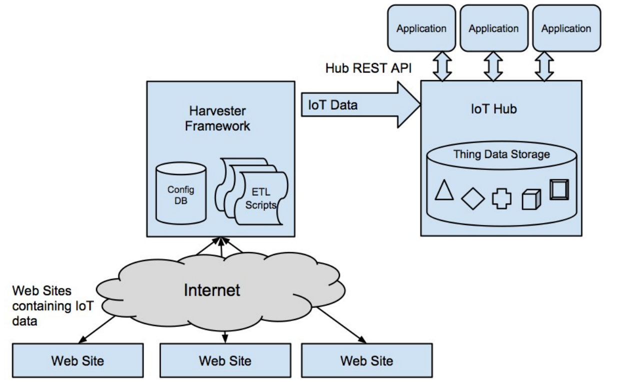

Figure 3. The Harvester pulls in data from web sites updating sensor streams in the WoTKit

ETL scripts to pull data from web sites and update sensors in the WoTKit data hub using its API as illustrated in Figure 3.

To add a new IoT ETL script to the Harvester, the user sets up a script configuration, then adds the script to the script folder or edits it directly in the user interface. The script configuration describes how often the script needs to run, which hub 'targets' to send data to, and the sensor registration information to register the appropriate sensor feeds. Because scripts are accessing external services, and may themselves have bugs, scripts are spawned as child processes. If they encounter errors, or are not responsive after a period of time, the framework will log errors and send notifications, then attempt to kill and restart the process for a number of retries.

5. EXPERIENCES

Both Smart City hubs have been deployed and used for a variety of applications. Of the two, the Smart Streets IoT hub has been operational for slightly longer, approximately 10 months, and currently manages over 64,000 time-series sensor feeds and a wide variety of static datasets. It includes a diverse set of both open and private data about transportation, road traffic, and highways, ranging from real time traffic data, to road asset condition, planned roadworks, air quality, weather and flooding information. These data sources have been pushed into the hub either via tools such as the Harvestor, by end users uploading data sets, or from physical devices, explicitly sending information to the Hub via its APIs. At a recent hackathon, 50+ participants from Switzerland, Germany and the UK developed a series of apps, and over a two day period generated more than 300K Hub API calls transferring over 9 GB of data.

In addition to these Hackathon apps, our group and others have built a variety of web and mobile IoT applications and experimented with both the abstractions and the Hub APIs.

To make it easy to find and install hub applications, both systems include 'app store' functionality to list featured applications and provide a way of rating and searching for applications that make use of the hub as illustrated in Figure 4.

Applications developed for the Smart Streets Hub have included a "Catalogue Explorer" to browse not only our hub catalogue, but that of the other seven hubs. Other roadworks-related applications include one to visualize correlations between drain blockage and road works called "Roadworks Gully Correlator", a predictive application called "Pothole Predication" for pot hole analysis, and "Cycle Spot", an app to allow cyclists to avoid road hazards, including roadworks, poor road conditions and winter issues such as ice. The Urban Opus hub includes several applications, e.g. a 311 visualization application allowing citizens to explore data about citizen requests and complaints, 'Bike Racks' to find and report on the condition of bike parking in the city, and 'Street Trees' to access and contribute data to a database about Metro Vancouver's urban forest.

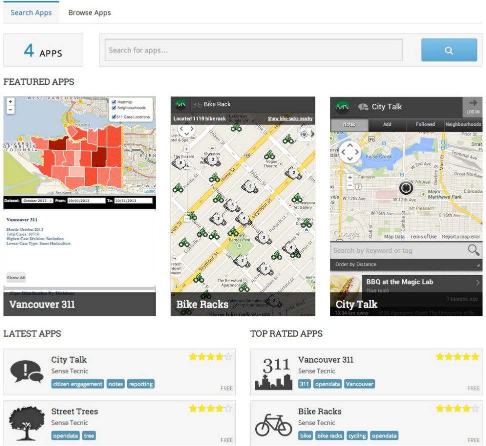

Figure 4. Urban Opus Hub 'App Store'.

The Smart Streets "Gully and Roadwork Correlator" application leveraged physical drain condition sensor data (silt and water levels) and data provided by work crews related to road repairs and gully levels. As such it provides an interesting example of physical infrastructure data with data "crowdsourced" from city workers

The Urban Opus 'Bike Rack' Application leverages the bike rack inventory of the city to display locations where cyclists can safely lock up their bikes. Users can report problems with these locations such as vandalism or full bike racks. The Hub logs interaction with the application, including where and when users search for bike racks. Using this data, our hope is that authorities in the region can prioritize investments in maintaining and purchasing more bike racks for cyclists.

6. RELATED WORK

Platforms for unifying IoT resources for a Smart City have been the focus of several Smart City testbeds [7]. IoT hubs and large-scale sensor networks have been used for making a variety of data streams from the physical environment available to application developers [23,24]. Finally, large-scale IoT projects have begun to address interoperability between IoT domains for industrial and Smart City applications.

The SmartSantander testbed includes a platform for experimenting with a variety of IoT technologies. One of the goals of the system is to address the inherent heterogeneity of IoT resources [16]. The testbed deployed in Oulo, Finland [15], aimed to provide systems infrastructure support for application developers of public space services via a set of middleware tools. The CitySense testbed provided a city-wide platform to enable large-scale sensor and wireless networking research in a real-world urban setting [14]. While all of these systems provide centralized platforms, they did not aim to provide a web-based hub acting as a centralized access point for accessing both real-time sensor streams and static datasets.

Large-scale IoT hubs allow web developers to integrate 'things' across a wide variety of domains. Our own work, the WoTKit [3], as well as Xively [23] aggregate collections of data streams called feeds to store information about sensors and the data they emit over time. Similarly, ThingSpeak [24] supports a data model of channels similar to Xively and WoTKit feeds. All three include applications for processing, visualization and integration and offer the ability to find and share sensors and data, allowing others to take advantage of the integration work of others. Each of these platforms offer a 'hub' model to provide a repository for 'things' (data and metadata) and a set of APIs for accessing and using 'things'. These hubs do not focus on supporting Smart City applications per se, and while they do support real time data streams typically do not support static data set management.

The Internet of Things Architecture project (IoT-A) is proposing an architectural reference model for IoT interoperability together with key components of the future IoT to enable search, discovery and interaction as one coherent network [22]. This work offers a comprehensive approach to building IoT platforms, potentially at city scale, rather than providing a single focal point for accessing the data of a Smart City.

7. LESSONS

While developing applications using our hub, we have identified a number of issues relating to data and sensors as well as the hub architectural model and APIs.

Hub architecture and APIs: Making a diverse set of data sets and IoT resources available on a hub using a common catalogue was our initial focus in creating Smart City hubs. Even within the limited scope of the HyperCat specification we found that integrating the CKAN and WoTKit systems was difficult. Some of the challenges included, the need to resolve different access control mechanisms, different query semantics, and dealing with large datasets and data catalogues.

Based on our experience with HyperCat, our hub containing both CKAN and WoTKit data feeds, and the diverse data already available on the web, we believe that it may be more practical to agree on how to describe domain-specific data formats rather than agreeing on one format for all IoT resources. Using a data format agnostic catalogue like HyperCat with a flexible metadata facility for describing both static data sets and IoT resources will allow application developers to decide whether they can consume the data exposed by a given resource.

Data: Firstly, the use of static data from open data catalogues often required cleaning and transformation for use by applications. Secondly, applications often did not access the API directly, but used scheduled jobs to periodically download data from the hubs into their own data storage for faster and more flexible queries into the data. In some cases, developers then modified the captured data with their own information collected by end users. Over time we expect that city data hubs will need to provide data cleaning, more flexible query capabilities, and data versioning to reduce the effort required for application developers to create apps, and collect data crowdsourced from citizens and other users.

We believe that the use of ETL tools such as the Harvester is a good interim step toward making a diverse set of data streams available on a Smart City hub. Once data has been made available on a hub, developers can focus on their applications rather than worrying about the location and formats of the real-time data they needed.

Sensors: a number of issues related to sensors and 'thing' management arose during the projects. One area of significant concern is around the taxonomy and semantics of sensors/things. We quickly found, when interoperating with other hubs, that the lack of an agreed taxonomy and semantics led to reduced interoperability. Another issue was the use of the hub as a proxy for real-time sensors/actuators – although the hub provided an excellent way to aggregate data, it sometimes caused problems interacting with objects that required real or near real-time response.

8. CONCLUSIONS

The use of a Hub centric approach as a basis for developing the Internet of Things is a promising one. In our work we have focused on the IoT and Smart Cities and used our IoT hub as a platform and testbed for two deployments, one in the UK and one in Canada. One of the significant challenges faced by the IoT is that of interoperability, i.e. how do devices/sensors/things work together, how are they found, represented on the Internet, accessed and controlled? Our approach has been to use IoT hubs to provide a well-defined API that provides a common way to find and control IoT objects. One significant advantage of this approach is that multiple hubs can be connected, or federated to build up a system of systems that can represent significant parts of the IoT ecosystem – for example the components of a Smart City. We have demonstrated some of the advantages of this approach through our 'in the wild' deployments both in the UK and Canada, and through the large number of applications and services developed for the hubs. Clearly, we still have many issues to resolve as we explore this approach to Smart Cities. However we feel our approach is a valid one and that some of the issues we have uncovered, and the lessons we have learned, will help others in the IoT community as we collectively develop a truly global Internet of Things.

9. ACKNOWLEDGMENTS

We are indebted to our colleagues in the Smart Streets IoT project team, especially those at In Touch Ltd and Lancaster University. The HyperCat work is the result of collaboration by the 8 IoT hub projects who participated in the interoperability working group. Partial funding for this work was provided by the TSB and NSERC.

10. REFERENCES

1. Benouaret, K., Valliyur-Ramalingam, R., and Charoy, F. CrowdSC: Building Smart Cities with Large-Scale Citizen Participation. *IEEE Internet Computing 17*, 6 (2013), 57–63.
2. Blackstock, M., Lea, R., and Friday, A. Uniting online social networks with places and things. *Workshop on the Web of Things (WoT 2011)*, ACM (2011), 5:1–5:6.
3. Blackstock, M. and Lea, R. IoT mashups with the WoTKit. *Internet of Things (IOT), 2012 3rd International Conference on the*, IEEE (2012), 159–166.
4. Blackstock, M. and Lea, R. Toward Interoperability in a Web of Things. *ACM Pervasive and Ubiquitous Computing Adjunct Publication*, ACM (2013), 1565–1574.
5. Boyle, D.E., Yates, D.C., and Yeatman, E.M. Urban Sensor Data Streams: London 2013. *IEEE Internet Computing 17*, 6 (2013), 12–20.
6. Difallah, D.E., Cudre-Mauroux, P., and McKenna, S.A. Scalable Anomaly Detection for Smart City Infrastructure Networks. *IEEE Internet Computing 17*, 6 (2013), 39–47.
7. Gluhak, A., Krco, S., Nati, M., Pfisterer, D., Mitton, N., and Razafindralambo, T. A survey on facilities for experimental internet of things research. *IEEE Communications Magazine 49*, 11 (2011), 58–67.
8. Guinard, D., Fischer, M., and Trifa, V. Sharing using social networks in a composable Web of Things. *2010 8th IEEE Pervasive Computing and Communications Workshops (PERCOM Workshops)*, IEEE (2010), 702–707.
9. Guinard, D., Trifa, V., Mattern, F., and Wilde, E. From the Internet of Things to the Web of Things: Resource Oriented Architecture and Best Practices. In D. Uckelmann, M. Harrison and F. Michahelles, eds., *Architecting the Internet of Things*. Springer, London, 2011, 97–129.
10. Josh Winn. Open data and the academy: An evaluation of CKAN for research data management. http://eprints.lincoln.ac.uk/9778/1/CKANEvaluation.pdf.
11. Kostakos, V., Ojala, T., and Juntunen, T. Traffic in the Smart City: Exploring City-Wide Sensing for Traffic Control Center Augmentation. *IEEE Internet Computing 17*, 6 (2013), 22–29.
12. Lee, E.-K., Chu, P., and Gadh, R. Fine-Grained Access to Smart Building Energy Resources. *IEEE Internet Computing 17*, 6 (2013), 48–56.
13. Michael Kehoe, Michael Cosgrove, Steven De Gennaro, et al. A Foundation for Understanding IBM Smarter Cities. http://www.redbooks.ibm.com/redpapers/pdfs/redp4733.pdf.
14. Murty, R.N., Mainland, G., Rose, I., et al. CitySense: An Urban-Scale Wireless Sensor Network and Testbed. *2008 IEEE Conference on Technologies for Homeland Security*, (2008), 583–588.
15. Ojala, T. Open Urban Testbed for Ubiquitous Computing. *2010 International Conference on Communications and Mobile Computing (CMC)*, (2010), 442–447.
16. Sanchez, L., Muñoz, L., Galache, J.A., et al. SmartSantander: IoT experimentation over a smart city testbed. *Computer Networks 61*, (2014), 217–238.
17. Schaffers, H., Komninos, N., Pallot, M., Trousse, B., Nilsson, M., and Oliveira, A. Smart Cities and the Future Internet: Towards Cooperation Frameworks for Open Innovation. In J. Domingue, A. Galis, A. Gavras, et al., eds., *The Future Internet*. Springer, 2011, 431–446.
18. Shane Mitchell, Nicola Villa, Martin Stewart-Weeks, and Anne Lange. The Internet of Everything for Cities. http://www.cisco.com/web/about/ac79/docs/ps/motm/IoE-Smart-City_PoV.pdf.
19. Zambonelli, F. Pervasive urban crowdsourcing: Visions and challenges. *2011 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, (2011), 578–583.
20. HyperCat: an IoT interoperability specification. 2013. http://www.research.lancs.ac.uk/portal/services/downloadRegister/53462399/Interoperability_Action_Plan_1v1_spec_only.pdf.
21. Cities in the Cloud: A Living PlanIT Introduction to Future City Technologies. http://www.cisco.com/web/about/ac78/docs/Living_PlanIT_SA_Cities_iWhitepaper.pdf.
22. Internet of Things - Architecture — IOT-A: Internet of Things Architecture. http://www.iot-a.eu/public.
23. Xively by LogMeIn. https://xively.com/.
24. The Internet of Things - ThingSpeak. https://thingspeak.com/