Robust and Trusted Crowd-Sourcing and Crowd-Tasking in the Future Internet

Denis Havlik, Maria Egly, Hermann Huber, Peter Kutschera, Markus
Falgenhauer, Markus Cizek

Robust and trusted crowd-sourcing and crowd-tasking in the Future Internet

Denis Havlik, Maria Egly, Hermann Huber, Peter Kutschera, Markus Falgenhauer, and Markus Cizek

AIT Austrian Institute of Technology GmbH, Austria
{Denis.Havlik, Maria Egly, Hermann Huber, Peter Kutschera, Markus Falgenhauer, Markus Cizek}@ait.ac.at

Abstract. A great majority of the EU citizens already owns a cellular phone. An increasing part of these phones are smartphones with a broadband internet connection. This growing network of smart internet enabled devices could act as a dense sensing network, as well as a tool for individual informing and tasking of mobile citizens and volunteers.

In order to fully harvest this new resource, we need to understand its rules and develop adequate tools. This paper lists some of the peculiarities of ad-hoc volunteer networks supported by smartphones. It furthermore compares the capabilities and limitations of these networks with other means of observation gathering and activity coordination. This includes: (1) a reflection on the motivation for users participation; (2) human and technical limitations of smartphone-enabled volunteer networks; (3) legal and ethical challenges; (4) reliability and usability issues; as well as (5) issues related to trust and quality of information.

The second part of the paper presents our experiences with design and prototypic development of the tools supporting volunteer efforts in the field of environmental monitoring, e-health and crisis management. This development is presented in a wider scope of the "Future Internet Public Private Partnership" research programme.

Finally, the paper summarizes our findings and recommendations for further developments.

Keywords: crowdsourcing, crowdtasking, human sensors, local situation awareness, volunteered geographic information, future internet, mobile applications

1   Motivation

According to a recent comScore whitepaper [1], the smartphone penetration has surpassed the 50% mark in US, Canada and the EU5 (UK, France, Germany, Spain and Italy) during 2012. Moreover, the mobile media already accounts for 37% of the total time spent online and for a majority of usage in categories well suited for use on smartphones and tablet PCs: maps, weather, music and social networking. These de-

vices combine (almost) "always on" broadband connectivity, ad-hoc local connectivity, significant storage capacity and computing power with intuitive user interfaces and a number of built-in as well as add-on sensors for various purposes.

Numerous researchers have been testing the usability of smartphones as platforms for mobile sensing, in the recent past [2-3]. From the point of view of user interaction, this resulted in two distinct application classes: "opportunistic sensing" [4] applications, where the user merely allows an application to report (interpreted) readings of selected sensors, and the participatory sensing applications [5], where users are an integral part of the sensing and quality assurance loop ("human sensors", "crowd-sourced quality assurance"), and actively provide information on their environment.

In this paper, we shall consider crowdsourcing of observations, data interpretation, dissemination and (crowd-) tasking as integral part of the mobile Volunteered Geographic Information (VGI) applications and present a prototype of the Mobile Data Acquisition Framework (MDAF) which simplifies the task of developing such applications. The word "observation" is used in-line with the definition of the Open Geospatial Consortium, to indicate geospatially and temporary referenced data reported by hardware sensors, human sensors or automated processes (nowcasts, forecasts, virtual sensors, and indicators).

The paper starts with a discussion of the challenges and opportunities of mobile VGI applications (section 2). Section 3 presents the MDAF framework and section 4 discusses MDAF's ability to answer these challenges. Finally, section 5 "conclusions" presents the main lessons learned in MDAF development.

2   Challenges and opportunities of mobile VGI applications

This section discusses some "peculiarities" of VGI applications as compared to a classical monitoring system where all sensors and networking equipment are maintained and calibrated by a central authority.

For the start, let us consider some challenges directly resulting from the technology used in VGI applications: inaccurate sensors, unreliable networks, sub-optimal positioning of the smartphones for taking measurements and the need to recharge the battery on (at least) a daily basis. On the whole, smartphones aren't optimal sensing systems, and in reality there are only three reasons for (mis)using them in this way: (1) the phones are already out there and we don't have to pay for them; (2) they are at the appropriate location to provide information on phone users and their surroundings; and (3) the phone users can provide us with information we could not easily get from hardware sensors.

In opportunistic sensing application, the users merely need to be sufficiently motivated to install an application and let it use the smartphone resources. On the other hand, the participative applications rely on users as active providers of observations or "human sensors"[1]. In some areas, human sensing surpasses the capability of hardware sensors and automated algorithms for sensor data interpretation: recognising and in-

---

[1]   In the simplest case, users may only be asked to use their own judgement to decide when to start recording and assure the optimal positioning of the smartphone.

terpreting combinations of visual, audible, and to some extent also olfactory patterns and anomalies is easy for humans and very difficult for hardware sensing systems. On the other hand, humans are notoriously bad at some tasks which can be easily performed by hardware sensing systems:

- Humans get easily tired and bored, and are notoriously bad at performing repetitive activities.
- Reported data will often depend on the individual sensitivity (e.g. some people are sensitive to weather changes or pollen, others aren't), cultural background (is a playground full of children noisy or music to ones ears?) and even on the current mood of the reporter.
- Finally, the humans are not good at quantification of observations (how loud is "very loud" in decibel? Which temperature corresponds to "very cold"?)

Fully fledged analysis of the participants' motives and the ways to enhance the motivation, as presented in e.g. [6], is beyond the scope of this paper. Nevertheless, it is important to keep in mind that the *users' motivation* is crucial for success of all VGI applications. Some key concepts influencing the users' motivation for participation in voluntary activities are summarized below:

- Participants can have very different reasons for being both constructive (providing observations, helping with quality assurance) and destructive (providing misinformation, misusing the system for spamming or for the personal gain).
- Direct or indirect personal gain, including both "monetary gain" and the possibility to improve own knowledge (intellectual gain) are very strong motivations –for constructive and for the destructive contributions.
- Another very strong motivation is the "fun factor" that originates from using the application.
- Knowledge that own contributions will be acknowledged, appreciated, seen by the peers and used for the well-defined and "good" purpose also greatly enhances the user motivation.
- The ease of use of the system and active solicitation of the information are crucial for attracting the participants with relatively low motivation levels.
- Concerns about misuse of users' observations and privacy-related information may lower the willingness of the citizens to participate.

As mentioned above, the participants can be motivated for both constructive and destructive contributions. In addition, the "quality" of the human sensors and of the inexpensive sensors built-in smartphones is a-priory unknown, may vary over time, and in-lab calibration is often unpractical.

Consequently, the quality of the information in VGI applications is usually estimated based on the combination of statistic methods and human judgment. In many cases, the application also takes users "reputation" into account – a numeric value indicating how likely it is that a certain user will contribute high quality observations. Some methods for calculating the users' reputation are discussed in [7].

3 MDAF framework

3.1 Vision and functional architecture

Mobile Data Acquisition Framework (MDAF) is a set of mobile and backend software components designed to simplify the task of implementing mobile VGI applications – independent on the contents and scope of these data. Our long-term goal is to develop an application framework which allows loosely organized groups to conduct measurement campaigns and to analyse and discuss the results online. For the reasons explained in section 2, the framework must also: (1) support automated and manual tasking of the volunteers; (2) provide mechanisms for interpreting the meaning and quality of the observations; and (3) assure that the privacy and security of the users and the data transferred and stored in the framework is handled carefully according to legal constraints.

The use of MDAF from the mobile user perspective is illustrated on **Fig. 1** below.

Fig. 1. MDAF mobile use cases

Users can: (1) view existing knowledge; (2) report new objects of interest[2] (OOI) and new observations on existing OOIs; and (3) receive alerts and event notifications on changes and requests for performing certain tasks pertinent to their areas on interest (AOI)[3]. The "offline mode" and "offline interface" are necessary to improve the overall reliability and usability of the system:

---

[2] OOI is a specialization of "thing" in internet of things and "feature" in OGC context.
[3] AOI is, as a word suggests, an area that the user is interested in.

- "Offline mode" is a consequence of the requirement that the application remains fully functional in slow and unreliable networks. As a welcome side-effect, the pre-loading and caching of the information pertinent to user-defined AOIs lowers the requirements on network bandwidth and latency during field trials.
- "Offline interface" refers to the requirement that users can provide input by means other than the smartphone GUI interface, e.g. using some of the built-in sensors to either directly provide the observation, or to record the user's input.

From the perspective of a service using the data collected and stored within the framework, MDAF must provide a way to: (1) combine the observations received from mobile users with data from external sources; (2) quality assure and interpret the resulting data heap; and (3) assure that the results can be queried and accessed by any service that needs them. For the sake of efficiency, the framework must both generate events, which can be used to trigger external processing services for each incoming observation and provide a service interface for accessing the observation ("pull") at a later time. Main MDAF use cases from the backend perspective are illustrated in the **Fig. 2** below.
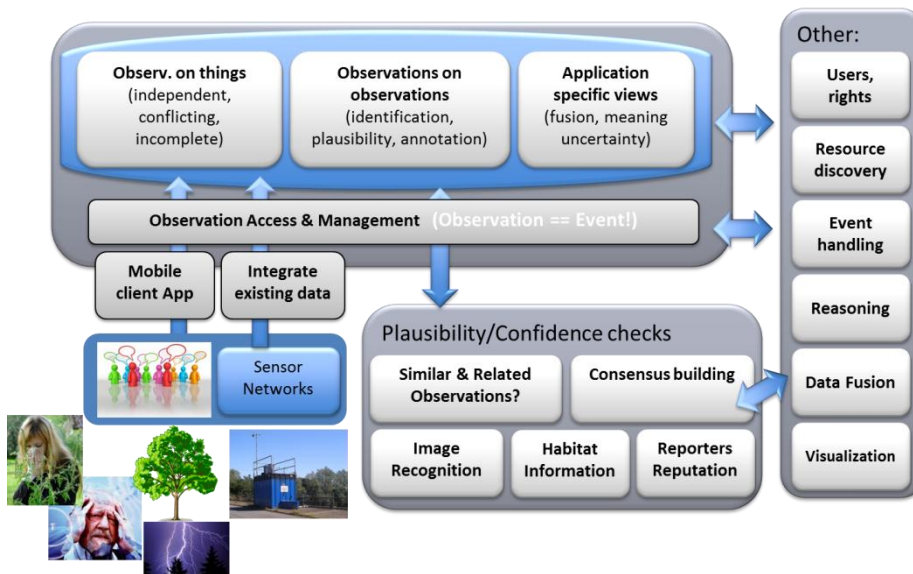

Fig. 2. MDAF backend use cases

This figure also illustrates the way MDAF organizes information and interacts with other services:

- Most of the data is considered "observations" – geospatially and temporary annotated information on some real-world or virtual "objects of interest". The observations can originate from our own application, from third party observation networks or from some processing service.

- "Objects of interest" themselves can be either simple data objects representing the real world objects[4] or other observations. The observations which refer to existing observations are called "observations on observations". They provide additional information on quality and meaning of the initial observations. This information can be provided by users as well as by automated processes.
- The framework does not prevent the users and automated processes from publishing incomplete or conflicting observations and a-priory does not make any judgments on the quality of this information. The task of interpreting the resulting observation heap and producing a useful view on this data is considered application specific, delegated to external processes and the results are stored in "application specific views".

The importance of this data model for VGI applications can be easily illustrated by the example of "consensus building" (**Chyba! Nenalezen zdroj odkazů.**). If several users are asked to report observations on a single OOI, the most likely result is a non-consistent heap of observations. The fact that all these observations can peacefully co-exist within MDAF allows us to decide which of the observations are most plausible at application level, possibly taking into account some additional information, such as users' reputations, observations reported by automated processes, statistical analysis of the nearby reports or the probability of reported observations in a local habitat.
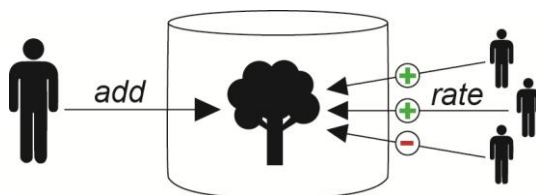

Fig. 3. Consensus Building

3.2   Implementation Architecture

Our initial idea was to follow the ideas of the SANY SensorSA architecture and re-use the existing OGC service interfaces, most notably the Sensor Observation Service (SOS) for serving of the observations [8, 9]. However, the initial tests with smartphones indicated that the combination of (Geo)JSON and RESTful service interfaces is more appropriate for mobile applications than the XML/OGC-style web service combination used by SOS. In addition, we needed a service that would run both on a backend and on mobile phone and synchronize the data to the users' device as needed.

CouchDB [10] offered exactly the type of functionality we needed: (1) a geospatially-enabled database which can be deployed both on a smartphone and on a backend; (2) RESTful http interface and a native support for (Geo)JSON; (3) Robust multi-master "eventually consistent" synchronization protocol allowing two-way partial synchronization of the data that fulfils the query criteria; and (4) Built-in event

---

4   In simplest case, the OOI will contain a unique ID and a reference to object type.

queue which can be used to trigger the sending of appropriate messages to users and other services.

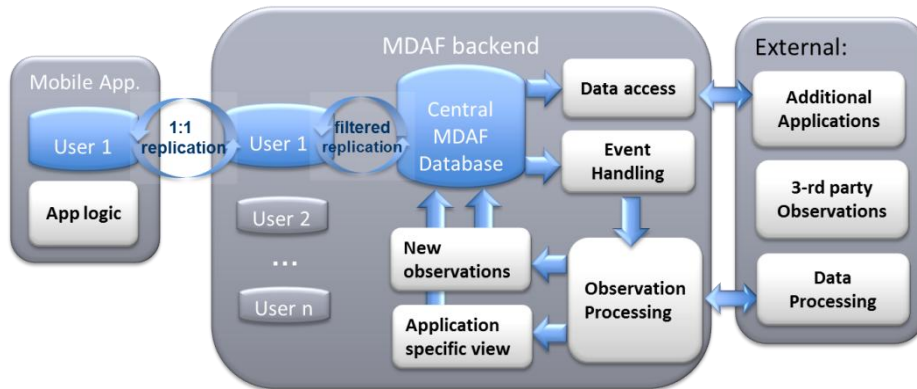MDAF implementation architecture is illustrated in **Fig. 4**.

Fig. 4. MDAF implementation architecture

- The MDAF backend maintains a 1:1 replica of the user database on each mobile device. Every database is connected to the central MDAF database via filtered replication. This architecture assures the eventual consistency between all databases.
- CouchDB generates internal events for each database change. MDAF event-hander monitors these events and triggers appropriate observation processing modules.
- MDAF observation processing modules are self-contained pieces of software which implement most of the MDAF server-side application logic. They usually receive new observations from the event handler, generate additional observations based on processing results and store them on the central database from where they can be distributed to mobile users. In many cases the modules act as connectors and the actual processing is delegated to external services.

Our main concern on the mobile application side was to assure portability. In order to achieve this goal, we opted for a hybrid development model where most of the application logic is realized using standardized platform independent technology. Platform-specific libraries are only used to access the underlying hardware.

As illustrated in **Fig. 5**, the hardware access is handled by PhoneGap/Apache Cordova. The application functionality build upon Cordova is offered by the Sencha Touch mobile application framework. An evaluation of the usability of the Cordova/Sencha Touch combination for cross-platform mobile application development can be found e.g. in [11].
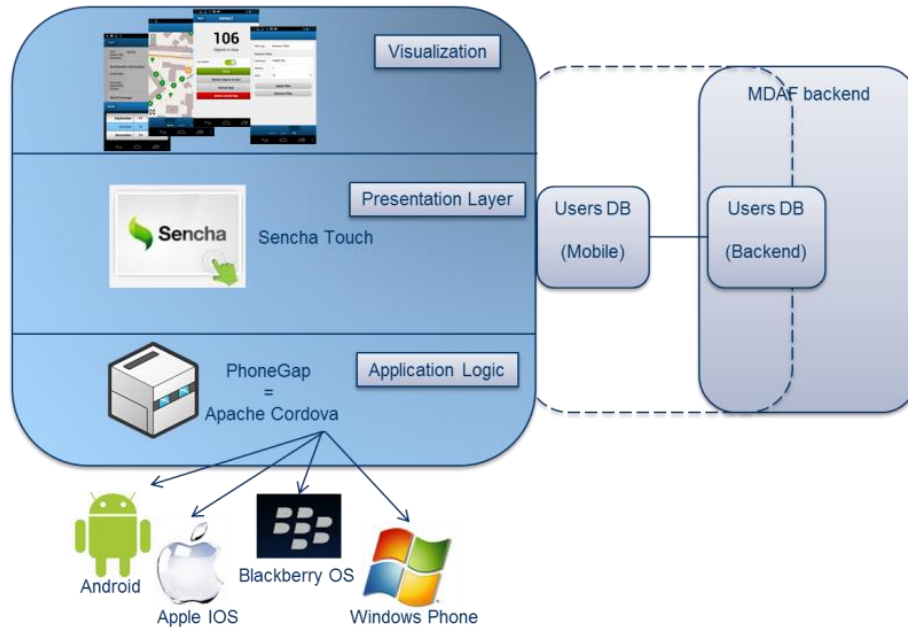
Fig. 5. Technologies used in MDAF mobile app development

3.3 Integration in ENVIROFI and FI-Ware architecture

The power of the MDAF framework originates from its ability to reuse the data and functionality offered by the third party data access and processing services. Within the scope of ENVIROFI, our primary goal was to assure that the MDAF framework can be used in combination with Generic Enablers (GEs) developed within FI-Ware project, as well as in combination with the environmental Specific Enablers (SEs) developed within ENVIROFI [12]. In fact, the MDAF framework is an integral part of the ENVIROFI architecture and implements three SEs:

- MDAF backend service is the reference implementation of the Environmental Georeferenced Observation Collection Service.
- The caching database deployed on user devices is the reference implementation of the Environmental Georeferenced Observation Proxy Service.
- Two mobile applications (see section 4.1) are considered implementations of the Environmental-Georeferenced Observation App.

The interaction of other enablers and other third party applications and services is achieved through implementation of the appropriate data access interfaces and observation processing/connector modules (see **Fig. 4**). The simplest type of modules acts as clients to third party data access services and can be used to import the data from external sources into MDAF. So far, the data can be imported from: OGC Web Fea-

ture Service [13], OGC SOS [14], Shapefiles and proprietary RESTful interface of the NILU air quality service.

More complex modules provide read-only or read-write access to MDAF observations. Primary data access to MDAF observation is the RESTful CouchDB interface. This interface allows retrieval of observations corresponding to specific geo-locations (within Bounding Box), observation IDs, OOI IDs, filtering on various object properties and publication of new observations. In addition, we have also implemented:

- A connector to TS-Toolbox and SUDPLAN's implementation of the Sensor Observation Service [15]. This allows us to provide access to MDAF observations over standardized SOS interface.
- A connector to ENVIROFI Environmental Image Sample Classification Service SE [16]. This allows us to automatically submit all observations of the "leaf image" type to this SE as well as to integrate the results (list of species with similar leafs) as additional observations.
- A connector to Google Cloud Messaging [17] provides an alternative channel (in addition to CouchDB replication mechanism) for pushing alerts and task requests to MDAF users.
- Finally, a two-way connector for exchanging events (observations) between MDAF and the Publish/Subscribe Context Broker GE [18] facilitates integration of MDAF in the Future Internet applications.

4   Development status and outlook

The development of MDAF started in Q4 2012, with the intention to support the development of the "conceptual prototypes" of the future internet enabled environmental applications. In the course of the project, the complexity and maturity of the MDAF framework grew way beyond a simple proof of concept. Nevertheless, further effort is required to reach the maturity level which would allow us to use this framework in real world applications.

4.1   ENVIROFI pilots

Within ENVIROFI, MDAF has been used in two sample applications concentrating on participatory sensing and on the human/environment interaction:

- The Personal Environmental Information System (PEIS) application allows users to track information on airborne pollutants, pollen as well as meteorology and correlates these with personal well-being information. Additionally, it enables the user to share current local environmental conditions as described in [19, 20]. The main user interface elements and use cases of the PEIS application are illustrated in **Fig. 6**.

Fig. 6. PEIS enables users to relate personal well-being to environmental observations

- The "biodiversity" application allows users to receive and publish information on biodiversity in area(s) they are interested in [21]. The current prototype has been designed and developed in cooperation with the Long Term Ecological Research (LTER) Austria [22] and specializes in the objects of the type "tree".

Both applications can be downloaded from ENVIROFI catalogue and tested on android smartphones. PEIS app is developed only to a proof of concept level, whereas the biodiversity prototype is almost completed in the terms of features.

Comparison of the MDAF-related features of the pilots is shown in the **Chyba! Nenalezen zdroj odkazů.**.

Table 1. MDAF-supported functionality in ENVIROFI pilots

	Biodiversity pilot	**PEIS pilot**
Observations from external sources	Tree databases of Vienna, Tuscany and LTER Eisenwurzen (WFS, shapefiles)	Air quality in Vienna, Stockholm and Karlsruhe (SOS, NILU RESTful)
Observations from mobile users	Various observations on trees, in-line with requirements of the LTER-Austria.	Barometric sensor reading and own reactions on environmental conditions
Observations generated by SEs	Leaf identification	Air quality coverage
Service interfaces	CouchDB RESTful	CouchDB, OGC SOS
Offline GUI	Assign NFC tag to OOI; identify OOI by NFC tag	-
Events	CouchDB, FI-Ware, Google cloud messaging	CouchDB
Filtering	By area, numeric parameters, text	Threshold
Key functionality implemented by FI-Ware GEs	Authentication; image store; event handling; web GUI interface	-

	Biodiversity pilot	PEIS pilot
Areas of Interest	User-defined AOIs	predefined regions: Vienna, Stockholm, Karlsruhe
Alerting	New observations in users AOIs. Can be sorted by time and by distance from the user.	Threshold for various air quality parameters
Tasking	Request additional observations (partially implemented)	-
Crowdsourced quality assurance	"consensus" (partially implemented)	-
Automated quality assurance	Leaf identification; common name/Latin name; probability within habitat (partially implemented)	-
Offline mode	Background maps, tree types and status within user-defined areas of interest	-

4.2 Beyond ENVIROFI

So far, the MDAF framework has been developed and tested primarily in the context of environmental observations and the Future Internet Public-Private Partnership initiative. Due to combination of time constraints, resource constrains and low maturity of the FI-Ware platform, we were able to integrate and test only a small subset of the FI-Ware GEs. In spite of the weak support for geospatial data and processing, which has been identified as main shortcoming of FI-Ware by the ENVIROFI team [23], our analysis indicated several GEs with "high potential" for future VGI applications.

Likewise, the ENVIROFI experience has thought us that the use of external data fusion and processing services can greatly improve the overall quality and usability of VGI applications. However, most of the algorithms used by these services are highly application specific and therefore not likely to ever become part of the core MDAF offering.

With ENVIROFI project approaching its finalization, our mid-term goal is to use MDAF to facilitate active citizen participation in applications combining environmental awareness with topics such as human health/well-being, traffic, energy or crisis management.

Our development priorities are therefore to: (1) develop robust and generic reputation, crowdtasking and consensus building modules; (2) stimulate the volunteer's motivation by various incentives (see [6]); (3) improve the overall quality, performance and scalability of the framework; (4) consolidate and standardize the MDAF data model and service interfaces; (5) improve the overall security of the application and assure the delicate information (e.g. information related to users health or sightings of rare species) is adequately protected per design; (6) further develop the concept of "offline use" in case of network failure towards fully autonomous pear to pear

exchange of information over ad-hoc networks; (7) simplify the task of developing new application to the level where this can be done by users with no software development experience; and (8) test the promise of OS-independent development by actually rolling out the application on non-Android devices.

5   Conclusions

The MDAF framework demonstrates the feasibility of developing robust and trusted crowd-sourcing and crowd-tasking applications in the context of Future Internet. Some of the concepts and technologies tested in MDAF and validated in the ENVIROFI applications are novel and have the potential for greatly enhancing the functionality offered by future mobile VGI applications. For example:

- The use of document-oriented CouchDB database with native http RESTful interface allows greater flexibility at the level of data models.
- The hybrid development model where most of the application is written in HTML5/Javascript lowers the cost of cross-platform development.
- The CouchDB replication mechanism and the possibility to run the same database on server backend and on the mobile device greatly improve usability of the application on slow and unreliable networks, offsets workload from the central server and minimizes the requirements on bandwidth and latency during field work.
- User-defined Areas of Interest completely eliminate the need for disclosing the current users' position to the central server, while still assuring the messages relevant for the user (e.g. alerts pertinent to his or her AOIs).
- The data model explained in section 3.1 allows MDAF-based applications to provide a consistent and easy to understand views on observations, even if the underlying dataset contains incomplete and conflicting observations.

Based on the experiences with development of the biodiversity and PEIS applications, we are confident that MDAF provides excellent answers to most of the challenges mentioned in section 2 of this paper and will develop into a fully-fledged framework for VGI applications. However, it is important to understand that the MDAF development is not yet finalized and that some important features (e.g. fully-fledged trust and reliability management, incentives for participating volunteers and secured data access) will have to be developed in the follow-up projects.

6   Bibliography

1. Lipsman, A. C. A.: 2013 Mobile Future in Focus, `http://www.comscore.com/Insights/Presentations_and_Whitepapers/2013/2013_Mobile_Future_in_Focus`
2. Lane, N. D. , Miluzzo, E., Lu, H., Peebles, D., Choudhury, T., Campbell, A. T.: A Survey of Mobile Phone Sensing, In: IEEE Communications Magazine vol. 09, pp. 140-150. IEEE Computer Society (2010)
3. Lane, N. D.: Community-Aware Smartphone Sensing Systems, in IEEE Internet Computing, vol. 16, no. 3, pp. 60-64. IEEE Computer Society (2012).
4. Campbell, A. T., Eisenman, S. B., Lane, N. D. , Miluzzo, E., Peterson, R. A.: People-centric urban sensing, In Proceedings of the 2nd annual international workshop on Wireless internet (WINCON), ACM (2006)
5. Burke, J., Estrin, D., Hansen, M., Ramanathan, N., Reddy, S., Srivastava, M. B.: Participatory Sensing, In: Workshop on World-Sensor-Web (WSW'06): Mobile Device Centric Sensor Networks, ApplicationsWoodrow, ACM (2006)
6. Coleman, D. J., Georgiadou, Y., Labonte, J.: Volunteered Geographic Information: The Nature and Motivation of Produsers, In: International Journal of Spatial Data Infrastructures Research, no. 4, pp. 332-358, Joint Research Centre of the European Commission (2009)
7. Bishr, M.: Trust & Reputation Models for Human Sensor, Ph. D. Thesis, University of Münster, Germany (2011)
8. Havlik D, Schade S, Sabeur ZA, Mazzetti P, Watson K, Berre AJ, Mon JL.: From Sensor to Observation Web with Environmental Enablers in the Future Internet, In: Sensors, Vol 11, pp 3874-3907, MPCI (2011)
9. Klopfer, M., Simonis, I., Eds.: SANY - An Open Service Architecture for Sensor Networks, SANY Consortium (2009)
10. Anderson, J. C. , Lehnardt, J., Slater, N.: CouchDB: The Definitive Guide, O'Reilly Media, Inc. (2010)
11. Heitkötter, H., Hanschke, S., Majchrzak, T. A.: Evaluating Cross-Platform Development Approaches for Mobile Applications, In Web Information Systems and Technologies, vol. 140, pp. 120-138, Springer (2013)
12. ENVIROFI Consortium: Catalogue of Specific environmental enablers and prototype applications, `http://catalogue.envirofi.eu/enablers` (2013)
13. OGC Consortium: Web Feature Service, http://www.opengeospatial.org/standards/wfs (2010).
14. OGC Consortium, Sensor Observation Service, `http://www.opengeospatial.org/standards/sos` (2007)
15. Kutschera, P., Bartha, M., Havlik, D.: SUDPLAN s Experiences with the OGC-Based Model Web Services for the Climate Change Usage Area, In: Hrebícek, J., Schimak, G., Denzer, R. (Eds.) Environmental Software Systems. Frameworks of eEnvironment - Proceedings of the 9th IFIP WG 5.11 International Symposium ISESS 2011, Springer (2011)
16. Middleton, S. E.: Environmental Image Sample Classification Service, `http://catalogue.envirofi.eu/enablers/environmental-image-sample-classification-service` (2013)
17. Google: Google Cloud Messaging, `http://developer.android.com/google/gcm/index.html` (2013).

18. Márquez, F. G.: Publish/Subscribe Context Broker - SAMSON Broker, `http://catalogue.fi-ware.eu/enablers/publishsubscribe-context-broker-samson-broker` (2013)
19. Kobernus, M., Pielorz, J., D2.3.2 Functional and Organisational Specification for PEIS Pilot II, Public deliverable of the ENVIROFI project, ENVIROFI Consortium (2012)
20. Van der Schaaf, H., Kobernus, M., Falgenhauer, M., Pielorz, J., Watson, K.: Data fusion in the environmental domain, In: proceedings of ISESs 2013, Springer (2013)
21. Schleidt, K., Egly, M., Havlik, D., Schade, S., Giacomelli, A., Mazzetti, P., Laurenne, N., Tuominen, J., Middleton, S. E., Arbab-Zavar, B.: D1.3.2 Functional and Organisational Specification for Biodiversity Pilots II, Public deliverable of ENVIROFI project, ENVIROFI Consortium (2012)
22. Umweltbundesamt Austria: Austrian Long Term Research Network, http://www.umweltbundesamt.at/umweltsituation/lter_allgemein (2013)
23. Usländer, T., Berre, A., Canut, C. G., Havlik, D., Lorenzo, J., Sabeur, Z., Modefferi, S.: The Future Internet Enablement of the Environment Informaiton Space, In proceedings of ISESS 2013, Springer (2013)