Advantages of an Integrated Open Framework for Immediate Emergency Response

Axel Bürkle, Florian Segor, Sven Müller, Igor Tchouchenkov, Matthias Kollmann
Fraunhofer IOSB, Karlsruhe, Germany
{axel.buerkle, florian.segor, sven.mueller, igor.tchouchenkov,
matthias.kollmann}@iosb.fraunhofer.de

ABSTRACT

Recent disasters have shown that wireless sensors and unmanned systems are increasingly becoming a valuable aid for first responders. Depending on the kind of incident and its extent, different assets are to be used. The more diverse these assets are, the more complex their simultaneous use and coordination. Therefore, integrated solutions are needed which comprise all necessary components such as power supply, communication infrastructure, data acquisition and processing, decision support and information dissemination. In this paper, an architecture for an open framework is proposed and its advantages over dedicated solutions are discussed. The flexibility of the universal control station presented here is demonstrated using the example of integrating a smartphone as an additional mobile sensor.

Keywords

Emergency response, open system architecture, universal ground control station, sensor network, smartphone.

INTRODUCTION

With the typical ad hoc character of crisis situations such as natural disasters (e.g., earthquakes, tornados, tsunamis, etc), terrorist attacks or major incidents (e.g., in a power plant), a fast and situation-specific response is crucial. Today, unmanned ground vehicles (UGVs) and unmanned aerial vehicles (UAVs) carrying sensors contribute to generating a quick situation picture. They help authorities and first responders in assessing the situation and making decisions. Furthermore, those systems are ideal means for tasks such as the search for injured people or the detection of possible causes of risk. Whereas in some cases, a single unmanned vehicle might be adequate, the aftermath of a typical crisis is often too vast to be covered by a single dedicated system. Those scenarios require a variety of sensors and sensor carriers working together. These systems should not only be able to function simultaneously without interference, but also to co-operate in order to perform complex tasks efficiently.

Another important aspect is the usability of these systems. UAVs, UGVs as well as many sensors usually have their dedicated control interfaces provided by their manufacturer. Adaptations to or exchange of these interfaces are usually very limited or impossible. Each system therefore requires its own trained operator. One reason for the hesitant use of robots in Fukushima, Japan was the high training effort for their operation.

To be able to control a multitude of vehicles and sensors in an ad hoc operation, a common easy-to-use interface is required. This leads to the conclusion that only with an integrated solution, consisting of an adaptable, universal ground control station, the use of diverse sensors and sensor carriers is feasible.

This paper presents a universal ground control station, which is based on an open integration framework for a wide range of sensors (both stationary and mobile) and sensor carriers (UAVs, UGVs, etc.). In the following section, related work is discussed. Thereafter, we present the architecture of the framework and demonstrate its versatility by elaborating on adding a new component (i.e., a smartphone). Finally, the paper concludes with a summary and a description of future work.

RELATED WORK

The integration of different sensors into one system has already been realized, e.g. for supervision systems. Those systems, however, are mainly customized individual solutions tailored to particular environments.

As for the control of various unmanned vehicles, related systems can be found in the defense sector. There has been a recent shift from dedicated solutions to universal control stations. The Pentagon, for example, began an effort in 2008 to break down the proprietary barriers between UAV systems and create a single ground control station that will fly all types of drones (Defense Industry Daily, 2011). The initiative tackles the problem that thousands of UAVs gather and distribute valuable data, but each system uses its own proprietary subsystem to control the air vehicle as well as receive and process the data. Yet commanders need access to information gathered by all types of UAVs that are flying missions in their area of operation.

Related approaches are the ground stations from Aerodrones (Aerodrones, 2012) and AII (AII Corporation, 2012). Both platforms are stand-alone control stations for multiple airborne drones. Aerodrones focuses on UAVs, whereas the AII Universal Ground Control Station (UGCS) is capable of controlling multiple unmanned aircraft, land vehicles or surface vessels. Further universal systems are the Raytheon Common Ground Control System (CGCS) (Raytheon, 2012) and the Defense Technologies common command and control software Open UMI (Defense Technologies, Inc., 2012). Open UMI is a core program which allows the end user to command and control multiple unmanned vehicles and/or sensors developed by other vendors from one station. The interfaces are based on military standards.

All of the above universal systems focus on controlling multiple UAVs, whereas some of them are also capable of handling other assets such as ground vehicles or surface vessels. Due to their intended purpose, the interfaces between control station and UAVs are based on a military standard, namely STANAG 4586 ("Standard Interfaces of UAV Control System (UCS) for NATO UAV Interoperability") (NATO Standardization Agency, 2007). This NATO standard defines interfaces for UAV command-and-control (C2).

Other than the above mentioned works, the following presented approach has a much broader focus. It provides a complete solution that brings together a wide range of heterogeneous sensors, both stationary and mobile, and a variety of unmanned vehicles. This framework is intended for civil applications and is based on open standards.

THE UNIVERSAL GROUND CONTROL STATION "AMFIS"

The use of very different sensors and sensor platforms at the same time is a challenging task. To deal with these problems, the complex surveillance and emergency response system AMFIS ("Aufklärung mit Miniatur-Fluggeräten im Sensorverbund", or reconnaissance with miniature aerial vehicles in a sensor network) (Leuchter, Partmann, Berger, Blum and Schönbein, 2007) was developed as a mobile and generic system. It delivers an extensive situation picture in complex surroundings - even in the absence of stationary infrastructure. In order to achieve maximum flexibility, the system is implemented open and mostly generalized so that different stationary and mobile sensors and sensor platforms can be integrated with minimal effort, establishing interoperability with existing and future assets. The system is modular and can be scaled arbitrarily or be tailored by choosing the modules suitable to the specific requirements.

The AMIFS system consists of a universal ground control station and a customizable set of sensors and sensor carriers (see Figure 1). In addition, there are interfaces to external exploitation stations and control centers. The ground control station is an adaptable prototype system for managing data acquisition with various sensors, mobile ad hoc networks and mobile sensor platforms. The main tasks of the ground control station are to work as an ergonomic user interface and as a data integration hub between multiple sensors and a super-ordinated control center. The sensors can be stationary or mounted on moving platforms such as micro UAVs (Bürkle, 2009), UGVs, aerostats or underwater vehicles. The system includes means to control different kinds of mobile platforms and to direct them to potentially interesting locations especially in areas with no prior sensor equipment. The actual AMFIS system is highly mobile and portable. It can be taken to and implemented at any location with relative ease. The sensor carriers of this multi-sensor system can be combined in a number of different configurations to meet a variety of specific requirements. The functions of the ground control station include: task management, mission planning, control of sensors and mobile platforms, situation awareness, fusion and exploitation of sensor data, reporting, generation of alarms and archiving.

The AMFIS system is predestined to support first responders in case of an emergency. Many of these events have very similar characteristics. They cannot be foreseen in their local and temporal occurrence. Situational in situ security or emergency response systems are usually not present. The data basis on which decisions can be made is quite slim at the beginning of a mission and, consequently, the present situation is rather unclear to the rescue forces. Especially in these situations, it is extremely important to understand the context as quick as possible to efficiently initiate the suitable measures. As an open system, AMFIS allows the integration of data from sources unknown up until the moment of the emergency.

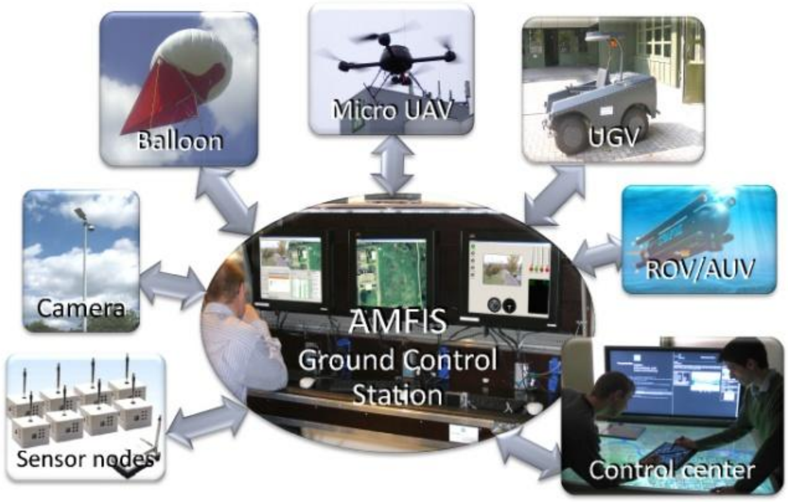

Figure 1. Overview of the AMFIS System

Applications of the AMFIS system are numerous. They include support of fire-fighting work with a conflagration, clarification of the debris and the surroundings after building collapses and search for buried or injured people. It increases the safety of rescue workers by helping with the detection of potential risks. Additionally, the system can be used to support the documentation and perpetuation of evidence during the cleaning out of the scene at regular intervals.

Advantages of an Integrated Solution

The benefits of an open and universal solution over dedicated platforms are manifold. First, an integrated system enables a coordinated use of different assets. By using several complementary sensors and/or sensor carriers a number of tasks can be performed in a shorter period of time. The search for casualties can be expedited by systematically scanning the area with multiple UAVs and therefore increases the chance of saving lives. Another example is the combination of aerial and ground vehicles. A UGV, which runs into a blockade, can be guided around the obstacle by a UAV that explores the surroundings.

Second, the integrated solution allows the fusion of collected data. The usually large amount of incoming data from different sources can be preprocessed, fused and displayed in a way that is easily comprehendible for the operator. The aggregated information helps provide a more comprehensive situation picture and supports decision making. The open framework also facilitates information dissemination through standardized interfaces.

Third, a scalable and extensible system such as the AMFIS control station has economic advantages. Components can be exchanged or updated at a relatively low cost without having to replace the whole system. The scalability allows it to grow with increasing demands. New technologies can be included without sacrificing assets already at hand. A unified user interface for controlling the integrated sensors and unmanned vehicles reduces the necessary training effort and allows simultaneous control of the assets with fewer operators.

THE OPEN ARCHITECTURE OF THE AMFIS SYSTEM

In order to create an open and generic system, one of the most important objectives was to design a sustainable architecture. One of the central demands for the system is the flexibility regarding new hardware, software and sensor components. The tasks for which the AMFIS system is developed are varied and the technological development during the recent years in related fields is enormous. Likewise, the number of micro UAV models is growing rapidly. Therefore, it is obvious that the AMFIS architecture must be able to master new demands resulting from future assets. To achieve this flexibility, the architecture was designed to not only be adaptable to components unknown today but also to be achieved with low expenditure.

The AMFIS ground control station's software architecture is basically 3-tiered following a pattern similar to the MVC (Model-view-controller) paradigm best known from web application development. The central application is the so-called AMFIS Connector (see Figure 2), which is a message broker responsible for relaying metadata streams within the network.

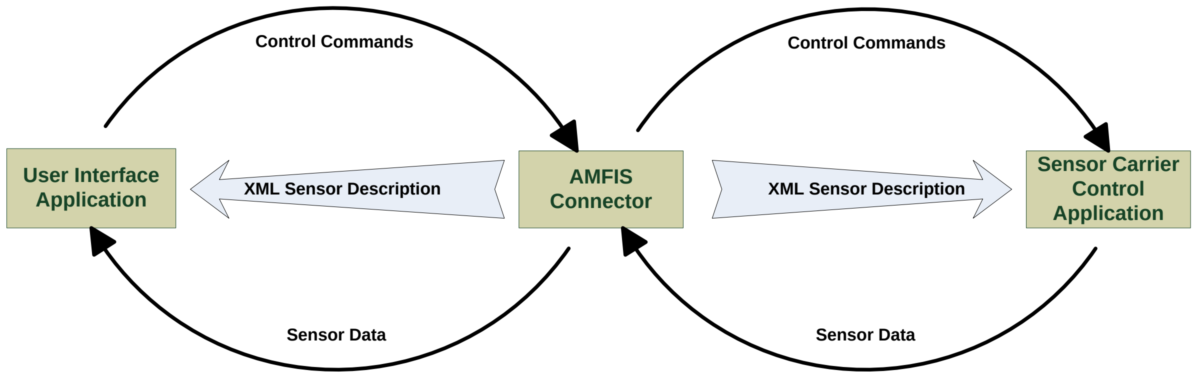

Figure 2. Software Architecture of the AMFIS Ground Control Station

To be able to manage a vast amount of sensors and sensor carriers, the physical sensors and sensor carriers are logically mapped onto the so-called sensorweb, a tree structure which contains virtual representatives of the actual units. The root node (sensorweb) is connected with a row of sensor networks, for example a set of PTZ (Pan-Tilt-Zoom) cameras. Each of these sensor networks consists of one or several sensor nodes, which correspond in each case to a physical sensor (for example a single PTZ camera). The sensor nodes themselves may again contain different sensors, for example a camera containing a compass.

The sensorweb is stored permanently in a database from which an XML document is generated at runtime. This is also done by the AMFIS Connector, which can be seen as the central information service of the ground control station.

The communication protocol within AMFIS is based on XML strings (see Figure 3), which are sent to TCP-Sockets. To simplify the use of this protocol and to provide different possibilities for software development, different implementations for different runtime environments (e.g. .NET, Java) are available, which enclose the XML management and allow an object-oriented view of the messages to the user.

```
<message key=”string” type=”string”>
  <parentmessage key=”string” />
  <timestamp>YYYY-MM-DD hh:mm:ss</timestamp>
  <originator type=”SENSOR|SENSORNODE|USER|SYSTEM”>
    <sensornodeid>bigint</sensornodeid>
    <sensorid>bigint</sensorid>
    <userid>bigint</userid>
  </originator>
  <subject
type=”SENSORNETWORK|SENSORNODE|SENSOR|SYSTEM”>
    <sensornetwork>bigint</sensornetwork>
    <sensornodeid>bigint</sensornodeid>
    <sensorid>bigint</sensorid>
  </subject>
  <value>value</value>
</message>
```

Figure 3. Example of an AMFIS XML Message

The implementation of the communication protocol is multicast-oriented; every incoming message is passed on by the Connector to all connected client applications. Each application decides itself if and how these messages are processed.

If a client application connects to the AMFIS Connector, it first receives the XML structure of the sensorweb followed by a steady stream of XML messages. Each of these messages contains metadata (e.g., sensor status data or control information), which originates from one of the sensors in the sensorweb or is meant to change the status of one or more sensor nodes. After successful establishment of a connection, the Connector supplies the client application with a constant stream of live sensor data.

A client application in this context is any application which includes one of the numerous AmfisCom implementations (.NET, Qt, Java) or implements the AMFIS message protocol directly.

The communication library (AmfisCom) builds the object tree from the XML data, providing the application developer with a type-safe and object-oriented view of the network of sensors and sensor carriers.

All applications within the AMFIS system, from graphic user interfaces to background system services, are designed as client applications:

- The various GUI applications of the user interface, most importantly the analyst's interface, the situation overview, the Photo Flight (Segor, Bürkle, Kollmann and Schönbein, 2011) or the pilot's interface. Those applications offer a visual representation of received metadata to the user, for example by displaying the current geographical locations of the various sensor carriers in the map and transmit commands to the sensor carriers e.g., a user-generated waypoint for a UAV.
- A number of services running in the background, notably the video server, offering time shifting and archiving for both video and metadata (more on video management within AMFIS see below) and the rule engine or the multi-agent system, both supporting the user by automating certain processes.
- Drivers for various sensor carriers e.g., a dedicated control software for UAVs, which translates high-level flight commands like waypoints into the proprietary RS232-based control protocol of the respective drone and in turn generates metadata XML status messages containing the current position, heading, remaining flight time etc.
- Interfaces to third-party applications or networks e.g., command and control centers.

While all metadata is sent as XML messages irrespective of their type (whether they are sensor measuring values, steering information or user generated announcements), the large amount of generated video data must be processed and stored differently. To do so, the Connector is tightly coupled with a server application called the video server. It is responsible for storing and distributing video streams, serving the dual purpose of providing time shifting capabilities to the network as well as reducing the load on the usually wireless links between sensor carriers and the ground control station. Since time shifting or archiving is not always required, this functionality was not integrated in the Connector itself in order to keep it as light-weight as possible. To store the generated mission data permanently, the video server is connected to a database, from which the data streams can be restored for playback.

In order to be able to transmit reconnaissance results to external systems, the stored video data or the live video streams can be accessed externally. The main disadvantage of this method is the lack of metadata generated along with the video stream. For reconnaissance tasks, additional data such as location, time, sensor carrier or sensor type are often vital. Hence, the video server offers the possibility to convert the video data into standardized video formats, which contain these metadata. This is done by the AMFIS Transcoder Process, which encodes the metadata and the video data into a STANAG 4609 compliant data stream. With this globally recognized NATO format, not only the imagery but also corresponding additional information is available to systems supporting this standard. The video streams generated by the AMFIS Transcoder can be stored as video clips in a CSD (coalition shared database) (Essendorfer and Müller, 2009) or be transferred in real-time to an exploitation system such as Fraunhofer IOSB's ABUL (Heinze, Esswein, Krüger and Saur, 2010).

To be able to receive messages or data information requests from an external system on top of the possibility to publish information, a communication module was implemented, which is called the XMPP Client. This client implements the Extensible Messaging and Presence Protocol (XMPP), a commonly used XML-based open-standard communications protocol for message-interchange. The XMPP Client translates incoming reconnaissance requests (e.g. a Region-Of-Interest) placed by external systems into the AMFIS message format.

A Smartphone as a Human-Carried Sensor

The software architecture described in the section above provides an easy way to integrate new systems and sensors as well as completely new technologies without the need to change the system's basic structure. In order to show how simple it is for a new technology to be incorporated, we describe the extension of the AMFIS system by a new sensor, which was not considered at the time of system design. The additional component encloses the full integration of a smartphone in the heterogeneous sensor network with the aim to use the mobile device's integrated sensors. It is embedded as a client and therefore also allows the user to access the data of the AMFIS system. This makes the person carrying the mobile device become a mobile sensor in AMFIS.

The basic functional structure of the new sensor can be divided into three modules. Besides the sensor itself, there is the web server, which is responsible for supplying services and the application server, which structures the communication between the sensor and the system (see Figure 4).

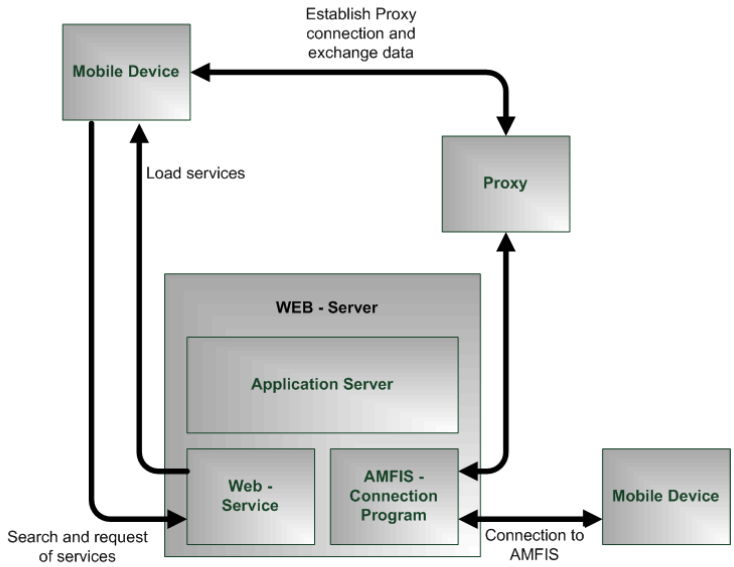

Figure 4. Smartphone Architecture

By providing the services through a web server, new functions can be added easily to the sensor system. Since data processing and communication infrastructure are separated, it is possible to have additional and more computationally intensive functions running on another computer system if necessary.

Splitting the architecture into component-based modules enables the development of the subsystems without affecting the already existing services. In addition, platform independence is achieved by using the functionalities of a web service. Thus, different types of smartphones with different operating systems can be used (e.g., Android, iOS, Windows Phone, etc.) without having to change the data processing functions. Only the client software on the mobile device has to be adapted.

Smartphone

To gain a maximum degree of independence from suppliers and other external influences and, at the same time, benefitting from the possibilities of an up-to-date smartphone, the Samsung Galaxy Nexus i9250 is used as a prototypical mobile device for AMFIS. It runs the Android™ 4.0 ("Ice Cream Sandwich") operating system and has a 1.2 GHz dual core processor.

The most important functionality which must be available on the mobile device is to provide the geo-specific context for the user. For this purpose, a GIS (Geographical Information System) (Tsou and Smith, 2011) was created as a basis for the user interface in the first stage of development. To be able to use the geographic information even if the data connection is lost, the application is designed as an off-line GIS with the ability to store the map data locally. However, with an available data link, additional information or new map tiles can be reloaded. The source of the data is variable; at the moment, the used data comes from the open source project OpenStreetMap (OpenStreetMap, 2012). These map data can be loaded from public web servers, but also from the AMFIS system's own Map Server. This allows, for example, quick spreading of highly up-to-date and geo-referenced aerial images as for example from an aerial reconnaissance flight (Segor et al., 2011)).

To provide the users with an overview of the whole situation they are dealing with, the sensor data of all stationary sensors is also visualized. In the future, this will be extended by regular updates of the positions of mobile sensors and sensor carriers.

In the current stage of development, the primary sensor of the smartphone to be used is the built-in camera. The user can both gather information and also transfer these data to the AMFIS system. Besides pictures, additional accompanying metadata, such as the origin of the image, the orientation and the footprint of the camera can be transmitted. To complete this workflow, the possibility to transfer short messages from the smartphone to AMFIS and vice versa was implemented.

To establish a connection between the smartphone and AMFIS, the available communication possibilities of the mobile device are used. In many scenarios, it can be assumed that the user stays in immediate vicinity of the AMFIS ground station. In these cases, the connection is established using Wi-Fi. However, a shortfall of connectivity cannot be excluded. Hence, the data transfer via GSM/UMTS is realized as an alternative communication method.

Nevertheless, the complete loss of the connection has to be considered as well. Therefore, the mobile device can also function as an independent stand-alone sensor. In this case, the accumulated data is buffered and transferred automatically with a re-established connection.

Web Server

Since the new sensor and its architecture are bound to already existing infrastructure and the independence from any operating system should be preserved, a service-oriented approach was chosen. By implementing this solution, a later integration of other platforms such as the Apple iPhone or a Windows Phone device is possible since only the application has to be adapted and/or compiled for the new platform. The functions and processes of the web service or the connection management system for AMFIS do not require any changes.

The web server provides the connection to the internet and offers the runtime environment for the application server. It can be used to pre-filter packages or carry out the authentication for the mobile device. The basic infrastructure is thereby provided by an Apache Http server.

Among other features, two web services were implemented to provide the basic functions. One service is used to transmit the user's data (e.g., photos and text messages) to AMFIS. The other one is an update service, which can be used by the smartphone to establish a connection.

The first service for transferring user's data is designed as a one way connection to be used exclusively by the smartphone to send data to the application server only. The application server sends an acknowledgment for the incoming data or messages. A transmission of data or messages accumulated by the application server is not possible with this web service. This service is only invoked if required e.g., if an image or a text message needs to be transferred.

The update service for the representation of the positions of sensors and sensor carriers is used to dispatch data from the application server to the smartphone (see next subsection). The smartphone uses this possibility to transfer not only its own position, but also any additional data accumulated so far. Another difference to the first web service consists of the fact that the smartphone calls this service periodically.

The AMFIS Connector software is therefore providing the suitable time slots according to the number of devices to be served. The smartphone is using the allocated time slot to change its own communication configuration accordingly. This is done to prevent a capacity overload, which might result in a data jam or even a breakdown of the communication infrastructure.

Within the update service, not only are messages dedicated to the mobile device fetched but also other information is transferred (for example the smartphone's position). The position information of the mobile device is communicated with every existing proxy connection but at the latest with an established connection to the update service.

Application Server

The application server provides the web services as well as the connection management to AMFIS. It contains the runtime environment for these functions and is realized by an Apache Tomcat server. The application server offers different web services which can be accessed by the smartphone. As soon as the smartphone has requested a service, a proxy is generated between the application server and the smartphone. This proxy runs only until the acknowledgement of the service and is terminated thereafter. The AMFIS Connector management software runs on the application server. The software contains an encoder which re-encodes the data received by the smartphone in suitable AMFIS messages and transmits these data to AMFIS. This process works also conversely using a suitable decoder to provide information to the mobile sensor. The management process is therefore able to decode AMFIS messages and to route the information to the receiving smartphone.

As it is to be expected that the mobile device will not be able to sustain a connection under all circumstances, an agent object for each registered smartphone is created. Within this agent, a stack of "messages to be transferred" exists, which contains all messages encoded in the smartphone application-readable format. Without any further transformation, necessary data can be transmitted faster in the event that a connection can be established. The messages remain in the list until reception of the data has been acknowledged. Even though, this increases the duration of the data exchange as well as the amount of data to be exchanged. However, this is necessary to prevent a possible loss of messages, which might be critical. If the connection is disturbed or terminated during the transfer process the data is not deleted but transferred with the next possible connection. In addition to these services, the agent object provides a certain supervision function. The agent is equipped with a timer, which is reset on each connection between the agent and the smartphone. If no connection to the smartphone can be established before timeout, the system can be informed about the connection state to the mobile device, which can be used for example, to notify the user of the ground station about the lost link.

Benefits of a Smartphone as Generic Mobile Device

By using smartphones as a mobile sensor within a heterogeneous sensor network new possibilities arise for emergency responders. Because of the wide acceptance of these technological platforms most people are already familiar with their use. The built-in sensors and communication capabilities can be used right on-site. The favorable price and the fact that they fit in every pocket makes it easy to use them for these kinds of applications.

The presented architecture used to integrate the smartphone into the AMFIS system allows the parallel use of a large number of different mobile devices, which makes it possible to cover a wider operational area at a minimum of costs.

Moreover, the use of the built-in sensors is just the first step in this development. With the different state-of-the-art interfaces available in these phones, it is conceivable that other mobile sensors, which lack a communication infrastructure of their own or suffer from a disturbed communication line, can be connected.

An example is the gas sensor payload AirSens (see Figure 5), which was developed at the Fraunhofer IOSB in cooperation with Leopold Siegrist GmbH (Leopold Siegrist GmbH, 2012). It is a small modular gas detection payload designed to be carried by a micro UAV. The development focused primarily on the modularization and interchangeability of the sensors as well as of the communication interface, so that the final product can be used for most different purposes.

Figure 5. UAV Payload "AirSens"

To allow data transmissions between the payload and the mobile device, the communication board, which is interchangeable on the data-processing board, was exchanged. In this way the sensor could quickly be altered with very low developing expenditure from a data transfer channel with 868 MHz into a Bluetooth device. With the altered interface a connection to the Smartphone can be established. Hence, the payload is transformed into a hand-held device with a smartphone as an analyst unit on-site. In addition, the onboard long range communication possibilities of the mobile device can be used to report back possible hazardous pollution and their expansion directly to the central controlling station or to other mobile devices in the network.

Conclusion and Future Work

To gain the most use out of the increasing number of available sensors and sensor carriers (e.g. unmanned aerial vehicles) in an emergency response scenario, their use has to be coordinated. Therefore, an open integration framework was proposed. The ground control station AMFIS presented in this paper is a prototype implementation of this framework.

As the latest add-on, a smartphone has been integrated into the heterogeneous sensor network of the AMFIS framework. The functions realized so far are only the most essential and basic processes within this subsystem to render it usable. With a smartphone, it is not only possible to give first responders mobile access to the AMFIS system, but also to connect other third-party sensors that lack adequate communication or processing capabilities to the network.

Future development of the AMFIS system will include further components. In addition to the binding of external sensors, the improvement of the communication range of an ad-hoc infrastructure is an important issue to be solved. The integration of a meshed Wi-Fi network is inevitable here. However, this is a feasible challenge as a solution has been described in (Seither, D., König, A. and Hollick, M., 2011, OLSR, 2012).

Another topic for future work is the security of the communication. It must be guaranteed that external communication partners cannot access system data, and that unauthorized communication devices are prevented from getting access to the infrastructure. Beside the menaces from the outside, a roles or access management process is inevitable to keep the data rates low and to avoid an information overload to a single person. Besides, it must be assured that only persons who need (and are allowed to see) these data are granted access. Thus, an access group distribution makes sense, such as, for example, the one implemented on the European digital radio network of the rescue forces TETRA BOS (Linde, 2008).

ACKNOWLEDGMENTS

The authors would like to thank Torsten Großkurth and Judy Lee-Wing for their contribution.

REFERENCES

1. AAI Corporation (2012) http://www.aaicorp.com
2. AERODRONES (2012) http://www.aerodrones.com
3. Bürkle, A. (2009) Collaborating Miniature Drones for Surveillance and Reconnaissance, *Processings SPIE Europe Security+Defence*, Berlin, Germany.
4. Defense Industry Daily (2011) It's Better to Share: Breaking Down UAV GCS Barriers, Oct 03, 2011, http://www.defenseindustrydaily.com/uav-ground-control-solutions-06175/
5. Defense Technologies, Inc. (2012) http://www.dtiweb.net
6. Essendorfer, B. and Müller, W. (2009) Interoperable sharing of data with the Coalition Shared Data (CSD) server, in: *North Atlantic Treaty Organization (NATO)/Research and Technology Organization (RTO): C31 in Crisis, Emergency and Consequence Management.*
7. Heinze, N., Esswein, M., Krüger, W. and Saur, G. (2010) Image exploitation algorithms for reconnaissance and surveillance with UAV, *Processings SPIE Vol. 7668, Airborne intelligence, surveillance, reconnaissance (ISR) systems and applications VII.*
8. Leopold Siegrist GmbH (2012) http://www.siegrist.de/en/product-overview.html.
9. Leuchter, S., Partmann, T., Berger, L., Blum, E.J. and Schönbein, R. (2007) Karlsruhe generic agile ground station, in: Beyerer, J. (ed.) *Future Security. 2nd Security Research Conference*, Karlsruhe, Germany. Fraunhofer Defense and Security Aliance, Universitätsverlag, Karlsruhe, pp. 159-162.
10. Linde, C.(2008) Aufbau und Technik des digitalen BOS-Funks, Franzis Verlag, ISBN 3-7723-4216-7.
11. NATO Standardization Agency (2007) STANAG 4586: Standard Interfaces of UAV Control System (UCS) for NATO UAV Interoperability, Edition 2.
12. OLSR (2012) Optimized Link State Routing, http://www.olsr.org/

13. OpenStreetMap (2012) http://www.openstreetmap.org/
14. Raytheon (2012) http://www.raytheon.com.
15. Segor, F., Bürkle, A., Kollmann, M. and Schönbein, R. (2011) Instantaneous Autonomous Aerial Reconnaissance for Civil Applications - A UAV based approach to support security and rescue forces, *The 6th International Conference on Systems ICONS 2011*, St. Maarten, The Netherlands Antilles.
16. Seither, D., König, A. and Hollick, M. (2011) Routing Performance of Wireless Mesh Networks: A Practical Evaluation of BATMAN Advanced, *Processings 11th IEEE International Workshop on Wireless Local Networks (WLN'11).*
17. Seither, D., König, A. and Hollick, M. (2011) Sicherheitsmechanismen und Bedrohungsanalyse des Standards für drahtlose Mesh-Netze IEEE 802.11s, *Processings Praxis der Informationsverarbeitung und Kommunikation (PIK).*
18. Tsou, M.-H. and Smith, J. (2011) Free and Open Source Software for GIS education, A White Paper, National Geospatial Technology Center of Excellence.