

# Vehicular Safety Platform Integrating MCC Based Emergency Context-Aware Management Architecture Enhanced With Crowdsourcing Filter

D. M. S. Sultan

Blekinge Institute of Technology, Sweden  
sultan@alumni.chalmers.se

## **Abstract**

*This paper aims to propose a realization of a new vehicular safety platform integrating emergency context-aware management (ECM) platform utilizing Mobile Cloud Computing (MCC) concept with efficient manipulation of wide spatial data using crowdsourcing filter (viz. SwiftRiver). ECM platform is now a hot research arena that is mainly focused upon emergency issues within environmental disaster. Interestingly, ECM platform may also be useful in vehicular safety like in deep winter in Scandinavian countries where extreme snow storm causes significant accidental occurrences every year. Here, the proposed platform is aimed to add a new dimension in vehicular safety utilizing simple internet connectivity within GPS enabled smart phones where every user within the platform will act as a sturdy source of emergency vehicular safety context.*

*Keywords: Vehicular Safety, Mobile Cloud Computing, Emergency Context Management, Crowdsourcing.*

## **1. Introduction**

Hazardous weather conditions might have a great influence on transportation sectors. Heavy snowfall and strong windstorm are common nature in these Scandinavian countries in deep winter that causes serious disturbance in road and rail

traffic. Due to the dense snowfall at 17<sup>th</sup> March, 2005 in Finland, 300 vehicles were in road crash with a severe damage of 3 people death and around 60 people's injuries [1]. In another study, it has shown that winter's induced road slippery caused 50% of all traffic in western Sweden [2]. The southern Sweden becomes more accident prone while there is rain or sleet drops upon frozen road surface. The combined effect of both dense snowfall and windstorm caused 824 accidental occurrences with 112 injuries [1]. Similarly, hoarfrost formation along snowfall also introduces severe hazardous road condition. In these cases, it is quite clear that a proven emergency context - aware management (ECM) platform is needed to mitigate the devastating phenomena to a large extent. The proposed ECM architecture here aims to provide robust responses at such emergency situations.

Sensor networking, VANET, ZigBee like several technologies is out there to mitigate the risk ratio in transportation. A continuous research for making more robust framework on vehicular safety issues is being held in collaboration with both industry expert and academy. The proposed platform here is aiming to improve the same safety issues but with integration of Mobile Cloud Computing (MCC) along SwiftRiver like crowdsourcing filter. The framework is expecting the presence of smart mobile stations along GPS and internet communication functionality at user end. In accordance to the latest report, the total mobile subscriber is about 4.6 billion [3]

around the globe where smartphone users among them is going to be around 468 million [4] by year 2011. It is also going to be exponentially increased by the time being. Hereby, it indicates a sound ground and feasibility of proposed framework deployment in current world. Besides, cloud computing platforms (i.e. Microsoft Azure, Amazon EC2 and Google AppEngine etc.) have shown a proven alternative platform of mass information processing and the possibility of integration of mobile devices as client and resource provider both [5]-[8],[9]. On an application of next generation vehicular safety communication, mobile cloud computing will be handy one to build emergency context-aware architecture for providing efficient swift and authentic action for robust and resourceful emergency responses.

Literally, context is any situation what can be used to characterize the situation of a person, place or object that is considered relevant to the interaction between a user and an application, including location, time, activities, and the preferences of each entity. And Context-awareness means that one is able to use context's information. Hereby, a system can be context-aware if it can extract, interpret and use context information and adapt its functionality to the current context of use [10]. The term context-aware computing is commonly understood by those working in context-aware, where it is felt that context is a key in their efforts to disperse and transparently weave computer technology into our lives. The goal of this paper is to comprise context-aware systems which will be used to acquire and utilize information (obtained from mass people used smart phones' GPS interface with internet connectivity within it) and to take swift, authentic and efficient action on vehicular emergency responses. The proposed architecture also includes an efficient manipulation of collected data that will be routed through mobile cloud infrastructure (driven by mobile operators or with third party cloud computing alliances) inheriting

SwiftRiver like intelligent crowdsourcing filter within the system.

## **2. Crowdsourcing Data and Mobile Cloud Computing-Based Services**

### **A. Crowdsourcing**

Crowdsourcing concept is literally outsourcing tasks, traditionally performed by an employee or contractor, to a large group of people or community (a crowd), through an open call [11]. Utilizing the concept, the mass correspondences (who will carry out the large amounts of emergency information) can be invited to be the part of the proposed platform. Those conveyed emergency contexts could be refined at later step following the systematic and analytic steps of crowdsourcing filter (viz. SwiftRiver) enabled at Web 2.0 technological interface. It would be quite natural to face unprecedented challenges in handling large amount incoming crowd-sourced data systematically and analytically within any emergent platforms except using crowdsourcing filter.

### **B. Mobile Cloud Computing-based Services**

Cloud services are the form of centralized web-based applications where applications and files are hosted on a "cloud" that is consisted of thousands of mobiles, computers and servers (where all are linked together and accessible via the Internet). From the point of resource service, ECM system may have thousands users and sensors (that run specialized software to acquire data or carry out data intensive tasks) in numerous locations. Recent evolutions of crowdsourcing information acquisition through MCC technology has led to a more distributed information process and computing environment. Through the currently evolved data and telecommunication network, MCC services are capable to obtain the necessary services easily by the on-demand. In proposed ECM

platform, the following advantages could be explored:

- *Resource virtualization:* MCC services support users' interaction to the system from heterogeneous wireless network platforms. The requested resources in ECM (i.e. emergency context web-service access, emergency context information web-service storage and other resources access & storage) may come from the cloud resource architecture rather than fixed entity resources. A client user won't need to know the specific location of the backend applications running at MCC interface except providing emergency contexts.

- *Large-scale pattern and high reliability:* MCC services are suitable to handle very large-scale scientific computation of a considerable data volume (i.e. The giant Google's cloud computing already has more than 100 million servers [12] to serve distributed data processing around the globe). Cloud computing based services may provide a reliable and safe storage centers for spatial data processing. These data-centers care the users' data loss by providing multiple replica of fault-tolerant, isomorphism of the computing nodes and interchangeable measurement. Besides, strict rights management strategy integrated in MCC helps to achieve data sharing facility.

- *Generality and high scalability:* MCC platform is capable of evolving multi-pattern applications under the cloud service support. Besides, the dynamic scalability feature of MCC met it to be a right candidate to fit the demand of the emergency management applications and the scale of users' growth.

- *Low-cost and on-demand services:* MCC services are capable of providing different types of services and so, the proposed EMC platform here can easily inherit the desired ones by selective or customization process among all offered services. Low-cost nodes used in MCC will help to measure special fault-tolerant and automated centralized management based MCC services.

### 3. Vehicular ECM System



Fig. 1: Schematic integration of MCC based ECM system for vehicular emergency context.

In the proposed architecture, it is also expected that a user will use a GPS receiver and internet facility enhanced Mobile Station (MS) where a customized client-end GPS application is already integrated. Within the application, an observer will see an additional triangular icon in 2 Km distances ahead from its current location (though the distance may vary by developer choice) as in Fig. 1, GPS application UI. Now, if an emergency scenario (i.e. accident or large tree fallen or ice blockage due to dense snow fall and storm on highway or railway) occurs, the observable driver or people will convey emergent context instantly just by pressing Yellow button or Red button shown at UI interface as in Fig. 1.

A concurrent information transmission will be made using mobile broadband service supported by mobile operator or wireless CPE (supported by any WAN provider). As soon as a press will be on Yellow or Red input button by an observer, the corresponding longitude and latitude information will be sent to MCC core. If the conveyed input seems feasible and verified by SwiftRiver, a Yellow triangle or Red triangle will be placed on that specific coordinate and will be highlighted for all vehicles (those are behind and subscribed into the same cloud architecture). Some

constraints have to be considered during the development of such customized GPS UI application especially in assignation of Yellow or Red triangle icon. As soon as, Yellow or Red triangle is appeared to UI at a vehicle at 2 Km ahead, the Green triangle (default) should be vanished until the vehicle reaches to the specific coordinate. Here, Yellow triangle will represent the probable danger ahead where the driver should take necessary steps for safety and Red triangle will represent severe danger ahead at that coordinate point where driver should stop or take other subway to avoid the danger.

While mass people will act as a strong source of emergency contextual input, many of them could be false or crime motivated, which may cost in terms of unnecessary traffic jam or collision and hijacking. Thereby, incoming crowd-sourced information have to be verified within an automated and manual intelligent filter system both before being finally adopted by the proposed platform. SwiftRiver like crowdsourcing filter is capable handling such challenges in efficient way. A proven example of this SwiftRiver's successful automation could be seen in Ushahidi project that was applied in gulf oil spill crisis in Kenya [14]. It is expected that the proposed ECM system enhanced with crowdsourcing filter (viz. SwiftRiver) will enrich a new dimension to overcome vehicular emergency challenges by proficient data filtration and swift awareness with centralize collaboration of MCC platform.

#### 4. Detailed Framework of Proposed Emergency Context-Aware Management

The proposed ECM Platform is aimed for efficient manipulation of large scale of incoming context-aware contexts [15]. ECM architecture proposed here could be segregated into the following four layers: network layer, middleware layer, application (& service) layer and UI layer (as, Fig. 2 shows each layer with detailed component as

insets). Context Provider (CP) and Context Consumer (CC), both of them will be embedded to Network layer that consists of protocol, sensor network requirement and network implementation. Middleware layer will be driven by Mobile operators like Context Broker (CB) in the proposed system. User infrastructure layer is going to be divided into interface and usability categories. Application and service layer will consist of smart space, Information Systems (IS), communication systems, and web service. A brief projection of each section has explained below:

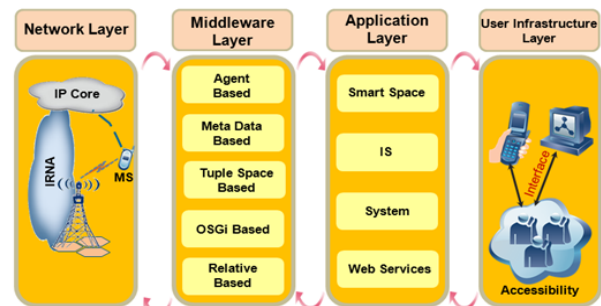


Fig. 2: Framework of emergency context management platform.

##### A. Network Layer

Mobile cloud computing vastly depends on wireless connectivity that often meets the challenges of network's scalability, availability, energy- and cost-efficiency. An intelligent mobile access scheme has to be adjusted in network model that will exploit context information provided by the Mobile Cloud. The location coordinate acquiring services require data collection from terminal sensors (i.e., GPS, gyro, and proximity detectors). The acquired data are used for measuring network status and load. Here, a classical so-called "Intelligent Access" concept has assumed in the proposed architecture where all categories of dynamic context information (user profiles, terminal status and sensor information, external sensor networks, and network status) will be adopted optimistically through mobile access. Interestingly, the network layer of proposed system will experience heterogeneous access

scenarios in real-life deployment (i.e. different type radio access technologies like GPRS, WCDMA/HSPA, LTE, WiMAX, cdma2000, WLAN etc.) around the globe. The context information, such as terminal locations and emergency awaking context from large source-participant can be accumulated by the help of intelligent mobile cloud controller (IRNA) to locally optimize the access management (Fig. 3).

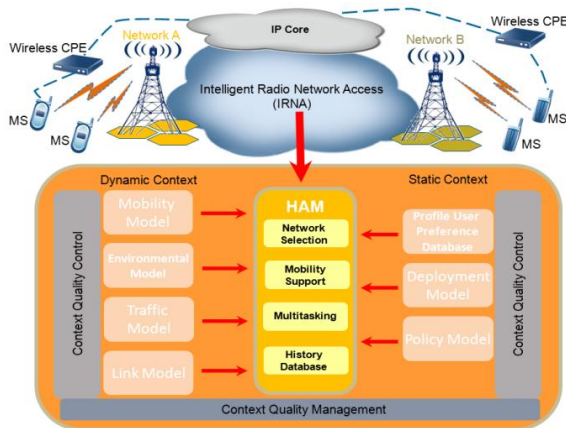


Fig. 3: Mobile Cloud Controller in a heterogeneous access scenario where MS means Mobile Station, CPE means (Customer premises equipment) and Network A and Network B mean two different types of radio network.

Add-on functionalities, such as AAA or charging, will be integrated in network by deploying Intelligent Mobile Subsystems (IMS) in favor of mobile operator revenue. Aside these, it should be assumed that the proposed infrastructure can support low data rate signaling cloud control channel (viz. in GPRS standard). Energy- and cost-efficiency also have to take an account while developing the required radio resources and post application and context-aware radio resource management at network layer. In terms of wide deployment of the proposed framework where majority of the MCC users can be part of it, the network model has to focus upon consumer use rather business, (viz., ABIResearch) [16]. In more and more fragmented and heterogeneous wireless network's landscape, the increasing demand for high data rates (viz. on-demand scalable transmission bandwidth at MCC scenarios)

need efficient network access management across different Radio Access Technologies (RATs); It also referred as Heterogeneous Access Management" (HAM). The fact is radio resources are limited due to physical, technology-specific and regulatory constraints. Thereby, the need for a more sophisticated radio resource management (RRM) for the proposed framework arises due to the presence of different wireless technologies in current world and their common & convergent supports within it and also from the growing importance of the future internet facilities. Wireless communication of proposed infrastructure will firstly enable the internet connection of objects, sensors or actuators and secondly extend the access possibilities to the internet. While there will be emergency situations' relevant responses from crowdsource, they may cost network congestion during communication among mass source (i.e. drivers, observers at probable accident place) and platform. So, demand for a ubiquitous internet access within ECM networks will further increase. Existing RRM techniques are not suitable for coping with the kind of challenging requirements of ECM because they most often aim at optimizing lower layer metrics rather focusing application layer or context information. The fact is there will always be many different types of applications while supporting emergency context-aware based services. Moreover, it will become a prominent challenge to provide same level of QoS while supporting robust and emergent services at higher degree of mobility (i.e. running vehicle in highway). So, it is necessary to adapt Intelligent Radio Network Access (IRNA) for efficient utilization of radio resources and providing end-to-end performance to user demands in emergency context-aware application.

Vehicular emergency context-aware services increasingly rely on data gathered by terminal sensors and (wireless) sensor networks deployed in highway or railway (i.e. integrated in lamppost, signal bar or bill-board by roadside). Information will be accumulated from client-end GPS enabled software interface through internet connection. These data are



complemented by information stored in databases. User and network context, in particular location information and movement prediction by sensor network response around the highway, can be used for optimizing mobility support. Furthermore, adapting the intelligent and dynamic system behavior according to current context of the network, the user, and the terminal can yield significant improvements in end-to-end performance and resource utilization. However, radio resource allocation and utilization decisions may be influenced by network operator policies, service level agreements, user preferences [17] and location information, or as a result of sophisticated resource utilization analysis [18]. Inherently related to RRM becomes concerning issue when the change appears for the user's point of attachment (PoA) of a specific network to another network due to "Handover". Some features such as "always on" and "always best connected" should be included here for efficient use of the heterogeneous landscape of RANs. Therefore, a context-enhanced RRM will be able to exploit not only radio network parameters but any available context information of the users. However, various parameters have to be accounted for ensuring a satisfactory end-to-end performance since each wireless access network provides a different level of Quality of Service (QoS), capacity, and coverage as mentioned earlier. For evaluating possible connections between terminal and PoA, relevant context parameters will be normalized and added up, where each context parameter will be weighted with a certain confidence level, relevance, and costs for information retrieval [19]. A handover of a current active connection of a terminal to a specific PoA is only performed if the maximum of a weighted sum for a specific PoA exceeds the current connection quality plus a pre-defined margin. This margin ensure a better connection quality in absolute terms and allows for trading off handover costs against enhanced connection quality where handover costs are not limited to signaling costs but also comprise costs for acquiring, processing, and evaluating context information. A likewise framework for

network context exchange is being standardized by IEEE's 802.21 working group [20]. The integration of middleware to network layer (mobile operators end to cloud computing end) is expected through IP backbone here which aims at strict separation of the processes of context acquisition and context management. This separation will improve system extensibility and reusability.

## B. Middleware Layer

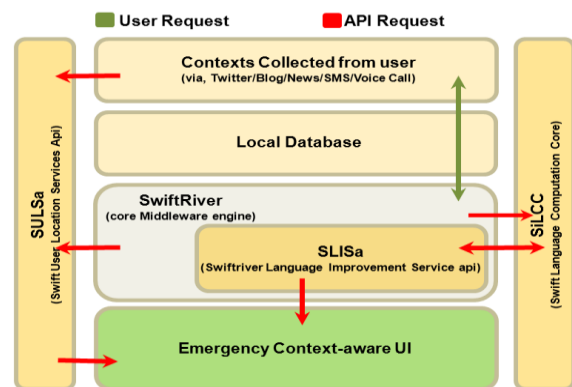


Fig. 4: SwiftRiver based middleware.

Middleware addresses many of the requirements of traditional distributed systems, such as heterogeneity, mobility, scalability, and tolerance for component failures and disconnections. In addition, it must protect users' personal information, such as location and preferences, in accordance with their privacy preferences. It ensures that automatic actions taken by context-aware applications on behalf of users can be adequately understood and controlled by efficient information filter. Fig. 2 shows a general reference of middleware where context-aware system may be built-in with several basic parts. Agent (i.e. Mobile-operator agents (MA)) shown there are regarded as most widely used and easily accessible for many areas. They are suitable to develop context aware services, and have been developed to implement the needed active infrastructure and the MA-based middleware design (Cloud computing agent in proposed framework) and implementation. Metadata provides concurrent binding management

actions in the basis of metadata, context and location visibility. Tuple spaces have been considered for use in sensor networks and as a blackboard with a mechanism that permits to retrieve partially known information. This is a shared repository of information, which can be accessed via well-defined primitives. OSGi (Open Services Gateway initiative) based middleware reliably manages context-aware services to support context acquisition, discovery, and reasoning. Reflective middleware possesses the unique ability to model itself through self-representation, such that the manipulation of its behavior may be changed. A middleware system with self-representation is causally connected if changes made to the self-representation directly affect the implementation of the middleware. Almost all the middleware parts shown in Fig. 2 gather the context information, process them and derive meaningful actions afterwards. In the proposed framework, middleware will allow mobile operators and cloud computing agents integrated way to acquire contextual information easily, reason about it using crowdsourcing filter (viz. SwiftRiver) and then adapt themselves to changing contexts. Decision making logic for effective manipulation and authentication of accumulated data from crowdsources will be adopted by conventional emergency management expert as well as technical expert of Mobile Agent (MA). Crowd-sourcing filter (viz. SwiftRiver) enhanced middleware operation blocks have schematically represented in Fig. 4.

SwiftRiver is an open sourced filter platform that can be adopted as hardcore decision making engine in middleware. It will apply an efficient manipulation of huge accumulated information coming towards middleware from user end. Main task of filter engine will be verifying crowd sourced reports by applying efficient logic dynamically along manual monitoring by admin panel. Whenever the emergency contexts will reach the cloud computing agent, it will store them into cloud data mine and a copy of each of them will be forwarded to SULSa API for extracting the coordinate or location of incoming context.

While incoming context arrives, a dual function named as ‘sweep’ starts intelligent manipulation and verifies authenticity of them (avoiding duplicates). SwiftRiver also offers all users a way to input relevant emergency context. Like as, the proposed framework here, user is going to interact with this intelligent filtration process just by pressing ‘Red’ and ‘Yellow’ button (mentioned at section 4) at client-end software. A concurrent feedback from client-end software is going to communicate with proposed MCC based ECM platform through internet or pre-formatted auto generated SMS (“Message|location|\*CODE#” where, in proposed framework, message is going to be ‘Red’ or ‘Yellow’, location will include GPS coordinates and CODE will add \*888# like predefined SMS on-Demand service token). Aside, automatic filtration of incoming emergency context, a manual monitoring system will be added developing web-based interface by PHP, SimplePie (rss aggregation and real-time), Python, Ruby, Protovis, Photochart application interfaces. The basic SwiftRiver workflow process has summarized in the Fig. 5.

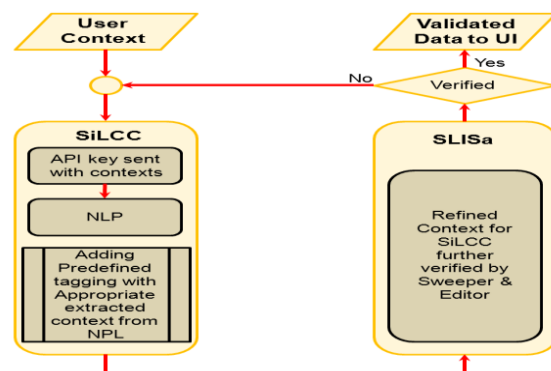


Fig. 5: Work flow diagram of SwiftRiver.

All users’ raw contexts first forward to SwiftRiver that redirect those to Swift Language Computation Core (SiLCC) for parsing the valuable information by tagging with internal code with the help of SwiftRiver API (for reducing the duplicate usage of SiLCC for same context location). Besides parsing, SiCDS (SwiftRiver Content Duplication Service), RiverID (distributed reputation

tagging) and Riverberations (weighting the context measuring the online influence) are also integrated in the same SiLCC block. The preliminary filtered data with predefined parameter are then tagged as successfully parsed and forwarded to SwiftRiver Language Improvement Service API block where sweepers (registered administrators who use monitor manually and approve submissions) and editors (who can add feeds, sweep and vote on source accuracy with more 'weight' than sweepers) review again the authentication of preprocessed emergency contexts. If the reviewed contexts become validated by sweepers and editors, they are ready to publish in UI. Failing to be approved by automatic filtration or manual filtration by Sweepers/Editors forwards the incoming emergency contexts again to be reprocessed through SiLCC [21].

### C. Application Layer

Application layer is responsible of proving smooth intermediary communication channel of UI to machine interface. Context-aware application and service should have independency of computer device to provide the user with services autonomously. Context-aware applications should be customizable by using user context and preference. Using Java-enabled script, a user or OS independent platform could be offered there. Besides, context-aware applications will act in anticipation of future goals or problems moving point and problem of user. In this case, various devices are going to be interconnected among each other in the proposed platform where they will be recognizable each other within a certain distance. Proximal devices distance between user and device or between devices will play an important role to offer the appropriate service to the users (especially to vehicle like nomads).

### D. User Infrastructure Layer

Fig. 2 shows the references according to the detailed categories in user infrastructure layer. It is characteristic for handheld devices

and their users that they are continuously moving in several simultaneous fuzzy contexts. The usefulness of a given context aware systems are most likely to suffer from the small screen and a dynamic context of users. Mobility feature is the very essence of context-awareness management platform. Therefore, dynamic environment sets special requirements for usability and acceptance of context-aware systems. In UI category, a content adaptation system for overcoming small interface is Smartphone available today need to be incorporated (A demo interface in shown as UI in Fig. 1). User modeling and human-computer interaction for considering the emergence of global and mobile computing environments have also been considered here [15]. The whole UI experience will be based upon simple technology where users will not have any kind of dealing with backend operations. The main goal of having UI is to employ human and machines interaction with effective operation. Users only experience here what they need to input and what they need to see as output. Likewise for admin point of view, the system will be pretty much simple as they would be able to add/edit mock up the whole data flow in their customize way and generate a review.

## 5. Summary

Vehicular safety is a hot research segment around the globe among all research interest. Among several righteous candidates, Mobile Cloud Computing (MCC) enhanced ECM platform has chosen here with integration of SwiftRiver like context filtration intelligence and GPS along wireless LAN embedded smartphone. Automation through the integration of such proposed framework will offer a simple but exciting step to mitigate the large risk at vehicular communication during disaster. Aside this, the complexity level of user end operation is low since it doesn't require any advanced IT qualification of a user except having a smart knowledge in operation of GPS enabled application. Thereby, it would be the high time to implement the proposed framework



that would set an example to rest of the world for next generation vehicular emergency challenges.

## References

- [1] I. Juga and M. Hippi (2011), "Snowfall induced severe pile-ups in southern Finland on 17 March 2005", 14 June [online], Available: <http://meeting.organizer.copernicus.org/EMS2009/EMS2009-389.pdf>.
- [2] Jonas Norrman, Marie Eriksson, Sven Lindqvist (2011), "Relationships between road slipperiness, traffic accident risk and winter road maintenance activity", 14 June [online], Available: <http://www.int-res.com/articles/cr/15/c015p185.pdf>.
- [3] Wikipedia , "Mobile Phone – Wikipedia, the free encycloedia" [online], cited: August 2010 [online], Available: [http://en.wikipedia.org/wiki/Mobile\\_phone](http://en.wikipedia.org/wiki/Mobile_phone).
- [4] Gartner Newsroom (2011), "Worldwide Smartphone Operating System Market by Year-End 2012", 18 June [online], Available: <http://www.gartner.com/it/page.jsp?id=1622614>.
- [5] J.H. Christensen (2009), "Using RESTful web-services and cloud computing to create next generation mobile applications," *OOPSLA*, New York, USA: ACM Press, pp. 627.
- [6] X. Luo (2009), "From Augmented Reality to Augmented Computing: A Look at Cloud-Mobile Convergence," *International Symposium on Ubiquitous Virtual Reality*, vol. 10, , pp. 29-32.
- [7] I. Giurgiu, O. Riva, D. Juric, I. Krivulev, and G. Alonso (2009), "Calling the cloud: enabling mobile phones as interfaces to cloud applications," *Middleware '09: Proceedings of the 10th ACM/IFIP/USENIX International Conference on Middleware*, New York, NY, USA, , pp. 1-20.
- [8] B. Chun and P. Maniatis (2009), "Augmented Smartphone Applications Through Clone Cloud Execution," *HOTOS workshop*, USENIX,.
- [9] E. Marinelli (2009), "Hyrax: Cloud Computing on Mobile Devices using MapReduce", Master Thesis Draft, Computer Science Dept., CMU.
- [10] Byun, H. E., & Cheverst, K. (2004). "Utilizing context history to provide dynamic adaptations". *Applied Artificial Intelligence*, 18(6), pp.533–548.
- [11] Wikipedia, "Mobile Phone – Wikipedia, the free encycloedia Crowdsourcing - Wikipedia, the free encyclopedia", August 2010 [online], Available: <http://en.wikipedia.org/wiki/Crowdsourcing>.
- [12] H. Liu and D. Orban. Gridbatch (2008), "Cloud computing for large-scale data-intensive batch applications", In *Proceedings of the 8th IEEE International Symposium on Cluster Computing and the Grid (CCGrid)*, IEEE Computer Society, , pp. 295–305.
- [13] Xiaosan Ge, Huilian Wang (2009), "Cloud-Based Service for Big Spatial Data Technology in Emergency Management", *Proceedings of the ISPRS*, vol. 38(7)C4, , pp. 126-129.
- [14] "Ushahidi Used to Create Oil Spill Crisis Map", December 8, 2010 [online], Available:<http://blog.ushahidi.com/index.php/2010/05/08/lab/>
- [15] Jong-yi Hong, Eui-ho Suh, Sung-jin Kim (2009), "Context-aware system: A literature review and classification", *Expert Systems with Applications*, vol. 36, no. 4, ,pp. 8509-8522.
- [16] "ABIResearch: Mobile Cloud Computing - Next Generation Browsers, Widgets, SIM, Network-as-a-Service, and Platform-as-a-Service", 3Q 2009 [online], Available: <http://www.abiresearch.com/research/1003385>
- [17] Ormond, O., Perry, P., Murphy, J. (2005), "Network Selection Decision in Wireless Heterogeneous Networks", *PIMRC 2005*, Berlin, Germany.
- [18] Blau, I., Wunder, G., Karla, I., Sigle, R. (2007), "Cost Based Heterogeneous

Access Management in Multi-Service, Multi-System Scenarios”, *PIMRC 2007*, Athens, Greece.

- [19] Mannweiler, C., Klein, A., Schneider, J., Schotten, H. (2010), “Context-Awareness for Heterogeneous Access Management, *Advances in Radio Science*”, Vol. 8,.
- [20] IEEE P802.21/D14, “Draft Standard for Local and Metropolitan Area Networks: Media Independent Handover Services”, 2009, New York.
- [21] SwiftRiver initiative, “SwiftRiver Platform Overview”, August 10, 2010 [online], Available: <http://www.slideshare.net/Ushahidi/swiftriver-overview>.