Towards a Ubiquitous Service-Oriented Architecture for Urban Sensing

(Position Paper)

Carlos O. Rolim¹, Anubis G. Rossetto¹, Valderi R. Q. Leithardt¹, Guilherme A. Borges¹, Tatiana F. M. dos Santos², Adriano M. Souza², Cláudio F. R. Geyer¹

¹Institute of Informatics, Federal University of Rio Grande do Sul, Brazil
{carlos.oberdan, agmrossetto, valderi.quietinho, gaborges, geyer}@inf.ufrgs.br

²Postgraduate Program in Production Engineering, Federal University of Santa Maria, Brazil
taty.nanda@gmail.com, amsouza@ufsm.br

Abstract. Cities are involving towards intelligent dynamic infrastructures that serve citizens with new services to improve their quality of life and to fulfill the criteria of energy efficiency and sustainability. In this context, an important question is how to instrument citizens and cities to promote the sensing of data about different aspects. In this work, we present early stages of our research towards a ubiquitous service-oriented architecture for urban sensing called UrboSenti. Our approach differs from other sensing platforms since it provides a set of services to collect data from several sources and to assist the development of new sensing applications. Moreover, our model encompass all sensing activities, ranging from the collection of data to generation of reports about events in the city.

Keywords: Urban Sensing. Smart Cities. Service-oriented architecture. Ubiquitous.

1 Introduction

Cities are involving towards intelligent dynamic infrastructures that serve citizens with new services to improve their quality of life and to fulfill the criteria of energy efficiency and sustainability. The concept of Smart Cities is a response to these challenges. [1].

In this context, researchers are raising important initiatives and questions about how to promote citizen participation and community engagement to better interact with urban ecosystem. Among the initiatives, social urban sensing applications is a promising way to approximate computational area to community.

In this position paper, we present early stages of our research towards a Ubiquitous Service-Oriented Architecture for Urban Sensing called UrboSenti. Our approach differs from other sensing solutions since it provides a set of distributed services to collect data from several sources and to assist the development of new sensing applications.
Moreover, our model couples social sensing with the traditional sensing and encompass all sensing activities, ranging from the collection of data to a high-level view of events in the city.

In summary, the contributions made by this paper is a seminal approach for urban sensing using an innovative ubiquitous service-based architecture. In addition to being original, it signals the direction for further research in this area.

This paper is structured as follows: The next section describes our motivation scenario and present existent computational challenges; Section 3 describes our proposed architecture; Section 4 presents related works and; and, we conclude this work in Section 5.

2 Motivation

Our motivation for research is the problem-scenario shown in Fig. 1. This scenario presents a city with several data sources being used for sensing. Human-carried, fixed or vehicle-mounted sensors are applied for obtaining sensing maps of transit, air quality, noisy levels, temperature, CO2 concentration, etc. Moreover, data from social networks are used in conjunction with sensors data to provide a holistic view of the city.

This scenario presents an urban sensing ecosystem where the following computational challenges needs to be addressed: (i) heterogeneous devices, fixed and mobile, are used to collect data and to access the resultant processed information. (ii) context-aware and adaptation mechanisms are needed to support application adaptation; (iii)
large amount of data are continuously generated and collected along the city. It demands high processing power, with data preferably processed in real time; (iv) uncertain about collected data require data fusion and analytics techniques to generated relevant and correct information for decision-making and to export to other systems; (v) communication infrastructure is not always available. It demands alternative ways for inter-devices communication, like ad-hoc networks or delay tolerant networks; (vi) data security about collected data and privacy about citizen and device used by sensing; (vii) possibility to reuse of software components to develop deploy new sensing applications.

The computational challenges listed above are based on findings from researcher like [2–6]. They observed an increasingly demand for new ubiquitous and pervasive solutions to provide better services to citizens in a Smart City.

Thus, we argue that a ubiquitous services oriented architecture could provide a set of components and services that attend all challenges (i to vii) aforementioned. In the next section, we will present our vision of such architecture.

3 Proposed architecture

In this section we will present an architecture called UrboSenti. UrboSenti comes from the union of two Esperanto’s words: Urbo = urban, city and Senti = feel, sense. We settled on this name, because we wanted to represent the idea of a computational solution able to “feel the pulse” of the city.

![Diagram of UrboSenti architecture](image)
Figure 2 provides a high-level view of UrboSenti collecting data from city, reasoning about them and providing feedback to citizens and to other systems. We are using Service Oriented Architecture (SOA) approach to guide our design. In this way, we do not use well-defined tiers (or layers). Such traditional approach would constrain the value and flexibility of modules functionalities, resulting in dependencies across unrelated components. Instead, we designed our architecture in services. This is the SOA mode to expose components functionalities used by other components or modules, providing flexibility and reusability.

Our architecture is split in two main modules: Sensing module and Backend module.

The heart and brain of UrboSenti is Backend module. It runs in a data center infrastructure and, in brief, it is responsible to receive sensed data, process it and to give feedback to the citizens and other systems.

Its internal components and services are outlined in Figure 3 and its behavior is explained bellow.

- Services Repository: aggregates all services available. The services are grouped into categories according to its purpose. The services available are: (i) Data services: are used to manipulate overall data used by architecture. It also provide...
services to retrieve and to store data and functionalities to “clean” collected data from inconsistency and noisy; (ii) Social services: are services related to handle relevant data from social networks; (iii) Sensing services: are services to interface with different sensors and to aggregate collected data. It use supports open standards to facilitate the data exchange; (iv) Concern services: are used to deal with security, encryption and privacy of data and users; (v) Analytics services: are services for mining, classification and report. In addition to derive information, these services are able to correlate data from different sources and to predict events in the city; (vi) Proxy services: are services to interact with external systems like vehicular or sensors networks and to exchange data with third-party systems; (vii) Foundation services: enfold all basic services.

- Communication Bus & Services Orchestration: It is the bus used for inter-components communication. It also, does the coordination and arrangement of calls and invocations to multiple services to expose it like a single aggregated service.
- Web Services Mediation: is an intermediary between external entities, like sensing devices and third-party systems, and Services Repository (invoked by Communication Bus & Service Orchestration component).
- Service-Based Applications: all applications built aggregating available services from architecture. In conjunction with Web Services Mediation, it composes the more external component from architecture, called Presentation, which interacts with users and other systems.
- SOA Registry: it is an identity-management system for available services. It internal services keep track of metadata about services, which gives each service a unique identity. The information stored for each service establishes ownership for the service and specifies how the service behave at run time (lifecycle of service). It also provide artifacts to handle UDDI data store and services registry/lookup.
- Run Time Tools: encompass tools to monitoring and management of services. It contain artifacts generated at run time, such as logged messages, archived performance data, archived health, heartbeat of main components and also provides Key Performance Indicators (KPIs) for dashboards and reports of services performance. Policy Enforcement ensures that messages are properly formed and services are executed in proper behavior according to service-level agreements.

The other main module of UrboSenti is called Sensing module. It is responsible for social and traditional sensing. This module encompass activities of intentional and non-intentional sensing. It runs in mobile devices (e.g. mobile phones, embedded in vehicles, etc) and in fixed sensors scattered along the city. The internal components is depicted in Figure 4 and its behavior is explained bellow. We highlight that such components are composed by other subcomponents to provide internal services, but it has been omitted due to clarity and space limitations.

- Micro-kernel: The core of this module. Its main function is to provide basic services for more external components. Internally, the Micro-kernel is structured in: (i) Device: provides basic information about running device (name,
network address, interfaces, GPS, internal sensors, etc); (ii) Communications: provide methods to send and receive messages using available network infrastructure, such as IEEE 802.11b/g/n (structured and ad-hoc), GPRS/EDGE/3G and Ethernet as underlying for TCP/UDP communications. When network infrastructure is not available, this module provide support for Delay Tolerant Network approach via Bluetooth interface; (iii) Data: handle operations of store and retrieve data; (iv) Events: capture external events of interest (a position change due to user movement, alteration of interface status, etc). The detected events are available for use by other components.

- Localization: handle localization issues such geopositioning information, Points Of Interest (POI) and Location Based Services (LBS).
- Resources: set of components and services to monitor local resources and to discovery resources from other devices
- Concern: handle security, privacy and encryption issues.
- Context: group components and methods for context-awareness (i.e. context reasoning, context knowledge, context discovery)

**Fig. 4. Sensing module**
- User: handle the user’s preferences, social networks profiles and basic knowledge of user.

- Adaptation: components and services to perform adaptation in device’s behavior. It uses a set of policies and knowledge to describe how to adapt the application and monitors basic data about quality of service to infer about when adaptation is needed.

At top are Application layer representing running sensing applications and at bottom are the Operating System and the Hardware of device.

We already started the UrboSenti deployment. Currently, we are working mainly in Sensing module, more specifically in Micro-kernel. The basic functionalities of all internal modules are done, except Communications module. We are coding this module to use several network interfaces to send and receive data according to existent infrastructure. We already have support for TCP/UDP communications using wireless and wired infrastructure. However, we are stuck in Bluetooth as underlying for Delay Tolerant Networks paradigm. At this point, we are facing difficulties about paring new devices in suitable time and exchange information with existent devices without timing out. We hope to solve this issue soon to start to coding external components of Sensing module. After this, our focus will be the Backend module. This module have many stuffs for coding and it will require a long time to be done. In other words, we have a hard work until the two main modules may exchange data and the proposed platform could be used in a real scenario.

4 Related Works

In this section, we will present some initiatives related to our architecture.

In AnonySense [7, 8] is presented a framework for opportunistic and participatory sensing with strong emphasis on privacy. It adopts a polling model for task distribution to anonymize the location of mobile device to the infrastructure. Furthermore, tasks are written using a domain-specific language called AnonyTL, which utilizes predicates based on the context of the mobile device, such as location and whether the device is moving or not.

Medusa [9] is a programming framework that provides a program language and a distributed system runtime with focus in crowd-sensing. It seeks to provide a common platform to realize any kind of task supported by smartphone sensors. For this, Medusa utilizes a programming language based on XML, called Med-Script. It specifies the workflow of sensing tasks to be performed in smartphones (workers), with coordination of cloud services.

PRISM [10] presents a mobile phone sensing platform to facilitate the development of large-scale sensing applications. PRISM seek to address some issues of security, privacy, scalability and concerns about controlling the resource access in smartphones.

MobiSens [11] presents the design, implementation and evaluation of a flexible platform for mobile sensing. It can be used at individual scale (e.g. monitoring falls of elderly people) or at community and public scales (e.g. collecting data from participants
to infer collective behavior). Furthermore, MobiSens seeks to meet some common requirements in these types of applications, such as privacy, energy optimization, interaction between server and mobile client and activities recognition, segmentation and annotation.

Pogo [12] proposes a middleware infrastructure for mobile phone sensing to facilitate the construction and testing of large-scale sensing applications. Furthermore, Pogo allows the controlling of granularity of resources at user-level to protect the privacy of volunteers. It uses the XMPP protocol to disseminate data sets.

Micro-blogs [13] presents the design and implementation of an application, that allows smartphone-equipped users to generate and share multimedia content data called microblogs. Such data can be browsed or queried through Internet map services to get different information about stored data.

MetroSense [14] is an architecture for large-scale urban sense using an approach of opportunistic sensing networks. It takes the advantage of the interaction with mobile devices and provides coordination between mobile and static people-centric sensors.

By analysis of presented works we can note that all are engaged in urban sensing area. However, most of them only perform sensing using mobile phones (smartphones), not considering to perform sensing task with a “mix” of different source, such as mobile and fixed devices. The only work that addresses other devices is MetroSense, but it does not consider data from social networks like our approach.

AnonySense have a large overhead arising from the used pull model. It ensure more privacy for the device than push-model, but does not scale very well on large-scale applications. Nonetheless, the process of anonymize data used by AnonySense and PRISM is a desired characteristic for crowd-sensing applications that need to publish public data in anonymous way, however, it is not a desired characteristic for all type of systems such as for healthcare and personal sensing services. In our work, we not bind such issues intrinsically at core of platform. Instead, we provide a set of services for this purpose and developer could use it as desired.

Moreover, most of works support both opportunistic and participatory sensing like our approach (the exceptions are Medusa and Micro-blog that only addresses the participatory sensing, and in contrast, MetroSense that only aims opportunistic sensing). However, only AnonySene and MetroSense consider the use of Delay-Tolerant Networks (DTNs) as paradigm for situations where infrastructure is not available. In our platform we will provide support for both opportunistic and participatory sensing with usage of several communication paradigms, including DTN.

Furthermore, a key point that differs our approach from all others is the usage of Service-Oriented Architecture (SOA) to design the services and components. It provides more flexibility for integration with other existent solutions, reduce the complexity of the proposed platform and leverages the reusability of existent services to develop new solutions rather than trying to “re-invent the wheel”.

Finally, we highlight that all works have focus on gathering sensing data. However, none of them worries to provide a complete solution for collecting, analyze and give feedback to citizens. We are filling this gap with a platform that encompass all these steps.
5 Conclusions and future work

This paper presented early stages of our efforts to build a new ubiquitous services oriented architecture for urban sensing called UrboSenti. We presented our initial design of software modules, their internal components and provided services. We argue that our approach will fill the gap of related works and address the computational challenges of our initial problem-scenario. Moreover, it motivate us for further researches in multidisciplinary area of Smart Cities seeking to improve services and applications for urban sensing.

As future works, we can appoint the coding of proposed modules and seek for a way to simulate the implementation before put it in a real test-bed.

Acknowledgment.
This research was supported under UbiArch project - Ubiquitous Architecture for Context Management and Application Development at UFRGS.

References.


