User-Centric Social Interaction for Digital Cities

Kåre Synnes[1], Matthias Kranz[2], Juwel Rana[1], Olov Schelén[1], Michael Nilsson[1]

[1]*Luleå University of Technology, Sweden*

[2]*University of Passau, Germany*

Abstract

Pervasive Computing was envisioned by pioneers like Marc Weiser, but has yet to become an everyday technology in our society. The recent advances regarding Internet of Things, social computing and mobile access technologies however converge to make pervasive computing truly ubiquitous. The key challenge is however to make simple and robust solutions for normal users, which shifts the focus from complex platforms involving machine learning and artificial intelligence to more hands on construction of services that are tailored or personalized for individual users.

This chapter therefore discusses Internet of Things together with Social Computing as a basis for components that users in a 'digital city' could utilize to make their daily life better, safer, etc. A novel environment for user-created services, such as social apps, is presented as a possible solution for this. The vision is that anyone could make simple service based on Internet-enabled devices (Internet of Things) and encapsulated digital resources such as Open Data, which also can have social aspects embedded.

This chapter also aims to identify trends, challenges and recommendations in regard of Social Interaction for Digital Cities. This work will help expose future themes with high innovation and business potential based on a timeframe roughly 15 years ahead of now. The purpose is to create a common outlook on the future of information and communication technologies (ICT) based on the extrapolation of current trends and ongoing research efforts.

Introduction

By the end of 2008 a milestone was reached, there was now more people living in the cities than outside. This has of course affected and will even more affect people's life in the future. A higher density of people creates its challenges, problems and needs.

Today's society and economy is totally depending on a working and always accessible Internet 24/7. This fact changes and creates opportunities among people in cities and elsewhere. In the cities there is a high density of almost everything and therefore the need of services is special - citizen centric services.

Early 2010 the topic Smart Cities was not very much known in the research community of Future Internet (FI). So far FI had focused on the next generation Internet, building large-scale test-beds, having in mind that this is a 30 year old design. Visions like 50 billion connected devices by the year 2020 (Ericsson), Internet of things creates new opportunities.

The last ten years another topic called Living Labs, sometimes also described as open user driven innovation, entered the European research scene. Methods, tools and processes have been developed in how to involve users as co-creators and this in parallel to FI research. Today there are tools ready to be used involving end users, many end users, to participate in the development of new services and products

as co-creators. Not in the end of the product development cycle but early, for, with and by, the users of the new services/products. These new services can be developed by the end-users in the cities (citizens).

Up to now many services that already from the start didn't attract a huge amount of potential customer was never created. Phenomena like the entry of the iPhones and Android mobile phones completely changed the game plan in the telecom world. Services for a very small group of users was possible to develop to a small cost and by the users of their own. Still though, you have to be a rather skilled 'programmer' to create a mobile 'app' or service so the challenge today is to lower the threshold, the barriers of becoming a 'programmer'.

By providing the users, the citizens with tools in order to make their own mobile services the expectation is booming regarding potential new needs to be solved by citizens developing their own services. Internet of things, 50 billion connected devices, open and accessible public data, both from static servers in the city but also by dynamic sensor data in the street will create a totally new scenario about a more intelligent use of smart technology creating a better quality of life and entrepreneurship in cities.

One definition of a Smart City is 'We believe a city to be smart when investments in human and social capital and traditional (transport) and modern (ICT) communications infrastructure fuel sustainable economic growth and a high quality of life, with a wise management of natural resources, through participatory governance' [6].

The trend is to allow anybody to become a developer of services, even for a small target group usually not in focus by telecom operators and thereby contribute to a better society in many aspects, step by step. Smart Cities is very close to the thematic research area of Digital Cities and the importance of citizen centric services can not enough be seen as a strong driver of new services, products and companies but to reach full effect, there is also a need to lower the threshold, provide the tools, and utilize peoples creativity and the cities advantage as a multicultural melting pot driving societal changes will reach its full potential.

This book discusses the creation of personal, social, and urban awareness through pervasive computing. Although pervasive computing services are foreseen as potentially revolutionary, there is yet little adaption in industry. This paradox is similar to the predicted potential of artificial intelligence and later machine learning, which are successfully applied within a few applications, but which are not generally adopted. Though, this topic recently received again a lot of attention within the context of embodied AI, that is AI within technical systems. The reason for the lack of real-world adoption of pervasive computing for social interaction is potentially due to the inherit complexity of such systems that needs to span both heterogeneous networks and organizations. How can then pervasive computing succeed better? The authors of this chapter believe that there are clear incentives, which will be discussed further later in this chapter. It however builds on three pillars: access to open data, novel interaction techniques and enabling end-users to visually compose mobile/pervasive components.

In the following, we introduce in Section 2 the characteristics for grid architectures for Internet of Things, which enables citizen-centric services based on open data. Section 3 presents a discussion of Social Web of Things. Section 4 presents initial work on a framework of social components enabling citizens to easily create their own mobile social apps. Section 5 summaries the chapter, highlights the trends, identifies future challenges and presents recommendations based on the presented work.

Grid Architectures for the Internet of Things

There are several proposed architectures for storing, indexing and presenting large scale sensor [40, 38, 39, 7, 12, 27]. However, there is more research needed to fully exploit the Internet of Things (IoT) and Crowd scenario across wireless domain and a distributed cloud of multiple players in providing application services. Experiences from these earlier architectures should be considered.

Earlier results in real-time and client-server scenarios should be reused. There are systems focusing on real-time streaming of sensor data to sinks [53]. More recently there is a trend towards advanced sensor nodes that have capabilities to act as servers providing their data through lightweight RESTFUL approaches [54]. Methods for advertising and discovering data are needed [1, 5]. Protocols for communications in resource-constrained environments are developed in the IETF [31, 32].

The objective would be to provide technology and infrastructure for an open business environment of both free data [59] and data provided commercially on equal terms. Security considerations must be explored [24, 37].

Grid Architectures for Internet of Things and Future Media

The Internet of Things is expected to grow quickly to 50 billion devices and beyond [14, 13]. The solutions will include sensor networks and machine-to-machine communication. Applications may range over dynamic services in smart cities, advanced annotated media, supporting industrial and business processes, etc. In such networks, sensors and other input-devices will produce vast amounts of data that need to be collected, disseminated, stored, classified and indexed in a scalable way for various application purposes. Sensor data must in some cases be provided in real-time to large numbers of receivers and in some cases be stored and indexed for retrieval of large numbers of applications.

Research has emerged on the above-described aspects but most proposals are point solutions for specific scenarios, requirements, and problem domains. Consequently, the objective of this focus area is to support more research on scalable solutions for Internet of Things scenarios. This includes supporting crowd services where the crowd can be both producers and consumers of data. Besides meeting technical requirements, a clear objective is to promote open and generic distributed cloud solutions where a diversity of players can interact on reasonable and equal business terms to jointly provide unprecedented end services.

Research direction includes generative grid architectures that can scale over the data collection domain and application domain involving multiple users and organizations cooperating for the common objectives. Some specific research issues are mentioned in the following sections and at the end some related work is listed.

Security (authentication, authorization, encryption)

The solutions must be completely open and promote equal opportunities (e.g. business terms) to different players, however this does not necessarily mean that all data is free and available to anyone. Therefore authentication, authorization and encryption are essential elements that need to be researched in this context.

Performance and scalability

The vast number of producers and consumers where many entities will assume both roles, require specific research on performance and scalability in this context. The scalability solutions must be applied across all other requirements as here listed.

Decentralization, open interfaces, business interfaces

A key objective is that there must be an open market place with open interfaces and open business terms. Much data may be free and unrestricted (possibly funded by advertisements), but diversity of players that refine data and services is normally increasing if there is support to charge for and protect access to such data.

Wireless aspects

Wireless devices may be the norm in Internet of Things and crowd scenarios. The architecture must support efficient resource usage (e.g. battery and power) and some degree of service continuity in scenarios where mobile devices (producers or consumers of data) only have occasional connectivity.

Resource discovery

The vast amount of data provided must be discoverable by entities that want to use them. It is expected that data offered in the crowd scenario may in some cases be hard to find and in some cases be very redundant. In either case it is a matter of resource efficiency and service availability to be able to determine how and where to retrieve data.

Storage, classification and indexation

Large scale (big data) storage of information in Internet of Things networks is a critical issue. For data that is timeless or of typical historical value this is a natural issue. However, even for typical realtime data it may be desirable to store some history for following up on failures etc. Existing data store technologies may need adaptation for effective storage and indexation of such data. Also, machine generated data from multiple sources is often hard to combine and interpret for humans. Automated methods to interpret data and classify it into real-world status and events are essential.

Applications

The key drivers for Internet of Things grid networks come from specific and creative applications in smart city scenarios, both addressing citizens and organizations/enterprises. Besides bringing clear values, the technologies mentioned previously should be evaluated both quantitatively and qualitatively in such application scenarios.

Social Web of Things

The on-going connection of appliances, the increasing adoption of smartphones, and emerging instrumentation of items with QR-codes and RFID provide the basis for a comprehensive layer of connectedness to objects, products, things and people. People are becoming part of digital social networks driven by personal interests and aspiration. The feeling of belonging to a community and the perpetual drive of getting connected from real life find it continuation in digital networks.

Both the digital integration of things and people starts to embrace our daily lives and enables for new interaction, new experiences and new behaviors. We can remotely query and control appliances of a smart home, we can participate in the experiences and opinions of our friends about product while shopping, and we can share our activities, our preferences instantly? [33, 26]

The technological developments with respect to computing power, sensing systems, communication technologies, identification systems, middleware systems and infrastructure have resulted in a large number of uniquely identifiable systems and objects allow the sensing and actuation of real world phenomena on a large scale, not possible before. These embedded systems and the information embedded therein [50] is mainly designed for machine-to-machine communication (M2M), but also facilitate human-machine communication (M2H), also called human-computer interaction (HCI).

This provides novel chances for future services, especially with respect to the integration into people's everyday life. Technology becomes, as envisioned by Weiser, part of it, indistinguishable and woven into the fabric of everyday life: *'In the 21st century the technology revolution will move into the everyday, the small and the invisible'* [56]. In the following, we discuss the novel challenges and potentials of the proliferation of these developments, just entering our environment and emerging markets.

Private and Public Sensing as basis for Open Data and Big Data

Networked sensing systems on all scales are on the verge of entering different environments of public and private life: intelligent power grids, vehicular communication systems such as vehicle-to-vehicle (V2V) and vehicle-to-infrastructure communication (V2I) [52, 47, 51], smart home sensing, or crowdsourced data, both acquired explicitly (manually inputted or automatically acquired) and implicitly (e.g. such as the traffic data generated by GSM base station changes from travellers using navigation services, such as TomTom HD traffic) or by explicitly incorporating the smartphone of the user [9]. This data is complemented by the release of governmental data, such as geo-referenced data sets (e.g. maps, pollution data, air quality data, health information, ...) or other institutional data. This data is shared, online (e.g. using services such as former Pachube) or offline, in near-real time or even real time. The data originates from both public (e.g. governmental institutions, research agencies, ...) and private sources (companies, private persons, ...). This can be use to create novel socio-technical networks optimizing e.g. personal mobility [10].

Solutions making use of the data will not only be required to make sense of all the sensing (such as employing techniques from machine learning), but also will need to handle this big data (immense amount of potentially heterogeneous data sources and data types that need processing, potentially in short periods of time). This requires novel tools and methods, especially novel middleware solutions [45, 2, 48]. While current social networks and social interaction platforms such as Facebook feature 1,000,000,000 users, the amount of sensor systems will be several magnitudes higher, assuming 50 to 100 devices per person will result in 50,000,000,000 to 100,000,000,000 data sources and sinks that produce data, at several Herz per second, and potentially of high dimension (multiple sensors per device) or size (full HD imagery).

The availability of data, provided that privacy and other legal issues are appropriately addressed, forms the basis for novel business models, services and markets by the end of 2020.

Embedded Interaction – Distributed Sensor and Actuator Systems

'Technological advances and new usage models can cause computing to undergo a stark transformation. Automatic object identification (such as RFID or Near Field Communication, and visual markers), ubiquitous connectivity, improved processing and storage capabilities, various new display technologies, sensor device availability, and decreasing hardware costs all lay the foundation for a new computing era. We can now build vehicles, devices, goods, and everyday objects to become a part of the Internet of Things.' [29].

The resulting artifacts are equipped with sensors and actuators that let users seamlessly manipulate digital information and data in the context of real-world usage. This means data is not only sensed, but also used for control purposes (such as intelligent heating and climate control (HVAC – heating, ventilation, air conditioning)). This development does not only increasingly show in the process industry (which, according to today's standards and the state of research is in urgent need for applied solutions to increase efficiency and environmental balance), but also the private and public sector. The control might occur automatically in many cases (e.g. based on machine-to-machine (M2M) communication) to control and steer systems, but it will often enough involve human users that want to modify their environment according to their social needs and social contexts [27].

This sensing and actuation will more increasingly be done via embedded systems and embedded devices. This trend is already immanent, looking e.g. at CPU sales where embedded CPUs have already by far outnumbered classical CPUs for desktop or enterprise computing. These embedded devices and systems will, in addition to their use in automated systems, become important points of interaction between humans and the environment, both in the private domain and in the public space. Miniaturization does not only allow us already to include technology e.g. in clothing (so-called wearables), but to further decrease the costs for a constant amount of computing power. The technological development of the ten recent years has made it possible to transfer this from prototypical objects [28] used in research into smart products for the mass market. This results in networked embedded systems being deployed in public infrastructure, from waste containers, to parks, public spots, etc., allowing data exchange not only with an a-priori known central infrastructure, but spontaneously with e.g. mobile devices of users, both via network-only connection, but also by more natural means such as public-private interaction. These services will be available citywide and moving and following their human users. Further research will be necessary on how to develop, deploy and maintain these large-scale services for citizens [34].

We expect, given the existence of initial field studies and case in 2012 (such as in the city of Oulu, Finland, public data sets from smart cities like Amsterdam, Netherlands, and several others), the availability of first citywide interactive services, enabled by embedded interaction sensor-actuator systems, by 2025 or even thereafter. One major issue to be resolved will be the legal framework that ensures privacy and the algorithms for ensuring trust (between service providers and consumers – both technical and human). An example for a challenge in this dimension is, e.g. ensuring or detecting the trustworthiness of sensor data as basis for calculations.

Crowd City Services

'A governance infrastructure is the collection of technologies, people, policies, practices, resources, social norms, and information that interact to support governing activities. Smart governance infrastructures augment society's ability to organize, interact, and govern. Novel instances of smart governance infrastructures already exist and are regularly emerging in distributed organizations and online (social) communities.' [21, 22]

The governments of Europe aim at integrating their citizens more directly in all administrative and governance processes. This trend, fostering community and social interaction, will not be limited to governance, but also be extended to all parts of the daily life. We in the following sketch some ideas for services that could evolve in future smart cities, given the current trends and potential future developments foresighted in this report.

The availability of distributed, networked sensor-actuator system will allow for a novel level of participation and social interaction in future smart cities [15]. This poses the question how future participatory systems need, from a methodological point of view, need to be designed to integrate with the societal goals and digital cities.

Future participatory services will, by combining both machine and human intelligence, allow for a faster and more efficient than the information systems and electronic services today [25]. We distinguish between services using information push (to the citizen) and information pull (from the citizen).

Classical participatory services for eliciting information are e.g. MobileWorks[1] (e.g. used for large-scale research studies to overcome the current limitations of user-involvement in research projects [26], or Amazon's Mechanical Turk[2] where not only pure information is solicited, but 'artificial

---

[1] https://www.mobileworks.com
[2] http://www.mturk.com

intelligence' is simulated, e.g. to overcome the limitations of current machine learning systems (e.g. when Steve Fosset's plane crashed, satellite imagery was bought, put online, and participants asked to find potential crash sites and respective reports were financially awarded).

Further examples of current solutions for crowd sourcing and participatory sensing are shortly presented here: The service Waze[3] uses user-generated geo-references movement data to build up maps e.g. of street networks and to generate and maintain additional information, such as the traffic information. User-contributed content is elicited in a gamified approach where points are awarded e.g. based on the type of data and its novelty (a road that hundreds of people have taken results in less points than a road only few have taken). WheelMap[4] uses also a user-based approach to generate information on accessibility of transportation networks (such as public sidewalks, places, etc.). The service EyeQuest[5] aims at providing up-to-date, on-demand imagery of physical locations. An example could be to ask the community about a photo of places to be visited soon. Users then take pictures of the desired spots and share them. The project 'Kleinwassersensor'[6] democratizes environmental and pollution sensing (today still a domain for the public authorities), at the example of water quality sensing. Finally, services like FixMyStreet[7] or SeeClickFix[8] try to include the user in identifying potentially problematic issues in the city and to raise awareness of the authorities to them. These services today do not 'match' to still pre-dominant governance style. Additionally, they usually lack integration in today's ICT systems. Future digital cities might very well benefit from these and similar services – if the challenges and hurdles can be overcome: lack of middleware and data exchange, lack of trust models of public and user generated/sensed data, scalability (e.g. of public responses to thousands of reports), etc., and finally a model how this more efficient reporting can result in more efficient solutions (demanding for cheaper fixes to the problems).

Data from future networked sensor-actuator systems will demand for an in-depth research on what value-added services can be composed, by e.g. providing the data and eliciting information from the citizen. This might, in a simple case be, e.g. the display of a picture of a spot in the city and the annotation of pollution.

But different data will put different demands on the users, some information will be harder to generate (and potentially not be possible to elicit from everyone), so novel approaches will be needed to develop these services. Gamification might be one of many possible solutions. This methodological research will be required prior to the availability of the digital city's data, so that to-be-identified day0 use cases can be implemented and further speed up development, acceptance and proliferation of these services.

Additional research questions will include a formalized development methodology, an identification of key parameters of these services (from input to output, users, fields of applications, etc.), and finally the societal, economical and social goals that shall be supported, from increasing social interaction if potentially anonymous mega cities, to increased community perception and awareness, well-being and health, citizen-involving governance, or many other possible goals. We are currently, after more then a decade of research, only in the beginning towards and understanding of future digital cities [20, 19].

---

[3] http://www.waze.com

[4] http://www.wheelmap.org/

[5] http://www.eyequest.de/

[6] http://kleinwassersensor.com/

[7] http://www.fixmystreet.com/

[8] http://www.seeclickfix.com/

The challenge includes finding methods, tools and approaches that e.g. increasing social interaction in society, by linking data from the digital city (digital networks), to e.g. Facebook (social networks) to physical human 'ad-hoc' networks (find a group to solve, in a community approach, specific problems, community networks).

As Virtanen and Malinen formulate the problem: *'there is a growing interest to use online communities to support social interaction also in geography-based communities'* [55].

How could these extension of current social networks be achieved with the goal of fostering and increasing social interactions in the real world (and not decreasing it by the introduction of technology and data)?

Personal User-Configured Services

Large scale deployments and large scale sensing solutions are enabled and supported by the expected technological developments. But the resulting services will need more than ever before be able to adapt to situations, contexts, and user-preferences. One size fits all solutions will no longer be adequate – we see here the same trend away from the 'personal computer' to 'ubiquitous computing' where users have multiple computational devices, probably even one device per task.

Personalization and Individualization – Individualization of the Society

Successful services will have to support end-user composition (see below in this report) and end-user configuration of the services, allowing the user to personalize the data, service and user interface. The need for personalization is driven by the societal trend of individualization of the particular members of our society. The larger mega cities become, the more mass production and consumption are at the centre, the more important it becomes for the people to live their individuality. This trend, so far, is at least visible in the western societies and in the growing generation E.

Many anthropologists state that there are great generational differences that can be forseen today, where the new generation is intrinsically accustomed to computers and mobile technologies. Ida Hult, CEO of Trendethnography, defines these as 'Moklofs' or 'Mobile kids with lots of friends'. This Generation E is used to getting rapid feedback on their opinions and actions, through a big flora of tools[9].

'Those ages 8 to 18 spend more than 7,5 hours a day with such devices, compared with less than 6,5 hours 5 years ago, when the study was last conducted. And that does not count the 1,5 hours that youths spend texting, or the 0,5 hour they talk on their cell phones. And because so many of them are multitasking – say, surfing the Internet while listening to music – they pack on average nearly 11 hours of media content into that 7,5 hours.' – Kevin Drum, MotherJones.com

The challenges here will include to develop novel tools that allow the development of these services, the education of users to compose their own services, and the development of mental models for end-users that foster the understanding of the underlying processes. Understanding will be a crucial part of the acceptance of the services, and in turn also of the acquisition of the underlying data. Shared ownership will be crucial and important to achieve for the stakeholders, the citizens, of future digital cities.

From personalized and individualized production to personalized and individualized consumption: Example for novel services

---

[9] Dagens Nyheter 2010-06-15 "Generation E går sin egen väg?"
http://www.dn.se/insidan/insidan-hem/generation-e-gar-sin-egen-vag

We currently see the personalization of mass production. The hot topics are how to facilitate one-of production scenarios (instead of mass market), that is, efficiently (with respect to resource and machine usage) produce one customized item for one customer after each other. Instead of producing the same product or service after another, one different service or product will follow another different product or service. This, as we currently see it, requires immense changes in the manufacturing industry, process optimization to facilitate cheap one-of production in generalized plants and fabrication; networked manufacturing (from initial production of raw materials, to refinement, to production, to delivery), etc., and includes the complete supply and manufacturing chain until the delivery of the good or service.

We see this individualization or socialization of production also in the rise of novel tools, allowing already the individual to produce his own goods: laser cutters, 3D printers, fab labs, etc. are getting more and more popular. Today, individuals have access to highly sophisticated manufacturing equipment. This trend in physical production is accompanied by first approaches to deliver personalized and contextualized services. Though, today's technology is not able to reliably sense or infer human contexts outside the laboratory yet.

Given current trends of e.g. cloud computing and the commercial availability of compute services for individuals (e.g. Amazon's Elastic Compute Cloud (Amazon EC2)) and extrapolating this, in several years it will be possible and economically feasible to reliably enough determine citizens' contexts and combine this with the data from the smart city. Socializing e.g. consumption (my quarter, my town) using e.g. social network data and open data, could result in a socialization again that in he end might result in more awareness. Other future trends will very certain include personal robotics, both as e.g. household helpers, but also as facilitators of social interaction. An other example could be unmanned aerial vehicles (UAV) that e.g. substitute current bike messengers, for both delivering physical goods (e.g. mail, pizza, ...) and virtual goods (e.g. call a movie – where a projector equipped UAV delivers the movie and the presentation service). In these examples again, the computational demands will be lifted to the cloud and combined there with the data from the networked distributed sensor-actuator systems.

Social Components for App Development by Empowered Citizens

This part presents a framework of social components for the Satin app development environment [49], which provides a systematic way of designing, developing and deploying social components, e.g. for social network applications. We discuss the life cycle of developing social apps (that is information applications designed for a great number of users and personalized towards specific social target groups), where the social app development environment is targeting end-users. We consider here persons that have no dedicated programming skills and more specifically have not programmed using languages such as HTML5, JavaScript or other similar scripting languages. As proof-of-concept, several social components have been deployed to the Satin Editor, which can be used to compose mobile social apps. We report on the specific results of this deployment, and extrapolate trends for social interaction in future digital cities.

Social Media

At present, there is a huge interest from the users in social media such as Facebook[10], LinkedIn[11], Google+[12], YouTube[13], Twitter[14], and others [16]. Smartphones are becoming more and more apps driven, with people using apps as specialized front end to different data sources and sinks. One of the major areas of apps development is social media, which covers different kind of communication and media

---

[10] https://www.facebook.com/

[11] https://www.linkedin.com/

[12] https://plus.google.com/

[13] https://www.youtube.com/

[14] https://www.twitter.com/

distribution needs [4, 8]. Moreover, popular social networking services such as Facebook, Twitter, Google+ are offering developer-oriented APIs to produce new apps on basis of these platforms.

End-users, though, are due to their lack of expert-level programming skills, excluded from 'developing' novel applications or personalizing existing services to their needs. This excludes an immense number of people (with other skills than programming) from contributing to social applications and services. Comparing 1,000,000,000 Facebook users to several thousand active developers highlights this imbalance. We argue that, next to user empowerment, including these people in a structured, selfdriven development process can leverage an immense potential.

Social web-based mash-ups are a means allowing for end-user to compose mash-up applications without programming dedicated knowledge [30, 58, 57]. In our proposed component-based social apps development framework, we aim to minimize this gap between traditional application developer and non-technical users with the goal to enable these end-users to develop social apps in a drag-and drop manner. For example, a component could retrieve a user's friends' birthday, process the information and e.g. compute an action, and eventually another component could sent a SMS with personalized birthday wishes. By combining these components, a user is able to form, from existing components, an novel and personalized app for this purpose. Social components show important aspects of social apps composition by offering new ways of managing contacts and initiating communication. For example, based on social components, users will be able to generate mash-ups to visualize global contacts (e.g. across several independent social networks), forming groups of people by adapting to contexts and by connecting participants, e.g. to organize a group video chat on a special topic (such as organizing a birthday present for the aforementioned friend).

Software structures are, also in other areas, evolving towards component-based architectures that support dynamic, high-level composition through wrapping and adaptors. We expect that open component libraries will be available for end-users that enable individual components to be reusable for visual end-user composition.

Another important problem in this context [42, 43, 44] is communication and social data aggregation. There are few theoretical models proposed to access social data generated from heterogeneous data sources in a unified manner. However, the proposed solutions do not cover appropriate cases where social data aggregation is essential and appropriate apps on that. By the proposed component based social apps creation framework, now it's being possible for the end-user to fuse social and communication data from different sources and develop attractive apps out of it. For example, collaborative social apps creation where collaborators from different networks need to be invited was complicated due to partition in social networking mechanism. However, compose-able component of such networks may cross that partition and invite each other to perform collaboration.

Social data is increasingly available in more and more areas of our daily life, which makes this data interesting for app developers. Data from these apps and their usage is also fed back to (groups in) social networks. Creating your own personalized apps based on the available social data by visual editing will, in the future, be offered by several platforms.

Note that the notion of a dynamic group can be used for many of these social components for media distribution [41, 18]. These dynamic groups could be created at need or be managed over time, and be used to control access to real-time collaborative applications [36]. Making components that are easily included in mobile apps support otherwise complex user management in a good way.

Social media distribution is often time critical (posing real-time demands on the delivery) and usually unpredictable beforehand (except examples e.g. from the sports domain, e.g. on a final baseball game). These circumstances inflict that it is complicated to handle, even for (temporary) homogenous social groups. Even despite the availability of a-priori knowledge on the sharing interest similarity allowing to generate patterns that define when and how users generally share content [17]. Therefore,

frequently access to a user's real-time content is very resource demanding in terms of data-capturing, or computational complexity of data processing. The research problem addressed here is: 'How does a framework support social components to capture social data and utilize those components for real-time social apps composition?'

Related Work on Social App Development and End-User Service Composition

The literature lists several graphical tools [43, 44, 46] where user can create different kind of (desktop) applications or (mobile) apps without any programming knowledge. In most cases, users can, by using the 'Drag-and-Drop' metaphor, or by simply connecting (e.g. by clicking via a direct manipulation interface such as a mouse or a tangible user interface) different visual or physical objects to create the desired apps. The source code of the components is shielded from the user, the functionality of the components are embedded within these visual or physical objects.

Liu et al. have proposed a mash-up creation architecture by extending a mash-up creating SOA models [30]. In this architecture, they present a mash-up component model allowing the user to create his own services by the composition of individual, smaller components.

Wong presented 'MARMITE', a visual tool that offers different kinds of graphical objects and allows users with no programming knowledge to create one's own mash-ups [58, 57]. Required programming scripts and logic are here, again, embedded within these graphical objects.

Web technology giants like Yahoo[15] and Microsoft[16] also provide environments for creating mash-ups by the end users. Using 'Yahoo Pipes' or 'Microsoft Popfly', end-users (with no programming knowledge) can create mash-ups. Both tools provide graphical editors where the user can drag and drop and interconnect the individual component to create the resulting mash-ups. Microsoft Popfly provides a platform where user can create apps ranging from games to small applications and eventually share these again. Ennals [11] presents the 'Intel Mash Maker' that is the web extension of the user's existing web browser. It allows the user to expand the current page that the user browsed to with additional information from other websites (which potentially poses security risks e.g. due to cross-site scripting etc.). A user can create a new mash-up and add it to the current web site with the 'Intel Mash Maker' tool. After learning about the new mash-up, 'Intel Mash Maker' suggests this new mash-up to other users. 'Intel Mash Maker' provides a platform for building and exchanging these new mash-ups. 'Intel Mash Maker' depends on the developer community who will 'teach' it about the structure and semantics of web pages (supported of course by the structuring HTML/XHTML/XML elements embedded in the pages).

Jung and Park proposed an ontology-based mash-up creation system that uses different kinds of web data sources to construct mash-ups by end-users without any programming knowledge [23]. This system proposed a 'mash-up rule language'. This system gets parameters from the users and uses a rule-based language to construct new mash-ups.

Berners-Lee and et al. proposed a platform called 'Tabulator' which links RDF data with the semantic web browser in order to create new applications based on the RDF data sources [3]. 'Tabulator' allows users to search a RDF graph in a tree structure and to browse nodes to find more information about them. An outliner mode is used which enables 'Tabulator' to create tables, Google maps, calendars, timelines, etc. Morbidoni et al. present 'Semantic Web Pipes' which is a tool to create RDF based mash-ups [35]. This tool aggregates and manipulates the content from different RDF data. This Semantic Web Pipe' can perform operations starting from straightforward aggregation to complex collaborative editing and filtering of distributed graphs.

---

[15] http://pipes.yahoo.com/pipes/

[16] http://www.popfly.ms/

Social Apps Development Lifecycle

This section describes a potential social apps development life cycle. The life cycle consists of four different conceptual steps of the component development process, namely the business model, the component development model, the composition model and evaluation model. Figure 1 depicts the iterative development flow of the proposed life cycle, which is similar to life-cycles for general system development but has an increased focus on the interaction between component developers and users/composers of apps (such as their creation of business models).
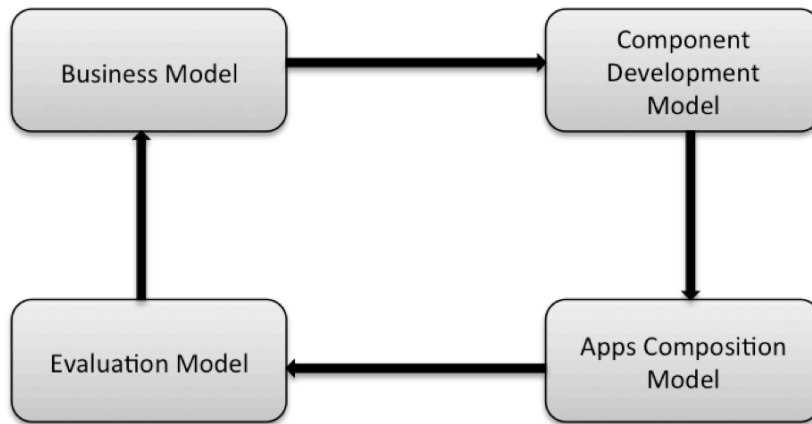

Figure 1: Social Application Development Life Cycle.

Business Model

There are different categories of mobile apps available in the various apps stores. We identified, based upon a qualitative investigation, a list of components from representative example apps in these app stores. Without loss of generalization we assume that comparable apps exist for all major platforms (such as Google's Android or Apple's iOS). When it comes specifically to social apps, their components have a lot of importance in user-centric mobile apps development. These components might include means to manipulate and process the users' social information, contact information, and communication history, and by utilizing this information, provide a means to create more personalized and contextualized mobile applications for improved communication experiences. An appropriate business model could help component developer and apps developer to understand the need for useful apps development and make use for a commercialization (e.g. by respective advertisement, or just by providing more appropriate services to the user).

Component Development Model

Components, within the context of this work, are considered as a building blocks of program logic or program code that performs a specific task in the context of social apps development. For example, a Facebook data aggregator is a component in the Satin Editor [49] that captures the user's Facebook data from this platform through the respective platform's API. In general, a black-box approach has been followed to develop the components within the Satin context. After the development, the components provide inputs/output interfaces to the users via a graphical object that can be configured using a direct manipulation interface. To perform the composition of a mobile app, the components need to be connected via their input/output labels of the components in a graphical editor. All of the processing

is hidden to the end-user. The details of the component development framework are discussed to the next section.

Apps Composition Model

The apps composition model describes how various components can be used for creating different kind of social applications. The individual social components could be used for different purposes in order to create various social apps that fulfill the user's specific requirements. A component can be composed through input/output interfaces. During the building session, the composition provides a web link to access the apps. In our apps composition model, there are several components which can be used for capturing social data, while there are other components for representing and filtering social data and finally there exist other components for visualizing data.
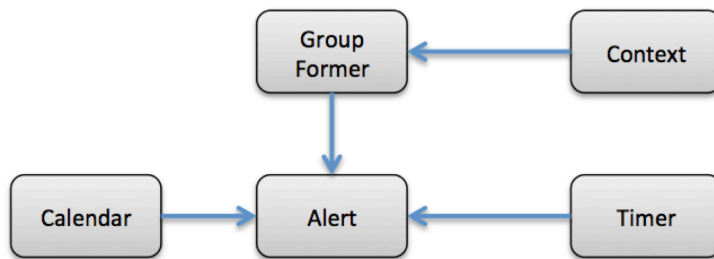
Figure 2: Apps Composition.

Figure 2 shows a high-level composition of an app for providing alerts for coffee breaks, where the blocks represent different single components. The 'Group Former' is a social data collector component that gathers and creates groups from a user's social connections based on a given context. 'Context' represents a supporting component which provides an input option for the apps user. The other three components 'Calendar', 'Alert' and 'Timer' performs specific tasks, as indicated by their name. In another scenario, a user might want an application that could help him to identify a location based on social connections. In this task, the user needs data adapters for instance for Facebook and/or LinkedIn (and/or other social network data adapters) in order to gather his social data from these data sources. A social data aggregator component will then aggregate the necessary information of these stored data. A data analyzing component will finally identify and sort the user's specific locations (e.g. Europe) based on the user's friends connected in different social networks and eventually suggest a location, e.g. to meet.

Evaluation Model

The evaluation model allows the users to study or provide feedback based on the app creation. The user's experiences of the app creation using these components will be collected here with the goal to improve the components in terms of compatibility, composition with other components, scalability, and simplicity. Moreover, in the evaluation model we have included some validation parameters. Evaluation for the proposed components will be carried out based on these parameters. Selected validation parameters are listed below:

- Social Acceptance - A user could evaluate if the social component based apps are acceptable from a societal point of view.

- Positive Affect - A user could evaluate if the social components allow for composition of any useful apps that make users' life more comfortable.

- Quality of Experience - A user could evaluate if the social apps are more useful and/or user friendly compared to commercial apps available.

- Control - A user could evaluate if he feels in control over the apps composition process.

- Ownership - A user could evaluate if the ownership of the newly generated apps remains with the user itself (and not with the platform provider).

Social Component Framework

Figure 3 shows a generalized, high-level framework for social components. It contains different layers of the social components development model that we previously described. In the lower layer, social data sources are connected to fetch the users' social networking data such as (e.g. Facebook, Twitter, LinkedIn, etc.). The middle layer provides a temporary storage of the users' social data, performs initial analysis on the data to offer different extended functionalities in the components placed on the top layer. Social data aggregation and analysis are performed in the middle layer. The social components itself constitute from the top layer.

During the component development, it is well recognized that different components have different levels of complexity (such as varying rights to use the contained data, richness of the data types and models, etc). A challenge is to express the components such that they can adapt to varying data sources, where the adoption naturally needs to be done per individual component. In the case of social components, most of the components are based on internal or external web APIs of social networks and social services (some of these APIs are based on open standards, such as OpenSocial). Therefore, we find it crucial to start with the identification what class of social components we need to develop. Primarily, two groups of classes components have been identified, which we call core components and supporting components (c.f. Figure 4).
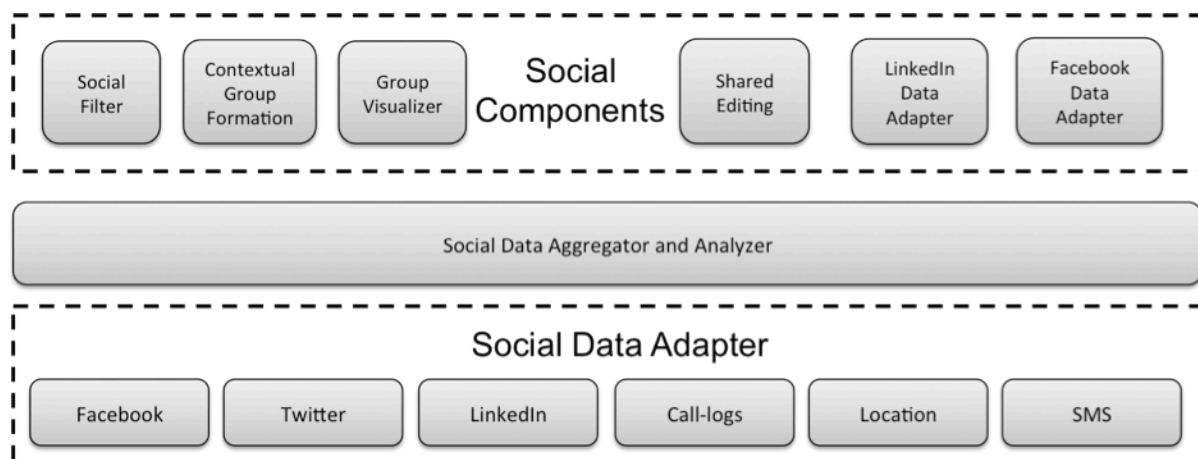
Figure 3: Social Component Framework.

Core components are the main social components such as social filters, social data adapter and so on. Many of these components have been implemented and deployed for illustration and research purposes in the Satin Editor. There are also some components that support these core components, e.g. in

triggering or labeling of web apps, which we therefore called supporting components. We find it important to provide a classification of the components to help social apps developers in assessing what is there and by providing a joint naming scheme. The classification is illustrated in Figure 4. The different types of core components that would be suitable for social apps creation are as follows:

- *Social Data Adapter* components: This class of components adapts social data from the social networking sites (which are made available through the users' credentials). The data is stored in a (computational) cloud for context reasoning.

- *Social Data Connector* components: This class of components provides an interface to communicate between data sources and other components that utilize data.

- *Social Data Processing* components: These components apply unified data representation to enhance data mining within social data components.

- *Social Data Reasoning* components: These components implement different program logic or semantic functions on the social data for the user's desired apps.

- *Visualization* components: Visualization components display different forms of social data. For example, if a processing and reasoning component forms a social group based on the user's social data, the visualization components can show this group in e.g. a grid view or a graphical tree view.

- *Smart Object* component: Smart object components provide interfaces for lightweight devices with messaging and web connectivity functionalities.

- *Messaging* component: These components provide different options of sending and receiving messages such as email, posting to social networks, SMS, and so on.

- *Location* component: These components use the location APIs and social networks location-based services, such as Checkin.

Implementation

As discussed above, different kinds of core social components along with some supporting components have been developed towards a fully working prototype in an explorative and constructive approach to investigate this important research field. The users' social data is embedded within a social data component and deployed in a platform that allows our target groups (end users without any programming knowledge) to create their own social application.

Additionally, other social components embed 'intelligence' that can exploit the social data to create social applications to provide novel services. In our case, the so-called Satin Editor is used as the test-bed for simulating and evaluating the proposed social component-based application creation environment. As the background and details of the Satin Editor are beyond the scope of this work, we would like to refer the interested reader to the research project's web site with the full documentation [49]. Social components are based on web technologies (i.e., HTML5, JavaScript, AXIS2 web Services, etc.), which enable users to run and test their applications regardless the specific type of devices used. Therefore any modern web browser (both in desktop and mobile versions) should be able to run Satin-based mobile applications.
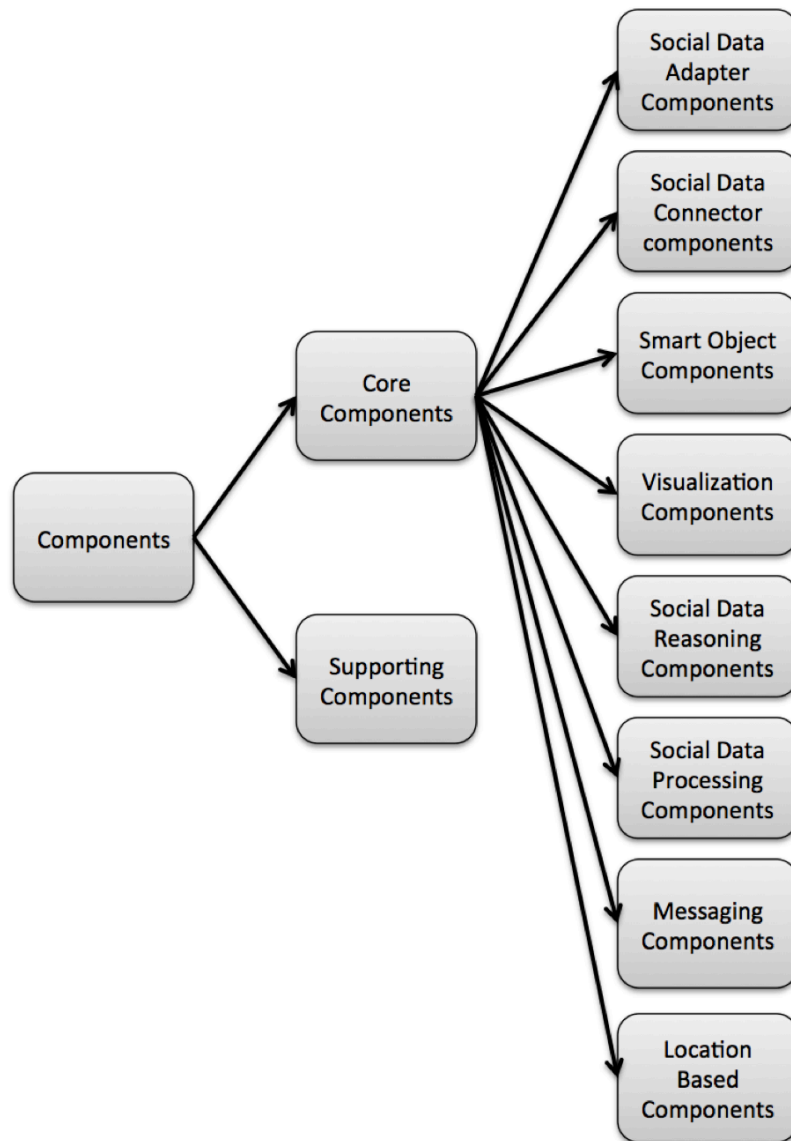
Figure 4: A classification of components for social apps development.

Data adapting (or collecting) components along with other supporting components aggregate the users' different social network(s) data. The data aggregating components 'understand' the format of the stored social data (e.g. by respective XML or similar resource descriptions) and aggregate it as a single data resource either based on context key or as a whole. Later, this data resource can be analyzed and reused by the data analyzer component in order to create personalized social applications. Data analyzer components are performing different kinds of data processing and analysis, such as the user's social interaction, his social behavior, interests, etc. Data visualization components are used to show e.g. user's aggregated social graph (a graph created from the relationships in various independent social networks, showing persons that e.g. have a temporary interest, e.g. as organizing a birthday present - c.f. the examples discussed above) either as graphical form or text form.

Different kinds of social data adapters (e.g., *LinkedIn Data Adapter*, *Facebook Data Adapter*, *Gmail Log Adapter*, etc.) have been developed and tested in order to collect users' data from these data

sources. In the implementation, all collected data from different data sources are stored in as common JSON properties. The same JSON data format is applied to other data-sources to e.g. solve data aggregation problem. Figure 5 shows an example of the data format of profiling friend's basic information through social components.

```
{
        "Friendsname": "abc" , "userid": 519817106, "username": "ab.c",
        "birthdate": "0",
        "email": "ab.c@facebook.com",
        "profileurl": "https ://www.1212121/ab.c", "movies": "",
        "interests": "",
        "picture": "https://a.akamaihd.net/hp/3 t.jpg",
        "contextkey": "p,q,r,s,t,u"
}
```

Figure 5: JSON properties for profiling friends basic information.

In this example, the *JSON properties Reader* component has been developed to parse important information from social data and developing interesting apps. Moreover, there are components that aggregate data from different sources and provide a personalized data source. To implement this, a new indexing based upon user's access in multiple social data sources is conducted. The JSON properties of the index file are then used to associate users multiple social identities and social data sources are shown in figure 6. Social data visualizer component could be used to visualize social data.

```
{"index":[{"asguser":"1323732759" ,"data":
        {"username":"1323732759" ,"socialIDs":
                {"facebookId":"1323732759" ,
                "linkedinId " : "UQxEWbiYX5" ,
                "googlePlusId " : " 109570210223523334209 " } ,
                "socialDataPaths"
                {"facebook":"../../jsonfiles/facebook/1323732759",
                "linkedin":"../../jsonfiles/linkedin/UQxEWbiYX5",
                "googleplus":"../../jsonfiles/googleplus/109570"}}}
        ]
}
```

Figure 6: JSON properties for associating user's identities in multiple social data sources.

Another social component was created which we called *Social Data Filter*, which filters users, aggregates social data based on the filtering parameters. For instance this *Social Data Filter* component could be used to create group with the users social connections based on users' interest. There are also other social components that are used to share user social resources with his/her connections.

Figure 7 shows the composition of an application in the Satin Editor. In this example one supporting component is used with three social components such as *Facebook Data Adapter*, *JSON Reader* and *Social Data Viewer*. Users can drag and drop the components in the canvas of the Satin Editor and compose the application. This application is used to collect the user's Facebook friend's data and view the whole collected data to analyze it further or to use this information with other components to create further apps. In a similar way, users can use different social data collector components to amass his/her social connections' information from different social data sources.
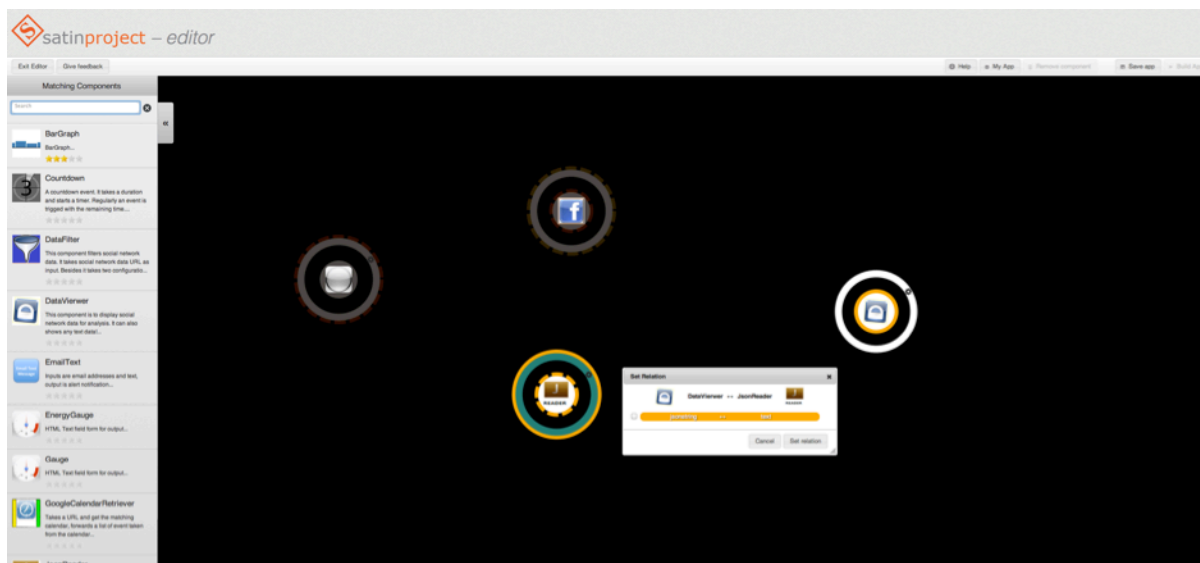

Figure 7: Sample Social Apps Composition in the Satin Editor.

Evaluation

The ultimate aim of this work is that any user after a 15-20 minutes introduction should be able to easily create simple social apps on their own. The objective of this study was however to indicate whether this would be achievable at all, as even visual composition can be challenging for normal users as data paths and sequential dependencies still are hard to understand, and thus provide feedback on social apps development for the next iteration of the Satin Editor. A small user study was therefore initiated, which was based on the evaluation model and which was limited to 10 users with some prior knowledge of mobile apps (they were for instance required to at least have used apps before). The limited set of users and their a-priori knowledge makes the results no more than indicative, but they are still very valuable as feedback for the next generation of the Satin Editor. To conduct the study, we prepared three different scenarios (described below) of apps development. Before the users start with the app composition, the available components for social apps were introduced to the users by providing written descriptions of the components, as well as demonstrating apps composition using the Satin Editor. In this subsection, description of the scenarios is presented.

Scenarios

The following three scenarios of app compositions were used to evaluate the described approach and concepts in the user study:

Scenario 1: Bob is planning an outdoor party in a newly explored and interesting place. He wants to invite all his Facebook friends to join the party from wherever they are. Thus, Bob intends to publish an app to his Facebook timeline in order to invite his friends, as well as a to provide a map to direct them to the party place in a comfortable way.

Scenario 2: Alice wants to know which places are of most interest to her. So, she creates an app that automatically logs her current location if she stays at a place for more that 30 minutes. At the end of the month, she is provided information about her most frequently visited places and potentially a respective visualization.

Scenario 3: Charlie is going for a coffee break, and he would like to send an alert message to his co-workers so that they can have the break at the same time.

Data Collection

The following parameters have been taken into consideration during the test for data collection:

- App composition time: The time that a user has spent to compose an app on a given scenario.

- Number of components: The number of components that have been selected to perform the final composition.

- App formation: The user is able to build an apps and able to run the apps

- App functionality: Logs whether the app that are composed by the users is functioning correctly with respect to the scenarios described above.

Figure 8 shows the snapshot of the apps generated during the user tests. Figure 8 (a) shows the apps based on scenario 1. Figure 8 (b) shows apps based on scenario 2 and Figure 8 (c) is based on scenario 3. The functionalities are not fully compliant with the described scenarios. For example, apps for the scenario 1 share the invitation through the Facebook 'Like' operation, which could also be done with other options.
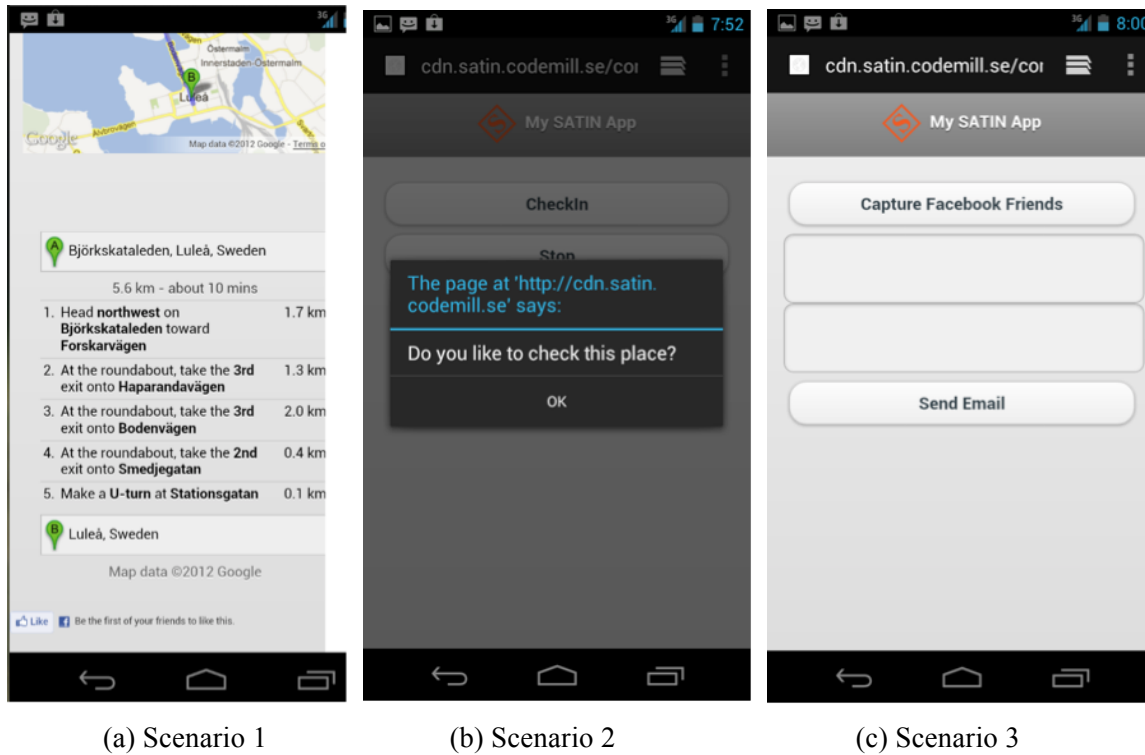
(a) Scenario 1   (b) Scenario 2   (c) Scenario 3

Figure 8: Social Apps generated by the Satin users for scenarios 1 to 3.

Evaluation Results

Figure 9 shows the users' ratings for evaluating social apps as gathered in the user study. The ratings are taken within the scale of 1 to 5, where 1 is the most negative response while 5 is the most positive response. In general, from most of users we received positive responses.

Figure 10 shows time durations for social apps composition. Although there are significant amount of assistance have been given before or during apps composition to the users, however the time duration varies highly from user to user. Average time is calculated to 29.3 minutes required per user to compose apps based on the given scenarios.
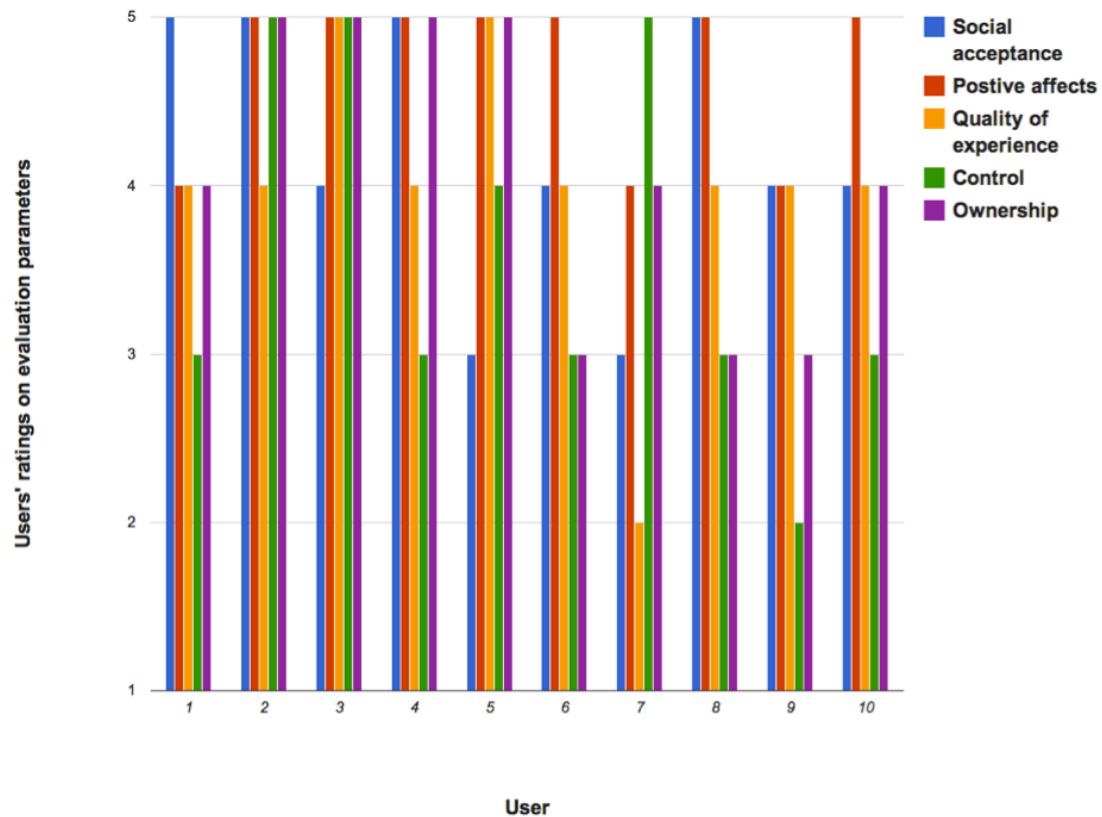
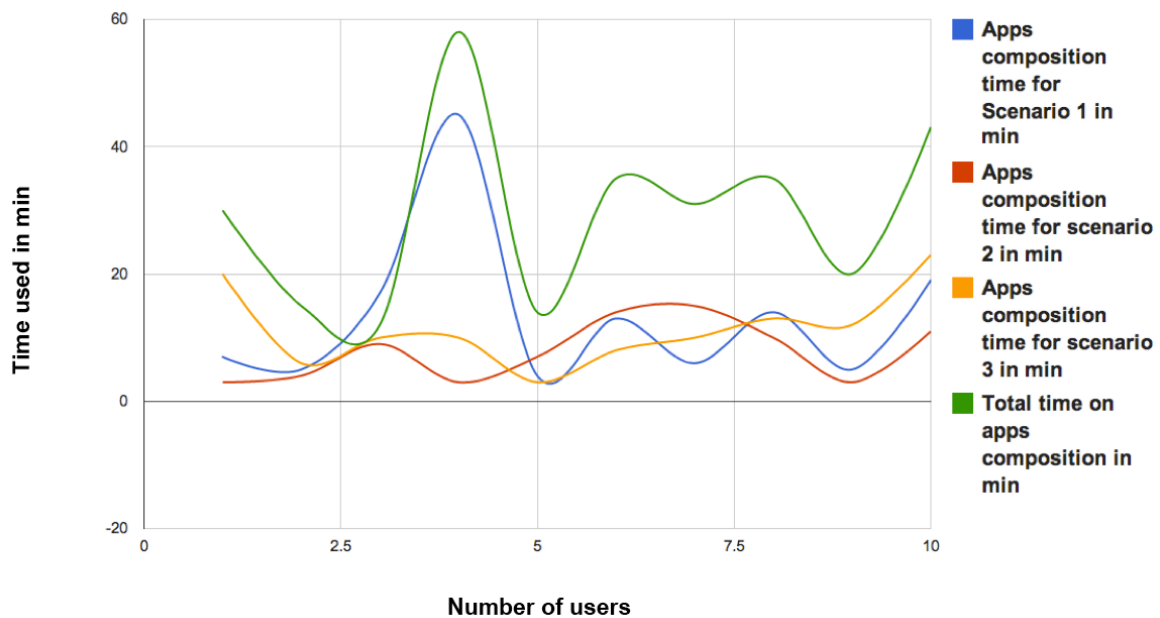Figure 9: Users' rating for evaluating social apps.

Figure 10: Time durations for social apps composition.

Moreover, we collected the users' individual opinions based on the described parameters from the evaluation model such as social acceptance, positive affect, quality of experience, control and ownership. Some of the general problems that we have identified are common amongst most of users during apps development. As of now, they are not comfortable enough with the editor environment, and they need more support to identifying appropriate components to accomplish functionalities. However, those are not directly connected with social components, but valid comments to fix those before re-running the study for larger user group and a more mature version of the Satin Editor. The positive impact that we got that after being to able successfully apps creation, the users are being relaxed and appreciates the environment as well as apps. Overall, the approach has been validated as effective and the participants of the user study have rated the concept positively.

Discussion and Future Work

We have proposed components for social app development environment via a high-level, four-phase social app development life cycle. Each of the phases of life cycle has been briefly discussed. The business model does not provide any technological challenges, however it describes the social components from the business and usability point of view. The business model is important, as it will help to understand the users' needs and wishes and thus be an important driving factor.

The component development model mainly addresses social aspects of the app development. It identifies different core components and supporting components to compose social apps. We also provided a classification of components to provide developers with an overview of what kind of components that could add value to the mobile apps development environment. We argue that there are huge domains of components that need to be developed and classified for diverse apps development, and that is one of the targets for the future work.

From the user study we identified the users' difficulties to understand the composition scheme of the Satin Editor, however having initial support from a instructors, users could soon get proficient in app composition. By our validation model, we achieved useful feedback from the users even though the scale of user study was limited.

The research question addressed in this work ('How can be the user's social and communication data captured from the social and communication data sources be utilized to compose real-time social apps?') considers social data as one of the important area of component development. Facebook and LinkedIn data components got users attractions as they being able to develop social components based on those. We have shown that the social component framework provides a standard way of developing social components to capture data from social media sources. It also shows different kinds of social data collector components developed in Satin platform. The framework could be used as model for other social component developer and could adapt our JSON properties to make their components possible to compose with Satin-based social data components. Thus, new component developers may be benefitted to design and implement their social component in Satin environment. Another aspect of future works is to cover different domains of apps developments such as games, mobile OS based native apps, and so on.

The challenge will be creating an open platform for components-based visual design of apps requires standardization or an industrial de-facto. For user-composed and personalized services to become ubiquitous, the challenge is to create such an open platform based on recognized (de-facto) standards.

Due to use of smart devices, social apps are becoming part of everyday life. Social apps could be very beneficial in social media distribution, group formation, lightweight collaboration and so on. If users being able to adapt social components to build their own apps, then it will open up a new and efficient way of social development. The user of the Satin Editor logically does not require any programming knowledge to develop mobile apps. From the user study and deployed social components in the Satin

Editor, it show that proposed social component development framework could be used to develop social component using different kind of social networks and social media data. The social component development life cycle and classification of component may help other developer to develop useful components to generate not only social components but also diverse type of mobile apps in general.

A key technological challenge is to manage to link openly available software components with smart devices and tangible artifacts. This challenge builds on open and standardized APIs.

Our recommendation is to study component-based visual editing of apps that utilize open data, social data and personal preferences, such that personalized apps can be easily constructed by any end-user for use with a smart device.

Summary and Discussion

We have presented three areas of research necessary to achieve personal, social and urban awareness through pervasive computing. The argument made is that pervasive computing services are likely to be driven by user-needs, where three pillars enable these services. These are free access to open data, novel interaction techniques based on the social web of things and the vast potential behind end-users able to easily visually compose mobile/pervasive apps. In other words, that awareness is accomplished by personalizing apps through easy visual composition of components based on open (urban) data connected to social facets that allows for effective filtering, prioritization and recommendation of information and services. What drives the next age of computing is then highly personalized services that harness information from the social web of things.

The vision of pervasive computing is thus likely to be achieved, at least in part, not by inherent complex architectures and services but on simple building blocks that just about anyone can combine into useful services. These services are naturally first deployed in mobile devices, but is likely later also deployed into smart environments when available. Techniques such as tangible computing may then bloom from the few applications available today, to a full range of personalized applications applied to various areas of society such as teaching tools for children, smart homes for elder care, environments for social and tangible communication, etc.

The path towards this development can be plotted by currents trends in this area, which also highlight challenges to overcome and thus recommendations for how to achieve the vision above. A few trends, challenges and recommendations based on the work above is identified in the following sections.

Trends

- Software structures are constantly evolving towards component-based architectures that support dynamic high-level composition through wrapping and adaptors. By 2016, open component libraries will be available that enable single components to be reused for visual end-user composition.

- Social data is increasingly interesting for app developers, so is feeding back information to (groups in) social networks. By 2018, creating your own personalized apps based on social data by visual editing will be offered by several platforms.

- Personalized Physical and Digital Services: Individualization of the members of the society will further increase the demand for personalized services, extending significantly the current services that are designed for the mass.

- The amount of data sensed (and available for control) will grow exponentially. This trend is mainly grounded in the development of the so-called Internet of Things and results in many challenges regarding capturing, filtering, storing, managing and utilizing Big Data.

Challenges

- Creating an open platform for components-based visual design of apps requires standardization or an industrial de-facto. For user-composed and personalized services to become ubiquitous, the challenge is to create such an open platform based on recognized (de-facto) standards.

- A key technological challenge is to manage to link openly available software components with smart devices and tangible artifacts. This challenge builds on open and standardized APIs.

- Provision of a European-wide legal framework for sensing and open data: future smart cities will be providing and collecting a lot data to and on their citizens. Before this data can be used as basis for novel services, clear rules have to be defined on the extent this data may be used and how e.g. the privacy of the digital citizens can be ensured.

- Physical and Digital Rural Depopulation: The size of cities will further grow. Today already more than half of the population lives in cities. In the future, more and more people will live in connected mega cities. The rural areas will need to develop services that, if not stopping this trend, make it at least convenient and possible for the remaining population to stay there. This will most probably concern elderly people that are more reluctant to move, but have special needs and demands, such as connected healthcare.

- The trend for individualization of inhabitants of mega cities requires the society to take countermeasures against a future 'digital loneliness'. The data and digital cities will need to use the digital data to create spaces that foster physical real cooperation, co-living and interaction. This can potentially also support virtual communities, sprung from a common need or understanding, that can offer the benefits of smaller communities in a mega city (digital villages).

- Social interaction and responsibility should, using the available data of the connected artifacts, be employed to increase the participation of the digital citizens in all social and societal issues, e.g. co-governance or to-be-developed services like FixMyDigitalCity 2.0.

Recommendations

- Study component-based visual editing of apps that utilize open data, social data and personal preferences, such that personalized apps can be easily constructed by any end-user for use with a smart device.

- Use Sweden, due its leading position with respect to widespread availability of high-bandwith internet connectivity (esp. in rural areas), the availability of electronic IDs for digital services, and spearhead projects for digital cities, like in Skelleftea˚, Sweden, as a test-bed for Europe due to the widespread acceptance and adoption of new technology

- The lack of a common legal framework across the borders of the EU member states will be hindering the development of future digital city services. As can be foreseen now, most probably it will be multi-national companies that will be developing these novel services, due to the challenges from big data and the from the Internet of Things. The associated costs are higher due to the different existing legislative frameworks. A 'legal standardization' will both provide security for the investments and also be valuable for the citizens as users as clear rules will have to ensure their rights, esp. privacy.

- Standardization of middleware and service APIs allowing the interconnection of services, data, etc., while ensuring trust, authenticity and privacy. Application areas benefitting from this range from e-government, citizen service, social interaction to novel business opportunities. This will include the need for modeling and developing a 'transparency' layer for open/big/user centric data in future digital cities and economies.

Conclusions

End-users in Digital Cities able to compose personalized services within a short span of time through component-based visual composition will drive the development of truly pervasive computing services, first deployed in smart phones and later in smart environments. Components will leverage of open data from both urban and private environments as well as on the advent of novel interaction techniques such as tangible interaction devices. Personal, social and urban awareness through pervasive computing is thus accomplished by a wide variety of novel personalized services, based on users' needs.

Bibliography

[1] Aloisio, G., Conte, D., Elefante, C., Epicoco, I., Marra, G. P., Mastrantonio, G., and Quarta, G. Sensorml for grid sensor networks. In Proceedings of the 2006 International Conference on Grid Computing & Applications, GCA 2006, Las Vegas, Nevada, USA, June 26-29, 2006, H. R. Arabnia, Ed., CSREA Press (2006), 147–152.

[2] Atzori, L., Iera, A., and Morabito, G. The internet of things: A survey. Comput. Netw. 54, 15 (Oct. 2010), 2787–2805.

[3] Berners-Lee, T., Hollenbach, J., Lu, K., Presbrey, J., Prud'hommeaux, E., and Schraefel, M. M. C. Tabulator redux: Browsing and writing linked data. In LDOW, C. Bizer, T. Heath, K. Idehen, and T. Berners-Lee, Eds., vol. 369 of CEUR Workshop Proceedings, CEUR-WS.org (2008).

[4] Böhmer, M., Hecht, B., Schöning, J., Krüger, A., and Bauer, G. Falling asleep with angry birds, facebook and kindle: a large scale study on mobile application usage. In Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services, MobileHCI '11, ACM (New York, NY, USA, 2011), 47–56.

[5] Botts, M., and Robin, A. Sensor Model Language (SensorML). http://www.ogcnetwork.net/SensorML, 2009.

[6] Caragliu, A., Del Bo, C., Kourtit, K., and Nijkamp, P. Performance of smart cities in the north sea basin. http://www.smartcities.info/files/13%20-%20Peter%20Nijkamp%20-%20Performance%20of%20Smart%20Cities.pdf, 2009.

[7] Castellani, A. P., Bui, N., Casari, P., Rossi, M., Shelby, Z., and Zorzi, M. Architecture and protocols for the Internet of Things: A case study. In PerCom 2010 Workshops: Proceedings of the 8th Annual IEEE International Conference on Pervasive Computing and Communications Workshops, Mannheim, Germany (2010), 678–683.

[8] Cui, Y., and Honkala, M. The consumption of integrated social networking services on mobile devices. In Proceedings of the 10th International Conference on Mobile and Ubiquitous Multimedia, MUM '11, ACM (New York, NY, USA, 2011), 53–62.

[9] Diewald, S., Möller, A., Roalter, L., and Kranz, M. Mobile Device Integration and Interaction in the Automotive Domain. In AutoNUI: Automotive Natural User Interfaces Workshop at the 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications (AutomotiveUI 2011) (Nov.–Dec. 2011).

[10] Diewald, S., Möller, A., Roalter, L., and Kranz, M. MobiliNet: A Social Network for Optimized Mobility. In Adjunct Proceedings of the 4th International Conference on Automotive User Interfaces and Interactive Vehicular Applications (AutomotiveUI 2012) (Oct. 2012), 145–150.

[11] Ennals, R., Brewer, E., Garofalakis, M., Shadle, M., and Gandhi, P. Intel mash maker: join the web. SIGMOD Rec. 36, 4 (Dec. 2007), 27–33.

[12] Enokido, T., Xhafa, F., Barolli, L., Takizawa, M., Uehara, M., and Durresi, A., Eds. The 13th International Conference on Network-Based Information Systems, NBiS 2010, Takayama, Gifu, Japan, 14-16 September 2010, IEEE Computer Society (2010).

[13] Ericsson. The Internet of Things Comes Alive through Smart Objects Interoperability. http://labs.ericsson.com/, 2012.

[14] Ericsson. Vision 2020 – 50 Billion connected devices. http://www.slideshare.net/EricssonFrance/vision-2020-50-billion-connected-devices-ericsson, 2012.

[15] Erickson, T. Geocentric crowdsourcing and smarter cities: Enabling urban intelligence in cities and regions a position paper for the 1st international workshop on ubiquitous crowdsourcing.

[16] Faloutsos, C., and Kang, U. Managing and mining large graphs: patterns and algorithms. In SIGMOD Conference, K. S. Candan, Y. Chen, R. T. Snodgrass, L. Gravano, and A. Fuxman, Eds., ACM (2012), 585–588.

[17] Gilbert, E., and Karahalios, K. Predicting tie strength with social media. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '09, ACM (New York, NY, USA, 2009), 211–220.

[18] Hallberg, J., Norberg, M. B., Kristiansson, J., Synnes, K., and Nugent, C. Creating dynamic groups using context-awareness. In Proceedings of the 6th international conference on Mobile and ubiquitous multimedia, MUM '07, ACM (New York, NY, USA, 2007), 42–49.

[19] Ishida, T. Understanding digital cities. In Digital Cities, Technologies, Experiences, and Future Perspectives [the book is based on an international symposium held in Kyoto, Japan, in September 1999, Springer-Verlag (London, UK, UK, 2000), 7–17.

[20] Ishida, T. Digital city Kyoto. Communications of ACM 45, 7 (July 2002), 76–81.

[21] Johnston,E.W.Governanceinfrastructuresin2020.PublicAdministrationReview70,1(2010), 122–128.

[22] Johnston, E. W., and Hansen, D. L. Design lessons for smart governance infrastructures. In American Governance 3.0: Rebooting the Public Square?, D. Ink, A. Balutis, and T. Buss, Eds. National Academy of Public Administration, 2005.

[23] Jung, H., and Park, S. Mash-up creation using a mash-up rule language. J. Inf. Sci. Eng. 27, 2 (2011), 761–775.

[24] Kapadia, A., Myers, S., Wang, X., and Fox, G. Secure cloud computing with brokered trusted sensor networks. In CTS, W. W. Smari and W. K. McQuay, Eds., IEEE (2010), 581–592.

[25] King, S. F., and Brown, P. Fix my street or else: using the internet to voice local public service concerns. In Proceedings of the 1st international conference on Theory and practice of electronic governance, ICEGOV '07, ACM (New York, NY, USA, 2007), 72–80.

[26] Kranz, M., Murmann, L., and Michahelles, F. Research in the Large: Challenges for Large- Scale Mobile Application research - A Case Study about NFC Adoption using Gamification via an App Store. IJMHCI (2013).

[27] Kranz, M., Roalter, L., and Michahelles, F. Things That Twitter: Social Networks and the Internet of Things. In What can the Internet of Things do for the Citizen (CIoT) Workshop at The Eighth International Conference on Pervasive Computing (Pervasive 2010) (May 2010).

[28] Kranz, M., and Schmidt, A. Prototyping Smart Objects for Ubiquitous Computing. In Proceedings of the International Workshop on Smart Object Systems in Conjunction with the Seventh International Conference on Ubiquitous Computing (Sept. 2005).

[29] Kranz, M., Schmidt, A., and Holleis, P. Embedded interaction: Interacting with the internet of things. IEEE Internet Computing 14, 2 (March-April 2010), 46–53.

[30] Liu, X., Hui, Y., Sun, W., and Liang, H. Towards service composition based on mashup. In IEEE SCW, IEEE Computer Society (2007), 332–339.

[31] McGregor, A., and Bormann, C. Constrained RESTful Environments (CoRE). Tech. rep., Internet Engineering Task Force (IETF), 2012.

[32] McGregor, A., and Bormann, C. IPv6 over Low power Wireless Personal Area Networks. Tech. rep., Internet Engineering Task Force (IETF), 2012.

[33] Michahelles, F., Kranz, M., and Mandl, S. Social Networks for People and Things (SoNePT). http://www.theinternetofthings.eu/social-networks-people-and-things-sonept, 2012.

[34] Möller, A., Michahelles, F., Diewald, S., Roalter, L., and Kranz, M. Update Behavior in App Markets and Security Implications: A Case Study in Google Play. In Proceedings of the 3rd International Workshop on Research in the Large. Held in Conjunction with Mobile HCI, B. Poppinga, Ed. (Sep 2012), 3–6.

[35] Morbidoni, C., Le Phuoc, D., Polleres, A., Samwald, M., and Tummarello, G. Previewing semantic web pipes. In Proceedings of the 5th European semantic web conference on The semantic web: research and applications, ESWC'08, Springer-Verlag (Berlin, Heidelberg, 2008), 843–848.

[36] Parnes, P., Synnes, K., and Schefström, D. Real-time control and management of distributed applications using ip-multicast. In Integrated Network Management, 1999. Distributed Management for the Networked Millennium. Proceedings of the Sixth IFIP/IEEE International Symposium on (1999), 901–914.

[37] Poolsappasit, N., Kumar, V., Madria, S., and Chellappan, S. Challenges in secure sensor-cloud computing. In Proceedings of the 8th VLDB international conference on Secure data management, SDM'11, Springer-Verlag (Berlin, Heidelberg, 2011), 70–84.

[38] Project Consortium. SENSEI - Integrating the Physical with the Digital World of the Network of the Future. http://www.ict-sensei.org/, 2008.

[39] Project Consortium. Smart Santander - Future Internet Research & Experimentation. http://www.smartsantander.eu/, 2008.

[40] Project Consortium. European Sensor Network Architecture (ESNA). https://www.sics.se/esna/, 2009.

[41] Rana, J., Kristiansson, J., Hallberg, J., and Synnes, K. Challenges for mobile social networking applications. In First international ICST Conference on Communications, Infrastructure, Systems and Applications in Europe (EuropeComm 2009), London, UK, August 11-13, 2009, Revised Selected Papers (2009).

42] Rana, J., Kristiansson, J., and Synnes, K. Enriching and simplifying communication by social prioritization. Social Network Analysis and Mining, International Conference on Advances in 0 (2010), 336–340.

[43] Rana, J., Kristiansson, J., and Synnes, K. Enriching and simplifying communication by social prioritization. In Proceedings of the 2010 International Conference on Advances in Social Networks Analysis and Mining, ASONAM '10, IEEE Computer Society (Washington, DC, USA, 2010), 336–340.

[44] Rana, J., Kristiansson, J., and Synnes, K. Modeling Unified Interaction for Communication Service Integration. In UBICOMM 2010, The Fourth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies, IARIA (October 2010), 373–378.

[45] Roalter, L., Kranz, M., and Mo¨ller, A. A middleware for intelligent environments and the internet of things. In Ubiquitous Intelligence and Computing, Z. Yu, R. Liscano, G. Chen, D. Zhang, and X. Zhou, Eds., vol. 6406 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2010, 267–281. 10.1007/978-3-642-16355-523.

[46] Roalter, L., Mo¨ller, A., Diewald, S., and Kranz, M. Developing Intelligent Environments: A Development Tool Chain for Creation, Testing and Simulation of Smart and Intelligent Environments. In Proceedings of the 7th International Conference on Intelligent Environments (IE) (july 2011), 214–221.

[47] Ro¨ckl, M., Gacnik, J., Schomerus, J., Strang, T., and Kranz, M. Sensing the environment for future driver assistance combining autonomous and cooperative appliances. In Fourth International Workshop on Vehicle-to-Vehicle Communications (V2VCOM) 2008 (June 2008), 45–56.

[48] Rusu, R. B., Maldonado, A., Beetz, M., Kranz, M., Mo¨senlechner, L., Holleis, P., and Schmidt, A. Player/Stage as Middleware for Ubiquitous Computing. In Proceedings of the 8th Annual Conference on Ubiquitous Computing (Ubicomp 2006) (Sept. 2006).

[49] SATIN2 Project. SATIN Editor. http://satinproject.eu/, 2012.

[50] Schmidt, A., Kranz, M., and Holleis, P. Embedded Information. Workshop Ubiquitous Display Environments at UbiComp 2004 (Sept. 2004).

[51] Schmidt, R. K., Leinmüller, T., and Böddeker, B. V2x kommunikation. In In Proceedings of 17th Aachener Kolloquium 2008 (2008).

[52] Schmidt, R. K., Leinmüller, T., Schoch, E., Kargl, F., and Schäfer, G. Exploration of adaptive beaconing for efficient intervehicle safety communication. IEEE Network Magazine, Special Issue on Advances in Vehicular Communications Networks 24 (2010), 14 – 19.

[53] Shneidman, J., Pietzuch, P., Ledlie, J., Roussopoulos, M., Seltzer, M., and Welsh, M. Hourglass: An infrastructure for connecting sensor networks and applications. Tech. rep., Harvard, 2004.

[54] Tsiftes, N., and Dunkels, A. A database in every sensor. In Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems, SenSys '11, ACM (New York, NY, USA, 2011), 316–332.

[55] Virtanen, T., and Malinen, S. Supporting the sense of locality with online communities. In MindTrek 2008, International Digital Media & Business Festival (October 2008).

[56] Weiser,M. The computer for the 21st century. Scientific American 265, 3 (January1991), 66–75.

57] Wong, J. Marmite: Towards end-user programming for the web. In VL/HCC, IEEE Computer Society (2007), 270–271.

[58] Wong, J., and Hong, J. I. Making mash-ups with marmite: towards end-user programming for the web. In CHI, M. B. Rosson and D. J. Gilmore, Eds., ACM (2007), 1435–1444.

[59] Yuriyama, M., and Kushida, T. Sensor-cloud infrastructure - physical sensor management with virtualized sensors on cloud computing. In Enokido et al. [12], 1–8.