**LIVE Singapore! - An urban platform for real-time data to program the city**

Kristian KLOECKL  
SENSEable City Lab  
Massachusetts Avenue 77  
02139 Cambridge (MA)  
USA  
Tel: +1-617-2537926  
E-mail: kloeckl@mit.edu

Giusy Di Lorenzo  
SENSEable City Lab  
Massachusetts Avenue 77  
02139 Cambridge (MA)  
USA  
Tel: +1-617-2537926  
E-mail: giusy@mit.edu

Oliver SENN  
SENSEable City Lab  
Singapore-MIT Alliance  
for Research and Technology  
S16-05-08, 3 Science Drive 2  
Singapore 117543  
Tel: 65-6516-8603  
E-mail: oliver.senn@smart.mit.edu

Carlo RATTI  
SENSEable City Lab  
Massachusetts Avenue 77  
02139 Cambridge (MA)  
USA  
Tel: +1-617-2537926  
E-mail: ratti@mit.edu

**Abstract:** The increasing pervasiveness of urban systems and networks utilizing digital technologies for their operation generates enormous amounts of digital traces capable of reflecting in real-time how people make use of space and infrastructures in the city. This is not only transforming how we study, design and manage cities but opens up new possibilities for tools that give people access to up-to-date information about urban dynamics, allowing them to take decisions that are more in sync with their environment.

LIVE Singapore! explores the development of an open platform for the collection, elaboration and distribution of a large and growing number of different kinds of real-time data that originate in a city. Inspired by recent data.gov initiatives, the platform is structured to become itself a tool for developer communities, allowing them to analyze data and write applications that create links between a city's different real-time data streams, offering new insights and services to citizens. Being a compact island based city-state metropolis, Singapore offers a unique context for this study.

This paper addresses the value of stream data for city planning and management as well as modalities to give citizens meaningful access to large amounts of data capable of informing their decisions. We describe the technology context within which this project is framed, illustrate the requirements and the architecture of the open real-time data platform to serve as a base for programming the city, and finally we present and discuss the first platform prototype (using real-world data from operators of cellphone network, taxi fleet, public transport, sea port, airport and others).

**Keywords:** real-time city, urban dynamics, control system, urban management systems, open data platform.
1. INTRODUCTION

Over the past years, urban space has been pervaded by networks and systems that generate digital bits. Bits, that are closely related to human activity, both informing human actions as well as reflecting its impacts - they are somewhat traces of human activity (accessing the Internet on a laptop from a café locates the user through the WiFi antenna, an online search or order of a product online establishes countless histories describing your preferences and interests, using a cell phone registers at the nearest cellphone network antenna, using a public transport smart card generates records on the origin and destination patterns of your trip,...); traces that remain in time and that can be explored from a distance and with the help of sophisticated digital tools. This generation of bits happens incidentally, being generated and memorized during the daily acts of a city’s life and as such reveals the manifold and complex facets of the city and its inhabitants in detail and in real-time.

The SENSEable City Lab has created partnerships with entities that manage large urban systems who’s normal operation generates such data streams, and we have developed innovative ways of interpreting these and inferring urban activity as it happens in real-time: The Real-Time Rome project consisted in the analysis of data streams originating from Italy’s largest cellphone network, GPS locations of the public transport bus fleet as well as the taxi fleet. The project’s objective was the exploration of how to use real-time data for the better planning and management of cities, with the main audience being planners and architects. A year later, Real-time Rome was succeeded by the WikiCity project in which our work with real-time data was taken further, exploring a way in which combinations of data streams could be turned into effective tools for citizens to support the decision making process when following their goals and activities in the city.

Most digitally managed urban systems generate data that is not directly accessible by other urban systems or the public audience. As a result, digital bits describing human activity in urban space related to these systems is accurately described by the generated data but remains locked in within each of these specific domains. The LIVE Singapore! project aims at developing an open platform for the collection, combination and distribution of large numbers of a city’s real-time data streams, allowing developer communities to join in creating applications that turn these data streams into meaningful and useful tools for people to make use of their cities.

The context in which this project is being developed is the city-state of Singapore. While this urban environment does represent several characteristics that we consider particularly favorable to the development of an urban data platform (a country that is all one city and that is clearly defined by being an island; highly developed infrastructures, tech-savvy population used to rapid change in urban environment,...), the platform in development is generic enough to allow several conclusions to be drawn generically and independently from the specific city context.

2. CONTEXT AND RELATED WORKS

2.1 Real-time Rome and WikiCity

The LIVE Singapore! project is a continuation of research carried out within several projects at the SENSEable City Lab which dealt with real-time data from urban systems and the exploration of new opportunities for city planning and
management. The Real-time Rome project in 2006 pioneered the use of real-time data from cellphone networks to infer aspects such as overall population distribution; predominant direction and speed of movement of people using different modalities of transportation (walking, car,...); and movement of foreign visitors in the city. It combined this data with complementary live feeds from the public transport bus system as well as the taxi fleet system to analyze urban dynamics reflected in these data as well as produce a number of real-time data visualizations, making data accessible in a meaningful way to an audience of planners and architects, exhibited at the 2006 Venice Biennale.

The subsequent WikiCity expanded upon the initial project, formulating the theoretical framework of a platform idea for urban real-time data and culminated in a first step in an urban installation which added live data types related to events as well as location based news feeds in the occasion of a large public event in Rome: the “notte bianca”, a long night of museums during which an estimated 2 million people participated in approximately 400 events that carried out all through the night. The installation consisted in a 10x5 meter large public projection in a central city square with the real-time data platform driving a live visualization of the data streams. The real-time data visualization was visible by Romans throughout the nightly event, representing an effective tool for deciding how to explore the city and its manifold activities.

2.2 Real-time information

There is an undeniable charm in working with real-time data in an urban context as it reflects “in this moment” specific dynamics directly or indirectly caused by human activity. Having access to data in the very moment it is being generated allows for its consideration in the process of decision making, leading to feedback loops as described below, capable of increasing the overall efficiency of a system such as a city.

Key to the consideration of real-time data in the context of the city is adopting a definition of the term which is somewhat different from the one used commonly. There, “real-time” relates to a system in which data is made available and processed within a “very small” time interval, a sensor that returns a measurement which can be processed and used to act upon the system in the fraction of a second for example. The “very small” here is a vague indication and while commonly agreed upon when contained within fractions of a second or seconds, doubts emerge as to whether call an interval from minutes or tens of minutes still real-time.

This poses difficulties in the context of an urban data platform such as LIVE Singapore which is confronted with a multitude of data systems which were not initially constructed to share system data externally and due to the structure of the system often have latencies of magnitudes larger than fractions of a second. Existing confusion becomes apparent when observing the usage of variants such as “near real-time”.

A more useful interpretation of “real-time” is the one which refers to “the actual time during which a process or event occurs” (Oxford, 2007). In this reasoning, a real-time process implies the existence of a deadline before which a given information can be considered as useful to a higher level system while that same information would be not useful or even destructive would it become available
after that deadline. While the deadline refers to a process, identifying the usefulness of respecting the deadline implies the existence of a higher level mission. Since therefore it is this mission which defines the parameters of the deadline this definition of real-time does not imply a stringent necessity of any a priori speed of data transmission and processing. Instead it draws attention rather to the identification of reasonable deadlines for specific missions that ought to be accomplished. This again combines favorably with applications within the context of the city and the use of data by people who move and act in a city to follow their various goals and objectives.

2.3 Open data initiatives and platforms

Data.gov

In May 2009 the US government launched the first of what has since been referred to as data.gov initiatives in which it made non sensitive historic data sets publicly available in order to increase the “ability of the public to easily find, download, and use datasets that are generated and held by the Federal Government”\(^1\). In this dynamic of giving data collected from the people back to the people, some of the goals are to spur economic, scientific, and educational innovation, further active citizen participation, as well as driving down government costs by inviting a larger public community to actively participate in exploring how new value can be created from existing sets of data and their novel combinations. Since its start, the this initiative has been replicated in various other countries and cities (e.g., UK, London, San Francisco) and multiple applications have been created based upon the data sets made available through these platforms.

Pachube

Pachube\(^2\) is a platform for sharing sensor data over the web (“data brokerage”). Pachube was developed as an infrastructure for the Internet of Things that allows individuals and organizations/companies to make their sensor data available online. The platform uses a RESTful, HTTP-based API for the interaction between the sensors, the platform and the clients which makes it easy to publish and consume sensor data.

Two different ways are supported for uploading data. The sensor can directly “push” data by issuing a HTTP PUT request along with the data. Alternatively Pachube can also poll data from the sensor in 15 minute intervals or when a user is requesting data (if the sensor is supporting this mode). Pachube is primarily focussing on supporting sensors with low sampling rates, since both mechanisms to upload data are based on HTTP requests and therefore are not suited for high rate data streams.

To access data points of sensors, clients have to send a HTTP GET request to Pachube. The response then contains the current value of the sensor in a format specified by the client (CSV, XML and JSON are currently supported). Pachube also supports the notion of “triggers” that notify clients with a HTTP POST request once data for a particular sensor is available and fulfills certain filter criteria.

\(^1\) http://www.data.gov/about, Accessed 01/03/2011

\(^2\) Pachube, Online at http://www.pachube.com/, Accessed 01/03/2011
Both mechanisms are only suitable if sensors don’t produce data with high
frequency. The overhead associated with the polling or with handling the trigger
HTTP POST request is certainly too big in order to handle hundreds of messages
per second.

Pachube has the advantage that it makes the data easily available over the Web
and that the protocol used (HTTP) is widely used. The disadvantage is though
that HTTP is not a messaging protocol and not suited for transporting data
streams since it doesn’t support any notion of streaming data. Furthermore, the
polling and simple notification mechanisms are not suited for high-rate data
streams.

Apart from acting as a data exchange platform between sensors and clients,
Pachube does not provide any data processing functionality such as aggregation,
advanced filtering or combination of data streams.

2.4 Semantic web and linked data

Ubiquitous sensors and sensor networks are increasingly providing data sources
of various contents, formats and qualities. Integrating diverse data sources allows
developing applications that would not be possible by using a single sensor
network.

When integrating data from heterogeneous sources, syntactic, schematic and
semantics diversities of the data schemas are challenging problems. Syntactic
heterogeneity refers to differences among paradigms used to format the data
such as Relational DB, XML or RDF documents. Schematic integration refers to
different aggregation or generalization hierarchy defined for the same real world
facts. Finally, semantic integration regards disagreement on the meaning,
interpretation or intended use of the data.

To conclude, the data should be syntactically and semantically described in order
to allow a machine to automatically search, combine and so construct
applications on the fly. In this context, a crucial role is played by ontologies, which
specify machine readable vocabularies that can be used to formally describe
specific domains (e.g. the city), their entities (for instance people, transportation
means, events) and the relations between entities that are relevant for the
domain under analysis (Bizer 2009).

A lot of effort has been done in the Semantic Web community to provide
standardized technologies, languages, and tools to define ontologies in a specific
domain and to semantically browse and search data entities and to link different
data ontologies (Hausenblas 2009). In the Semantic Web Community, the RDF
Vocabulary Definition Language (RDFS) and the Web Ontology Language (OWL)
(McGuinness 2004) provide a basis for creating vocabularies, and link entities
belonging to the same or different domains.

In the specific context of sensor networks, the Open Geospatial Consortium
(OGL) and Sensor Web Enablement (SWE) have invested efforts in the
standardization of the Semantics Sensor Web (Henson 2008), to annotating
sensor data with spatial, temporal, and thematic semantic metadata. In particular,
sensor networks and sensor data are described using the Sensor Web
Enablement (SWE) standard. The meaning for sensor observations is given by
adding semantic annotations to existing standard sensor languages of the
SWE, in order to increase interoperability and provide contextual information for situation awareness.

2.5 Mash-ups and combining data

In a Web context, Mashup applications use two or more sources of data combined to create certain functionality or to simply derive new data sets. Mashup tools help in acquiring, combining, filtering and aggregating the data. Furthermore, often these tools allow users to easily build (i.e. without programming skills) mashups and visualize or share the outcome of the mashup process.

One example of an online mashup tool is Yahoo Pipes\(^3\). In analogy to Unix pipes, the idea is to apply several simple operators after each other to one or more data sources in order to extract the data that is needed. The tool supports several mechanisms to fetch data such as reading from CSV files, accessing RSS and Atom feeds or getting images from Flickr (an image hosting website). Yahoo Pipes does however not offer mechanisms to fetch data from data streams, it only polls the data sources in certain time intervals.

Once the data is loaded, different operators can be used to modify and combine data from different sources. There are separate operators that filter, sort and aggregate data. Other operators for example allow to replicate data (so that different operations can be performed with it afterwards) or to reformat data from different sources. Another feature allows to geocode the data feeds and then display them on an interactive map.

To build the mashups, user employ a visual editor rather than a query or scripting language. The editor lets users drag operators onto a canvas and then specifying the data flow by linking them together. After the mashup was finished, it can be viewed and reused by other users.

The functionality of Yahoo Pipes allows to quickly combine various static data sets. The goal in LIVE Singapore is to have similar functionality but for real-time data streams. This increases the complexity of the operators significantly since data streams are infinite streams of messages but most operators (say the sort operator) rely on a limited set of messages to work. One way to deal with this problem is to introduce new operators that specify rules to select a certain group of messages from a data stream (called a window). The issues with those operators is to make them easy to understand and use.

2.6 Sensing

An increasing amount of digital devices is distributed in today’s cities that records, stores and transfers data triggered by ambient conditions and use dynamics - sensors of different kinds which are distributed among people as mobile devices, vehicles as well as building structures and products. With their wired or wireless telecommunication capabilities these devices result in interconnected sensor networks, able to reflect a large variety of a city’s dynamics linked to aspects such as transportation, environmental parameters, communication and social patterns, structural conditions of a city’s built infrastructure just to name a few (see

---

2.7 User generated data

The development of online tools that allow for easily accessible and widely participated information sharing, commonly associated with the term “web 2.0”, has lead to an enormous number of data generated consciously by large numbers of web users. People share personal information through social network sites, short comments on twitter, photographs on sites such as flickr as well as music, art and other user kinds of data. Common and distinctive of this data is that users are active in both the generation as well as distribution of these kinds as data and often publish them associated with their real identity or at least a specific user account. Many of the web tools enabling such user participated authoring provide API's which subsequently allow for the further elaboration and processing of data contained within their content domains.

2.8 Urban computing

Over the past decade, cities have been pervaded by digital devices such as sensors and mobile electronic devices often connected via telecommunication technologies to form networks. Urban computing is an emerging field of study addressing the impact which these distributed and embedded technologies have and can have on the way cities are planned, managed and ultimately lived in, “exploring how people respond to, adopt, and understand these technical conditions, and appropriate them for their own uses” (Greenfield and Shepard, 2007). Beyond being just about the city, as envisioned in (Bassoli et al., 2007), urban computing increasingly binds in the city and its inhabitants as active elements of a common platform for the creation of novel social interactions more than just the resolution of urban dynamics considered as problematic. For this to happen, (Trevor and Hilbert, 2007) suggest an emphasize on the seamless integration of interconnected wireless devices into everyday life by way of appropriate networked service access and the sharing of information through multiple types of interfaces.

3. LIVE SINGAPORE

The LIVE Singapore! project leverages all of the above described dynamics. It is a flexible and scalable open platform structure enabling the collection, combination as well as the distribution of large and growing numbers of real-time data streams originating from a city. Inspired by the data.gov initiatives, LIVE Singapore! is not a limited set of specific applications as such but an enabling platform on top of which developer communities internal as well as external to the project group can subsequently build applications that employ combinations of the data feeds available through the platform. In this way, instead of limiting the application development to a restrained group, we trigger a dynamic of crowd sourcing the actual ideas creation and development of applications that become possible for the city, firmly believing in the creative potential of cities and their inhabitants. As in the case of the data.gov initiatives and their historic datasets, also in this project’s context, letting citizens, city planners, companies, authorities actively work with live data streams has the objective of: fostering crowd sourcing, multiplying the creative potential, and ultimately setting the base for an emerging economy based on new ways to extract value from data. For this reason, the
LIVE Singapore! platform aims at becoming somewhat an ecosystem with toolboxes that enable developers to write applications that turn urban real-time data into useful information for citizens. While working with real-time data from any one single technology network has shown interesting results in recent years, it is becoming increasingly clear how the combination of very different types of data can bring about even richer insights into how cities work and how people make use of space. The data that we are considering for this project can be divided into three groups: data as a “by-product” from existing networks, data collected with tags or sensors, and data actively shared by people.

Dealing actively with a multitude of data that originate from very differently structured networks brings one main attention to the conception of connector elements as outlined in the next section. While we are actively developing code to include data streams in this first prototype phase, we will be moving towards providing indications as to how external data providers can provide data to the platform in a more standardized way. It is clear that a platform such as LIVE Singapore! needs to tackle and resolve the seemingly conflicting requirements of providing connectors that require as little as possible data processing and conversion on the provider side (in order not to discourage participation), yet receive data streams that are as coherently structured and described as possible for a maximum of interoperability and the possibility of combining different data streams.

Another of the platform’s declared requirements that orients development is to provide easy access and easy programmability in order to capture an as large as possible active developer audience, interested and able to program applications that valorize the available data streams.

Currently, much of the value that is being attributed to many online platform projects is the data that these platforms accumulate and which give them deep insights into their user’s activities and preferences. The situation for LIVE Singapore! is of a fundamentally different nature: This project deals with stream data and as such does not store any data within the platform - it is always the data provider that has control over what data is made available. We believe it would be difficult for any one operator to propose itself today to centrally host a wide variety of a city’s data streams for difficulties related to data management performance but more so for questions of data ownership, privacy and security. The platform is therefore structured to make data streams available in a coherent form but not to store this data. Historic data can be made available in much the same way as the live streams, being hosted on the providers’ side and made available through connectors via the platform.

LIVE Singapore! in this way sees the value of data not in it’s centralized accumulation but focuses on keeping track of the connections made between real-time data streams by developers and users creating applications. While the semantics of data streams is a difficult task to resolve, even more so is the semantics of data connections: what is the value and meaning of the combination of any group of data streams? Our approach lets the users make and describe these connections while saving this information within the platform in order to gradually grow an internal knowledge about the semantics of stream connections to inform the platform’s growth.

4. PLATFORM ARCHITECTURE

This section covers various technical aspects of the platform. From a high-level view the
platform functionality falls into two categories. The first and most basic part of the platform is the messaging part, i.e. transporting data from a provider to a set of consumers. The platform is designed to handle data in the form of (real-time) data streams. A data stream consists of an infinite sequence of messages and each message has a number of data fields (usually key-value pairs).

We will describe the requirements for the messaging in the next paragraph. The second part of the functionality consists of data consumers being able to manipulate and process the data. The platform supports various kinds of data processing further described in paragraph [Data Processing Functionality] below.

4.1 Data transport requirements

LIVE Singapore has the goal of making many data streams available to the public. In the first phase of the project already more than 10 government agencies and companies provide their data. Apart from that, the platform shall also support data streams that are coming from individual devices (like smart phones) in the future. This implies that our platform needs to be able to handle a large number of data streams. At the moment we assume hundreds of data streams, with individuals contributing this number could go into thousands.

The data streams from data providers vary widely in terms of data rates. A large telecom company partnering with us provides us with a data stream with a rate of around 300 messages per second. At the same time, the platform also has to be able to handle data streams with only a few messages per second. This is especially true if we consider individuals, contributing data from their smart phones. Considering a growing number of providers in the future means that the platform is required to ultimately handle tens of thousands of messages per second in overall input load. On the output side, once the data of the platform is made publicly available, many people will be using the data (for now we assume several thousands) resulting in each data stream to be distributed to a potentially large number of clients.

Taking into account the overall input load and the numbers of potential clients, and doing a quick approximation, we easily end up with up to one million messages per second. Since the initial load of the system will however be much smaller, it must be designed in a way to run on just a few servers initially but scale up near linearly with every server we add (we cover the approach to scalability in a separate section below).

4.2 Architecture

The overall architecture can best be described by following messages of a data streams through the platform (see Figure [1] from left to right).
Each data provider usually has its own format for the data. In order to allow any data processing, the data needs to be converted into a common data format (see section [Data Format]). The platform provides connectors to the data providers that allow to convert data into the platform data format and then forward the data to the platform (i.e. connecting to the platform messaging system).

The messaging system is then responsible for receiving the data from connectors and routing it to the client applications that need it. A client application can subscribe to individual data streams. After that every message of that data stream is then pushed to the client.

The processing modules implement the data manipulation and processing features of the platform (further described in paragraph [Data Processing Functionality]). They take one or several data streams as input (via the messaging system), do the processing/manipulation and output a modified data stream (to the messaging system). Client applications can then subscribe to the modified data stream in the same manner as they would for an unmodified data stream coming from a data provider (i.e. through the messaging system).

4.3 Scalability and flexibility aspects

As mentioned in the requirements paragraph above, we expect that the platform has to deal with a considerable amount of messages (this causes CPU, memory and network load). Furthermore, the processing modules impose additional load. In order to handle this load the work has to be distributed to many machines.

After the platform was launched, the load will be small, so we designed our system to be able to run on a few servers initially. At the same time, the system is designed to scale to hundreds of machines.

Our approach to achieve this scalability is that each part of the system (we call the parts components) can be run on separate machines if necessary. Each component can also be replicated on many machines if needed. Components are for example the individual processing modules and the parts of the messaging system.
We illustrate this by describing a common scenario: It is very likely that some particular data streams will be very popular with client applications. Many client applications will subscribe to this data streams and a single machine may not be able to distribute the messages to all the clients anymore. A component called Forwarder (see figure [2] below) is then used to replicate the data stream to many (in the figure, two) machines. This distributes the load among two machines and enables us to support many client applications. The cost of this solution is an increased latency for messages since the messages have to be transported over one link more.

![Figure 2 - Forwarders replicating data stream to many machines.](image)

The concept works the same for other components of the system such as the processing modules. This very flexible way of running components on different machines and therefore distributing load works because all the components have a similar interface: They consume data from a number of data streams and output a data stream.

So far an administrator has to configure which components to run on which machines but we plan to develop an automatic mechanism that starts and stops components depending on the current overall load of the platform and the load on individual machines.

**Client applications executing components locally**

Client applications subscribing to data intensive data streams and/or using complex filter modules cause a lot of processing load on the platform. An advanced concept for limiting the load is to offload some of the computation to the client applications themselves. For example the client application requesting to filter a data stream could run the filter module locally, thereby freeing platform resources.

The platform could even be outfitted with a cost model that balances computation costs and costs for transporting messages to clients. If, for example, two client applications execute the same filter over a data stream it might be beneficial to run the filter on the platform to reduce the number of messages that need to be forwarded to the two client...
applications. The cost model could be made dynamic so that it considers the current platform load in terms of computation and networking load.

4.4 Data processing functionality

Besides transporting data from providers to consumers, LIVE Singapore! allows clients to easily process, manipulate and visualize the data of data streams. Several modules were already implemented in a prototype:

A predicate filter module: Allows to filter messages of data streams. The user specifies an arbitrary complex predicate that is then evaluated for each message of a data stream (and the message is then either discarded or forwarded). As an example consider a data stream with a data field for humidity and temperature. The predicate could be temperature>=10 meaning that the filter would discard all messages where the temperature is below 10 degrees.

A point-in-polygon filter module: Often data streams contain geo-tagged information and many client applications may only be interested in data from a certain area. This module allows users to specify a polygon (using lat/long coordinates) and the filter then only forwards messages that have a location within that polygon.

A truncating module: This module allows users to remove unwanted data fields from data streams.

Our system can easily be extended with new modules. We plan to implement several modules in the future:

A module to support “windowing” (well known from stream processing engines). Windowing is needed for example when a user wants to calculate the average value of a data field for some messages. Since a data stream is an infinite sequence of messages, the user needs to be able to select a meaningful subset of the messages to calculate the average on.

A module that converts data from our internal data format to different well known formats (such as XML, CSV, JSON) requested by client applications.

One very specific module that we would like to implement is a map matching module which would be of great use for all transport related data.

Since all the modules are implemented as components, they can all be linked together, allowing users to build processing chains out of simple operators. This idea is again very similar to stream processing engines. The downside with stream processing engines is that the operators (especially when it comes to windowing) are very complex and hard to understand and use. We plan to limit the number of modules and their functionality to make them easy to understand and use.

A second advantage of modules having data streams as outputs is that all client applications can potentially subscribe to, say, a filtered data stream created by another application. This allows users to build up a repository of useful modified data streams together. Also, the platform will keep track of all the derived data streams so that we can add a mechanism that advertises popular derived data streams to users (since those are most likely meaningful ones).
4.5 Prototype implementation and evaluation

Over the past six months, a first prototype of the platform was implemented. The messaging system was implemented using a low-level (TCP) messaging library called ZeroMQ\(^4\). The advantage of that library is that it allows us to implement a fast messaging fabric both internally as well as between the platform and data providers / client applications.

The messaging system underwent some initial tests which showed that even a normal desktop machine is capable of distributing a data stream with a data rate of 1000 messages per second to 300 clients in parallel with an average latency of around 1.3 seconds. This result shows that the overhead imposed by the messaging system is relatively low. However, a lot more tests will be necessary, especially to see whether scaling the messaging system works in a nearly linear manner (meaning twice the machines can handle almost twice the load).

Apart from the messaging part, two processing modules (predicate and point-in-polygon filters) were implemented. Also, a management console is provided allowing to manage the network of components that are used. Another feature of the prototype is that the new HTML 5 WebSocket protocol\(^5\) is supported. This means that messages of data streams can be pushed to a Web browser (or any other HTTP/HTML client). This is an important feature because, unless the low-level TCP, it works for clients behind a firewall/NAT.

In the future, we will do a more extensive evaluation of the platform under real-world conditions and implement additional processing modules (as mentioned above). Furthermore, we plan to provide users with a toolkit that makes it easy to subscribe to data from the platform, to handle and visualize it.

5. DATA FORMAT AND SEMANTICS

The proposed approach for describing the data stream is the following:

The data model for describing city entities and the city data is defined using the standardized SensorML Observation and Measurement Language.

The semantic description of the city entities and all the sensory data sources is specified using the ontologies defined in the OWL. A Domain Ontology connects all sensory data ontologies.

The semantic description of the data is decoupled from the data per se, which can instead be represented in JSON, to reduce file size and parsing overhead due to the unnecessary descriptive tags. Figure 3 shows the data model generation process.

---

\(^4\) ZeroMQ, Online at http://www.zeromq.org/, Accessed 03/03/2011

\(^5\) WebSocket API, Online at http://dev.w3.org/html5/websockets/, Accessed 03/03/2011
Figure 3 - O&M data model generation process

Users can use the platform to search, browse and integrate data streams based on their semantic descriptions, and create applications. The system overview is illustrated in Figure 4. Through the platform, applications can also be published and shared among users.

Figure 4 - Central user generated data flow process

6. CONCLUSION

The LIVE Singapore! project is a five year research initiative started in mid 2010 and has reached the following state: A platform prototype has been implemented as outlined above allowing at present the development of applications on top of the platform which is at this stage being carried out internally.

In the month of April, the LIVE Singapore! exhibition was staged at the Singapore Art Museum in which we showcased six applications developed within the LIVE Singapore! framework that combine multiple real-time streams, making them accessible to the public by means of visualizations (two of which allow user interaction) of the data that give insights into the city’s dynamics as they happen and demonstrate the usefulness of real-time data in concrete urban contexts. These applications are realized with data streams coming from an initial set of data sharing partners that include the city state’s
largest cellphone operator, the sea as well as airport, the largest taxi fleet operator managing 16,000 vehicles, the country’s environmental agency, and Singapore’s major electricity provider. For example, in “Raining Taxis” the rainfall data from the environmental agency is combined with the location of free taxis, enabling taxi drivers to better cater for upcoming high trip demand areas due to frequent tropical rainfall as well as clients to quicker find available nearby taxis. In “Isochronic Singapore” the city’s geographic shape distorts around a user selected trip origin location according to the road traffic situation - parts of the city that take longer to reach by car move further away while other parts that can be reached in shorter time move closer to the origin location. Using traffic data from 16,000 taxis the map of Singapore gets distorted dynamically during the time of the day as well as according to the trip origin location chosen. A user can grasp in one glimpse the state of road traffic from his current location while at the same time, observing this dynamic map over several hours one can observe how strongly road traffic fluctuates due to rush-hour impacts during peak hours - in this way, this map gives a visual tool for the evaluation of how well cities cope with peak hour traffic load (if coping well, the city shape would distort little, while if coping worse, the city shape would expand due to a prolongation of trip durations).

On the basis of these developments several conclusions can already be drawn which inform the next steps of the project’s development:

It emerged in the collaboration with all partners sharing data with the platform that a considerable effort was needed on their side to prepare the data to be shared. The effort is manifold and some of the core points relate to: anonymization, extrapolation of relevant and non-sensitive fields as well as structuring the data in a coherent format. This confirms the common impression of the the existence of “data silos” which are difficult to integrate - these systems have not been set up with the purpose of sharing operational data with an external entity such as an open urban data platform. The

Figure 5 - The six visualization applications shown at the LIVE Singapore exhibition at the Singapore Art Museum between April 7 - May 1 2011. From left to right and top to bottom: Isochronic Singapore, Hub of the World, Urban Heat Islands, Real-Time Talk, Formula One City, Raining Taxis.
process of interacting with our initial partners will substantially inform the formulation of standard procedures, policies and data structures for the next phase of the platform, keeping in mind the above outlined contrasting objectives of high level of data coherence as well as reduced requirement of data preparation on the side of the providers. On the positive side however is the high level of commitment on the side of the data sharing partners and their motivation to address these issues, indicating an awareness of the overall benefit of an urban data platform such as LIVE Singapore! - in some cases, the project has been taken by partners as a stimulus to resolve internally existing data incoherences and incompatibilities.

It is commonly known in the community dealing with projects that involve data sharing partnerships that one of the most difficult parts is to actually formulate such partnerships. Our experience in Singapore has been very encouraging for the fact of having found highly motivated and cooperative partners but has also enhanced our understanding of the necessity to formulate a clear incentive framework for an urban real-time data platform. Simply put: what is the incentive for any one data provider to provide its data to such a platform? And how is it possible to ignite the process when the full potential is unleashed only when there are numerous such data contributors? Again, the experience in Singapore suggests that there is in fact today an increased awareness among institutions and companies that the combination of their data with other data sources holds compelling potential and sharing their data in order to have access to data shared by others is starting to be perceived as a viable approach. Since urban actors such as the current data sharing partners of LIVE Singapore! have in some parts expressed interest to share some data publicly while explore other kinds of their data on a “reserved” part of the platform, this may suggest considering such a possibility as part of the platform’s structure. Another interesting possibility for further development of the platform is to allow people to contribute data through mobile devices such as smart phones or personal sensor networks (e.g. sensor in a home). This would further help to close the feedback loop mentioned above.

Singapore is a lively business environment and we have seen an eagerness to take a project such as LIVE Singapore quickly towards a city wide implementation level and explore new business opportunities that this would open up. While such an implementation will require still further research and development steps, it does anticipate the mandatory question of a viable and sustainable business model for a platform such as LIVE Singapore!. Will data be made available free of charge by the providers? Shall resulting applications be without charge too? If developers charge for applications that make use of several real-time feeds, will the data providers participate in the earnings? And finally what kind of entity (private, public,...) would best be indicated to operate the platform which at present is managed within our academic research group?

Summarizing the next steps of LIVE Singapore! in one word, these will focus predominantly on “accessibility”. After the initial phase of laying the basis and developing a working prototype of the platform architecture, activating real-world real-time feeds together with our partners as well as programming first applications on top of this framework, LIVE Singapore!’s explicit emphasize is on creating an easy access for external data providers as well as an easy access for developers and users to the city’s real-time streams and allowing for the programmability of these in order to allow the joining in of an extended developer and design community in exploring the future of programming the city.
REFERENCES

Balan Sethu Raman, Mohamed H. Ali: Spatial data streaming or streaming spatial data: just stream it the way you like. COM.Geo 2010.


