CityHub: A cloud based IoT platform for Smart Cities

Rodger Lea
Human Communication Technologies Laboratory
University of British Columbia
Vancouver, Canada
rodgerl@ece.ubc.ca

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

***Abstract*—Cloud based Smart City hubs are an attractive approach to addressing some of the complex issues faced when deploying PaaS infrastructure for Smart Cities. In this paper we introduce the general notion of IoT hubs and then discuss our work to generalize our IoT hub as a Smart City PaaS. We briefly describe our approach and discuss our experiences deploying two cloud-based Smart City hubs, one in the UK and the other in Canada.**

Keywords-component; Cloud based IoT, Data Hubs, Smart Cities

I. INTRODUCTION

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. Within this broad set of research, some researchers have focused on the specification and development of platforms that have sought to exploit the Internet of Things (IoT) paradigm as the basis for Smart Cities. This has included work by partnerships between local public authorities and private companies e.g. IBM, Cisco and Living PlanIT [11], reference initiatives like the IoT-A [9], and large-scale urban testbeds e.g. [14,15,16].

One area that we feel offers a promising approach is the use of cloud-based IoT "hubs" that provide an easy-to-use service access point to the emerging data infrastructure of a city. By providing a cloud-based 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 application developers. In this paper, we detail our 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. We describe the architecture of our cloud-based Hub and discuss our use of Hybrid public/private cloud and application partitioning to support Hybrid Cloud infrastructure. Finally we describe our future work that looks to extend the notions of Hybrid cloud partitioning to end points, i.e. IoT devices, and outline some of the technical issues we face.

II. BACKGROUND AND RELATED WORK

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. Experiments with large-scale sensor networks that enable real-time monitoring of critical infrastructure have included work on the urban water supply [6] and defining key interfaces to buildings to allow smart-grid managers to interactively manage energy use for the city [12].

Smart City testbeds have included the SmartSantander testbed which 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 CitySense testbed provided a city-wide platform to enable large-scale sensor and wireless networking research in a real-world urban setting [14].

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] and explore cloud technologies as the basis for open platforms [8, 1].

The Internet of Things Architecture project (IoT-A) has proposed 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 [9]. 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.

Work on Cloud based IoT platforms – in some cases focused on Smart Cities – includes the work of the OpenIoT project [7] as well as Li et al [8] and ClouT [2]. In our work we have focused on how to build and scale a cloud-based IoT middleware that can be used across a broad range of Smart City research and in particular, the use of Smart City data hubs as a central approach to building urban-scale IoT systems [4].

978-1-4799-4093-6/14 $31.00 © 2014 IEEE
DOI 10.1109/CloudCom.2014.65

A. *IoT Hubs for Smart Cities*

Cloud-based hubs offer a promising approach to developing an IoT centric framework for smart cities and address two of the key issues identified above. Firstly, they offer a consistent and easy-to-use interface for emerging IoT infrastructure within the city that systems integrators and application developers can use.

Secondly, they support the system-of-systems approach to smart cities whereby a cloud-based hub can integrate a number of sub-systems that collectively make up the complete smart city software infrastructures [5].

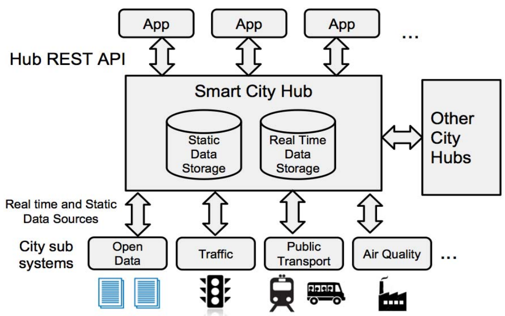

Figure 1. An IoT hub acts as a portal for Smart City Infrastructure as well as other hubs.

In addition to infrastructure management, cloud based hubs also offer a framework to integrate 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.

B. *Cloud considerations*

While Smart City hubs will offer the promise of a centralized and easy-to-use access point for Smart City data, there are a number of complexities driven by the realities of Smart City infrastructure and services.

1. Hybrid-Cloud requirements. It is unrealistic to expect that future Smart City Hubs will be built as single stand-alone cloud-based hubs. Rather, they will need to address the reality of existing private infrastructure. Today, most cities run their own infrastructure software managing a variety of areas such as traffic management, licensing, water and sewerage, 311, citizen engagement etc. This infrastructure is unlikely to completely migrate to public cloud infrastructure and so any eventual Smart City hub will have to address hybrid public-private cloud.

2. Cloud-Cloud or Integrated Cloud. As Smart City infrastructure evolves, it will need to address the issue of federated cloud services, or integrated clouds. City infrastructure is generally a system-of-systems approach with a number of self contained services supporting different aspects of a smart city, e.g. water, transport etc. While it is possible that in some cases, such sub-systems can be supported by a single cloud framework - as multiple services in a multi-tenant environment, it is also likely that a number of PaaS platforms will exist catering to different aspects of the smart city. There is a need therefore, to support integrated clouds to ensure new services can leverage across city infrastructure.

III. SMART CITY DATA HUBS

Since 2012 we have developed and deployed a number of cloud based Smart City technologies that leverage our underlying IoT platform. Two of the most significant are one in the UK, focusing on road and highway infrastructure (Smart Streets) and the other in Canada (Urban Opus) focused more generally on Smart Cities.

A. *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 which gathered data from a variety of sources related to the UK's national and regional road network. 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 [19]

B. 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 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.

C. Core IoT services and city centric data flows

In both projects, a critical challenge that surfaced early 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. This requirement extended the traditional scope of an IoT platform which generally focuses on real-time systems and control of infrastructure nodes (things) embedded in the physical world. A mundane but illustrative example from our work would be planning application data. In most cities, such data is of significant importance to planners and citizens alike, but is generally not a real time data feed. Usually, applications are captured in static files created by developers and city officials. Any smart city platform needs to be able to handle these and make them available in some form to developers.

To address this need we extended our core IoT platform called the Web of Things Tool Kit (WoTKit) [3] with an open data system called CKAN [10] designed to support static data and metadata storage as illustrated in Figure 2.

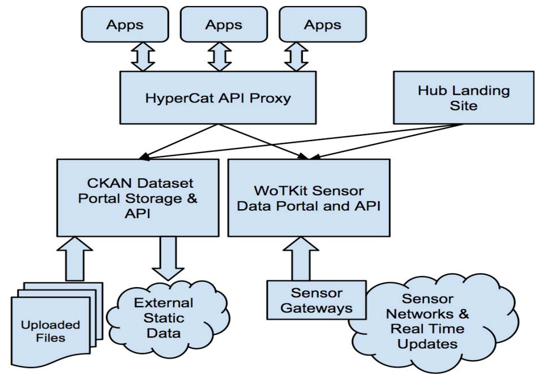

Figure 2. The core IoT infrastructure supports access to real-time and non-real time urban data streams.

Managing Real-time Sensor Data

The WoTKit, under development since 2009, is a web-centric IoT toolkit, focused on managing 'things' that exhibit real-time behaviour. 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 platform is sensor or data feed agnostic and treats data from roadside traffic sensors in the same way as data from social network feeds such as Twitter. The WoTKit includes a UI for viewing sensor data using customizable dashboards, managing alerts, and creating real-time sensor data mashups.

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.

IV. CITYHUB

CityHub extends our initial work on the use of data hubs for Smart Cities by formalizing our IoT framework as a Smart City PaaS framework. The goal of CityHub is to provide a framework that makes it easy for 3rd parties to develop and deploy Smart City applications. These 3rd parties can be citizen groups aiming to provide a service for fellow citizens, local companies offering hyper local services, or even national companies offering new services to citizens. The framework leverages cloud infrastructure and offers a core IoT capability to manage and interface with Smart City infrastructure. As discussed above, the core of the CityHub is the IoT Data Hub that provides a common API for access to and management of Smart City data streams. As can be seen in Fig 3, the core Hub API provides a common set of capabilities for access and control of infrastructure as well as 'softer' data such as citizen contributed data and open data. The CityHub cloud infrastructure layers a cloud service layer on the IoT framework making this available to application developers. This offers multi-tenant support allowing multiple application providers to instantiate services and manages resources associated with those services. Lifecycle management supports the creation of new services and their use of platform and IoT resources to meet service requests (and constraints). Lastly, a resource usage and billing capability is provided to ensure efficient usage of resources and appropriate charge out to service providers – and eventually their end customers.

Service developers use the Cloud infrastructure (PaaS) to instantiate new services that sit within the framework,

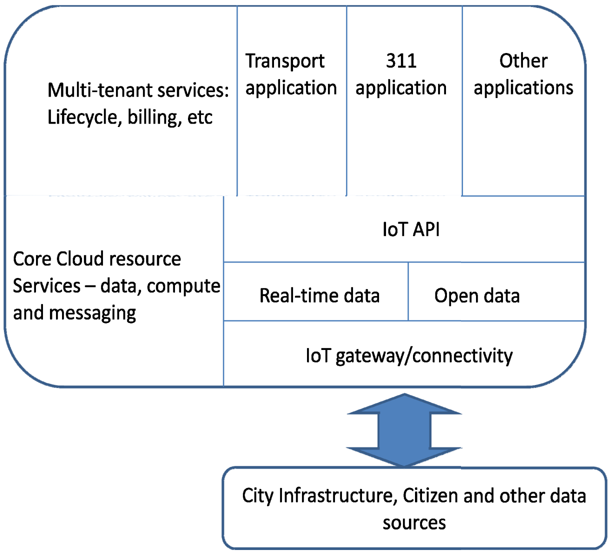

Figure 3. An IoT centric PaaS framework for Smart Cities

accessing the common IoT APIs as well as other service PaaS components and providing service end points.

A simple example of a Smart City service could be a citizen focused traffic/transportation app that aimed to provide to end users (citizens) comprehensive information on the status of the city's road, rail and bus network offering information on best routes, problems etc. Such an application would use data from a variety of city sources including public updates from transportation services, citizen reported information, information gathered from social media feeds and even perhaps, if available, sensed information from the city traffic monitoring facilities. Using this wide range of data the Transport App would be able to update citizens on status of the transportation network, offer real-time advice on routes (and re-routing) and even give advice on ride sharing, or sustainable transportation choices. In CityHub, this application would be instantiated within the cloud framework using the PaaS capabilities, would use the core data hub to gather data from a variety of registered data feeds and would be managed and billed via the multi-tenant facilities of CityHub.

A more complex example of a city service is a citizen reporting application that allows citizen to report issues to the City. Examples of these include popular mobile apps such as SeeClickFix or FixMyStreet. These applications are more complex in that they can be provided by 3rd parties as a managed application within the CityHub, but they will interact with city services running on City IT infrastructure. For example, a citizen report of a pothole in a local street would be reported from their mobile device to the CityHub, but then, once classified, would need to be sent on to the city for acknowledgement and action. One approach to this exploits the fact that in N. America, many cities are moving towards an open 311 framework that provides a single interface to city services. Users access the 311 service through phone or a web browser and their service request is routed depending on its nature, e.g. garbage complaint versus development permit. Open 311 offers a well-defined API for web based provisioning of 311 related services offering a common service API. One way to exploit this open 311 API is for the CityHub to provide, as part of its PaaS, a proxy API for applications that forwards API requests from CityHub application through to the appropriate city server. Interestingly, in cities that have not implemented their own open311 API, it would be possible to host the open311 front end entirely in the CityHub, thus distributing the implementation of a city service between the public cloud (CityHub) and private premises, i.e. the City IT servers.

V. CITYHUB IN THE CLOUD

In section II.B we indicated two key research areas when utilizing cloud infrastructure in support of Smart City data hubs, i.e. how to support hybrid cloud and how cloud-to-cloud federation can be achieved. Of these, we have actively explored Hybrid Cloud support via work on application partitioning. Integrated or federated cloud issues remain an area for future work.

A. Hybrid cloud and application partitioning

As discussed above, it is clear that any cloud platform designed to support smart cities will need to address the complex set of existing services that any city infrastructure entails. While it is possible to design and build a 'greenfield' cloud-based hub for smart city management, in reality any such hub will have to contend with significant legacy software systems. As such it is necessary to ensure that our hub architecture is able to support Hybrid cloud situations where smart city applications and services reside on both public cloud infrastructure and private (e.g. city) infrastructure. An example of this would be a 311 service where the core web portal is migrated to the public cloud – for cost and performance reasons, but the core backend services remained on City IT premises for security and management reasons.

To support this scenario we have explored application partitioning for hybrid cloud [13,18] with a particular focus on combining code (application) and data partitioning. To address partitioning, we follow the following methodology. The application is initially profiled by measuring execution time on a reference machine and collecting the data exchanges between software functions and data entities. Using this profile information, a dependency graph is developed which can then be analyzed to examine the effects of applying different cost models and placement constraints. Finally, the dependency graph can be converted into an optimization problem and solved using binary integer linear programming (BIP) to partition the application.

Of particular note within the context of Smart Cities has been our interest in developing partitioning algorithms that exploit cost models offered by public cloud infrastructure. Specifically, we have developed a flexible approach that allows City IT planners to explore the cost implications of moving service and application code from private on-premises to public cloud which often has significantly lower costs. One aspect of many public cloud providers is that they adopt an asymmetric charging model for data, i.e. data pushed to the cloud is either not charged for, or charged at a lower rate than data pulled from the cloud. Our partitioning algorithms allow for the exploitation of this and other aspects of cloud cost models as developers search for the correct trade-off between performance and cost that their application needs. As City IT departments face increased cost constraints, this approach allows them to balance the need to maintain legacy or privacy constrained code within their private infrastructure but exploit, where appropriate, the lower costs of public cloud infrastructure. In our initial experiments, we have seen cost savings of up to 40% for typical 3 tier applications as we have deployed across public and private cloud. Full details of our approach to Hybrid Cloud partitioning can be found in [18].

VI. DEPLOYMENT EXPERIENCES

Both Smart City hubs have been deployed and used for a variety of applications. The hubs are deployed on different cloud services, with the UK hub running in the Amazon EU (Ireland) region and the Canadian hub running in academic cloud infrastructure. Of the two hubs, Smart Streets has been operational for slightly longer, approximately 18 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 roadwork, air quality, weather and flooding information. These data sources have been pushed into the hub either via tools that harvest data from the web, by end users uploading data sets, or from physical devices, explicitly sending information to the Hub via its REST 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.

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 roadwork related applications include one to visualize correlations between drain blockage and road works called "Roadwork Gully Correlator". This application leveraged physical drain condition sensor data (silt and water levels) and data provided by work crews related to road repairs and gully levels. It correlates the two and allows city managers to understand the impact of roadwork on city drains and the need for adaptive street cleaning. Other apps include, a predictive application called "Pothole Predication" for road pothole analysis, and "Cycle Spot", an app to allow cyclists to avoid road hazards, including roadwork, 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.

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, authorities in the region can prioritize investments in maintaining and purchasing more bike racks for cyclists.

VII. CONCLUSIONS & FUTURE WORK

The use of a Cloud-based Hub as a basis for developing Smart City infrastructure 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. Our architectural approach, whereby the core IoT infrastructure is exposed as a data hub via a PaaS framework, addresses some of the core technical issues in building cloud-based IoT frameworks for smart cities. 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 [20]. We have demonstrated some of the advantages of this approach through our deployments both in the UK and Canada and through the large number of applications and services developed for the hubs.

There are two areas that we feel require further work, firstly the area of integrated or federated cloud and secondly the area of application development.

Federated Cloud: With respect federated cloud, our initial experiences have shown that we need to accept the reality of existing city infrastructure that is currently managed by City IT departments. Our work on application partitioning for hybrid cloud is aimed at addressing this constraint. A second constraint, which we have not yet addressed, is the fact that as new cloud infrastructure becomes more frequently used, we will see an increasing number of smart city cloud offerings. It will be necessary to ensure that our cloud based smart city hub is able to accommodate a number of peer PaaS services and that we can offer application developers a framework that allows them to exploit services and functionality resident in other clouds.

IoT application development tools: A second area of work that we feel is critical for widespread adoption of cloud based smart city hubs is the complex issue of IoT application development. In contrast to typical web based applications, IoT applications, by necessity, often require code to run at infrastructure end points, i.e. in city infrastructure. To accommodate this, we need an application framework that not only supports Hybrid public/private cloud but extends to end devices. To address this issue we have begun extending our application development tool, the WoTKit processor [3]. This tool provides a visual programming tool that allows IoT application developers to 'wire-up' data sources with processing logic and outputs (in a similar manner to LabVIEW or Yahoo pipes). We plan to extend our tool to allow code components to be migrated not only to private cloud, but also to end nodes. Our approach looks to combine some of the benefits of our own processor with Node-RED[1], an open source processor focused on IoT edge nodes from IBM [21].

Clearly, we still have many issues to resolve as we explore this approach to Smart Cities. However we feel a cloud-based hub approach to providing a Smart City PaaS is a valid one and that some of the issues we have uncovered, and the lessons we have learned, will help others in the Cloud and IoT communities as we collectively develop a truly global Internet of Things.

VIII. ACKNOWLEDGEMENTS

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 and we acknowledge the contribution of Nima Kaviani for his work on Hybrid Cloud computing. Partial funding for this work was provided by the TSB and NSERC.

REFERENCES

[1] Zanella, A; Bui, N.; Castellani, A; Vangelista, L.; Zorzi, M., "Internet of Things for Smart Cities," Internet of Things Journal, IEEE , vol.1, no.1, pp.22,32, Feb. 2014

[2] Kenji Tei; Gurgen, L., "ClouT : Cloud of things for empowering the citizen clout in smart cities," Internet of Things (WF-IoT), 2014 IEEE World Forum on , vol., no., pp.369,370, 6-8 March 2014

[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 (UbiComp 2013) 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] Soldatos, J.; Serrano, M.; Hauswirth, M., "Convergence of Utility Computing with the Internet-of-Things," Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2012 Sixth International Conference on , vol., no., pp.874,879, 4-6 July 2012

[8] Fei Li, Michael Voegler, Markus Claessens, and Schahram Dustdar. 2013. Efficient and Scalable IoT Service Delivery on Cloud. In Proceedings of the 2013 IEEE Sixth International Conference on Cloud Computing (CLOUD '13). IEEE , USA, 740-747.

[9] Internet of Things - Architecture — IOT-A: Internet of Things Architecture. http://www.iot-a.eu/public.

[10] J. Winn. Open data and the academy: An evaluation of CKAN for research data management. http://eprints.lincoln.ac.uk/9778/1/CKANEvaluation.pdf.

[11] Living PlanIT, Robert G. Eccles, Amy C. Edmondson, Susan Thyne, Tiona Zuzul Source: Harvard Business School, 29 pages. Publication Date: Feb 08, 2010. Prod. #: 410081-PDF-ENG

[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] Kaviani, N.; Wohlstadter, E.; Lea, R., "MANTICORE: A framework for partitioning software services for hybrid cloud," Cloud Computing Technology and Science (CloudCom), 2012 IEEE 4th International Conference on , vol., no., pp.333,340, 3-6 Dec. 2012.

[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] Kaviani, N, Wohlstadter, E & Lea, R 2013, 'Cross-tier application and data partitioning of web applications for hybrid cloud deployment'. in Middleware 2013: ACM/IFIP/USENIX 14th International Middleware Conference, Beijing, China, December 9-13, 2013

[19] HyperCat: an IoT interoperability specification. 2013. http://www.research.lancs.ac.uk/portal/services/downloadRegister/53462399/Interoperability_Action_Plan_1v1_spec_only.pdf.

[20] Blackstock, M. and Lea, R. IoT Interoperability: A Hub based. *Internet of Things (IoT), 2014 4th International Conference on the*, IEEE (2014)

[21] Blackstock, M. and Lea, R. Towards a distributed dataflow platform for the web of things. *Web of Things (WoT), 2014 5th International Workshop on the*, ACM (2014).

[1] www.nodered.org