Privacy-preserving Ubiquitous Social Mining via Modular and Compositional Virtual Sensors

Evangelos Pournaras, Izabela Moise, Dirk Helbing
Chair of Sociology, in particular Modeling and Simulation
ETH Zurich
Zurich, Switzerland
Email: {epournaras,imoise,dhelbing}@ethz.ch

Abstract*—The introduction of ubiquitous systems, wearable computing and 'Internet of Things' technologies in our digital society results in a large-scale data generation. Environmental, home, and mobile sensors are only a few examples of the significant capabilities to collect massive data in real-time from a plethora of heterogeneous social environments. These capabilities provide us with a unique opportunity to understand and tackle complex problems with new novel approaches based on reasoning about data. However, existing 'Big Data' approaches often turn this opportunity into a threat of citizens' privacy and open participation by surveilling, profiling and discriminating people via closed proprietary data mining services. This paper illustrates how to design and build an open participatory platform for privacy-preserving social mining: the Planetary Nervous System. Building such a complex platform in which data sharing and collection is self-determined by the user and is performed in a decentralized fashion within different ubiquitous environments is a challenge. This paper tackles this challenge by introducing a modular and compositional design approach based on a model of virtual sensors. Virtual sensors provide a holistic approach to build the core functionality of the Planetary Nervous System but also social mining applications that extend the core functionality. The holistic modeling approach with virtual sensors has the potential to simplify the engagement of citizens in different innovative crowd-sourcing activities and increase its adoption by building communities. Performance evaluations of virtual sensors in the Planetary Nervous System confirm the feasibility of the model to build real-time ubiquitous social mining services.

I. INTRODUCTION

Ubiquitous systems bring new opportunities for mining massive amount of data in real-time from physical and digital environments using mobile, home or environmental sensors. While more information may improve the understanding of various social phenomena or societal problems such as disease spreading [1], economic recessions [2], energy consumption [3], etc., the question that arises is who manages this information and for the benefit of whom[1]. Existing 'Big Data' systems are often designed as closed, proprietary and privacy-intrusive that surveille, profile and discriminate people [4]. In contrast to this current practice, this paper envisions a open, decentralized, privacy-preserving and participatory system designed to provide ubiquitous social mining services engineered as public good: the Planetary Nervous System [5]. How to design such a complex system is the research question addressed in this paper.

Social mining is defined in this paper as the process of discovering information from data sensed in one or more social environments so that a social phenomenon is understood or a societal problem is tackled. Ubiquitous systems, wearable computing and 'Internet of Things' technologies make our social environments data intensive. For example, understanding and regulating carbon emissions require data from a large number of heterogeneous sensors that provide information about human activity such as mobility, energy usage, etc. A ubiquitous system that can provide an abstraction of all these diverse sensors is capable of involving a broader range of data sources in the social mining process. Moreover, designing extensible and reusable social mining processes via compositional data flow of sensors simplifies application development [6], [7], [8], [9]. This design principle has the potential to simplify crowd-sourcing activities and increase the engagement of building communities.

This paper introduces a model of virtual sensors that provides a modular and compositional design approach to build ubiquitous social mining services. In contrast to a physical sensor based on hardware, a virtual sensor can be realized by software components that aggregate a set of input data streams from an environment and generate an output stream. The input streams may originate from physical and virtual sensors. The output stream becomes the input stream of other virtual sensors in their environments, resulting in a recursive composition of system functionality. A filter in each virtual sensor manages privacy by controlling the availability of the output streams to other environments.

This paper shows how the model of virtual sensors can be used to engineer the core components and applications of the Planetary Nervous System. More specifically, sensor data from mobile platforms are collected and managed locally by users. Two privacy levels are introduced to allow users to self-determine which data are logged locally in their phones and which are shared with others as a contribution to social mining services performed system-wide. Both privacy levels are designed with the model of virtual sensors. The performance of virtual sensors is experimentally evaluated within the Planetary Nervous System by emulating two users that collect data from several virtual sensors with a high frequency. Results show that real-time ubiquitous social mining via data-intensive virtual sensors is feasible.

This paper is outlined as follows: Section II introduces the model of virtual sensors. Section III illustrates how the model of virtual sensors can be used to realize the Planetary Nervous

[1] Some ethical issues are discussed in the FuturICT blog: http://futurict.blogspot.ch (last accessed: October 2014)

System. Section IV discusses how the modular and compositional design approach of the virtual sensors model can be used in the Planetary Nervous System to build applications as virtual sensors. Section V evaluates the performance of virtual sensors within the Planetary Nervous System. Section VI illustrates related work. Finally, Section VII concludes this paper.

II. The Virtual Sensor Model

Figure 1 illustrates the model of virtual sensors introduced in this paper. This model can be realized as a generic programming interface, with which open participatory platforms for privacy-preserving ubiquitous social mining can be software engineered.

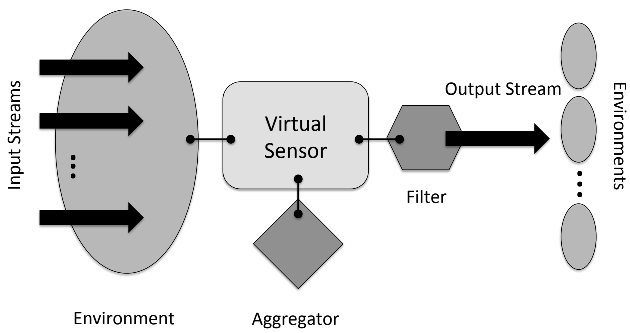

Fig. 1. The virtual sensor model introduced in this paper.

The core component of this model is the *virtual sensor*. A *virtual sensor* is defined by its *environment*, an *aggregator*, a *filter* and its *output stream*.

The environment of a sensor is a set of input streams of data generated from physical or virtual sensors. The environment defines the context within which the virtual sensor operates to generate its output stream.

The aggregator processes the input streams from the environment of the sensor in real-time and transforms them to the output stream of this sensor. An aggregator can be part of different types of sensors. For example, an aggregator may perform summation of input streams with numerical values, with each input stream having a given weight. The values of the weights may vary depending on the type of the sensor in which the aggregator is applied. Similarly, different aqgregators may operate within the same type of sensor, for instance, a sensor that computes the error of the input streams in its environment can be realized with aggregators that compute the absolute, relative or root mean square error.

The output stream of a virtual sensor is a type of real-time data signal generated by the aggregator of the virtual sensor. An output stream can be part of one or more other environments.

The filter controls the availability of the output stream to all other environments in real-time. In other words, the filter introduces privacy-by-design within the virtual sensor model. A filter can be realized by a scheduling algorithm or even by a user interface through which users have full control on which sensor information they make available.

The information flow of this model is designed to be recursive: It starts from an environment sensed by a virtual sensor. The output stream of this virtual sensor can form new enhanced environments for further sensing. This recursive design in the information flow of the model enables a highly modular, compositional and extensible environment for building data-driven ubiquitous platforms for social mining. The rest of this paper shows how these properties of the virtual sensor model are put into practice to build an open participatory platform for privacy-preserving ubiquitous social mining: the Planetary Nervous System [5].

III. The Planetary Nervous System

The Planetary Nervous System[2] is a large-scale distributed platform that provides ubiquitous social mining services as a public good. Users are provided with freedom and incentives to share, collect and, at the same time, protect their data. Data are collected in real-time from different heterogeneous sources such as mobile phones, environmental sensors, home sensors, etc. In contrast to most existing 'Big Data' systems designed to be closed, proprietary, privacy-intrusive and discriminatory, the Planetary Nervous System is an open, privacy-preserving and participatory platform that does not rely on any centralized computational or data storage entity for its operations.

The development[3] of the Planetary Nervous System is an ongoing work with a large supporting community originated from the FuturICT project[4]. The project aims to increase innovation by evolving as a citizen web via crowd-sourced participation[5]. Citizens can contribute in the development of the core functionality and the development of applications. This section shows how all the different components of the Planetary Nervous System can be realized end-to-end according to the model of virtual sensors. This holistic approach provides a shared community view of the project. It increases community awareness about the different components of the system and how they can interact with each other. It defines a common 'design language', with which the activities of building communities can be coordinated. Therefore, the model of virtual sensors has the potential to simplify the engagement of citizens in different crowd-sourcing activities and increase its adoption by building communities.

Figure 2 illustrates the design of the Planetary Nervous System according to the model of virtual sensors. The first observation is that all software components of the Planetary Nervous System are elements of the virtual sensor model. Data are collected from different ubiquitous environments with both physical and virtual sensors. The current implementation focuses on mobile phones such as Android and iOS systems, however, an extension to the physical sensors of the Arduino platform is ongoing work.

Smart phones provide access to various physical sensors, such as accelerometer, humidity, battery, temperature, etc. An aggregator of a virtual sensor can control the frequency of data sampling in the output stream of the virtual sensor. Moreover,

[2]http://www.nervous.ethz.ch (last accessed: October 2014)
[3]https://github.com/mosgap/nervous (last accessed: October 2014)
[4]http://www.futurict.eu (last accessed: October 2014)
[5]Some channels are available at http://www.nervous.ethz.ch/trac and https://twitter.com/nervousnet (last accessed: October 2014)

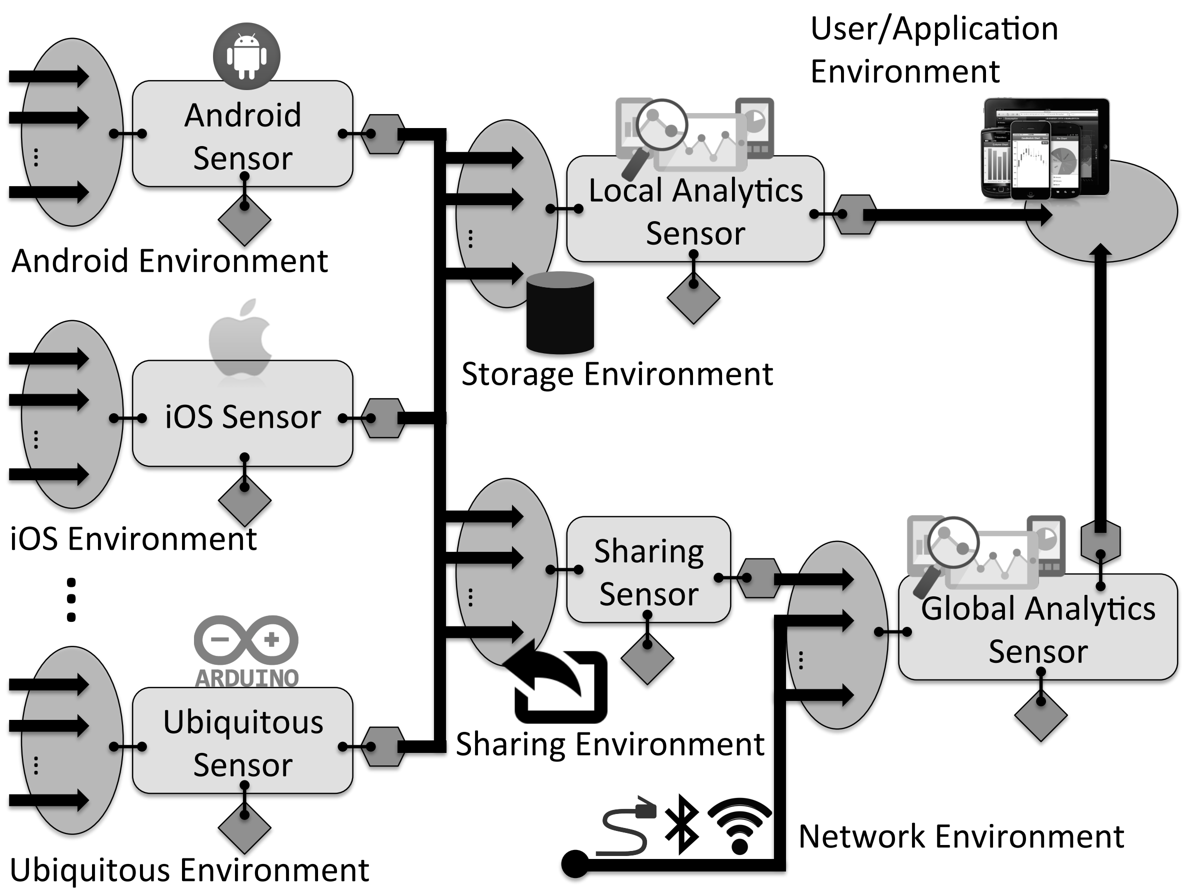

Fig. 2. The Planetary Nervous System designed according to the model of virtual sensors.

the output streams generated by these virtual sensors on a user's device come with a self-determining privacy control in two levels. The *first privacy level* provides user control for storing the data of the output streams in the phone. Users can select to log or not data from each sensor, but they can also schedule logging at certain time periods. This privacy control functionality is implemented in the filter of the Android and iOS virtual sensors. However, a user may desire different privacy control for storing data locally and sharing data with other users. This specialized privacy functionality can be engineered as a virtual sensor, the sharing sensor as shown in Figure 1, whose aggregator controls the streams shared in the network environment. This is the *second privacy level* introduced in the Planetary Nervous System. Both privacy levels are designed via the same model of virtual sensors. Figure 3 illustrates an example of a user interface that implements the two privacy levels of control.

Data sensed from sensors are stored in the storage environment shown in Figure 2. The storage environment implements an efficient method for serializing structured data with the Protocol Buffer library[6]. Data are stored within a data structure of a Red-Black tree that indexes data for fast retrievals based on range queries that define a period of time. The data stored in phones act as a data pool over which lightweight data analytics are performed. Such analytics are implemented in the aggregator of a local analytics sensor and include aggregation functions such as summation, average, maximum, minimum, standard deviation, but also data mining algorithms such as clustering. The aggregator interface of the local analytics sensor defines a toolkit for real-time operations performed in time-series data, with which application developers can further build other virtual sensors. Local analytics are performed over the data of a sensor type for a defined period of time. The purpose of the local analytics sensor is twofold: (i) It provides data for an engaging, interactive and gamifying visual experience to users in order to understand and explore in real-time their own social environment and activity. (ii) It provides intuition for users and developers to build their own applications with virtual sensors. Figure 4 illustrates two examples of real-time interactive visualizations[7] performed in mobile phones using the local analytics sensor.

While local analytics provide interesting information about

[6]Available at https://code.google.com/p/protobuf/ (last accessed: October 2014)

[7]The interactive charts of the Planetary Nervous System are implemented with Chart.js library that is available at http://www.chartjs.org (last accessed: October 2014).

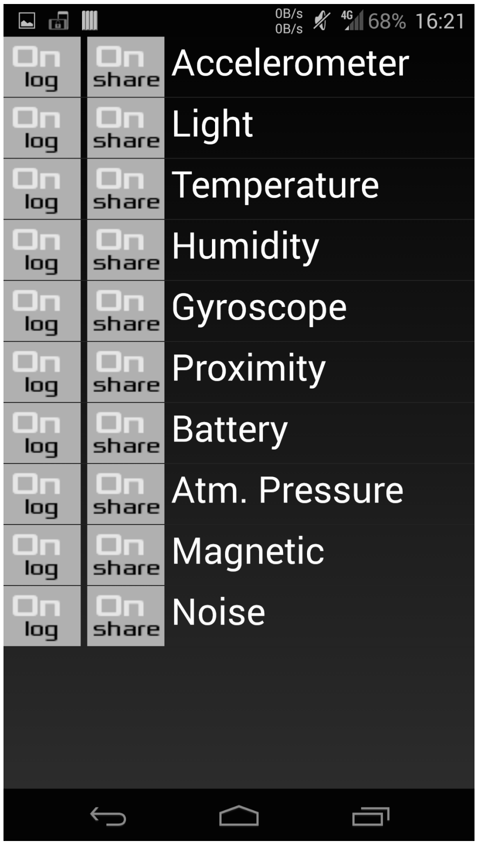

Fig. 3. An implementation of a user interface with the two privacy control levels in the Planetary Nervous Systems.

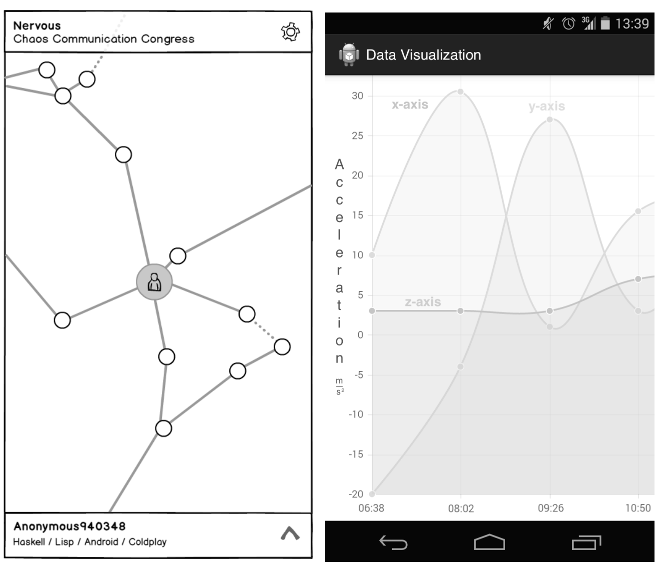

Fig. 4. Two examples of interactive user interfaces in the Planetary Nervous System that use the local analytics sensor. The first interactive visualization plots a social graph in real time based on the phsyical proximity of users. The graph is built with data collected from bluetooth beacons. The second interactive visualization plots data analytics from sensor data, e.g. accelerometer data.

single users, collective information about the status of the participatory community cannot be captured in real-time via the local analytics sensor only. System-wide analytics is the objective of a global analytics sensor, currently work-in-progress in the Planetary Nervous System. A global analytics sensor is ambitious and challenging to realize as computations should be performed in a decentralized fashion without reinventing the wheel of another 'Big Data' system. Distributed privacy-preserving aggregation services, such as DIAS, the *Dynamic Intelligent Aggregation Service* [10] and OpenPDS [11], can realize a global analytics sensor. More details about the implementation of this virtual sensor are out of the scope of this paper and part of future work.

IV. Applications of Virtual Sensors

Building applications according to the model of virtual sensors is based on the principle of aggregating a set of input streams from a defined environment and transforming them to a new output stream. This transformation can take place at several stages with multiple virtual sensors bound in an application graph of data streams.

Building applications with this modular and compositional approach can be straightforward in several cases. For example, although mobile phones do not have a physical noise sensor, a virtual noise sensor can be built using the physical sensor of a microphone. An aggregator samples sound information from the input stream of the microphone sensor and computes the sound power level for different frequency bands. Such a noise sensor is implemented in the Planetary Nervous System. The virtual noise sensor can be further enhanced to support higher privacy. For example, the design of a microphone as a virtual sensor with a filter that implements a low-pass filter for removing voice frequencies can provide an additional privacy level for noise detection applications. Spatial data from a population of users with such virtual noise sensor can be used to create real-time noise pollution maps of cities.

There are also other more complex applications that may involve collective crowd sensing, cognitive tasks and detection of complex human activities or physical phenomena. Some examples of such applications may include the following: earthquake detection [8], evacuation/emergency support systems [12], ambient assisted living [13], etc. A number of supporting virtual sensors are introduced in the Planetary Nervous System that generate meta-information used to reason about the design of complex virtual sensor applications. For example, a sentiment and activity sensor used under controlled experiments enable users to tag their status in real-time. Application developers can use this information to reason about their application design. The advantage of these supportive virtual sensors in this case is that information collected from this supervised learning or classification process is managed and stored in a universal way as a user input interface can be engineered with virtual sensors. Moreover, the privacy-by-design approach in the model of virtual sensors provides personalized and self-determining control in data sharing for every type of application developed with virtual sensors.

V. Performance Evaluation

This section evaluates the performance of virtual sensors by emulating data collection and retrieval by two phone user. The emulated users run the Planetary Nervous System in the following two Android 4.4 devices:

- *Samsung Galaxy S II*: Dual-core 1.2 GHz Cortex-A9 CPU, 1 GB RAM and Li-Ion 1650 mAh battery
- *LG Nexus 5*: Quad-core 2.3 GHz Krait 400 CPU, 2 GB RAM and non-removable Li-Po 2300 mAh battery

The users are assumed to run 20 virtual sensors. Each virtual sensor senses its environment every 5 seconds. The size of each measurement is assumed to be the same with the one of the noise virtual sensor implemented in the Planetary Nervous System that is 53 bytes. Therefore, during a day, each virtual sensor performs 86400/5=17280 measurements that require 915.84 KB of storage in the phone. The total storage requirement each day for all 20 sensors is 18.3 MB for 345600 measurements.

The goal of the experiments performed in this section is to evaluate the following performance indicators:

- *Insertion time*: The total period of time required for the storage environment to store the total number of measurements during each day.
- *Retrieval time*: The total period of time required by another virtual sensor, e.g. the local analytics sensor, to retrieve the total number of measurements of a day.
- *Battery level*: The battery consumed for storing in the storage environment the total number of measurements in each day and the battery consumed to retrieve by another virtual sensor, e.g. the local analytics sensor, the total number of measurements of a day.

The evaluation focuses on the performance overhead caused exclusively by the insertion and retrieval of measurements of the 20 virtual sensors. This isolation is achieved by letting the virtual sensors generate dump measurements in order to minimize the performance overhead caused by other operations of the virtual sensors. The experiment runs recursively 10 times to evaluate the performance indicators over an emulated period of 10 days. During runtime, all applications and activities of the phone are turned off besides the Planetary Nervous System application that runs the benchmark.

Figure 5 illustrates the insertion and retrieval times for the two phones during the emulated runtime of 10 days. The average daily insertion time for the LG Nexus 5 is around 4 minutes that is 57% faster than the Samsung Galaxy S II that is over 9 minutes. Similarly, the average daily retrieval time for the LG Nexus 5 is less than 34 seconds that is 72% faster than the Samsung Galaxy S II that is over 2 minutes. Note that the CPU performance of the LG Nexus 5 is almost double the CPU performance of the Samsung Galaxy S II and therefore, a significant difference in the insertion and retrieval times between the two phones is expected. The faster retrieval time compared to the insertion time is justified by the indexing structure implemented by a Red-Black tree. These results show that real-time social mining via the local analytics sensor is feasible as retrieving all measurements of a day for a certain type of virtual sensor can be performed in only a few seconds.

Figure 6 illustrates the battery level during the emulated runtime of 10 days. The main values indicate the battery level before retrieval, whereas, the error bars indicate the battery level before insertion and at the end of the experiment.

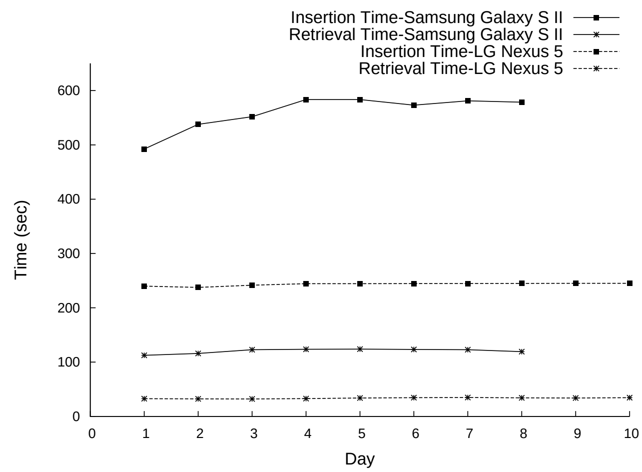

Fig. 5. Insertion and retrieval times for the two phones during the emulated runtime of 10 days.

Storing and retrieving data in the application of the Planetary Nervous System causes daily 1.5% of battery consumption in average for the LG Nexus 5, whereas, the average daily battery consumption for the Samsung Galaxy S II is 10.1%. The results of the last two days in the experiment of the Samsung Galaxy S II are missing as the runtime is interrupted because of a critically low battery level. The significantly higher battery consumption in the Samsung Galaxy S II cannot be justified by the battery quality exclusively. The significantly higher insertion and retrieval times cause a higher overall runtime of the experiments resulting in a higher overhead in battery consumption. However, the results of battery consumption show that the challenging emulated scenario is feasible for both phones.

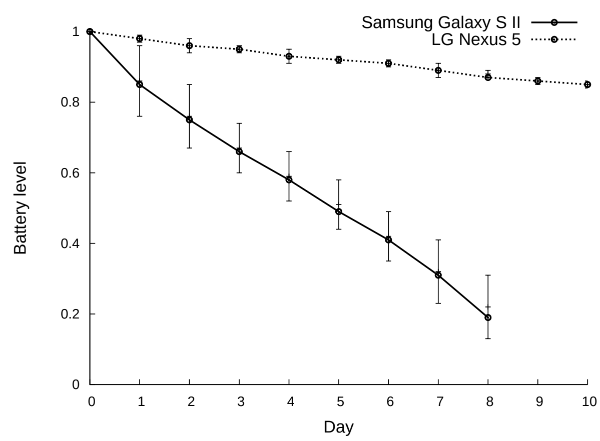

Fig. 6. The battery level for the two phones during the emulated runtime of 10 days.

The model of virtual sensors allows several performance trade-offs that can assure the scalability when the number of virtual sensors increases. For example, the logging period can be adjusted on-the-fly according to the total number of virtual sensors running or according to the performance profile of the phone. These enhancements are part of on-going work.

VI. Related Work

Authors in [14] provide an overview of the existing virtual sensing techniques used in several application domains. Virtual sensing is defined as a technique that relies on physical sensing and a model to aggregate information from certain areas, without deploying physical sensors in those locations. In other words, virtual sensing enables the estimation of information at one location based on the data obtained from other locations. Virtual sensing techniques are often used in domains such as active noise control [15], intelligent transportation systems, gas sensing systems [16] and robot design [17]. Several research efforts in these domains focus on defining virtual sensors based on a common principle: decoupling sensor deployment from application development that allows applications to dynamically discover, access and compose sensor services.

In [18], a virtual sensor is defined as a software component that provides measurements that are not physically measurable by combining sensed data from a group of heterogeneous physical sensors. A programming interface is introduced that enables applications to define tailored aggregation through virtual sensors. A prototype implementation of the middleware includes the creation of virtual sensors enabling adaptive and efficient in-network processing that dynamically adapts to applications needs. A declarative specification of the virtual sensors allows a programmer to describe the desired behavior. Applications and sensors share knowledge of a naming scheme for the low-level data types the sensor nodes can provide, e.g. 'location', 'temperature'. The programmer specifies the input data types of the physical measurements, an aggregator to calculate the desired measurements, the resulting data types and the aggregation frequency. In contrast, the virtual sensor model applied in the Planetary Nervous System forms a different approach: the aggregator component of the local analytics sensor exposes a generic interface defined by a set of functions with certain parameters, based on which social mining application are built. Another advantage of the virtual sensor model introduced in this paper is the self-determination of data sharing for every type of application developed with virtual sensors.

In [18], there are two types of queries that can be performed on a virtual sensor: one-time queries that return a single result or persistent queries that return periodic results. When the query cannot be made over the available physical sensors, the developer constructs and deploys a virtual sensor using knowledge of the available data types. The middleware is in charge of discovering the physical sensors needed by the application, based on the specified input data types. The high-level specification of the virtual sensor is translated into low-level code that runs either locally or relayed to a resource-constrained sensor within the network. A virtual sensor is deployed only when there are active queries, and the information from the virtual sensor is accessed on-demand. When the result is available, a listener is activated and it forwards the results to the application asynchronously.

The virtual sensor system for environment observation, namely radar rainfall [9], is designed as a community tool that facilitates real-time customization of physical sensor data. This virtual sensor system provides customization and collaboration through the publication of both the aggregated data and the workflow composition involved in obtaining them. By making available to users the template of the workflow used in the development of the virtual sensor, the data users can select from a broad range of virtual sensors to further build new ones and support their research. The prototype is designed in a three-layer architecture: the bottom layer contains a variety of remote heterogeneous sensor; the middle layer defines the virtual sensor abstraction layer; the top layer is a web-based collaboration interface through which users deploy and visualize new instances of the virtual sensors by using the published workflows. In contrast to this three-layer approach, the model introduced in this paper allows the composition of multiple layers and application graphs of virtual sensors. Moreover, an additional component, namely the filter, introduces the option to control the availability of the output stream to all other environments, in other words a privacy self-determination that is not captured in [9].

The authors in [19] propose a distributed and scalable mechanism for enabling virtual sensors in intelligent environments by relying on service-oriented sensor networks. In a Service-oriented Sensor Network (SOSN), each sensor is represented as a service object in a service framework that allows their dynamic discovery and composition into applications. A user issues a query to the virtual sensor framework that triggers the dynamic composition of physical sensor services into virtual sensors. Similar to [9], the framework is also designed as a layered structure. However, the architecture consists of four layers, instead of three: the physical layer that contains a variety of sensors that monitor different aspects of the physical space; the node layer that is built by a distributed set of hardware nodes, which integrate sensors from the physical layer and export their service representations to the layers above; the service layer that contains service representations of all sensors and actuators connected to the hardware nodes; the application layer that access sensors via their respective service objects. The node layer is the additional layer introduced for implementing aggregation by virtual sensors, which, in contrast, is provided by the service layer in [9] via a distributed and scalable approach. Privacy issues, raised by data collection and sharing, are not addressed.

VII. Conclusion and Future Work

This paper concludes that the model of virtual sensors is a promising design approach for building ubiquitous social mining services that are by design, open, decentralized, privacy-preserving and participatory. This paper shows how the Planetary Nervous System can be engineered according to the model of virtual sensors to provide such complex services. Two privacy control levels for logging and sharing data are realized in the Planetary Nervous System using the model of virtual sensors. Moreover, social mining applications can be incrementally developed by linking data streams of reusable virtual sensors in application graphs. This modular and compositional approach for application development is relevant for crowd-sourcing activities as it stimulates engagement and innovation by building communities. Performance evaluations of virtual sensors in the Planetary Nervous System confirm the feasibility of the model.

Future work focuses on further designing applications of virtual sensors such as the sentiment and activity sensor that will allow crowd-sourced building of more complex social mining applications using real-time data exchanged across inter-connected ubiquitous devices. We envision the emergence of an open virtual sensor ecosystem through which participatory citizens can contribute and acquire social mining services as public good.

Acknowledgment

The authors would like to heartily thank the core development team of the Planetary Nervous System alphabetically, for their dedication and enthusiasm to work on this project: Marica Bertarini, Simone Forte, Patrick Misteli, Petr Neugebauer, Siddhartha, Ramapriya Sridharan, Samir Sulaimanov and Fabian Tschopp.

References

[1] J. Manitz, T. Kneib, M. Schlather, D. Helbing, and D. Brockmann, "Origin detection during food-borne disease outbreaks-a case study of the 2011 ehec/hus outbreak in germany," *PLoS currents*, vol. 6, 2014.

[2] D. Helbing and S. Balietti, "From social data mining to forecasting socio-economic crises," *The European Physical Journal-Special Topics*, vol. 195, no. 1, pp. 3–68, 2011.

[3] E. Pournaras, M. Vasirani, R. Kooij, and K. Aberer, "Decentralized planning of energy demand for the management of robustness and discomfort," *Industrial Informatics, IEEE Transactions on*, 2014.

[4] D. Helbing and S. Balietti, "Big data, privacy, and trusted web: What needs to be done," *Eur. Phys. J. Special Topics*, no. 195, pp. 3–68, 2011.

[5] F. Giannotti, D. Pedreschi, A. Pentland, P. Lukowicz, D. Kossmann, J. Crowley, and D. Helbing, "A planetary nervous system for social mining and collective awareness," *The European Physical Journal Special Topics*, vol. 214, no. 1, pp. 49–75, 2012. [Online]. Available: http://dx.doi.org/10.1140/epjst/e2012-01688-9

[6] E. Pournaras, M. Yao, R. Ambrosio, and M. Warnier, "Organizational control reconfigurations for a robust smart power grid," in *Internet of Things and Inter-cooperative Computational Technologies for Collective Intelligence*, ser. Studies in Computational Intelligence, N. Bessis, F. Xhafa, D. Varvarigou, R. Hill, and M. Li, Eds. Springer Berlin Heidelberg, 2013, vol. 460, pp. 189–206. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-34952-2_8

[7] E. Pournaras, M. Yao, and R. Ambrosio, "Dynamic composition and reconfiguration of internet-scale control systems," in *Digital Ecosystems and Technologies Conference (DEST), 2011 Proceedings of the 5th IEEE International Conference on*, May 2011, pp. 233–240.

[8] M. Faulkner, R. Clayton, T. Heaton, K. M. Chandy, M. Kohler, J. Bunn, R. Guy, A. Liu, M. Olson, M. Cheng, and A. Krause, "Community sense and response systems: Your phone as quake detector," *Communications of the ACM*, vol. 57, no. 7, pp. 66–75, July 2014.

[9] D. J. Hill, Y. Liu, L. Marini, R. Kooper, A. Rodriguez, J. Futrelle, B. S. Minsker, J. Myers, and T. McLaren, "A virtual sensor system for user-generated, real-time environmental data products," *Environmental Modelling & Software*, vol. 26, no. 12, pp. 1710 – 1724, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1364815211001988

[10] E. Pournaras, M. Warnier, and F. M. Brazier, "A generic and adaptive aggregation service for large-scale decentralized networks," *Complex Adaptive Systems Modeling*, vol. 1, no. 19, 2013.

[11] Y.-A. de Montjoye, E. Shmueli, S. S. Wang, and A. S. Pentland, "openpds: Protecting the privacy of metadata through safeanswers," *PloS one*, vol. 9, no. 7, p. e98790, 2014.

[12] E. Asimakopoulou and N. Bessis, "Buildings and crowds: Forming smart cities for more effective disaster management," in *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2011 Fifth International Conference on*, June 2011, pp. 229–234.

[13] A. Dohr, R. Modre-Opsrian, M. Drobics, D. Hayn, and G. Schreier, "The internet of things for ambient assisted living," in *Information Technology: New Generations (ITNG), 2010 Seventh International Conference on*. Ieee, 2010, pp. 804–809.

[14] L. Liu, S. Kuo, and M. Zhou, "Virtual sensing techniques and their applications," in *Networking, Sensing and Control, 2009. ICNSC '09. International Conference on*, March 2009, pp. 31–36.

[15] C. D. Petersen, R. Fraanje, B. S. Cazzolato, A. C. Zander, and C. H. Hansen, "A kalman filter approach to virtual sensing for active noise control," *Mechanical Systems and Signal Processing*, vol. 22, no. 2, pp. 490 – 508, 2008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0888327007001069

[16] Z. Ankara, T. Kammerer, A. Gramm, and A. Schtze, "Low power virtual sensor array based on a micromachined gas sensor for fast discrimination between h2, {CO} and relative humidity," *Sensors and Actuators B: Chemical*, vol. 100, no. 1?2, pp. 240 – 245, 2004, new materials and Technologies in Sensor Applications, Proceedings of the European Materials Research Society 2003 - Symposium N. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925400503009365

[17] F. Bellas, J. A. Becerra, J. S. Reyes, and R. J. Duro, "Applying synaptic delays for virtual sensing and actuation in mobile robots," in *IJCNN (6)'00*, 2000, pp. 144–152.

[18] S. Kabadayi, A. Pridgen, and C. Julien, "Virtual sensors: Abstracting data from physical sensors," in *Proceedings of the 2006 International Symposium on on World of Wireless, Mobile and Multimedia Networks*, ser. WOWMOM '06. Washington, DC, USA: IEEE Computer Society, 2006, pp. 587–592.

[19] R. Bose and A. Helal, "Distributed mechanisms for enabling virtual sensors in service oriented intelligent environments," in *Intelligent Environments, 2008 IET 4th International Conference on*, July 2008, pp. 1–8.