Open Geospatial Consortium

Publication Date: 2016-10-05

Approval Date: 2016-10-05

Posted Date: 2016-10-05

Reference number of this document: OGC-16-014r2

Reference URL for this document: http://www.opengis.net/doc/PER/{short_doc_name}

Category: Engineering Report

Editors: Greg Schumann, Josh Lieberman

# Incident Management Information Sharing (IMIS) Internet of Things (IoT) Architecture Engineering Report

**The research in this presentation was conducted under contract with the U.S. Department of Homeland Security (DHS) Science and Technology Directorate (S&T), contract # HSHQDC-13-C-00119. The opinions contained herein are those of the contractors and do not necessarily reflect those of DHS S&T.**

## Warning

This document is not an OGC Standard and may not be referred to as an OGC Standard. This document presents a discussion of technology issues considered in an initiative of the OGC Interoperability Program. The content is subject to change without notice. The content of this document does not represent an official position of the OGC.

License Agreement

| | |
|---|---|
| Document type: | OGC® Engineering Report |
| Document subtype: | NA |
| Document stage: | Approved for public release |
| Document language: | English |

Contents Page

Figures Page

## Abstract

The Incident Management Information Sharing (IMIS) Internet of Things (IoT) Pilot established the following objectives:

- Apply Open Geospatial Consortium (OGC) principles and practices for collaborative development to existing standards and technology to prototype an IoT approach to sensor use for incident management.
- Employ an agile methodology for collaborative development of system designs, specifications, software and hardware components of an IoT-inspired IMIS sensor capability.
- Develop profiles and extensions of existing Sensor Web Enablement (SWE) and other distributed computing standards to provide a basis for future IMIS sensor and observation interoperability.
- Prototype capabilities documented in engineering reports and demonstrated in a realistic incident management scenario.

## Business Value

The IMIS IoT Pilot aimed to develop, test and demonstrate the use of networked sensor technologies in a real-world scenario, developed in collaboration with the Department of Homeland Security (DHS) Science and Technology Directorate (S&T) and first responder stakeholders. This pilot demonstrated an IoT approach to sensor use for incident management. Prototype capabilities included ad hoc, nearly automatic deployment, discovery and access to sensor information feeds, as well as derivation of actionable information in common formats for use in Computer Aided Dispatch (CAD), Emergency Operations Center (EOC) and Geographic Information Systems (GIS), as well as mobile devices.

**Incident Management Information Sharing Internet of Things Architecture Engineering Report**

# 1 Introduction

## 1.1 Scope

The Incident Management Information Sharing (IMIS) Internet of Things (IoT) Pilot developed, tested and demonstrated the use of networked sensor technologies in a real-world scenario developed in collaboration with the Department of Homeland Security (DHS) and first responder stakeholders. Among the types of sensors tested were in situ environmental sensors, wearable sensors and imaging sensors on mobile platforms such as Unmanned Aerial Vehicles (UAV) and autonomous vehicles. The key objectives of the IMIS IoT Pilot were:

- Apply IoT principles to sensing capabilities for incident management.
- Test the feasibility of ad hoc sensor deployment and exploitation by first responder groups (e.g., law enforcement, fire, emergency medical and emergency management).
- Prototype a standards-based architecture for sensor-derived situational awareness that is shared across multiple responder organizations.
- Create IoT specifications and best practices for incident management through a process of broad collaboration among stakeholders, rapid iterative development and running code.

## 1.2 Document Contributor Contact Points

All questions regarding this document should be directed to the editor or the following contributors:

| Name | Organization |
|---|---|
| Greg Schumann | Exemplar City, Inc. |
| Josh Lieberman | Tumbling Walls |
| Simon Jirka | 52°North Initiative for Geospatial Open Source Software GmbH |
| Marcus Alzona | Noblis |
| Farzad Alamdar | The University of Melbourne |
| Mike Botts | Botts Innovative Research Inc. |

| | |
|---|---|
| Roger Brackin | Envitia |
| Chris Clark | Compusult |
| Flavius Galiber | Northrup Grumman Corporation |
| Mohsen Kalantari | The University of Melbourne |
| Steve Liang | SensorUp |

## 1.3 Revision History

| Date | Release | Editor | Primary Clauses Modified | Description |
|---|---|---|---|---|
| 2015-11-05 | 0.5 | Greg Schumann | All | First draft |
| 2016-03-08 | 0.6 | Greg Schumann | All | Posted draft |
| 2016-08-05 | 0.7 | Josh Lieberman | All | Editorial update and fill-in |
| 2016-08-18 | 0.8 | Josh Lieberman | All | Response to DHS review |
| | | | | |

## 1.4 Future Work

This Engineering Report (ER) is intended to provide recommendations on the development of IMIS profiles of different OGC standards. Thus, the recommendations on future work can be found at the end of each section.

## 1.5 Pilot Overview

This pilot explores the emerging IoT enabled technologies as a means to provide first responders with better situational awareness and communications. Evolving IoT technologies now make it possible to establish basic network connectivity automatically with these sensors as soon as they are deployed. Basic connectivity is not enough, however. Actionable observations, analysis, alerts and predictions are needed. They must be easily discoverable and accessible from emergency response information systems and mobile devices to provide a dynamic and shared view of changing conditions. Standards are needed to make sensors easily and immediately identifiable, accessible, usable and useful across all teams (on-scene and Operation Centers) and information management platforms joining an incident response.

### 1.6 OGC Features and IoT Things

Although IoT principles and practices have been developed largely for the Worldwide Web and Industrial Internet communities, an IoT approach to sensors and sensor observation data is a natural one for a standards organization such as OGC that has been developing standards for publishing geospatial Web services and enabling webs of sensors for more than a decade. The key bridging concept is that the “things” which IoT connects to the Internet and Web are precisely the defined, located real world features that form the basis of almost all OGC standards. IoT eases the difficulty of working with networked sensor information, while OGC contributes the necessary rigor to define the Things being observed and the properties of Things being measured. The combined SWE-IoT architecture implemented in this Pilot activity is intended to leverage and reconcile both sets of standards and engineering practices.

### 1.7 Forward

This Engineering Report (ER) documents the SWE-IoT Architecture used in the IoT Pilot demonstration. The SWE-IoT Architecture includes networked sensors to quickly make a wide range of pertinent observations of an incident environment and its effects on people, including responders themselves.

## 2 Architecture

### 2.1 Layered Protocols – Segmentation and Integration

Establishing interoperable standards are a central requirement for enabling ad-hoc integration of sensor resources (e.g., catalogs, data access services, portrayal services). To ensure coverage of a broad range of use cases and application domains, many OGC standards were intentionally defined in a flexible manner. This leads to the need to specify profiles for specific application domains to restrict this flexibility and thereby further increase interoperability. Recommendations on such profiles are provided in the IMIS Profile Recommendations for OGC Web Services ER. During the IMIS IoT Pilot, the following standards were used:

- OGC Web Service Common (OWS Common; OGC 06-121r9): Specification of common aspects for all OGC interface standards. These include, for example, the definition of the GetCapabilities operation and the Capabilities response structure, as well as the definition of principles for XML (eXtensible Markup Language) and KVP (Key-Value Pair) encodings of operation requests and responses.

- OGC Catalog Service Implementation Specification (OGC 07-006r1): This interface specification enables clients to publish and/or discover geospatial datasets as well as services. It provides the metadata needed to decide whether specific datasets or services can be used by clients to fulfil a certain goal. The standard specifies interfaces and bindings for publishing and accessing digital catalogs of metadata for geospatial data, services and related resource information. The interface provides operations for managing the metadata records, e.g., for harvesting records, discovering metadata records, describing record types or querying certain records.

- OGC Web Map Server (WMS; OGC 06-042): The OGC defined the WMS standard for publishing and retrieving maps as images (e.g., providing background maps or pre-rendered satellite data). A WMS server lists its available map layers in the Capabilities document and allows retrieval of these layers with several query parameters (e.g., Bounding Box, using the GetMap operation). The optional GetFeatureInfo operation provides additional information about the features located at a certain pixel.

- OGC Web Feature Service (WFS; OGC 09-025r1/ISO 19142): The WFS standard specifies a service interface for retrieving geographic features (vector data) encoded in GML (Geographic Markup Language). The supported feature types are listed in the capabilities document. The DescribeFeatureType operation provides a description of a specific feature type. The central operation is the GetFeature operation which allows users to query features from a WFS server. Further optional operations are available, such as the Transaction operation for inserting, updating or deleting features.

- The OGC Sensor Observation Service (SOS; OGC 12-006): The OGC SOS standard was developed to provide a standard web service interface for accessing sensor observations. The standard also provides the ability to retrieve a sensor system description using a DescribeSensor request which typically returns a SensorML document. Two alternatives exist for retrieving observation values from SOS. The GetObservation operation is the core operation for retrieving observations using several filters for different observation properties. The response of a successful GetObservation request is an Observation which includes metadata about the observations as well as one or more measurement values. An alternative approach for retrieving archived or real-time measurements is to use the combination of GetResultTemplate and GetResult requests. The latter approach was designed to provide maximum efficiency for accessing results, including complex tuple or time series data, provided as single values, large blocks of ASCII or binary data, or streaming data. The response of a GetResultTemplate is Sensor Web Enablement (SWE) Common Data description of the data components, data structure and data encoding. This only needs to be called once for the client software to then understand how to parse the data values. The GetResult request returns only the requested data values with no metadata.

  Several extensions exist for transactional data publication or result handling in case the same request and response metadata should not be repeated in each request and response message. The Transactional Sensor Observation Service (SOS-T) transactions operation enables a sensor and/or sensor hub (S-Hub) to push measurements into a local or remote SOS-T instance, residing perhaps in the cloud.

- 52 North (52N) and Open Sensor Hub (OSH): Two open source software stacks that implement and provide ease of deployment of OGC SWE standards such as SOS, Sensor Planning Service (SPS), SensorML, Observations and Measurements (O&M), and SWE Common. Both were deployed by various teams to meet the needs of this scenario. This highlights one of the advantages of using open standards such as SWE. One is not solely dependent on one software stack or one software vendor, as long as the various software components are built upon and are compliant with the approved standards.

- OGC Sensor Things Application Programming Interface (STA): This standard (currently in the adoption vote process) provides an interface for the retrieval of observation data relying on O&M. In contrast to SOS, STA services are based fundamentally on Representational State Transfer (REST) principles and specify JavaScript Object Notation (JSON) as the default encoding for observations. As such, it is particularly Web-friendly and lightweight. STA services make it relatively easy to develop browser-based IoT client applications.

## 2.2 Sensor Hub (S-Hub) Services

S-Hub Services are gateways between one or more local sensor devices on one side and Internet users of the sensors on the other. S-Hubs provide standard, predictable and interoperable access to minimally connected, often proprietary sensor devices and are vital to both SWE and IoT. Within the SWE-IoT architecture, S-Hubs are components implemented as software stacks that conform to OGC SWE standards as well as other industry standards, in order to fill this mediation role. S-Hubs provide standard protocols and encodings for accessing real-time or archived observations, as well as for tasking sensor or actuator systems.

Two overlapping flavors of S-Hubs were developed in the IoT Pilot: those mainly supporting the original OGC SWE suite of services and those focusing on the REST-based approach of STA services. Both S-Hub types supported real-time and archived observations and complemented each other’s capabilities.

Software stacks that implement S-Hub services can typically be deployed on a range of platforms and at a range of scales. For instance, OSH has been deployed on platforms ranging from Android smartphones, tablets, and microcontroller boards (e.g. Raspberry Pi and Arduino), to Linux/Windows/OS-X devices, and the Amazon Web Service (AWS) Cloud. Specific computing hardware may be required to support specific network protocols and sensor devices. It is primarily the software implementation of open standards, however, that enables the interoperability required for easily deployed sensor web and IoT.

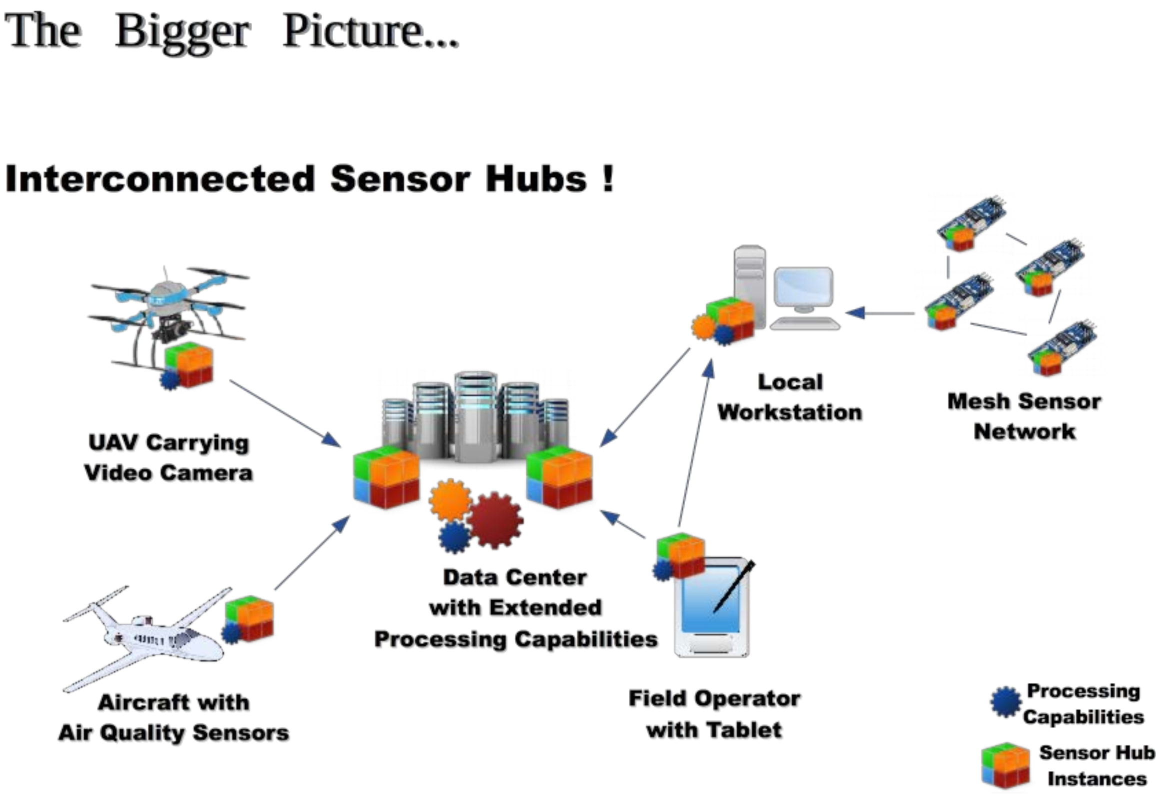

The Bigger Picture...
Interconnected Sensor Hubs !
UAV Carrying Video Camera
Local Workstation
Mesh Sensor Network
Data Center with Extended Processing Capabilities
Aircraft with Air Quality Sensors
Field Operator with Tablet
Processing Capabilities
Sensor Hub Instances

**Figure 2-1: The Distributed and Hierarchical Nature of S-Hubs**

One of the powerful aspects of the design and implementations of OGC-standard S-Hubs, as shown in Figure 2-1, is that they can be distributed throughout the global environment and can be hierarchically deployed. Thus, an S-Hub might be deployed onboard an Arduino-based sensor system providing tasking and observation capabilities. This S-Hub might be one of hundreds of S-Hubs that are managed and made accessible to the public through a local or regional S-Hub, while an S-Hub deployed in the cloud might receive observations from a collection of locally deployed S-Hubs in order to provide large-scale persistent storage and advanced processing. Processing and data storage can occur anywhere within such a distributed architecture thereby allowing one to configure particular data access, tasking, processing and storage capabilities on whichever S-Hubs make the most sense.

The design of these S-Hubs provides important scalability. This scalability allows the S-Hub software to be deployed and configured on platforms ranging from simple microprocessor boards to cloud-based services, and allows the S-Hub to support very simple, every day, mass market sensors to highly specialized and complex national sensor systems.

Another important aspect of the S-Hubs is their ability to support the original purpose of the sensor deployment while still being able to meet additional needs. For example, with proper authorization, a video camera deployed for store security could be repurposed to view an emergency event. Similarly, a laser rangefinder to measure remote locations could be repurposed to task a web camera to look at a particular geospatial location or guide an Unmanned Aerial System (UAS) to a particular place.

## 2.3 IoT, Web of Things (WoT) and O&M

The essence of IoT principles is that real world things have corresponding identities on the Internet, i.e. Internet Protocol (IP) addresses, that support access to the values of Thing properties, whether those are temperature, color, or simply how the Thing appears in an image. The most common implementations of this, however, focus on the use of URL's and Web protocols to provide identity and access; this specialization of IoT is known as the Web of Things or WoT. The connectedness of WoT carries additional benefits, however, in expressing the OGC O&M model for the various types of interrelated information that make up the process of sensing and measuring properties of real world Things. The SWE-IoT Pilot architecture leverages these benefits in several ways, especially in the context of STA services that provide URL links explicitly for such connections.

## 2.4 Registration and Discovery

Key to the IMIS IoT is the registration and subsequent discovery of sensors allowing them to be exploited. A number of key components work together to provide the infrastructure for discovering sensors; these are the S-Hub, the Hub Catalog (HubCat) and the Web Registration Processing Service (WRPS).

- The S-Hub provides either an OGC SOS compliant interface or an OGC SensorThings API (STA) compliant interface. SOS includes the key request 'GetCapabilities' which reports both the functional capabilities (supported functions) and the primary content of the service. SensorThings services have systematically defined URLs which also allow users to retrieve the functions and content of the service. This allows SOS and STA service metadata information to be harvested and catalogued.

- The second component, the HubCat, provides the metadata record management and discovery interfaces that enable clients to identify relevant services. It is not enough to just find a service; for a client to operate effectively it needs to be able to qualify the relevance of sensors accessible through a given S-Hub quickly; the HubCat is the key to this capability.

- The Publishing Service WPS, which is a sort of utility service, facilitates the population of HubCat metadata records. Although it would be possible for an S-Hub to populate the HubCat itself, this is a relatively complex process requiring strict metadata record formatting and interpretation of service information to generate key semantics, such as the phenomenon type that a particular sensor measures. The WRPS service takes the SOS or STA URL, harvests the required metadata and populates the HubCat, returning the Universally Unique Identifier (UUID) of the entry to the client. The trigger to initiate this service is typically the S-Hub, although it could be another actor depending on the architecture. The WRPS is also involved in "de-registering" sensors and S-Hubs in dynamic ad hoc sensor network environments.

The HubCat is the primary focus for clients (either user interfaces or other web services) to discover S-Hub services, sensors, and sensor observations. The HubCat for the Pilot was a component of Compusult's Web Enterprise Suite (WES). This component is an implementation of the OGC Catalog Services for the Web (CSW) standard, v2.0.2. As its datastore model, it uses the OASIS-standard ebXML Registry Information Model (ebRIM 3.0) and so is termed a CSW-ebRIM service. The flexibility of this standard means that there is no bespoke element required to be coded into the catalog software for a particular type of metadata, simply the loading of an information model configuration, known as an ebRIM Registry Extension Package (eREP). The eREP defines classification schemes, record types, associations, and other structural elements needed to catalog sensors. It can be thought of as similar to a relational database Schema definition (DDL).

Ingestion of sensor and service metadata then involves mapping the metadata elements into the eREP defined elements. Both the eREP and the mappings evolved during the Pilot, but these may be formalized as a standard profile of the ebRIM mode for HubCat implementations to maintain broad interoperability among themselves.

### 2.4.1 Registration Process

The overall registration process is shown in the sequence diagram below (Figure 2-2). When a S-Hub boots/comes online, it sends a request to the publishing service which then harvests the S-Hub capabilities and populates the catalog as necessary. The publishing service returns the UUID of the entry so that the S-Hub can subsequently update or remove the entry as its status changes.

This workflow depends on the S-Hub knowing to which catalog or publishing service it needs to connect. An alternative is an external trigger which performs the 'Add' request, which might be r ances.

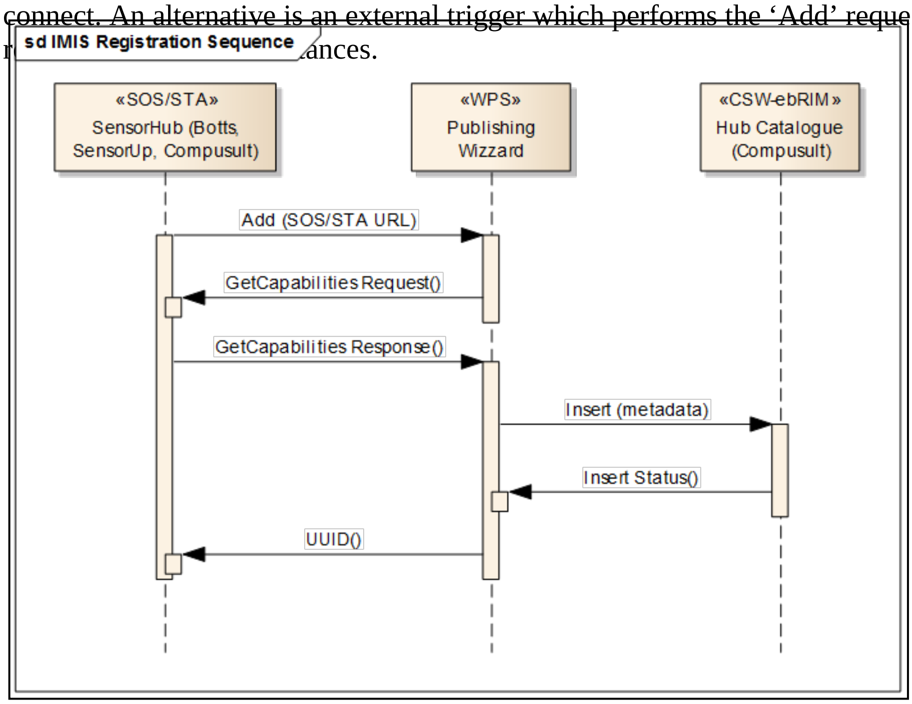

sd IMIS Registration Sequence
«SOS/STA»
SensorHub (Botts, SensorUp, Compusult)
«WPS»
Publishing Wizzard
«CSW-ebRIM»
Hub Catalogue (Compusult)
Add (SOS/STA URL)
GetCapabilities Request()
GetCapabilities Response()
Insert (metadata)
Insert Status()
UUID()

**Figure 2 -2: Registration Process**

### 2.4.2 Update Process

The update process is initiated by the S-Hub requesting an update using the ID returned during the registration process, as shown in Figure 2-3.

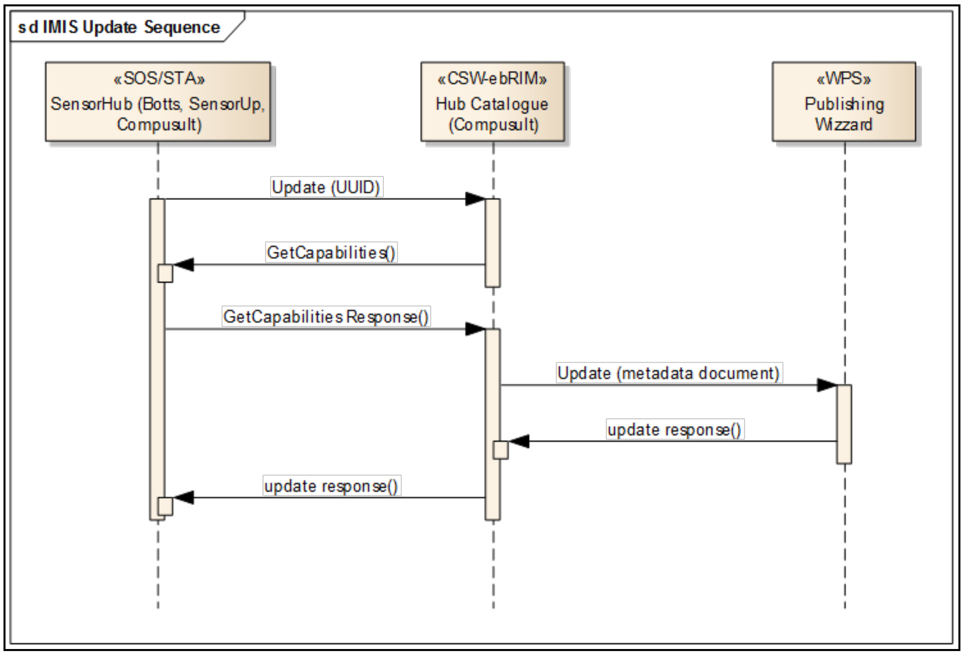


**Figure 2-3: Catalog Update Process**

### 2.4.3 De-Registration Process

A similar process occurs when S-Hubs shut down. They will initiate a de-registration process using the ID returned during registration (Figure 2-4). The result is that the HubCat will only show sensors that are currently registered (and by implication operational).

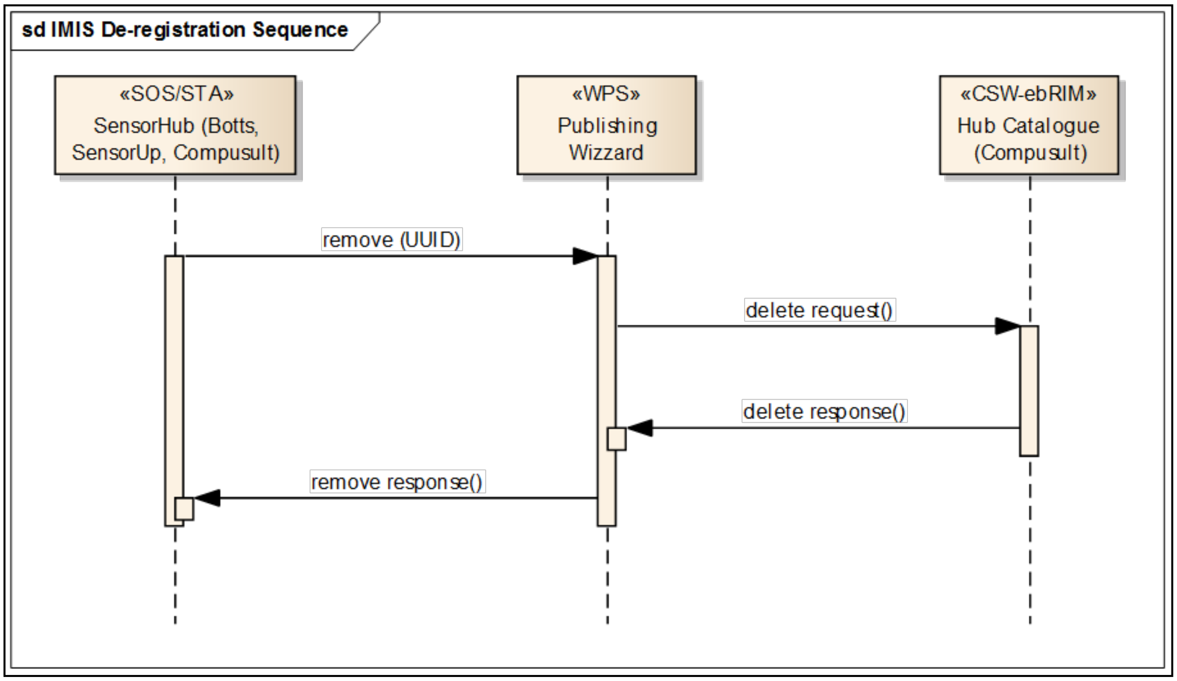


**Figure 2-4: Sensor De-Registration Process**

Of course, while de-registration could remove an S-Hub record from the HubCat (the method used in the experiment), it could also just mark it as offline or manage its availability in other ways. This is a decision related to the permanence of the sensor and the need to keep records of sensor availability/use.

Compusult's implementation of the HubCat will poll any registered services at a configurable rate, and change the status of the service from online to offline or vice versa if required.

### 2.4.4 Catalog WMS

Both Compusult and Envitia CSWclients were used to access the HubCat directly, but the limited number of clients able to interact with a CSW-ebRIM service was mitigated by the additional provision of a CSW-linked Web Map Service (CSW-WMS) which provided, in effect, access to the catalog contents as phenomenon-specific map layers. Clients could select specific phenomena and visualize all sensors from all registered S-Hubs providing measurements for those phenomena. The WMS GetFeatureInfo operation returned more detailed information for a given measurement on the map, including links to invoke a complete SOS Web client to view the observation records. Envitia exploited this in the demonstration for use in lightweight and mobile clients even though a CSW-ebRIM client search was available in the Envitia Client. The CSW-ebRIM query is useful for advanced queries, but the CSW-WMS provides a useful graphical shorthand access to available data for each phenomenon type.

### 2.4.5 Discovery Process via Catalog Services

Once 'auto-registered,' all SOS services, STA services and WMS services deployed before or during an incident can be discovered via the HubCat. Because the HubCat implements the CSW-ebRIM service, it is extremely flexible and technically able to catalog significantly more artifact types than just sensors. In fact, the HubCat was used to also catalog available maps, OGC Web Services (OWS) Context documents and other artifacts discussed later.

A wide range of other service catalogs may exist, of course, keeping track of a wide range of framework or other geospatial data.  Envitia also provided a CSW-ebRIM service to catalog geospatial data and imagery as background to the incident location (HubCat2). Because its interface conformed to the same standard as that provided by the HubCat, and the discovery client already accessed the Envitia CSW-ebRIM service, it was possible with very little configuration to allow it to access both vendors' services. Situations with multiple catalogs are likely in actual deployments and so the configuration offered a useful demonstration.

The sequence in Figure 2-5 shows the initial discovery of information from both HubCat and HubCat2, and the subsequent storage of an OWS Context document (describing the collection of information assembled from the discovery process by the client user) in HubCat2, although this could also be stored in the Compusult HubCat.

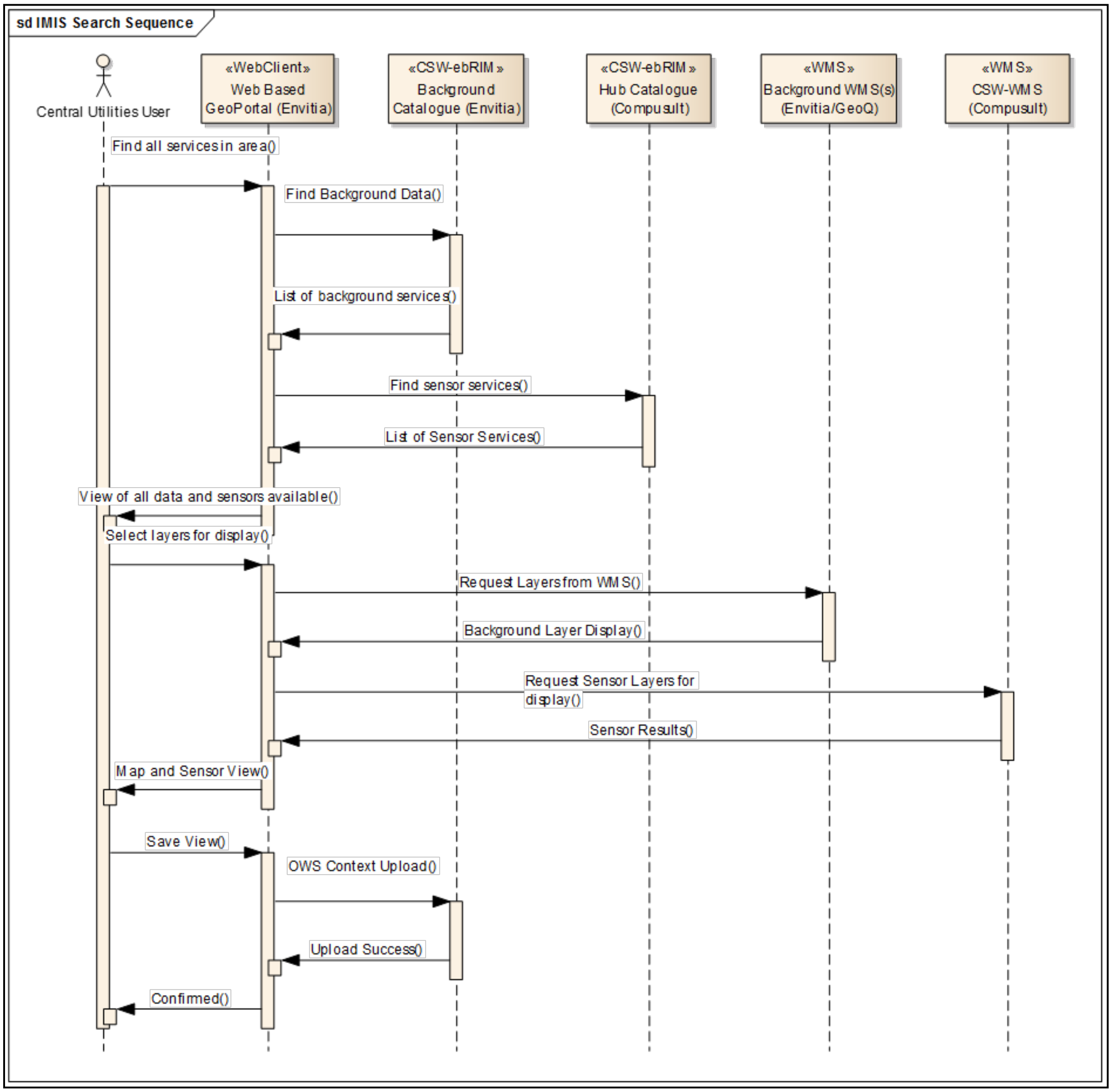

sd IMIS Search Sequence
Central Utilities User
«WebClient» Web Based GeoPortal (Envitia)
«CSW-ebRIM» Background Catalogue (Envitia)
«CSW-ebRIM» Hub Catalogue (Compusult)
«WMS» Background WMS(s) (Envitia/GeoQ)
«WMS» CSW-WMS (Compusult)
Find all services in area()
Find Background Data()
List of background services()
Find sensor services()
List of Sensor Services()
View of all data and sensors available()
Select layers for display()
Request Layers from WMS()
Background Layer Display()
Request Sensor Layers for display()
Sensor Results()
Map and Sensor View()
Save View()
OWS Context Upload()
Upload Success()
Confirmed()

**Figure 2-5: Discover Process from the Catalog**

Note the Envitia Client actually performs a relatively complex federated query. The user asks for all data in a given geographic area. The query is issued independently to both catalog services which subsequently respond. The Envitia Client then combines the results and presents them on the map display and in list form. This avoids the user needing to query each catalog in turn.

An alternative model, which is common to simplify the client, is to have a federating catalog service (again a CSW-ebRIM service) which is a single point of presence but, when a query arrives, distributes it out to all connected catalogs.

### 2.4.6 Discovery via OWS Context Documents

Another key method of discovery used during the experiment was discovery using contextual views. This delineates discovery and search. The concept is that one user or group of users, typically in a command center, prepare views and save them into the catalog. These views can then be discovered and loaded by mobile users, for example. Search in this case can be very simple; for example, finding all context documents in a specific area would be much more limited that finding all data. This was the process used in the experiment where the utilities user, using the Envitia InSight Client, simply searched for views and loaded them (Figure 2-6).

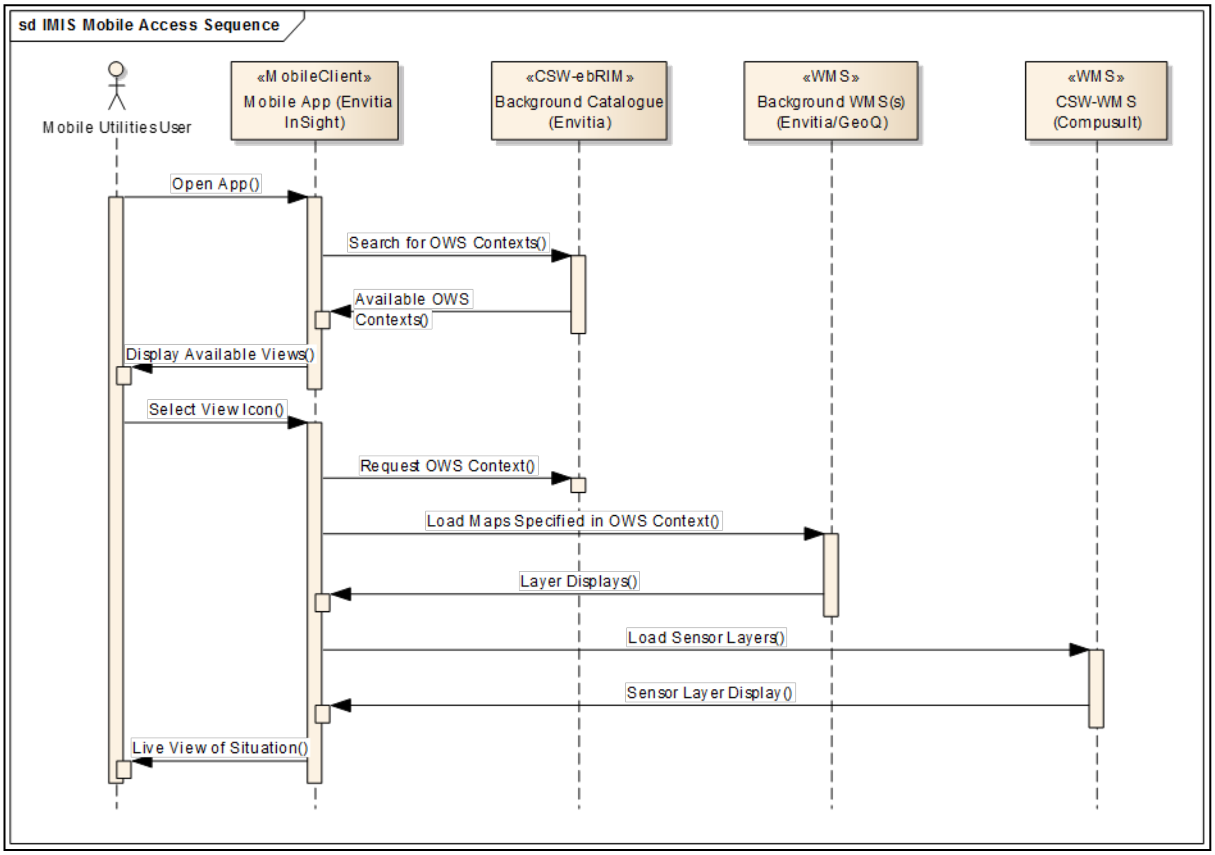

sd IMIS Mobile Access Sequence
Mobile Utilities User
«MobileClient» Mobile App (Envitia InSight)
«CSW-ebRIM» Background Catalogue (Envitia)
«WMS» Background WMS(s) (Envitia/GeoQ)
«WMS» CSW-WMS (Compusult)
Open App()
Search for OWS Contexts()
Available OWS Contexts()
Display Available Views()
Select View Icon()
Request OWS Context()
Load Maps Specified in OWS Context()
Layer Displays()
Load Sensor Layers()
Sensor Layer Display()
Live View of Situation()

**Figure 2-6: Discovery via OWS Context Documents**

To move away from search altogether, other options include direct emailing of an OWS Context document to a user, or the more advanced action of setting up Communities of Interest (COI) within the catalog. The mobile user can log in as a community member and will see icons which show the views in that community, providing immediate access the relevant operational view. This process was demonstrated in the desktop environment with the Envitia Horizon GeoPortal using a single COI.

## 2.5 Events and Notification

Figure 2-7 shows an overview of the event notification architecture. Central element is the Web Processing Service for Event Processing (Web Event Processing Service, WEPS) which controls the overall workflow. On the one hand, the WEPS receives from the client event subscriptions through WPS Execute requests. On the other hand, it controls the event processing module which performs the analysis and pattern matching of incoming sensor data streams against the event pattern rules contained in the event subscriptions. To push all relevant new observations into the event processor, a feeder is used which regularly checks a data source (in this case SOS servers) for new observations. As soon as a new observation is available, it is pushed into the event processor. Finally, the output of the event processor (i.e., all detected events that match to a subscription) are sent to the Notification Store. This is an RSS-based component so that clients can consume RSS-feeds containing those notifications that correspond to their subscriptions. A more detailed explanation of this architecture is provided in the IMIS Profile Recommendations for OGC Web Services ER.

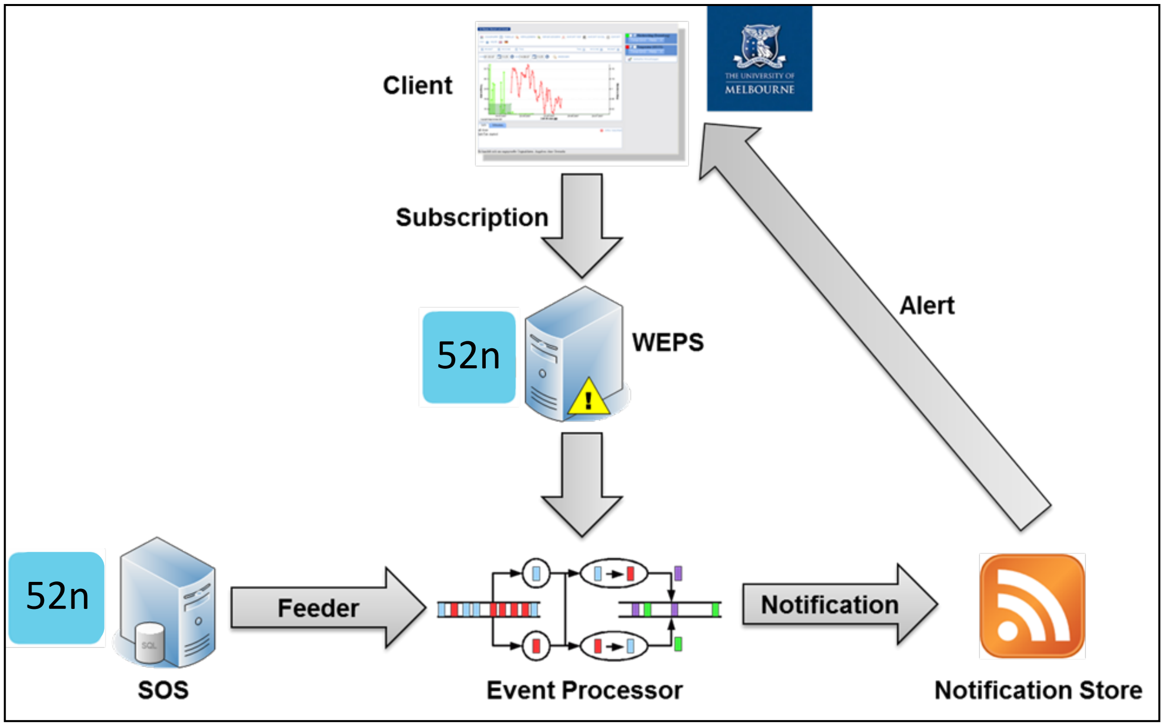


**Figure 2-7: Overview of the Event Processing Architecture**

# 3 Scenario Overview

## 3.1 Context

The incident occurs in the late summer, during the workweek at the beginning of rush hour. A cold front is approaching and expected to impact the vicinity within two hours. A tractor trailer truck carrying unknown cargo is traveling north on Memorial Parkway (the main north/south artery through town) near the Airport Road intersection that parallels a rail track (Figure 3-1). The truck collides with another vehicle as it approaches the Airport Road overpass, loses control and crashes through the barrier at the crest of the overpass. The truck tumbles some 15 feet onto the congested intersection below. The truck lands on the traffic in the intersection; the cargo it is carrying dislodges knocking down power lines and hits numerous cars, buildings and a transformer. The truck comes to rest near the railroad track and the cargo which appears to be numerous (15-20) very large cylinders (approximately 7-8 feet in length) is strewn about the crash scene. Some cylinders are dangerously close to a multi-story masonry constructed commercial building adjacent to an apartment complex. There are several vehicles trapped under and blocked by the truck and cargo, resulting in a complete traffic stoppage on both sides of the highway as well as local roadways. Several other cars and trucks are also involved in the accident, both on the northbound side of the Memorial Parkway overpass and on the Airport Road intersection below. There are numerous injuries at both crash sites. One or more vehicles are in danger of catching fire and several cylinders of the truck’s unknown but possibly hazardous (i.e., toxic, volatile, flammable) cargo begin to leak. The Department of Transportation placard on the truck is absent or not visible due to the wreckage and location of the other vehicles. North and southbound traffic on Memorial Parkway comes to a halt as does the west and eastbound traffic on Airport Road. People are beginning to emerge from their vehicles; some nearest the crash scene are gasping for breath.

Bystanders immediately call 911 and begin tweeting photos and descriptions. City law enforcement arrives a minute later and calls dispatch for fire, hazmat and emergency medical responders from the main city jurisdiction. The dispatcher notes the proximity of the rail line involved and alerts the proper authorities that the track is involved in the accident scene.

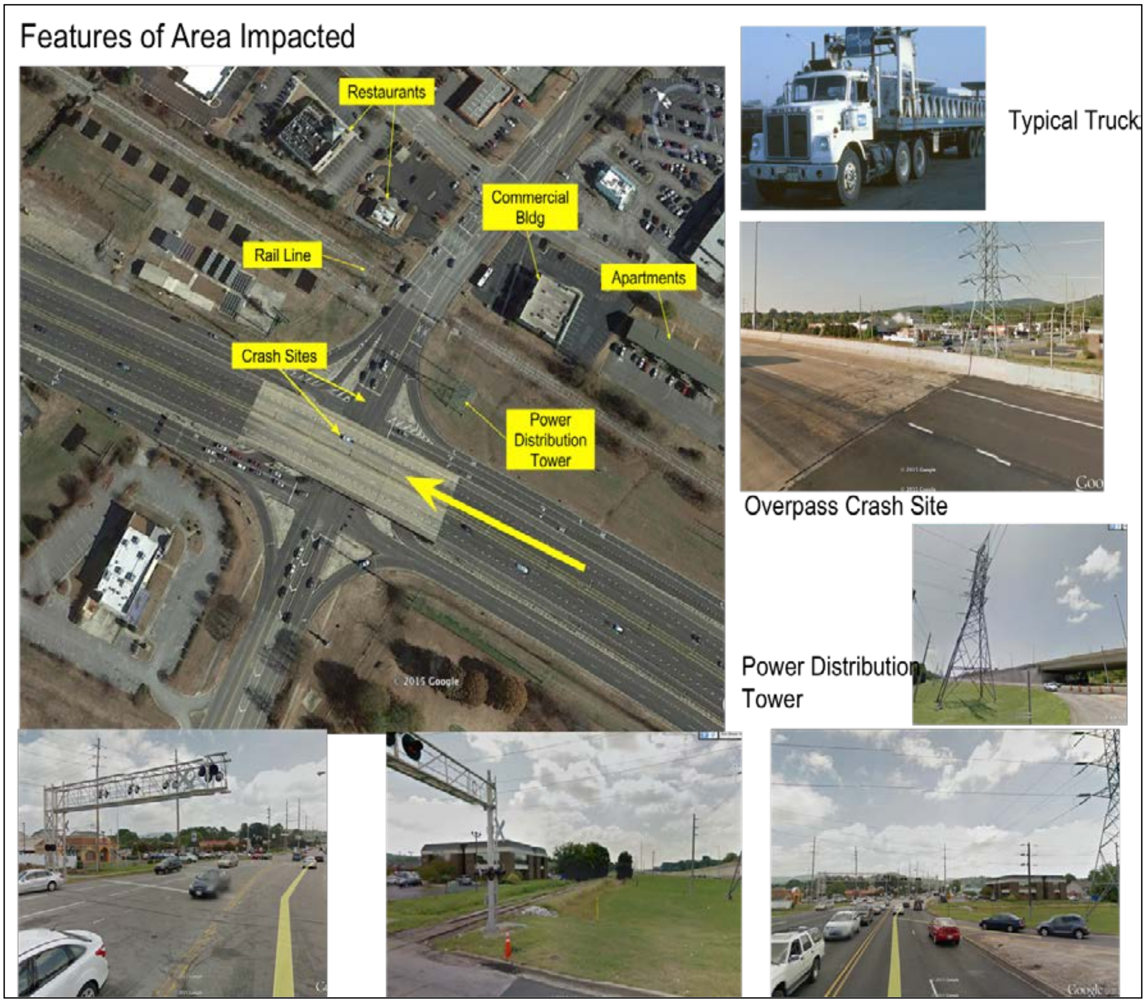


**Figure 3-1: Scenario Overview**

The initial priorities are to alert the rail line; contact the trucking company to determine the type and amount of cargo being transported by the truck; triage the scene and establish awareness of the situation; determine lifesaving requirements; and determine the extent of the incident and its magnitude as law enforcement begins to secure the scene and deal with the traffic. A site for the on-scene incident command is established by the city fire district chief at a safe setback distance from the scene (Joe Davis Stadium). The district chief makes an initial situation report to the city Public Safety Access Point (PSAP) describing the situation as a commercial truck accident with dislodged cargo that is leaking (yet unknown) contents, threat of fire, and multiple injuries at both crash sites. The district chief requests additional alarms, hazmat response, an emergency medical services (EMS) task force and for the police commander to report to the command post. The chief quickly develops the initial strategy for this incident which includes deactivating the downed power line, rescuing the walking wounded that can be safely reached, extinguishing fires, and a top priority of containing spilled/leaking cargo and the diesel fuel that is now leaking from the truck’s onboard fuel tanks. The command post, scene perimeter and hot zone are immediately established. Available staff quickly pull up data from multiple sources to characterize and visualize both incident scenes including resource staging, incident perimeter, access and egress points, as well as potential spill and flow patterns and/or smoke plume size and direction. The wind is currently blowing out of the west, across Memorial Parkway and onto nearby John Hunt Park; there are several soccer teams practicing in the park and hundreds of people in the area. The National Weather Service (NWS) forecast office has been monitoring an approaching cold front expected to hit the area within the next few hours and is now working with local hazmat teams and the emergency management agency (EMA) to generate continually updated plume models based on current and changing weather conditions. It is anticipated that the cold front could cause the winds to shift to a more westerly flow, further impacting populated areas around the incident scene.

Responding organizations contribute a variety of sensor resources to aid in awareness of the situation. City law enforcement and fire service task camera mounted vehicles and personnel, and request a video-equipped UAS as well as access to other existing cameras (both public and private) such as property surveillance/security, TV station weather cameras and traffic cameras to contribute periodic imagery of the incident area. Hazmat teams begin to arrive onsite, donning wearable biometric sensors for monitoring personnel entering and exiting the scene, placing environmental monitoring sensors around the scene, and deploying a UAS equipped with video and air quality sensors. All begin to transmit data about location and concentration of the hazardous materials. Initial investigation reveals the cargo is approximately 15-20 one-ton cylinders of chlorine gas. An unknown number of cylinders dislodged from the truck on impact and are strewn about the crash scene. A mutual aid request to the adjoining county for additional hazardous material resources is issued. Each of the respective jurisdictions (city and county) dispatch their first responder resources [via Computer Aided Dispatch and stand up their respective Emergency Operation Centers (EOCs)]. Further requests go out for any information regarding the truck's payload and number of cylinders onboard. It is confirmed from the UAS video that the payload is chlorine gas and 17 cylinders are quickly located. The hazmat teams have deployed portable hazmat sensors around the incident perimeter to track migration and concentration of the gas. The data are used to evaluate shelter-in-place strategies and safe evacuation routes of the immediate area. In the EOC, GIS analysts work closely with the hazmat teams and NWS local forecast office to monitor the changing weather conditions and develop situation products such as migration models to share with all stakeholders (and set triggers/alerts on the air sensor observations to guide evacuation planning). City emergency managers activate agreements with managers of a nearby building and access building sensor systems to monitor internal environmental conditions for a possible evacuation and/or determine suitability as shelter-in-place sites.

The multi-building apartment complex is evaluated for evacuation as there is concern about both hazardous material and fuel seepage into the underground infrastructure and explosion/fire impact to the buildings. Downed power lines interrupt electricity to the building as nightfall approaches.

Public observations on social media begin to report symptoms and locations of citizens impacted in the area. Social media serves to detect and map out a migration of leaked fluid into nearby underground infrastructures (mainly storm sewers) and a natural creek, triggering a revision of the incident perimeter, re-deployment of medical responders in the vicinity, and call up of an environmental management unit with hazmat cleanup capabilities. Some report a strong smell of diesel fuel emanating from storm drains west of the incident site approaching the nearby Spring Branch waterway. Social media is used by the authorities to alert citizens of the impacted areas, provide public evacuation routes and shelter-in-place safe zones. The media monitors the authoritative social media feeds (EMA, PD, FD and NWS) to stay informed of rapidly changing conditions.

As the threat of fires and hazardous materials are contained, accident victims are treated and evacuated, fire and rescue are required to canvas buildings in the immediate area to search for and treat victims that have been sheltering in place. Northbound traffic on Memorial Parkway is temporarily re-routed (through the Martin Road gate of Redstone Arsenal); the situation evolves from response to recovery. Deployed sensor units are recovered, maintained and stored for future use. Links to incident data are organized as a record of the incident response for use in retrospective training/learning activities and for use in tracking any subsequent effects on incident responders or victims.

## 3.2 Roles

- Citizens – Report the accident, provide situational awareness through social media updates and are evacuees. Citizens also represent accident victims.

- 911 operator (s) – Respond to 911 calls and gather additional situational awareness. Operators connect to the appropriate first responder dispatch to respond to the accident.

- First responders – Includes firefighters, police/law enforcement and EMS personnel. Hazmat crews are specially trained firefighters.

- Public utility crews – Neutralize risks from downed powerlines, and leaking gas and water lines.

- On-scene commander – Provides command and control of the incident response. The on-scene commander is a senior first responder, usually a fire chief.

- Emergency Operations Center (EOC) – Carries out the principles of emergency preparedness and emergency management, or disaster management functions at a strategic level during an emergency. This centralized command and control facility ensures continuity of operations. The EOC is responsible for the strategic overview, or big picture, of the disaster, and does not normally direct field assets, instead making operational decisions and leaving tactical decisions to lower commands. The common functions of all EOC's is to collect, gather and analyze data; make decisions that protect life and property; maintain continuity of the organization within the scope of applicable laws; and disseminate those decisions to all concerned agencies and individuals. In most EOC's the emergency manager is the individual in charge.

### 3.3 Narrative Cycle

The IMIS IoT Pilot demonstration was arranged in five stages, consistent with a standard disaster response.

1. **Notification –** 911 call, build incident, citizen observations, backup response, tracking, imagery;
2. **Build awareness –** establish scene, secure scene, utility operations, establish command post, hazmat on scene, EMS on scene, assess weather;
3. **Response –** UAS monitoring, traffic and crowd control, evacuation/shelter-in-place, building monitoring, threshold, hazmat planning, EOC activation;
4. **Mitigate event –** configure plume models, sweep area/buildings, health hazard detection, environmental hazard detection, traffic control update, EOC operational; and
5. **Recover –** coordinate EOC, conduct triage, mitigate cylinders, recalculate plume model, EMS/search and rescue (SAR) sweep, social media monitoring, transition to recovery, normalize traffic, spin down EOC, stand down sensors, close incident.

## 4 Technology Themes

Innovations that were advanced and deployed during the pilot activity can be organized into a set of technical capability themes. Although the technologies themselves are not necessarily of direct interest to first responders, the capabilities they represent are responsible for the user features that do have value to IMIS. Each of these themes relates in a significant way to the overall goal of getting the right information to the right person at the right time so they can be aware of the situation at hand and take the right action.

### 4.1 Shared Awareness – Incidents and Contexts

A number of technologies deployed in the pilot addressed the issue of finding sensor and information resources through dynamic catalog registration and search. Participants also recognized that traditional catalog search methods do not work for fast-paced responders. Virtual incident folders and context documents allow resources to be organized by an analyst or commander for each incident and responder role, then shared back and forth with multiple responders who receive the benefits of discovery without writing queries on their tablets or smartphones. Exchange of these link documents with connected responders literally represents the critical shared awareness of incident information that has mainly been communicated in patchwork fashion by voice communications up until now.

## 4.2 Driven by Events – Publish-Subscribe-Notify

Even when organized by event folders and context documents, ad hoc availability of incident sensor data quickly transforms not enough awareness into too much information to stay on top of. It then becomes necessary for IoT information systems to provide filtering, analysis and delivery of observations so that responders are notified of events when and if they are critical to their work without forcing them to wade through all the less significant data in between. This event-driven awareness has three important phases for the responder:

1. Publication of criteria and availability of observation events that someone needs to know about;
2. Subscription to those events by the users who have the interest or need to know; and
3. Notifications pushed to subscribers without delay when a published event occurs.

The appropriate scope of event recognition and degree of user involvement can vary tremendously, from an ad hoc user filter in a mobile app to configuration of a Message Queue Telemetry Transport (MQTT) service topic, to mandatory delivery of preconfigured health alerts from a complex event processing workflow. The common denominator is an informed awareness where users know what events will be generated, when events will be delivered to them and how they will be notified.

## 4.3 Tracking Resources – Resources and Responders

Given the present ubiquity of GPS receiver chips, it would seem trivial to count location as an observation. As with many other sensors, however, turning location observations into useful incident information for responders has its pitfalls, from the exact identity of the thing being located to the quality of GPS fixes to whether a location has gone “stale” from inadequate measurement frequency. The identity of the thing being located is critical, since this is the connection to other sensors, from videocams to health sensors, observing the same thing or from it. To provide open system federation of sensors, tracking information also may need to be “liberated” from proprietary Automated Vehicle Location (AVL) systems that hold tightly both the location data and the identity of the vehicle, person or resource being tracked.

## 4.4 Sightful Sensing – Imagery Sources and Targets

A wide variety of imaging sensors are practical (or soon will be) for incident response use. The pilot sought to georeference, time-index and otherwise transform imagery from vehicle, body, fixed and airborne cameras into searchable observation data that an incident manager or responder could effectively use as “eyes on the scene” to record a particular place at a particular time. This is a departure from the traditional linear paradigm of viewing and perhaps reviewing video footage one stream at a time. The opportunity to search for all available/captured views of a specific feature at a specific time from any imaging device is often imagined in TV procedurals, but rarely automated in public safety practice. Sensor Web and IoT capabilities advanced in the pilot have the potential to make this a reality.

## 4.5 Environmental Sensing – Air Quality

An essential component of assuring public health and safety is an awareness of environmental conditions and impacts around an incident scene, both for responders and for the public. The pilot scenario focused on air quality and aerosol hazards from a chlorine cylinder spill. Adequate sampling, resolution, reproducibility and reliability are all technical challenges even for fixed air quality sensors. An IoT approach to sensor integration can provide a weight of evidence of air quality conditions that helps responders have confidence in the decisions they make to protect the public and themselves. The pilot also integrated the results of air dispersion modeling into the incident management awareness as “future” observation results able to be tasked, recorded, queried and sent in notifications in the same way as current or historical sensor observations.

## 4.6 Health Sensing – Wearables and Physiology

Sensing human physiological parameters took on two challenges in the pilot. The first was to provide objective assessments of responder health and safety, alerting fire chiefs, for example, if a firefighter’s heart rate or body temperature became dangerously high even when their radio communications insisted, “I’m just fine, don’t pull me out.” Essential to this benefit is an easy-to-use, low-maintenance system for capturing such measurements for each responder at risk and generating reliable alerts of critical measurements.

The other challenge the pilot addressed was triage of incident victims (possibly including responders themselves). Health sensors have the potential to monitor automatically for vital signs and condition changes in a large number of patients while triage examinations and dispositions are carried out, usually, by inadequate numbers of EMS personnel. Again, ease of use, non-intrusiveness and reliability where lives may be at imminent risk are all success factors for this capability. Quickly alerting EMS which patient’s pulse is weakening and skin temperature is decreasing, as well as where they are situated at the scene, is exactly the type of challenge that the IMIS IoT Pilot was in a position to address.

# 5 Technical Implementation

## 5.1 Station 1

Demonstration Station 1 covers the time period from zero to five minutes and is focused on preparation and notification. Preparation takes place long before an event occurs and is often referred to as “left of boom.” Notification takes place when information is first received that an event has occurred; this information can come from a variety of sources including 911 calls and social media observations. This is often referred to as “right of boom.”

### 5.1.1 Preparation of Sensor Capability

To provide IoT support to an incident, several components need to be prepared in advance. These components can be broken into three categories.

S-Hubs

S-Hubs need to be available and ready to deploy. The job of the S-Hub is to speak to the individual sensors using whatever proprietary language it supports and make the data available using open standards such as SOS and STA. When an S-Hub is activated it will register itself with the catalog, making it discoverable.

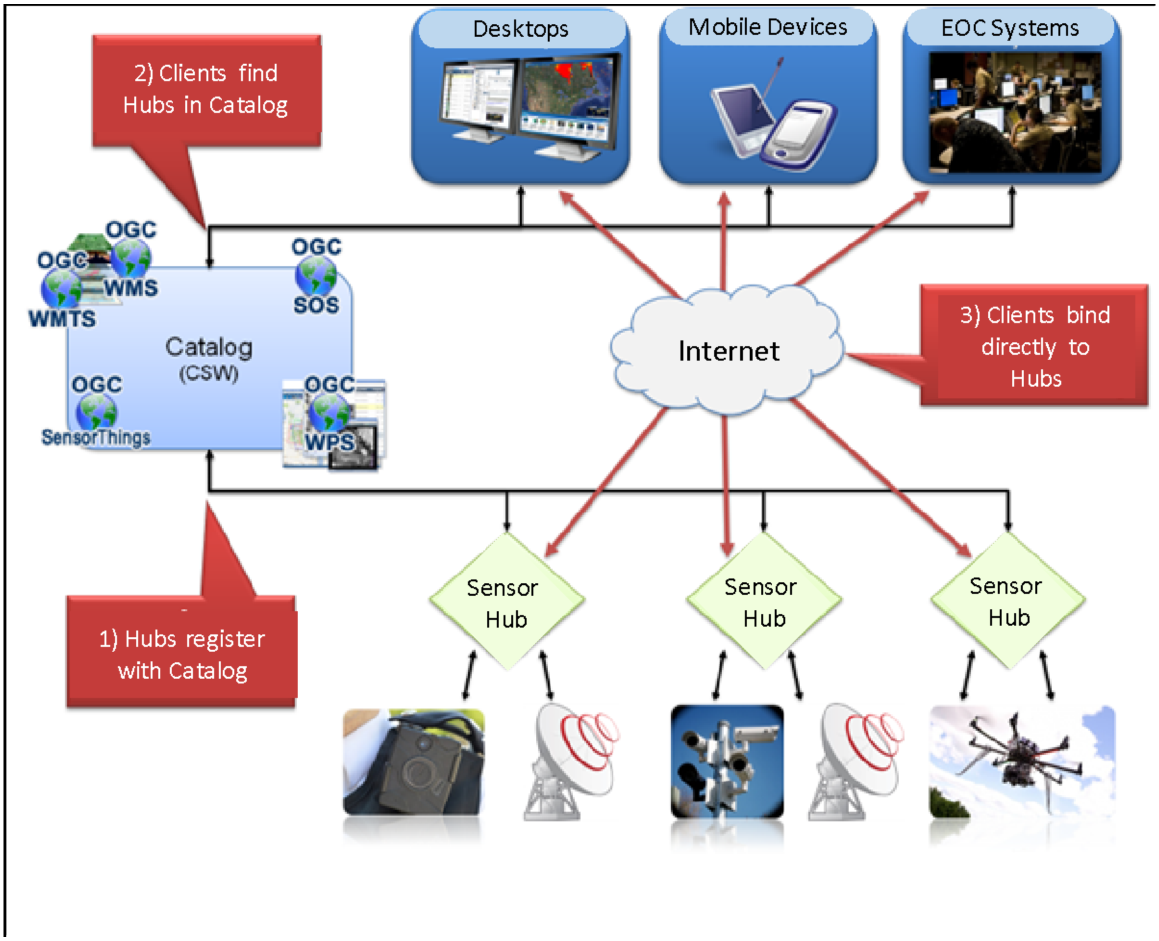
Catalog

The catalog (HubCat), which in this case was implemented using the CSW 2.0.2 specification, needs to be active and its address URL known by all of the organizations who may want to use it. The job of the catalog is to allow for the registration of S-Hubs as they become active, as well as the discovery of framework data services. Framework data is discussed in Section 5.1.2.

Clients

Clients can come in many different forms, whether it is desktops, mobile apps or EOC systems. These clients need to support discovery from the catalog as well as maintain support for SOS or SensorThings. The clients need to be preconfigured with the location of the catalog and be ready for use.

Figure 5-1 illustrates how these three components come together:



- When S-Hubs are activated, they automatically register with the catalog.
- Clients can be used to discover sensors of interest from the catalog and obtain the corresponding service URL.
- Clients communicate directly with the S-Hubs to retrieve data from the sensors.

### 5.1.2 Framework Data Services and Catalogs

Framework data refers to any data that can be made available in advance, which can help provide context to an event. This framework data is published to the catalog in preparation for an event and can come in many different forms including WMS, Web Map Tile Services (WMTS), SOS, etc.

In preparation for the event simulated during this pilot, the framework data included, but was not limited to:

- WMS layers
  - Fire hydrant locations
  - Fire station locations
  - Building footprints
- WMTS layers
  - Open Street Map (OSM) derived basemap
  - Satellite imagery
- SOS
  - Real-time weather

### 5.1.3 Social Media Observations

Social media platforms have the potential to provide another source of situational awareness in emergency response activities. For example, this idea was investigated in the OGC Testbed-11 and documented in a corresponding ER (OGC Testbed-11 Incorporating Social Media in Emergency Response Engineering Report, OGC 15-057).

One approach to handle social media observations is to consider humans as sensors. A corresponding O&M-based data model is described in more detail in OGC 15-057. In summary, the main concepts of O&M can be mapped to social media observations as follows:

- Feature of interest – The location/object/place for which the observer has published an observation;
- Observed property – The property which is described by the observation (e.g., in the case of a photo, this would be the visual perception);
- Procedure: The observer who made the observation (e.g., the photographer or a Twitter user who posted a Tweet); and
- Result: A reference to the social media content (e.g., a link to the photo on Flickr).

Based on this mapping it is possible to use an OGC SOS for handling human observations provided through social media platforms.

Figure 5-2 gives an overview of an architectural approach implemented in OGC Testbed-11. To avoid redundancy, the SOS for social media does not store any social media content. Instead, the SOS stores information about available metadata content. For example, it manages references to Flickr images, Tweets and photos on Instagram. For building this index of social media information, the SOS is connected to dedicated harvesters for the different social media platforms. These harvesters can be tasked with specific parameters (e.g., spatial or temporal extent, keywords). Subsequently, the harvesters query the social media platforms to find all content that matches with this query. Finally, references to the discovered content are inserted into the SOS database using the transactional SOS operations (InsertSensor and InsertObservation).

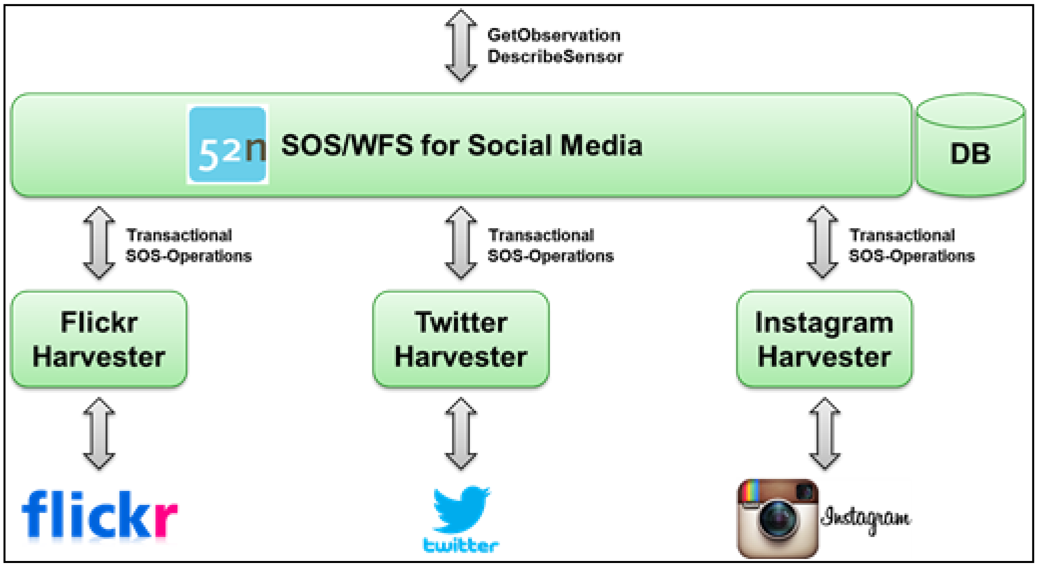


**Figure 5-2: SOS for Social Media Architecture**

Although this approach was not implemented within the IMIS IoT Pilot, it is worth mentioning as it might be a future extension to integrate additional sources of information.

### 5.1.4 Tracking and AVL

The incident commander and first responders can benefit greatly from real-time knowledge of the whereabouts of any personnel and emergency vehicles. In addition, the real-time geospatial awareness provided by streamed georeferenced camera video provides “eyes-on-scene” that can be vital to appropriate and timely decision making.

AVL data for Huntsville Fire Department, Huntsville Police Department and Huntsville Emergency Medical Services vehicles were served from the SOS interface of an OSH S-Hub providing location, ID and status. For this demonstration, these data were simulated using the actual AVL data structure used in Huntsville, Alabama, but with the routes to the scene calculated on-demand using Google routing services, as seen in Figure 5-3.

It is important to note that geographic locations *are* observation data just as much as temperature measurements and should not be treated simply as metadata in a catalog. Locations are measured by sensors with various degrees of accuracy and response time. Sometimes it is the location of the Thing itself that is important (e.g., person or vehicle), but other times the location of a sensor identifies geospatial context for measurements. Either way, it is important to treat the measurement of both location and time as sensor measurements that can be accessed from a S-Hub by themselves or as part of a more complete collection of information (e.g., a measurement tuple of time, location, wind speed, wind direction). The same could be said for other properties often treated solely as metadata, such as uncertainty, sensor calibration, etc.

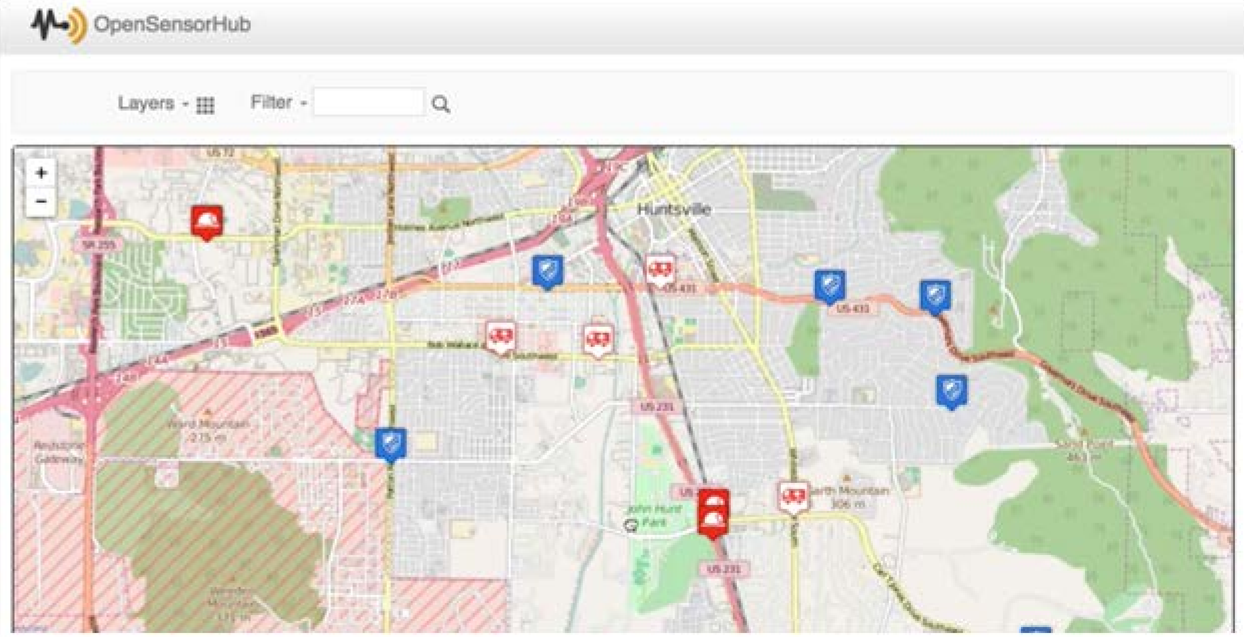


Figure 5-3: AVL of First Responder Vehicles

The demonstration also included remote access to real-time vehicle-mounted, high-definition Pan-Tilt-Zoom (PTZ) video cameras and personnel-mounted “body-cams.” In addition to live streaming video, the real-time locations and orientations of these cameras were provided by an SOS powered by an OSH node on the cloud.

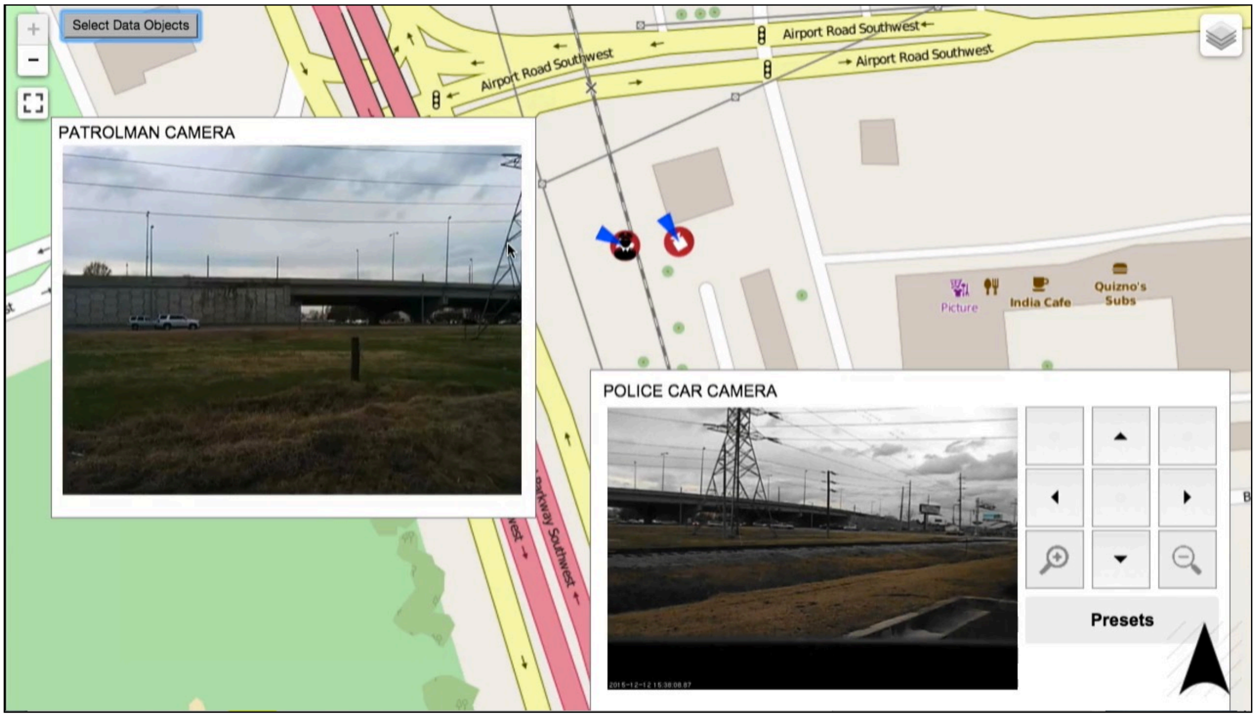


**Figure 5-4: PTZ Image Overlay**

For the vehicle-mounted video, an Axis PTZ video camera was mounted on a truck, providing video and PTZ settings to an OSH node running on a RaspberryPi2 board. The observations were streamed in real time to another OSH node on the cloud to make it easier to access through clients from anywhere in the world. Communication between OSH nodes was via a Long-Term Evolution (LTE) data plan on a T-Mobile hot spot. Likewise, the truck location and orientation were provided in real time by an OSH node running on an Android phone on the dash of the truck; again these data are streamed through LTE to the OSH node on the cloud.

A SensorML-encoded processing chain was running on demand in real time on the OSH node on the cloud providing geospatial awareness to the Axis camera. In addition to providing look angles for display on the map, it can also task the camera to look at a particular latitude, longitude and altitude (Figure 5-4).

For the “body-cam” video, an Android phone is worn as a prototype with real-time streaming of video, location and orientation data. An OSH node deployed on the Android phone gathered the data and streamed it to the OSH node running on the cloud which, again, allowed immediate web-based access to geospatially-aware personnel video. In addition to the video and location from the Android phone, all sensors on the Android are supported, as are various other sensors communicating to the phone through Bluetooth.

## 5.2 Station 2

Demonstration Station 2 covers the time period from five- to 20-minutes and is focused on building shared situational awareness among all those who are reacting to the emergency. The approach exploits the sensors and sensor catalog in which they register during the initial phase, and allows these to be brought together with other information to provide correct and consistent information to all who need it. This focusses on two of the key goals of the Next Generation First Responder (NGFR) Program, ‘Connected’ and ‘Fully Aware.’ It also optimizes collaboration between responders as is needed to achieve the overall goal of improved response. Figure 5-5 shows the overall context of the station.

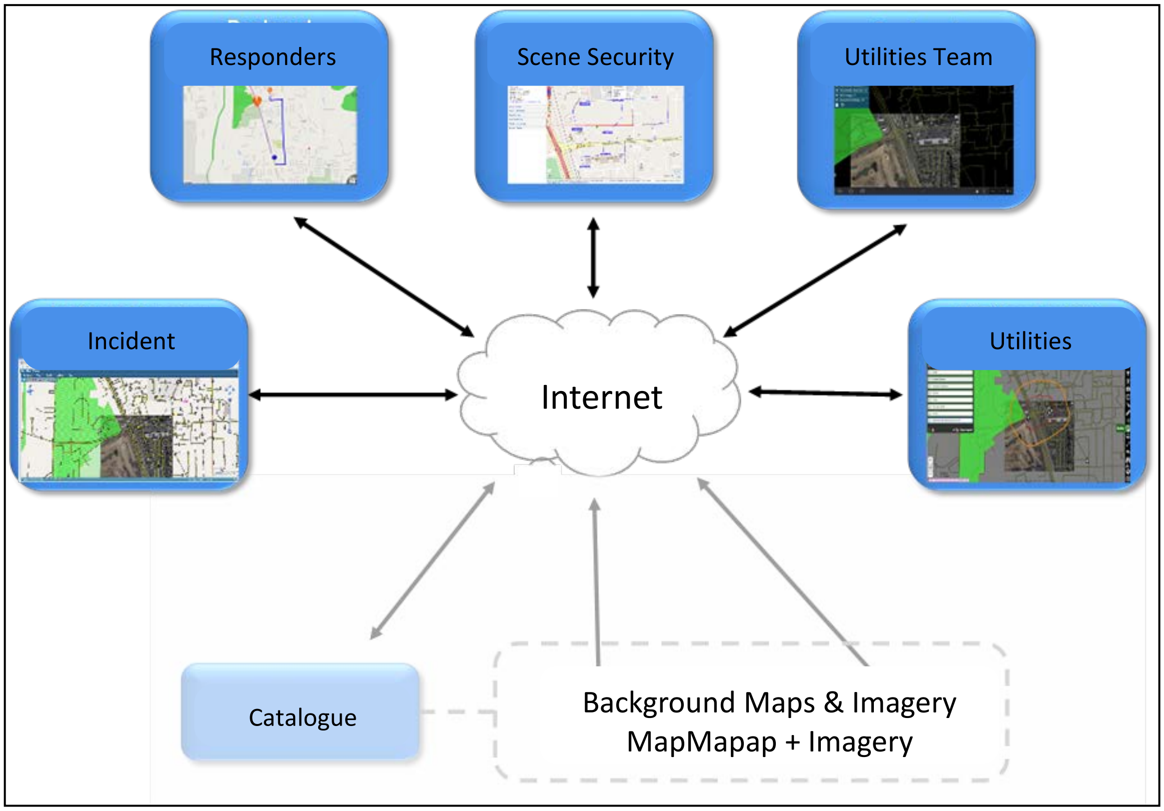


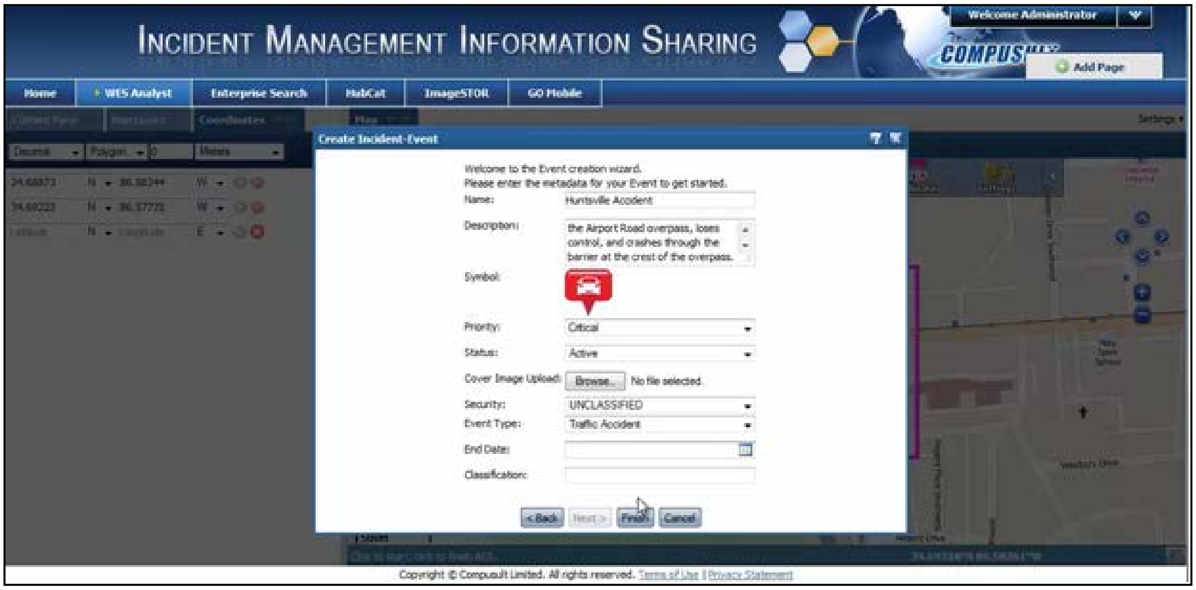
**Figure 5-5: Overall Context of Station 2**

Within the scenario, a range of responders address the situation. They have a range of resources including background information (maps, imagery, building plans), as well as a range of information from deployed sensors.

Using incident scene reports, the command center-based incident manager assembles a basic set of information in relation to the incident, in particular the background imagery, key geographic features such as road detail and names, and the positions of the response vehicles in the area. This is the basic view which complements the incident report.

### 5.2.1 Incidents Folder Creation and Management

As the event unfolds, notifications and information start to flow into the command center, requiring information collection and dissemination to those who can use it. The incident manager creates an incident-event record to capture the information. This record is created using Compusult's Incident-Event Manager (IM), which allows the incident manager to capture content from disparate data sources to provide a Common Operating Picture (Figure 5-6). The content captured can include map layers, imagery, reports, services and other files (Figure 5-7).



Welcome to the Event creation wizard.
Please enter the metadata for your Event to get started.

Name: Huntsville Accident

Description: the Airport Road overpass, loses control, and crashes through the barrier at the crest of the overpass.

Symbol:

Priority: Critical

Status: Active

Cover Image Upload: Browse... No file selected.

Security: UNCLASSIFIED

Event Type: Traffic Accident

End Date:

Classification:

**Figure 5-6: The IM Permits the Manager to Capture Content from Disparate Data Sources**

Now that the incident manager has created the incident-event record, it can be saved as an OGC context document and published to the catalog. Other clients can now discover this context and view/modify the Common Operational Picture it provides. Essentially, this allows multiple users/clients to share and maintain a situational view of the event as it unfolds.

### 5.2.2 Shared Tasking, Availability and Status

<u>Browser-Based Client</u>

The incident manager in this scenario uses a browser based interface (for this phase the Compusult WES Client is used) to assemble the above information. This queries the HubCat for sensor information and background. The benefit of a browser-light client is that as long as the incident manager is using a system connected via reliable communications, he/she can log in and access the information from any terminal and any location.

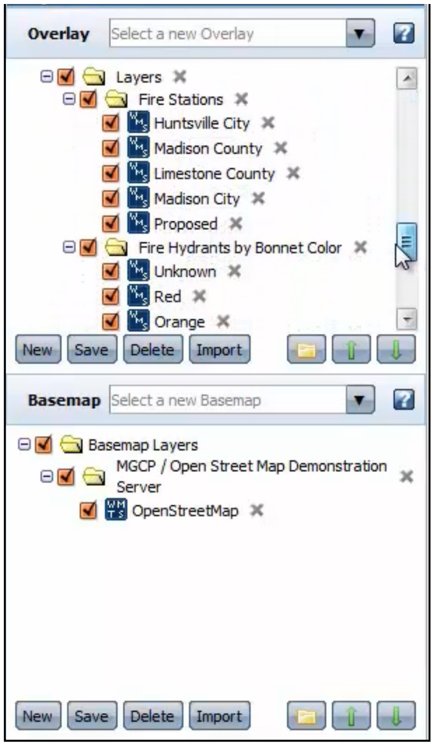


**Figure 5-7: Captured Content Includes Map Layers, Imagery, Reports, Services, etc.**

<u>Cataloging</u>

For the purposes of this experiment, some background information was also registered in the HubCat. In a real deployment it is likely that there would be multiple catalogs, for example one or more foundation catalogs containing base mapping, imagery etc., and one or more deployed HubCats (potentially one per emergency service). The goal should be that these catalogs are registered centrally and can all be searched via a single catalog, however. Alternatively, an operational HubCat could come online to support an incident (triggered by the incident manager) and be configured to connect and harvest from other catalogs so that a single catalog is presented. The implementation is a reasonable simplification of the real situation.

Base Mapping/Imagery

The incident manager's client is, via the catalog as a broker, reading base layers from multiple servers; in the case of the demo, the client is base mapping from both general Internet servers and from custom servers.

Base Department of Interior imagery from United States Geological Survey (USGS) was available via online WMS Services. During the pilot, however, we found that this service would potentially show invalid data tiles (perhaps showing daily collection). Envitia's Discovery Product, which takes base imagery and automates the generation of WMSs and catalogues them, was used to deploy a cache of USGS data in the area of interest in addition to shapefile data providing topographic data (e.g., road center lines, park areas). Geo-Huntsville WMS Servers provided additional layers, such as the locations of emergency services stations and building outlines and purposes. The Compusult client also exploited publically available OpenStreetMap Data to provide a detailed map backdrop.

Overall, the incident manager has access to a wide range of background information, both open source and authoritative, and is not limited to what is available in the system.

Sensors

Using the Compusult Browser Client, the incident manager can discover and add information on the emergency vehicles responding to the incident as they provide geo-locations via a sensor observation service. These are then added to the display. In the pilot scenario, the vehicle sensors were a mixture of tracks simulated from the Compusult SOS services and real sensors connected to the Botts Innovative SOS services which were in vehicles in the vicinity.

A further capability developed by Compusult, which visualizes SOS Services registered in the catalog as WMSs, is used to provide the 'Emergency Vehicle' layer actually used in the display. This aggregates multiple SOS contents which deliver similar information (e.g., emergency vehicle locations). The WMS services are also registered in the catalog.

### 5.2.3 Context Exchange and Collaboration

Context View

The incident manager constructs a geospatial view of the situation using the browser ‘Situational Awareness’ Client. To be fully useful though, it needs to be distributed to interested responders (i.e., to achieve ‘Shared Situational Awareness’).

To do this, the command center user can create an OGC Compliant OWS context document (Figure 5-8). This describes the area of interest (as a GML envelope) and the set of layers accessed by the client. These layers are, in this case, all WMS calls. The OWS Context document is created from the Incident Manager Client and is a simple XML document – small, so it can be easily emailed or stored in the catalog and read by a range of clients (including other browser-based clients and Android clients).

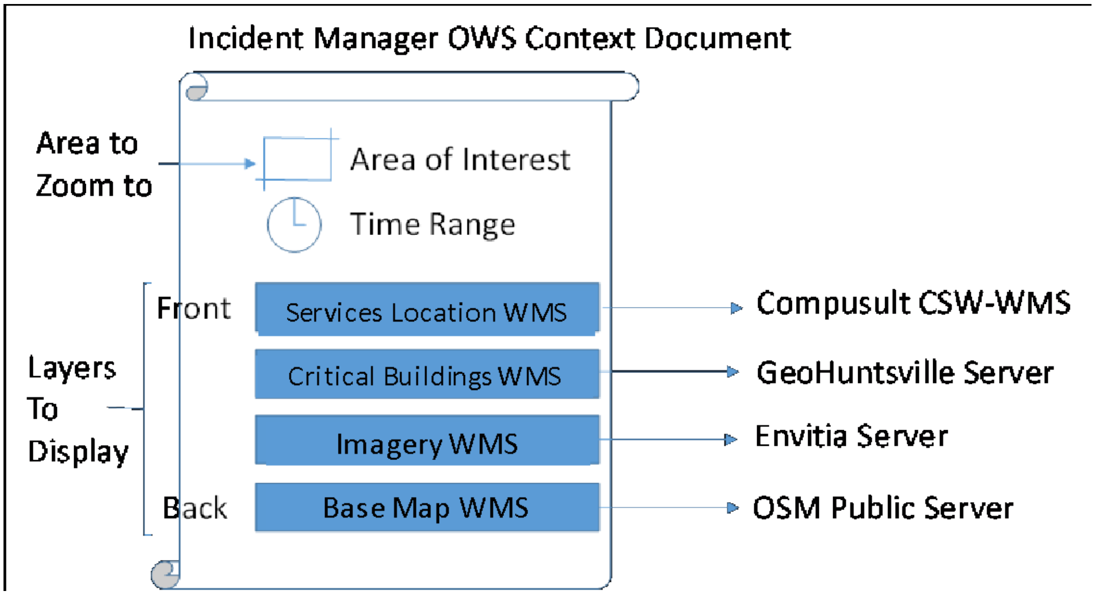


**Figure 5-8: OWS Context Document Structure**

The OWS Context document is transmitted to the mobile user on the scene using Compusult GoMobile. The user defines bounding extents of the incident area and other key critical areas using drawing tools on the mobile device (Figure 5-9). These geometries are uploaded to the server and available via WMS and WFS services, which can also be included in the OWS Context document but was not used during the pilot. The same process is used by law enforcement on a tablet to define road closures.

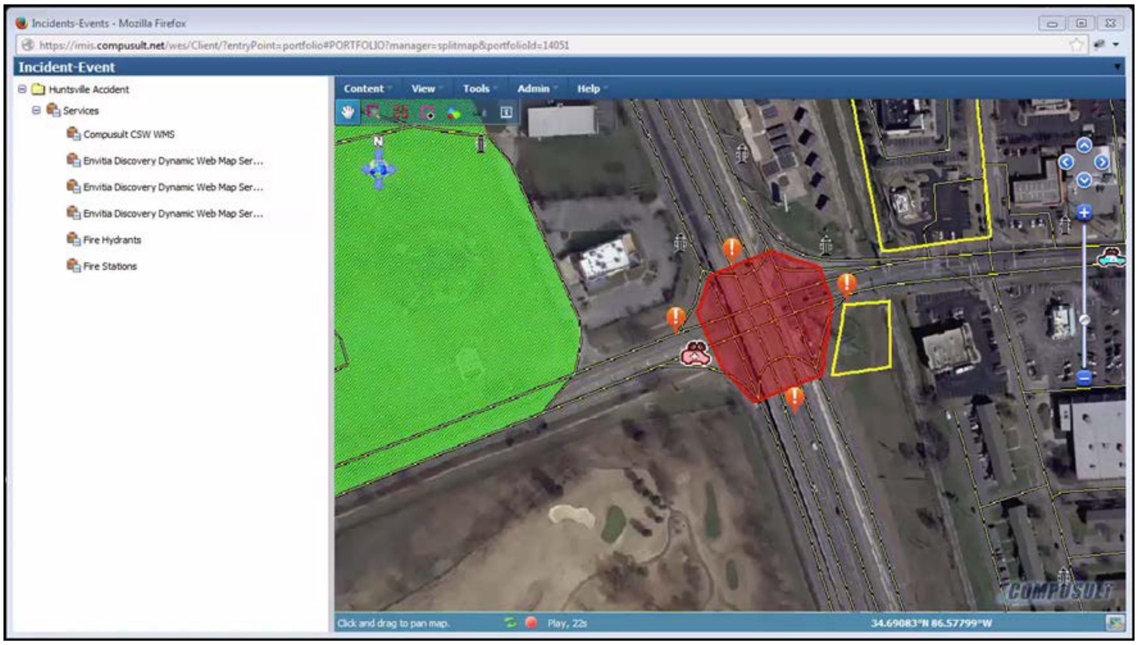
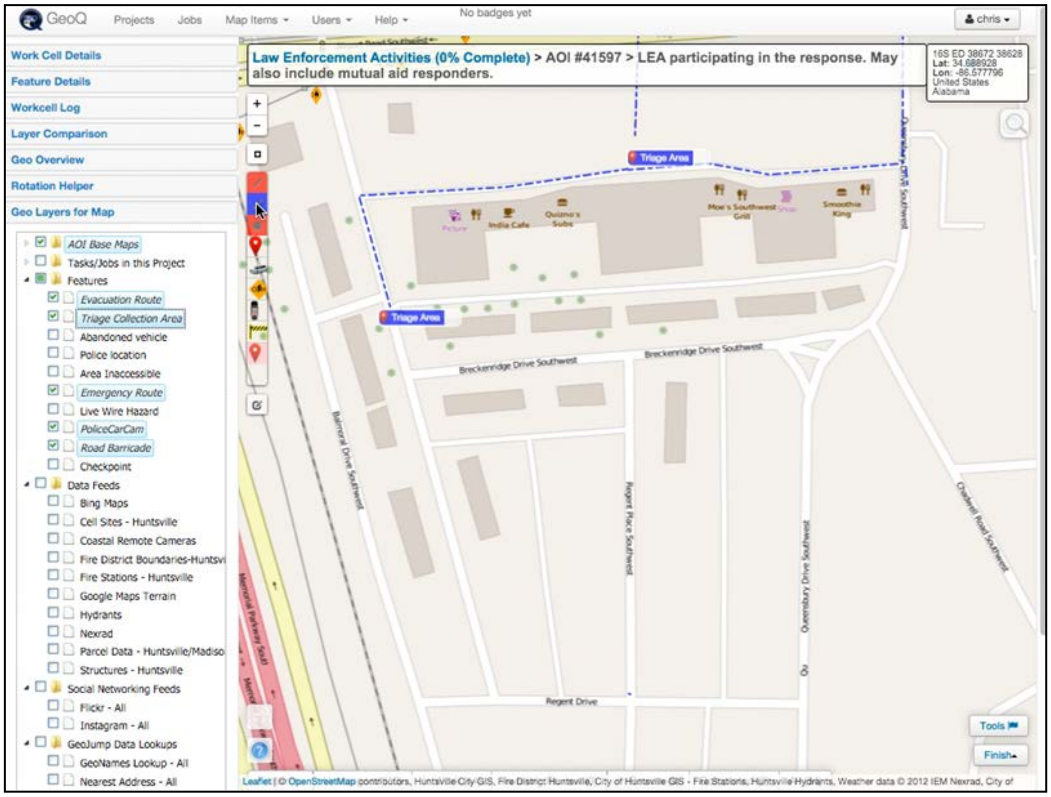

Incidents-Events - Mozilla Firefox
https://imis.compusult.net/wes/Client/?entryPoint=portfolio#PORTFOLIO?manager=splitmap&portfolioId=14051
Incident-Event
Huntsville Accident
Services
Compusult CSW WMS
Envitia Discovery Dynamic Web Map Ser...
Envitia Discovery Dynamic Web Map Ser...
Envitia Discovery Dynamic Web Map Ser...
Fire Hydrants
Fire Stations
Content
View
Tools
Admin
Help
N
COMPUSULT
Click and drag to pan map.
Play, 22s
34.69083°N 86.57799°W

**Figure 5-9: Annotation of the Scene in Incident Manager Client View after Update from the Field**

Another client, GeoQ from Exemplar City, Inc., is then used to pull in the same information and define the Law Enforcement Traffic Pattern (Figure 5-10) ensuring the effect of the incident is controlled. This client can draw on the same layers as the Incident Manager Client, providing consistent information.



Law Enforcement Activities (0% Complete) > AOI #41597 > LEA participating in the response. May also include mutual aid responders.

- ☑ AOI Base Maps
- ☐ Tasks/Jobs in this Project
- ▪ Features
  - ☑ Evacuation Route
  - ☑ Triage Collection Area
  - ☐ Abandoned vehicle
  - ☐ Police location
  - ☐ Area Inaccessible
  - ☑ Emergency Route
  - ☐ Live Wire Hazard
  - ☑ PoliceCarCam
  - ☑ Road Barricade
  - ☐ Checkpoint
- ☐ Data Feeds
  - ☐ Bing Maps
  - ☐ Cell Sites - Huntsville
  - ☐ Coastal Remote Cameras
  - ☐ Fire District Boundaries-Huntsvi
  - ☐ Fire Stations - Huntsville
  - ☐ Google Maps Terrain
  - ☐ Hydrants
  - ☐ Nexrad
  - ☐ Parcel Data - Huntsville/Madiso
  - ☐ Structures - Huntsville
- ☐ Social Networking Feeds
  - ☐ Flickr - All
  - ☐ Instagram - All
- ☐ GeoJump Data Lookups
  - ☐ GeoNames Lookup - All
  - ☐ Nearest Address - All

**Figure 5-10: Law Enforcement Pattern Established**

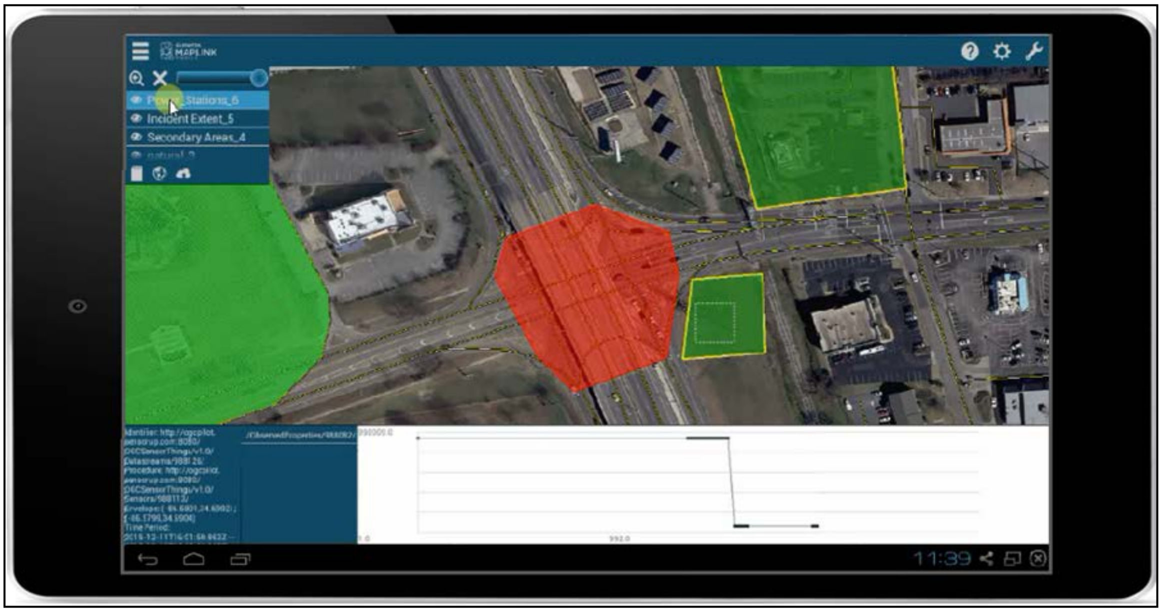
It should be noted at this point, that the OWS Context document exchanged between clients is not a static picture, but instead references to information; whenever it is loaded, it allows the client to show and keep up to date with the latest information relating to each layer. Thus, any vehicle positions will update as the document is passed around.

<u>Utilities Use of OWS Context</u>

Another use of the OWS context document is in the exchange of the view to the utilities crews who in the scenario were required to assess any electrical hazard and interruption. The base OWS Context document is distributed to the mobile crew, who are using a different application, the Envitia InSight Android App on a tablet. The app reads the OWS Context document created by the Incident Manager Client. It can receive the OWS Context document in several ways. In this pilot, it searches the HubCat for OWS Context documents.

Upon opening, the app displays the background information WMSs, the incident overlay information (also WMS) and lastly the CSW-WMS, which provides sensors. The app needs to interact with the original SOS service so it can query more details of the observation, however. The mobile client user can either search the catalog for an SOS service or query the SOS service from the CSW-WMS. The latter is possible because the link to the source SOS is available in the WMS via the GetFeatureInfo service.

Once the SOS has been loaded (Figure 5-11), the client allows the user to display the electrical power level observation over a time period. It is clear that the power level has dropped to zero in the recent history, due to the destruction of the tower carrying the cables. The line along which the powerline has been destroyed.



Identifier: http://ogcpilot.sensorup.com:8080/OGCSensorThings/v1.0/Datastreams/988126/
Procedure: http://ogcpilot.sensorup.com:8080/OGCSensorThings/v1.0/Sensors/988113/
Envelope: [-86.6801,34.6002]; [-86.1795,34.9904]
Time Period:
2016-12-11T16:01:56.8632 --

**Figure 5-11: Envitia InSight Showing SOS Service**

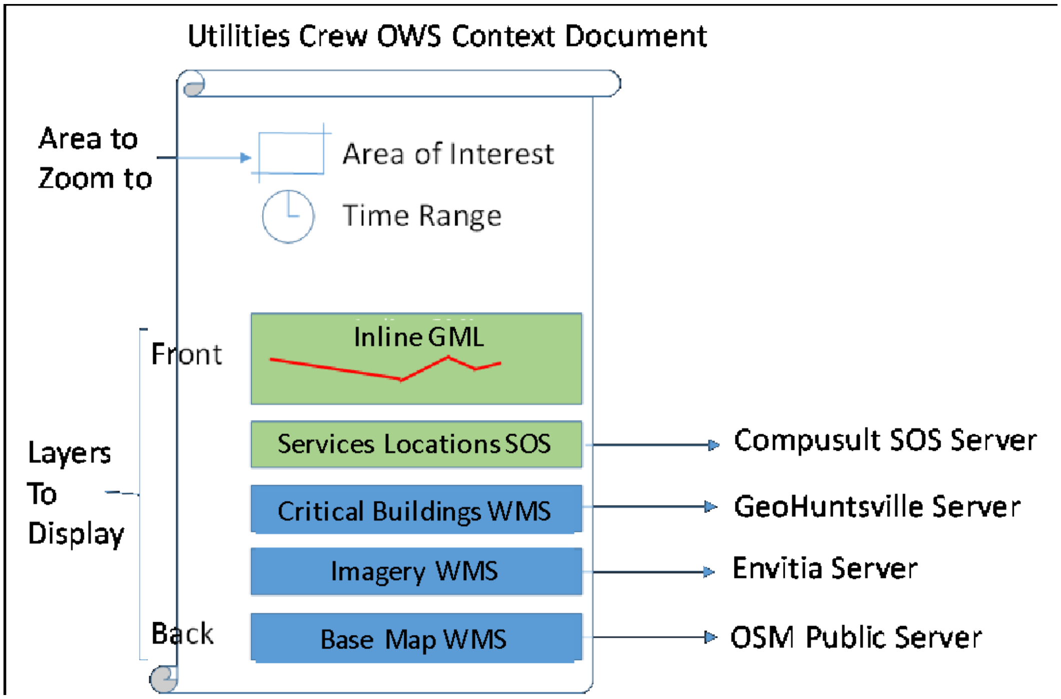


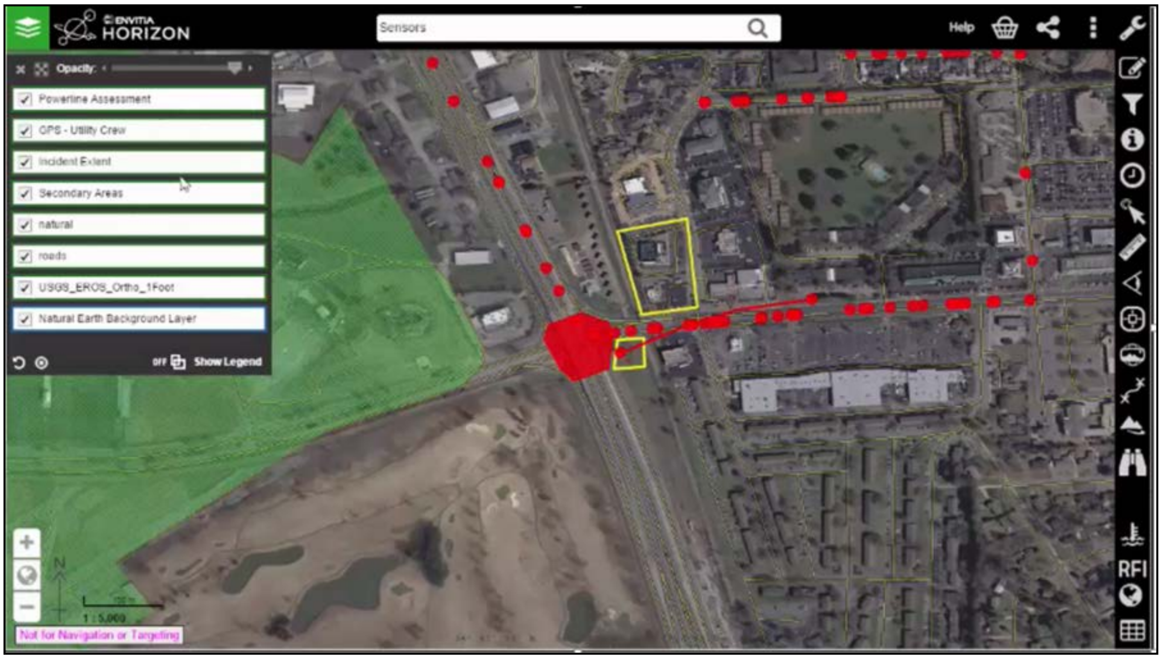
**Figure 5-12: Utilities OWS Context Document Returned**

The mobile user of the Envitia Insight App can re-publish the OWS Context document with the embedded SOS service, and with the annotation showing the powerline problem back to the server. This can be done by publishing a new version to the catalog or emailing a copy to the relevant user. The latter is done in this case (Figure 5-12).

The OWS context document contains the additional annotation rather than publishing it to the server. This is an alternative approach to the example described earlier.

Command Center Utilities Analysis

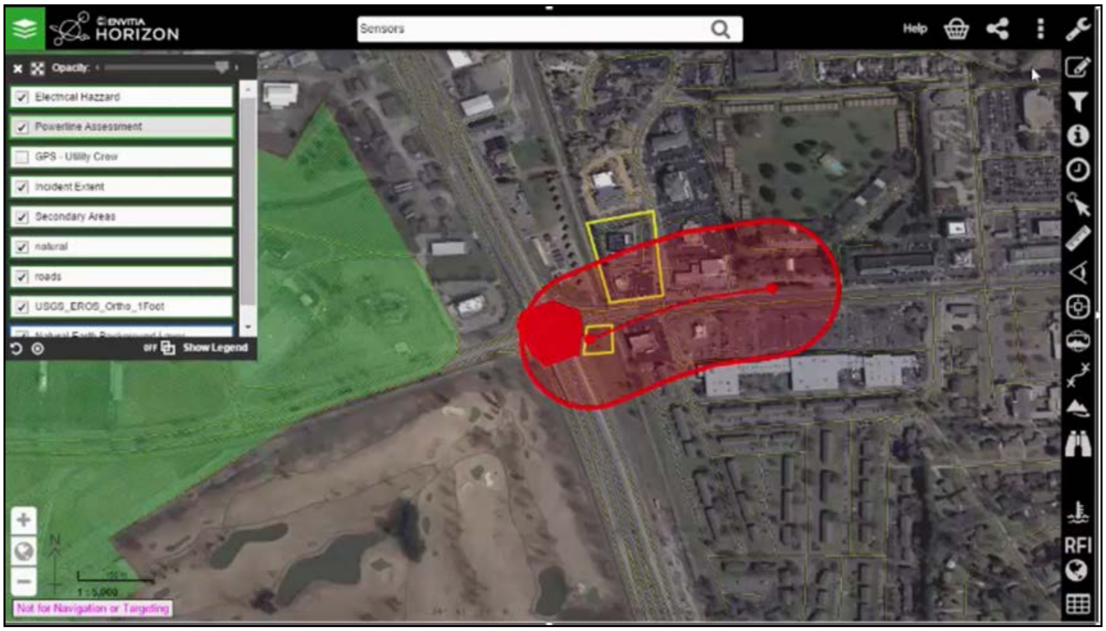
At the command center, the utilities analyst using the Envitia Horizon Browser Client ingests the utilities crew OWS context document. The utilities analyst can also use the HubCat to discover additional sensors. In this case, the areas surveyed by the crews are of particular interest to the utilities analyst. The utilities analyst loads the SOS and visualizes this (Figure 5-13).



- ☑ Powerline Assessment
- ☑ GPS - Utility Crew
- ☑ Incident Extent
- ☑ Secondary Areas
- ☑ natural
- ☑ roads
- ☑ USGS_EROS_Ortho_1Foot
- ☑ Natural Earth Background Layer

**Figure 5-13: SOS Visualization in Envitia Horizon**

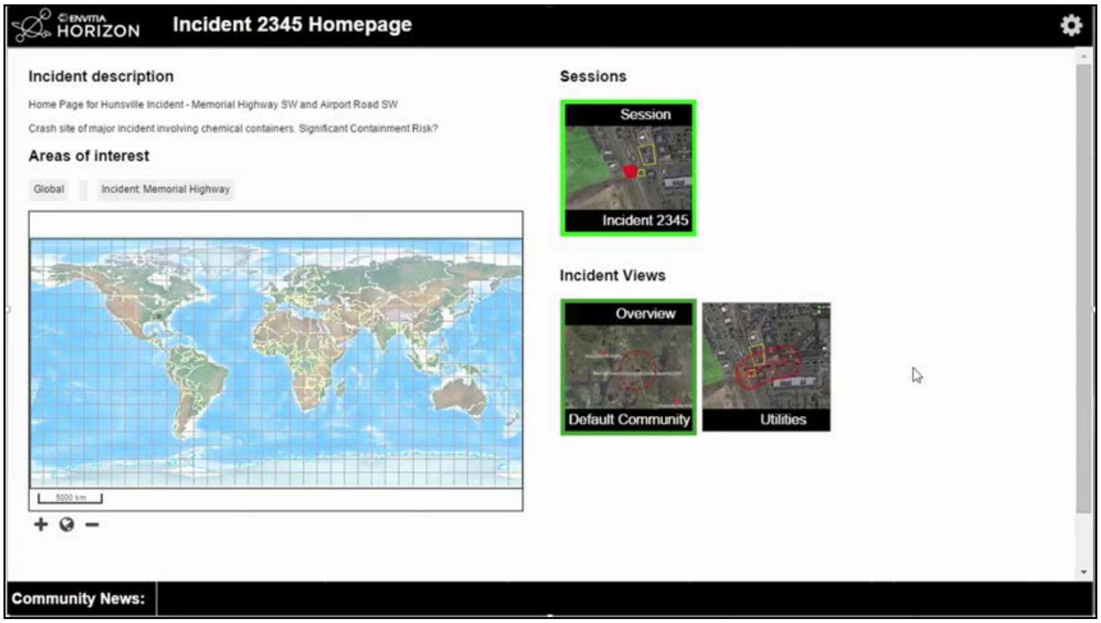
After loading the OWS Context document, the analyst also has a view containing the background information and the line along which the crews have identified a downed cable. The analyst decides to define a safety zone, a 100 meter buffer around the downed power line (Figure 5-14). This can be done using the buffer facility in the Envitia Horizon Portal. This uses the Envitia WPS-compliant Buffer service. The result is another view which can be once again re-published as an OWS Context document.



- ☑ Electrical Hazzard
- ☑ Powerline Assessment
- ☐ GPS - Utility Crew
- ☑ Incident Extent
- ☑ Secondary Areas
- ☑ natural
- ☑ roads
- ☑ USGS_EROS_Ortho_1Foot

**Figure 5-14: Buffer Created from Utility Crew Analysis**

The various views created in the client can be registered within the catalog to create views relevant to a specific community. The Envitia Horizon Portal is capable of storing the OWS context documents within a CSW-ebRIM compliant HubCat (Figure 5-15).


Incident 2345 Homepage

Incident description

Home Page for Hunsville Incident - Memorial Highway SW and Airport Road SW

Crash site of major incident involving chemical containers. Significant Containment Risk?

**Figure 5-15: Community Page Created by Querying the Catalog**

### 5.3.1 Environmental Sensor Deployment, Discover, Access

1. Sensors were deployed within the incident area, particularly around buildings, to monitor a number of environmental parameters relevant to public health and safety, specifically nitrogen dioxide, carbon monoxide, temperature, and humidity.

2. Sensors registered themselves to an S-Hub (with STA)

3. The S-Hub then published the registered sensors to HubCat, the sensors then became discoverable by web clients and accessible through the SensorHub's SensorThings API interface

4. Web clients subscribed to real-time STA datastreams topics and real-time readings were then pushed to the clients via MQTT as they were generated.

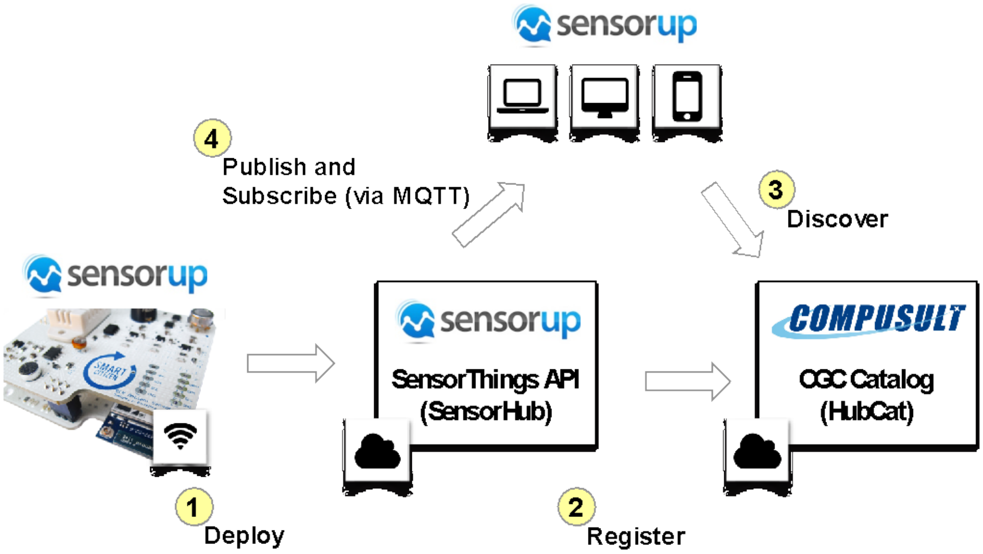


**Figure 5-16: Deployment of Environmental Sensors**

### 5.3.2 Health Sensor Deployment, Discovery, Access

Health sensors were deployed in two roles. Firefighters wore health sensors along with their trackers and body cams. Health sensors were also fitted to trauma victims to monitor their vitals and alert busy EMT's of any changes in condition. In both cases, as soon as health sensors were donned and switched on, they immediately connected to a personal STA-enabled S-Hub that in turn registered with the HubCat. Dashboard Web widgets connected to the S-Hub to provide continuous monitoring of many sensor systems at once. The Internet of Things organizing principle recognizes each sensor wearer as a real world feature (Thing!) that is able to tie together diverse sensor measurements into a coherent picture of what is occurring, where and when.

EMS responders were then able to select health sensors from a HubCat sensor map view and configure an event condition in their application for heart rate measurements exceeding 150 bpm. When the event condition occurred, the responder received a local “vital information” alert. Some firefighters received STA health alerts on their smart watches and were able to drill across to further information right on the watch, such as which responder triggered the alert, the heart rate reading and other parameters, even the responder's map location right on the watch.

### 5.3.3 Crowd/Traffic Sensing and Analysis

Law enforcement is typically on scene directing vehicle and pedestrian traffic but analysis of crowd count observations can increase and broaden the effectiveness of their activities. An analyst during the pilot was able to define a threshold for live crowd count observations that were then retrieved from an SOS service and visualized on the map by color. The analyst also examined a timeline of readings using chart-based situational awareness. A similar approach was used to analyze traffic count observations, visualize road closures and establish safe traffic routes. Traffic observations visualized on the map was able to show routes around the closed areas and towards the incident area becoming congested as traffic built following the closures. With information about traffic levels and road closures easily available, traffic could be routed efficiently around congested areas.

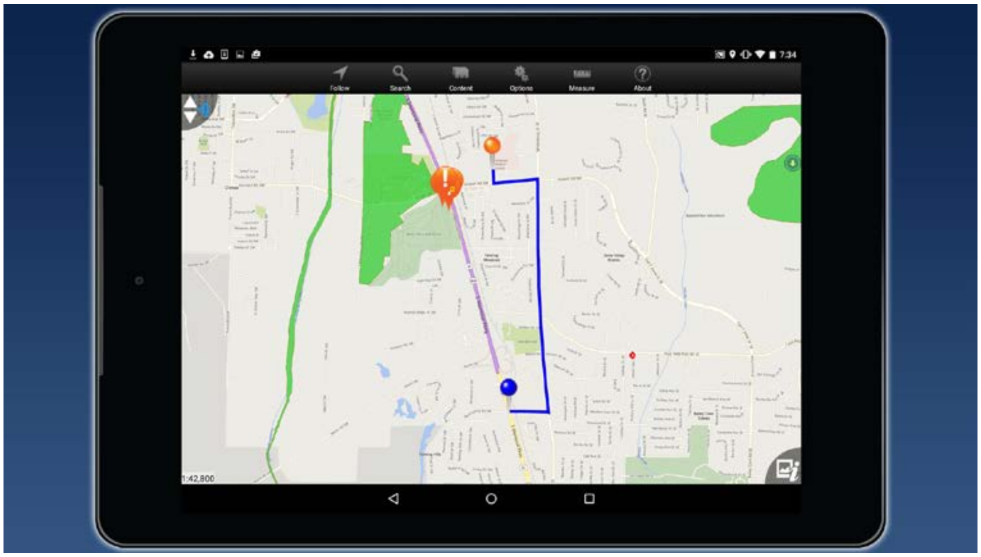


**Figure 5-17: Deployment of Environmental Sensors**

### 5.3.4 Rangefinding

The remote tagging of chlorine gas cylinder locations was accomplished using an off-the-shelf TruPulse 360R Laser Rangefinder. Through Bluetooth, the TruPulse 360R Laser Rangefinder provided distance to target, inclination angle and azimuth angle about north to an Android running OSH. The OSH node on the Android phone also ingests the phone’s location and streams these data in real time to another OSH node running in the cloud. Using the phone’s location and rangefinder observations as input, a SensorML-enabled process on the OSH node calculates the “target” location on demand and offers this as an observable from the SOS.

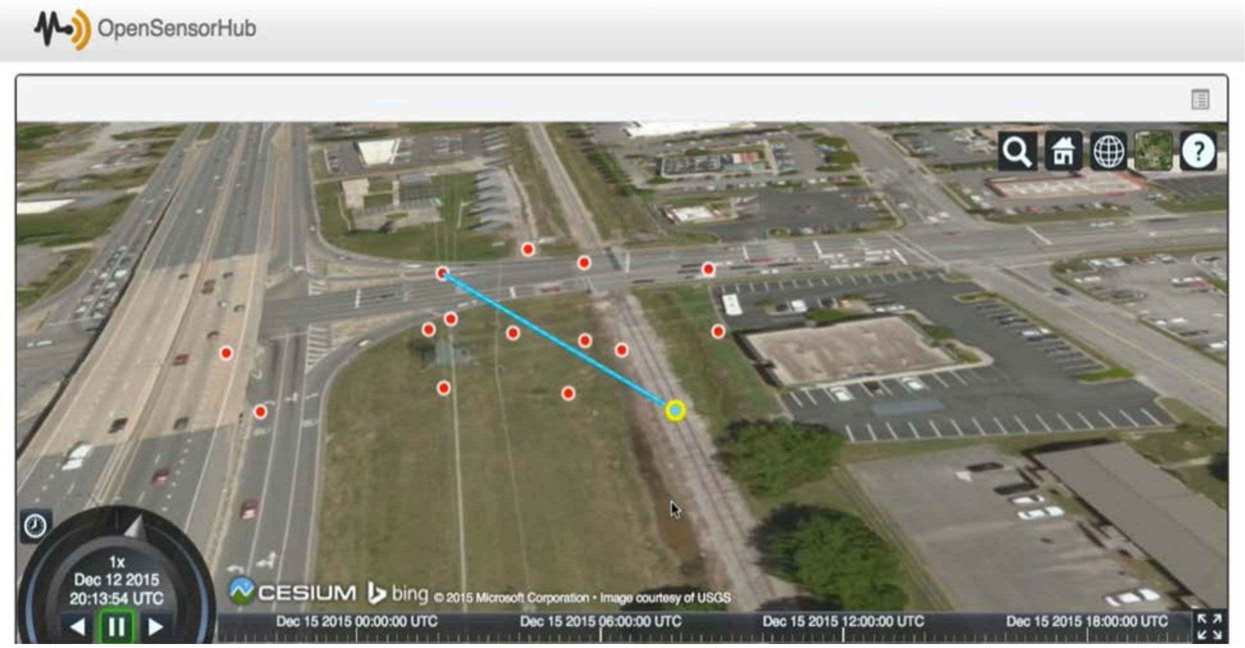


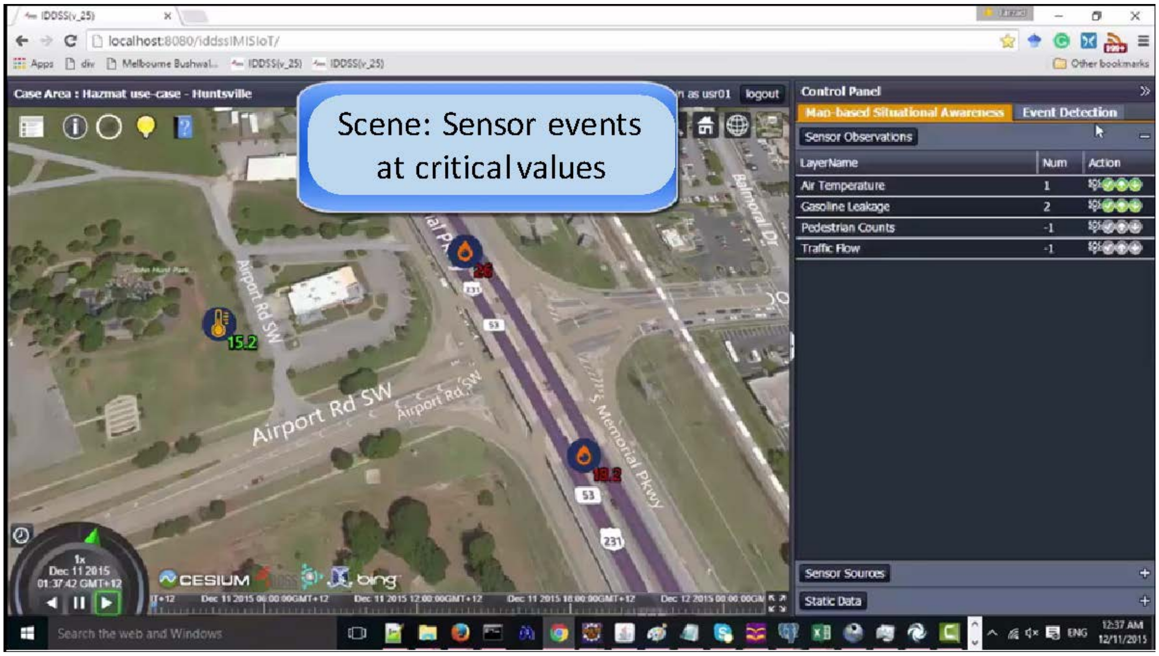
**Figure 5-16: Rangefinder Permits Geolocation of Chlorine Cylinders**

Although the laser rangefinder target locations in Figure 5-16 were used strictly for remotely determining the cylinder locations, the TruPulse 360R Laser Rangefinder can also be used to task other sensors (e.g., skewing the view of a video camera or guiding a UAS). This is because any OSH node or any client speaking "SWE" can use these locations and submit them as tasking parameters to any appropriate sensor that is taskable though an SPS (e.g., one supported by OSH).

## 5.4 Station 4

### 5.4.1 Event Creation and Notification

The OGC IMIS IoT Pilot developed a solution for complex event processing and notification. The resulting architecture is illustrated in Figure 5-18. The entry point for the user into this solution is a client application developed by the University of Melbourne (UM). A screenshot of this client is shown in Figure 5-17.



| LayerName | Num | Action |
|---|---|---|
| Air Temperature | 1 | |
| Gasoline Leakage | 2 | |
| Pedestrian Counts | -1 | |
| Traffic Flow | -1 | |

**Figure 5-17: Screenshot of the Event Notification Client**

This client application enables the retrieval of live sensor data streams. In addition, the client is fully bound to a Web Event Processing Service (WEPS). As a result, the client provides an interactive interface for the user to define the desired notification rule in addition to delivering the events that are detected by WEPS to the subscribed user.

After the user has entered the parameters for an event subscription, an execute request containing these parameters is submitted to the WEPS. This request contains the following information:

- Event filtering rule: Encoded as specified in the OGC Event Pattern Markup Language (EML) Discussion Paper (OGC 08-132), this rule specifies which events are of interest to the user so that a notification messages shall be dispatched if they occur.
- Sampling rate: A value indicating how often new observations are published by the sensor, the sampling rate is used by a feeder to determine how often the data source shall be queried for new observations.
- Runtime: The duration that the subscription shall be active.
- SOS Endpoint: The URL of the SOS server that shall be used by a feeder to retrieve new observations which are relevant for the subscription.
- GetObservation Template (KVP): A KVP-encoded GetObservation request that delivers the observations required for processing the subscription, the feeder automatically adds a temporal filter to this URL. This temporal filter is dynamically generated based on the time stamp of the last observation that was pushed into the event processor.
- GetObservation Template in Plain Old XML (POX): Values for the GetObservation request parameters procedure, observedProperty,

featureOfInterest and responseFormat (equivalent to the corresponding parameters in the KVP GetObservation Template).

After the WEPS has received the execute request containing the subscription, it parameterizes an event processor (in this case Esper, encapsulated by the 52°North Epos framework). In a next step, a feeder process is started which queries the SOS server referenced in the execute process for new observations that are relevant for the subscription. As soon as the feeder discovers new observations, it forwards them to the event processor which checks them against the event pattern rules of the subscription. If the event processing module detect an event that fits to a registered subscription, the notification is pushed to a notification store. This notification store offers an RSS feed which can be consumed by the client so that detected events can be displayed to the user.

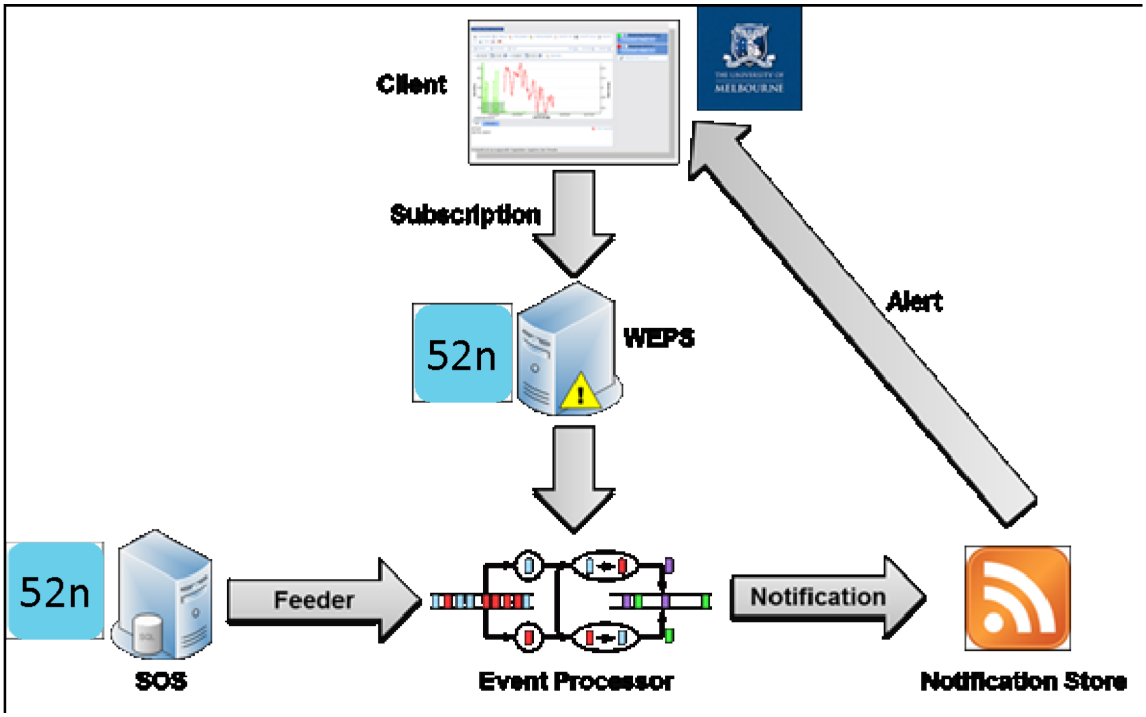
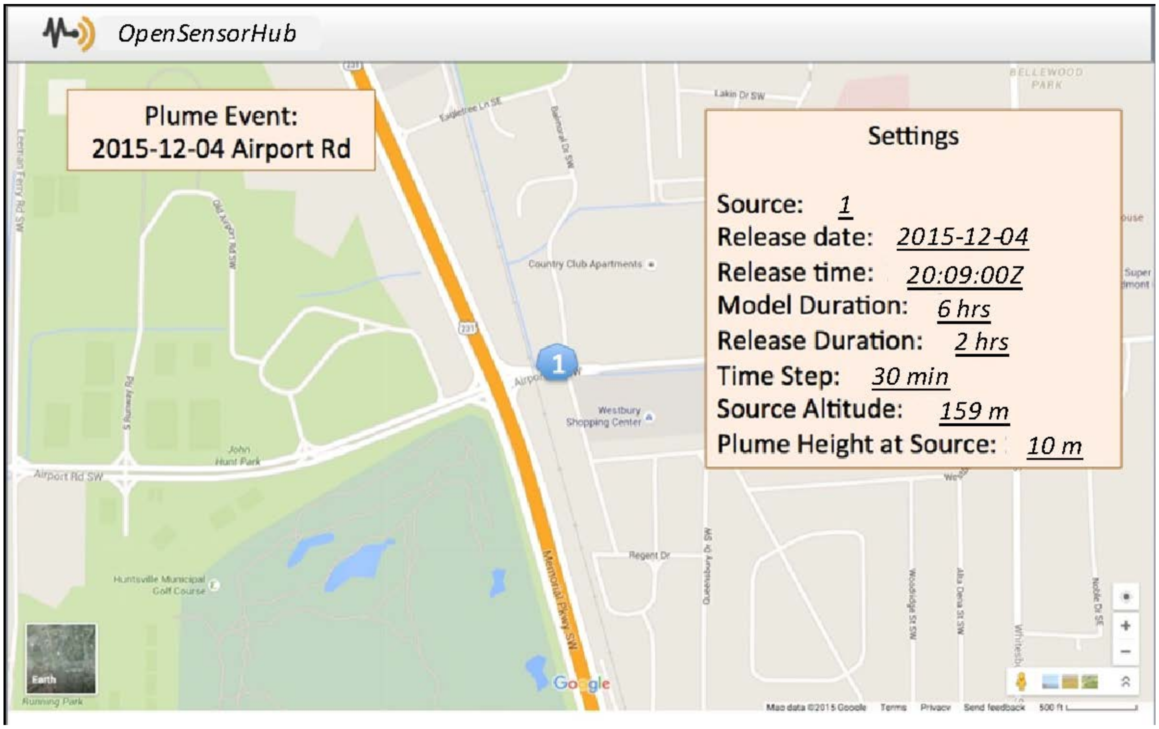


**Figure 5-18: Overview of the Event Processing Architecture**

### 5.4.2 Tasking Model Simulation Processes

Using the SPS Web service interface, one can task any supported sensor, actuator or processing system. The SPS interface is self-describing with regard to tasking parameters available and the requests implemented. Using the DescribeTasking request, one receives a SWE Common Data description defining the required and optional parameters that can be used to task the connected asset. A client can use a pre-configured user interface to provide values for the task, or it can create a tasking interface on-the-fly based on the tasking parameter provided.

Since in SWE we model any sensor, actuator or process as different forms of process, the SPS interface is appropriate for tasking any of these. For the demonstration, an SPS was configured in an OSH node at the University of Alabama at Huntsville (UAH) to enable the tasking of a Lagrangian Plume Model used for forecasting the distribution of a smoke or gas cloud in the atmosphere over a period of minutes to several hours (Figure 5-19). The model generates particles representing a parcel of atmosphere containing aerosols (e.g., smoke, contaminant) and tracks the movement of each of those particles over time; the plume model utilizes the MM5 atmospheric forecast model run twice per day.



Plume Event:
2015-12-04 Airport Rd

Settings

Source: 1
Release date: 2015-12-04
Release time: 20:09:00Z
Model Duration: 6 hrs
Release Duration: 2 hrs
Time Step: 30 min
Source Altitude: 159 m
Plume Height at Source: 10 m

**Figure 5-19: Lagrangian Plume Model**

Once tasking parameter values are set using the interface, the task to execute the model run can be started using the submit request in SPS. The OSH node running at UAH then initiates the model execution and when available, makes the model output results available through an SOS instance running on the same OSH at UAH. The output consists of a large collection of points consisting of model time, particle release time, particle 3-D location and particle source.

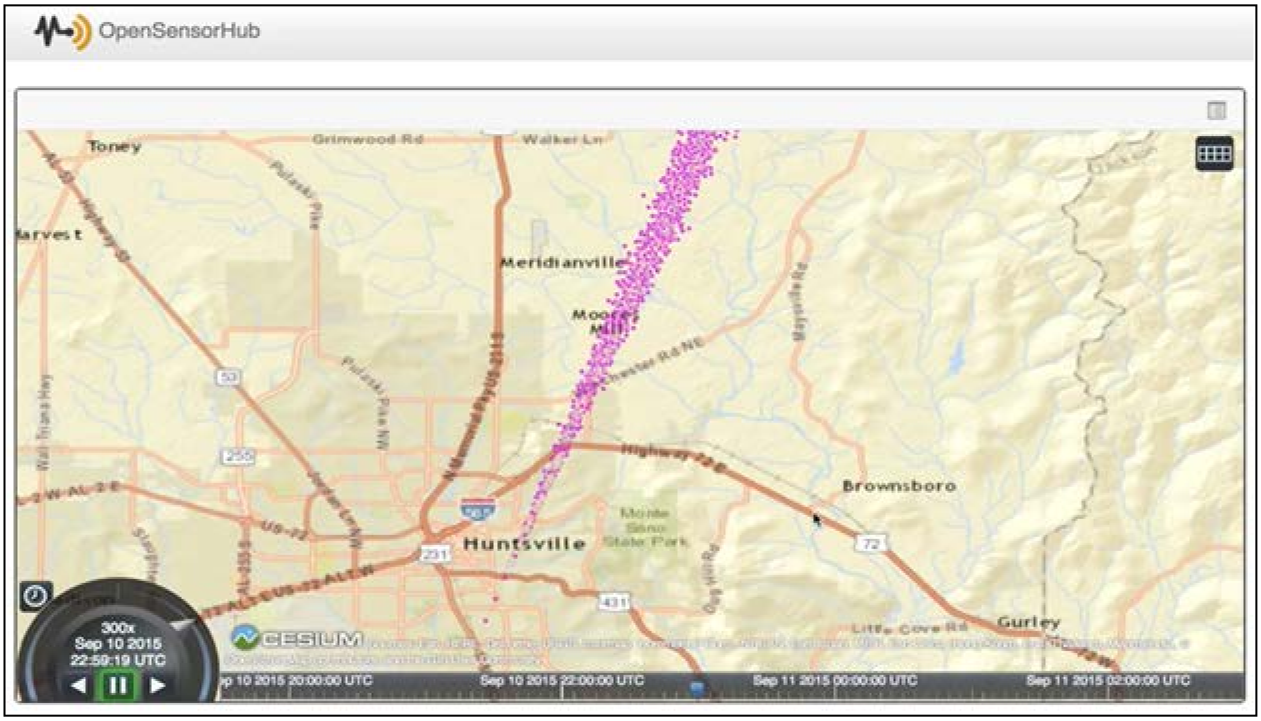


**Figure 5-20: Plume Dispersion**

The plume model was tasked to run for a particular day and time and at the scenario spill location. The plume forecast results were accessed from the SOS with a replay speed of 300x and displayed in a Cesium3D Web browser client (Figure 5-20).

### 5.4.3 Tasking Observation Processes

Similarly, an SPS can be used to task sensors. Tasking can include continuous real-time tasking or tasking the sensor to perform a certain action at some time in the future. The tasking can include a human in the loop typically acting on the sensor through a graphical user interface or can involve machine-to-machine interaction. Furthermore, tasking can involve workflows that utilize measurements from one sensor to task another disparate sensor (as discussed with the laser rangefinder above).

For the demo scenario, a 3DR Solo Drone was SWE-enabled using OSH. The 3DR Solo Drone was selected because in addition to being of sound design, it was also developed as an Open System/Open Standard drone. This allowed us to add software to the drone that would capture all data needed for georectifying the video or imagery onto terrain, and enable the direct streaming of video and navigation parameters into an OSH node on a laptop on the ground. We have the option to deploy OSH on the drone itself or on the flight controller (an Android pad), although we have not yet determined a need for that.

Tasking the drone can be performed manually and in real time using the controller interface delivered with the drone. This interface could be replaced with one that is SWE/OSH enabled allowing for enhanced capabilities, such as the ability to view the real-time footprint on the ground (see Figure 5-21 below). The tasking of the drone is inherently done using the ArduPilot autopilot standard. By creating an SPS adaptor supporting the ArduPilot standard, we could enable an SPS on OSH capable of real-time tasking and flying any drone supporting this common standard.

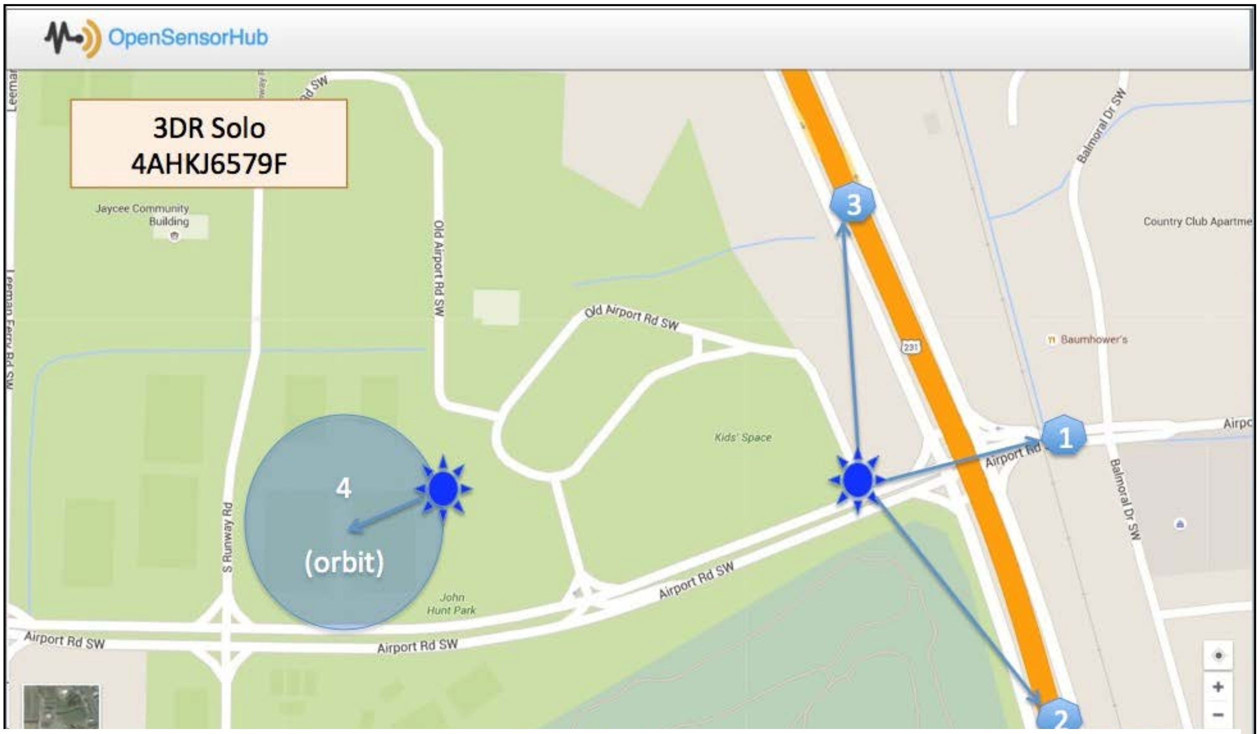
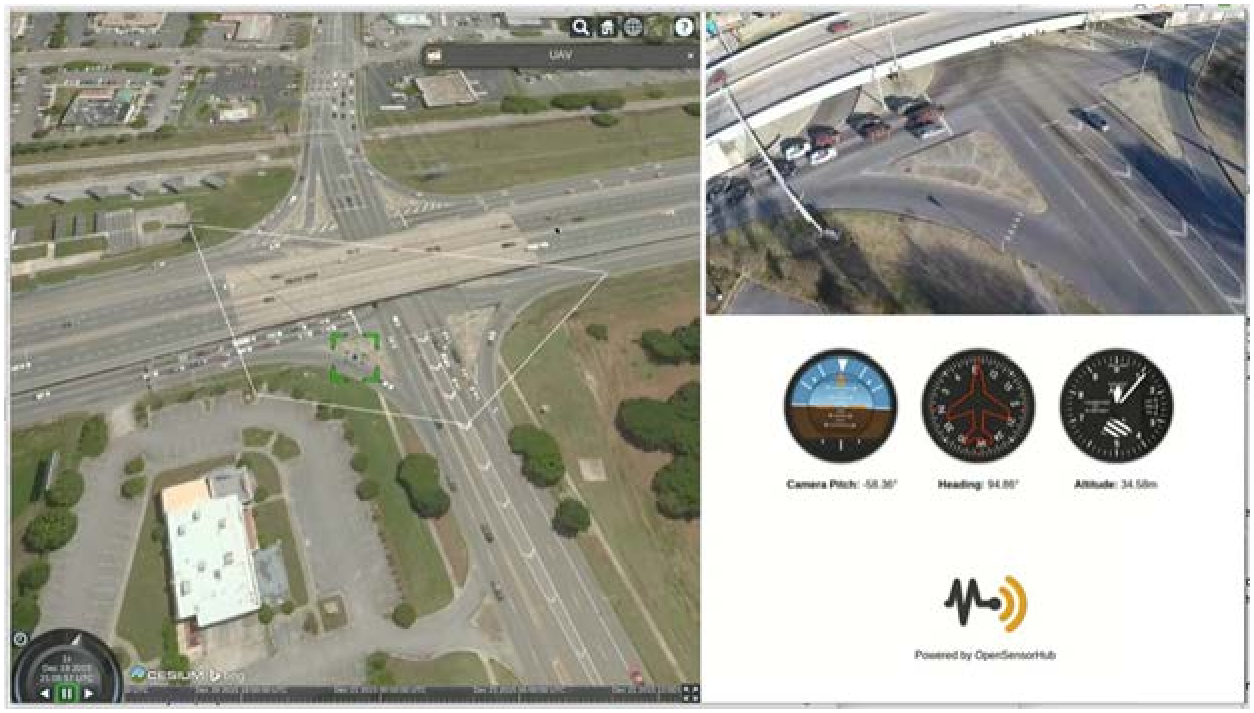


**Figure 5-21: Drone Tasking**

Data ingested by the OSH node includes video (H.264 encoded), drone location, drone orientation, gimbal positions and camera settings. Currently, the data streams to an OSH node on a laptop where it was processed on demand through a SensorML-enabled process chain that determined the geospatial location of the remotely-sensed video observations. The observations and the processed footprints were then available for display on any SWE-enabled client, including the web-based Cesium3D JavaScript client used for the pilot (Figure 5-22).

Subsequent development now allows georectifying the drone video and imagery directly onto terrain in real time (i.e., image draping). Providing fully georectified and draped imagery in near real time would not only allow the incident commander and first responders to have significantly increased situational awareness, but would also allow them to digitally mark geospatially tagged features in the imagery, such as areas of maximum damage, roads blocked by debris or best ingress/egress routes.

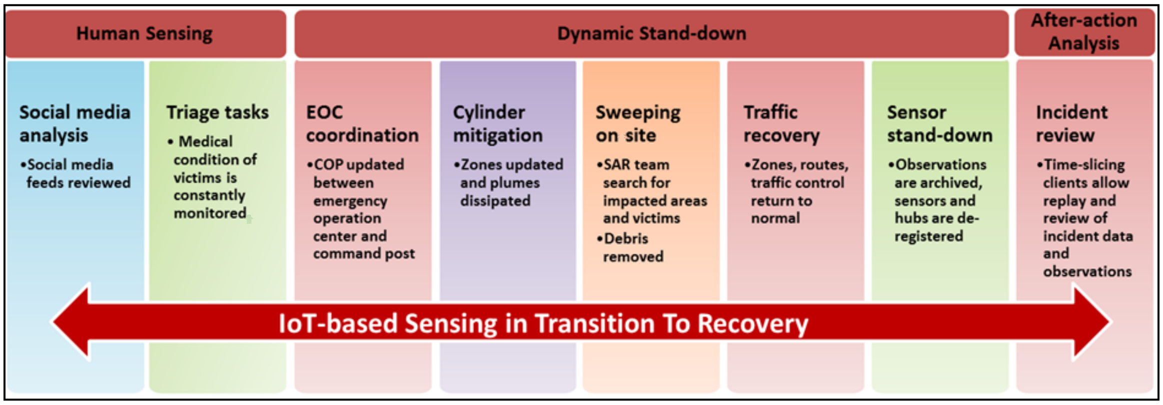


Camera Pitch: -58.36°
Heading: 94.86°
Altitude: 34.58m

**Figure 5-22: Drone Operator Display**

## 5.5 Station 5

Station 5 focuses on the activities in transition to recovery phase that are undertaken during the time period from two to 12 hours in the considered use case. The emphasis of the operations is on returning the situation back to normal, such as removing debris, triaging casualties and recovering traffic. Integrating IoT sensors into the transition to recovery phase could significantly enhance the situational awareness, as well as monitoring and coordinating recovery activities. Figure 5-23 provides an overview of activities conducted in Station 5. Within the IMIS IoT Pilot scenario several participants considered the role of IoT-based sensing during this phase and consequently provided a variety of IoT-based solutions for supporting transition to recovery activities.



Human Sensing

Dynamic Stand-down

After-action Analysis

Social media analysis
- Social media feeds reviewed

Triage tasks
- Medical condition of victims is constantly monitored

EOC coordination
- COP updated between emergency operation center and command post

Cylinder mitigation
- Zones updated and plumes dissipated

Sweeping on site
- SAR team search for impacted areas and victims
- Debris removed

Traffic recovery
- Zones, routes, traffic control return to normal

Sensor stand-down
- Observations are archived, sensors and hubs are de-registered

Incident review
- Time-slicing clients allow replay and review of incident data and observations

IoT-based Sensing in Transition To Recovery

**Figure 5-23: Overview of Station 5 Activities**

### 5.5.1 Human Sensing

During an incident, humans can sense or be sensed. In the former case, the people in the proximity of incident area act as observers and share their observed data through social media channels. In the latter case, the medical condition of the people who are injured or affected during the incident is monitored and prioritized. In both cases, IoT-based sensing can play a significant role to enhance first responders' ability to manage the emergency situation. As part of Station 5, the participants focused on the role of IoT in human sensing and provided solutions for the use cases of social media analysis and triage tasks.

#### Social Media Analysis

Alongside authoritative data, social media observations present an additional source of situational awareness in incident management activities. The considered approach to review social media contents in an emergency situation is described in Station 1. In summary, the approach first defines components for harvesting social media content from different social media platforms (e.g., Instagram, Flickr and Twitter). The retrieved content is then offered through the SOS interface for SOS clients to access. As a result, the SOS offers interoperable access to social media resources as a new type of observation. The same approach can be applied in Station 5 to review the social media observations while transitioning to recovery.

#### Triage Tasks

One of the main activities undertaken during incident recovery phase is the tasks pertaining to casualty triage. The medical condition of those who are injured or affected during the incident needs to be constantly monitored. IoT-based sensors created a great opportunity in emergency medical care through integrating low-power, wireless vital sign sensors and interoperable interfaces to access the sensed health data in real time. These bring the potential to enhance first responders’ ability to assess patients on scene, ensure seamless transfer of data among caregivers and facilitate efficient allocation of hospital resources.

In consideration of triage task in Station 5, SensorUp provided prototype capabilities to incorporate vital sign sensors in emergency medical care. The prototype considered a scenario in which multiple IoT vital sign sensors are deployed in the field. The sensors automatically register themselves to an OGC SensorThings service by sending sensor metadata, location and other relevant contextual data. After the contextual data are established, the sensors then streamed their real-time heart rate and body temperature readings to the SensorThings server via the power-efficient MQTT protocol. At the same time, any web client (e.g., dashboard, mobile phone or smart watch) can subscribe to these sensor data feeds, so that real-time data feeds are pushed to subscribers as soon as they are collected. Figure 5-24 shows the dashboard developed by SensorUp in which first responders can filter and sort the real-time data streams according to configurable thresholds.

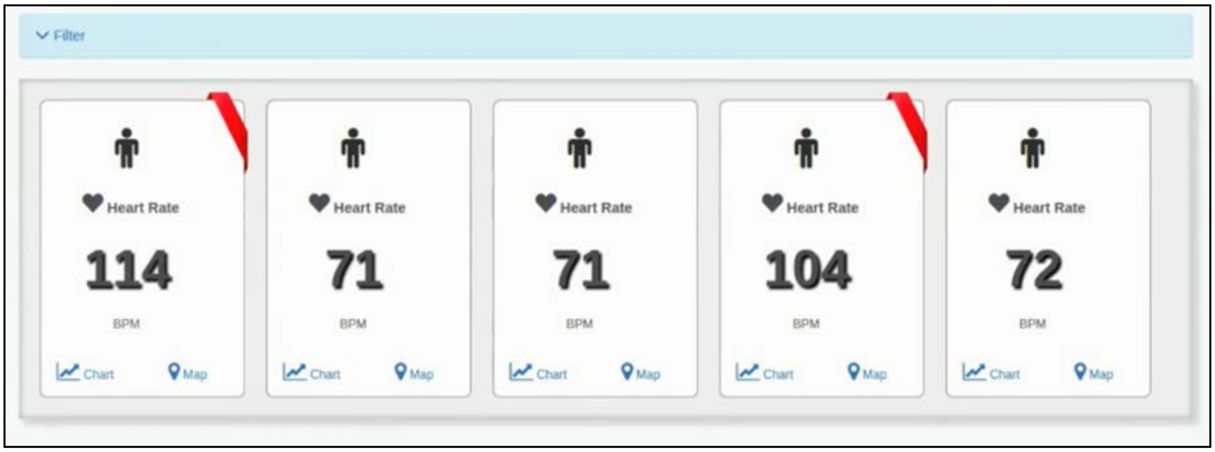


**Figure 5 -24: SensorUp Dashboard Showing the Sensor-derived Heart Rate Data**

### 5.5.2 Dynamic Stand-down

Within the Station 5 scenario, several collaborative activities need to be performed during incident stand-down, such as debris removal, traffic recovery, updating incident zones, etc. These tasks should be performed in a coordinated manner through the collaboration of contributing performers. The IMIS IoT participants aimed to use IoT-based sensors to enhance the process of stand-down activities. The following summarizes the exercised tasks.

#### EOC Coordination

Having performed mitigation activities in Station 4, the current emergency situation needs to be shared amongst all those who are reacting to the incident. A number of IMIS IoT participants focused on performing this task and developed approaches to construct and then exchange a geospatial view of the incident situation (see Station 2 for detailed description). For this purpose, OGC Compliant OWS Context document was applied as a pragmatic solution.

Within the Station 5 scenario, incident reports were exchanged between contributing actors for enhanced coordination of recovery activities. To achieve this, Envitia, Compusult and Exemplar City successfully exercised the task of exchanging the created OWS Context documents between a variety of clients including GeoQ from Exemplar City, Compusult GoMobile, Envitia InSight Android App on Tablet and Envitia Horizon Browser Client. As a result, the same incident reports can be imported, used and updated at different clients that are designed to be used both on scene and at operation centers.

#### Chlorine Cylinder Mitigation

One of the stand-down activities is mitigating the zones that are created during the incident response. Therefore, the status of incident zones needs to be updated and shared between responding teams. Two of participants addressed this necessity within the station scenario. Envitia provided client capabilities for on-the-fly update of incident zones (Figure 5-25).

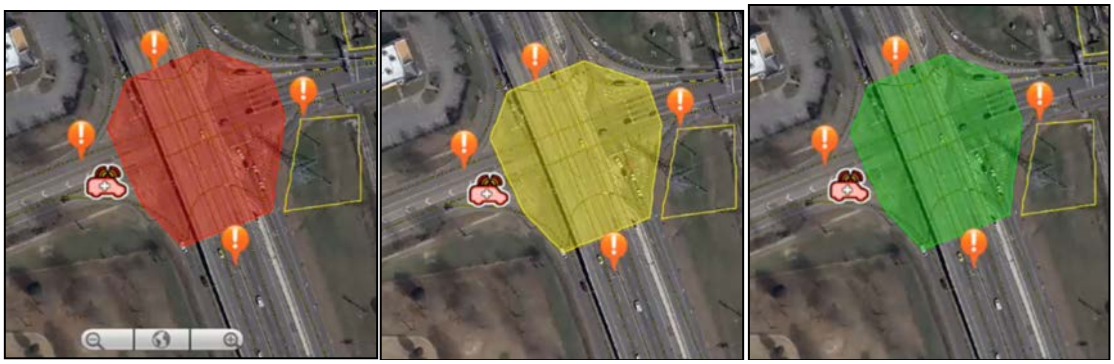


**Figure 5-25: Transition of the Incident Zone from Red to Yellow to Green Using Envitia InSight App**

Also, using GeoQ software tool, Exemplar City monitored the hazmat team's progress in updating the status of each chlorine cylinder and then share the Hazmat Map layer in real time using OGC standards (Figure 5-26).

## Plume Model Re-tasking

One of the indicators determining the mitigation of emergency situations is the dissipation of atmosphere containing aerosols (e.g., smoke, contaminants). Botts team developed an approach based on SPS and Lagrangian Plume Model to simulate the movement of such a contaminating gas in the atmosphere (see Section 5.4 for details).

The plume model was re-tasked based on the Station 5 scenario, so that it provided forecast results showing the dissipation process of the plume (Figure 5-27).

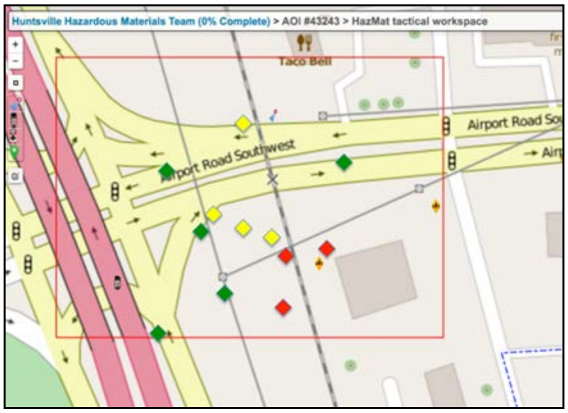

Huntsville Hazardous Materials Team (0% Complete) > AOI #43243 > HazMat tactical workspace
Taco Bell
Airport Road Southwest
Airport Road So
Air

**Figure 5-26: GeoQ Shows Hazmat Teams Progress in Updating Cylinder Status**

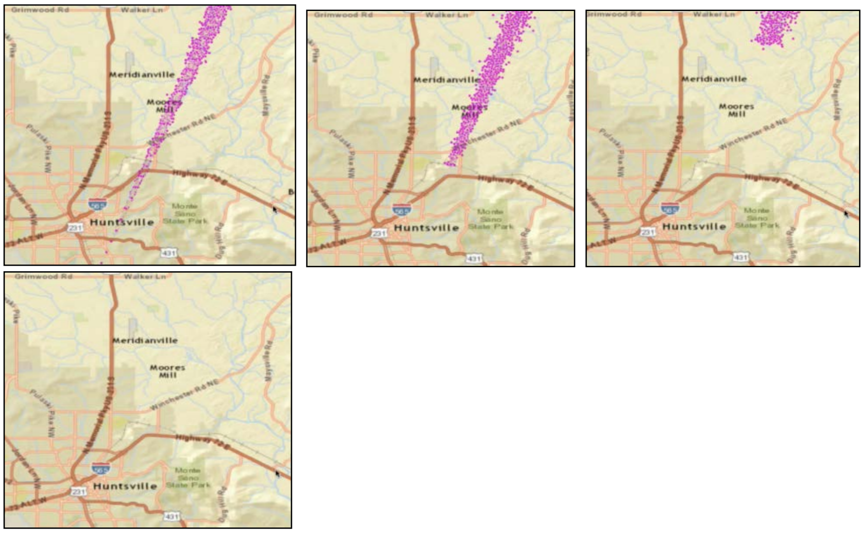


**Figure 5-27: Plume Model Shows Plume Elimination and Dissipation**

## Sweeping On Site

As a major activity in transition to recovery phase, field work needs to be performed by crews from several teams such as SAR and EMS. The crews are shown onsite to search for any overlooked or sheltering victims and annotate the areas of impacted victims. In consideration of this requirement, two of the participants focused on harnessing IoT capabilities to enhance sweeping on-site activities.

Considering utilities as the end-users, Envitia provided capabilities that the utility mobile user can apply to sweep the scene and upload information on the areas of impacted victims, such as downed power lines and nearby a retirement community. Using the Envitia mobile application, the crew can draw annotations of victim areas and upload new context back to catalog. In consideration of SAR teams, GeoHuntsville provided capabilities in which the SAR personnel can update searched locations and status from the field as areas are cleared. Then, the SAR Map layer is shared and updated in real time (using OGC standards) with others in the incident response and available in other situational awareness platforms (Figure 5-28).

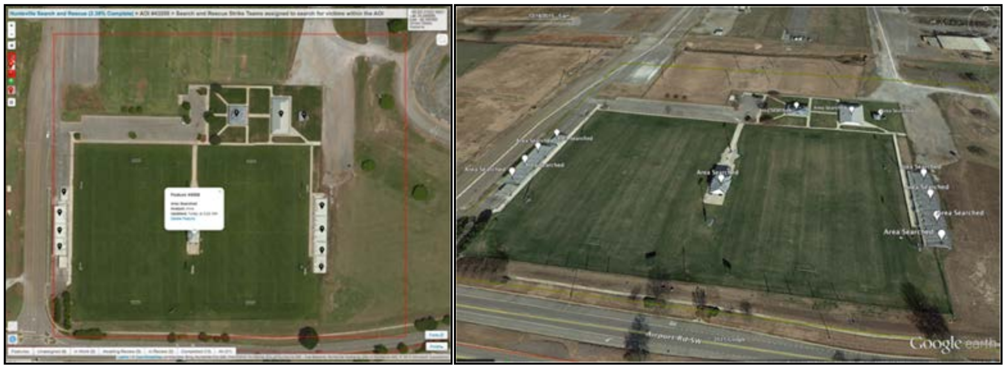


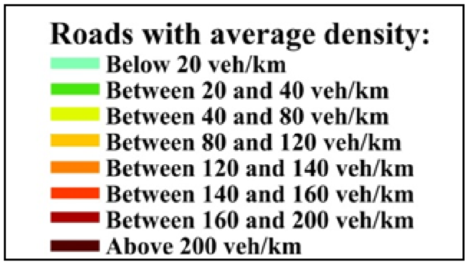
**Figure 5-28: Status Progression as SAR Personnel Sweep the Site**

## Traffic Recovery

In a major incident, traffic control should be conducted in a coordinated and planned manner in different phases for incident detection and response as well as restoration of traffic capacity. Undertaking this coordinated process demands contributions by several actors, including traffic departments, transportation, law enforcement, fire and rescue, medical services, public transport operators, towing and recovery, etc. Integrating IoT-based sensing in traffic incident management procedures could significantly improve the coordination between the contributing actors. In this context, live traffic observations produced by the traffic department could be shared among participating actors. Then, relying on the shared traffic observations, each actor could further process the data to dynamically extract its traffic information requirements.

Within the IMIS IoT scenario, UM provided capabilities related to harnessing IoT sensing in traffic incident management. The main emphasis of UM activities in the pilot reflected consideration of using OGC SWE standards for enabling interoperable access to traffic count observations. To achieve this, a number of steps were undertaken by the UM team. First, the UM client was adjusted to support loading traffic observations described using O&M. For visualizing time-series traffic observations in UM client (which is based on Cesium[1]), the CeZium Markup Language (CZML)[2] format was used. CZML is an open JSON schema for describing properties that change value over time in a Web browser running Cesium. As a result, when the client retrieves flow of traffic observations from SOS server, the roads as the feature of interests are color-themed based on the observations' result value (see Figure 5-29) for density observations.

Having developed the functionality for retrieval and visualization of SOS traffic observations, the step for sensor data simulation within IMIS IoT use case scenario was undertaken. To do this, Aimsun[3] was used for simulating traffic for the roads in the proximity of the collision. The traffic model was then simulated with an incident on Memorial Parkway (the main north/south artery through town). The start of the simulation was defined at 12:00 p.m. The time for collision was set at 12:10 p.m. and it takes six hours to respond, clear the hazmat zone and finally restore the traffic. This collision resulted in blocking the lanes in Memorial Parkway and Airport Road near the incident location. Then, the output of simulation results was prepared and loading into the UM SOS using 52°North SOS Importer.



**Roads with average density:**

🟩 Below 20 veh/km
🟩 Between 20 and 40 veh/km
🟨 Between 40 and 80 veh/km
🟧 Between 80 and 120 veh/km
🟧 Between 120 and 140 veh/km
🟥 Between 140 and 160 veh/km
🟥 Between 160 and 200 veh/km
🟫 Above 200 veh/km

**Figure 5-29: Color Themes to Visualize Traffic Density Observations in UM Client**

Figure 5-27 shows the deployment result for the above-mentioned process within the Station 5 scenario (i.e., from the time when the roads near the hazmat area are closed to the time when the roads are open and traffic is restored).

[1] https://cesiumjs.org/

[2] https://github.com/AnalyticalGraphicsInc/cesium/wiki/CZML-Guide

[3] https://www.aimsun.com/wp/

Sensor Stand-down

As the penultimate activity considered in Station 5, the sensors and S-hubs need to be de-registered and recovered from the field for future use, but data should be cached on cloud services. As a result, development of capabilities related to de-registration of S-hubs from the HubCat was required. Compusult addressed this necessity and provided the functions for insertion, update and de-registration of various S-hubs from the Compusult HubCat. Figure 5-30 shows the Compusult's HubCat automatically de-registering S-hub's from the HubCat. Although the S-hubs are deregistered, the data remains for incident review and future use.

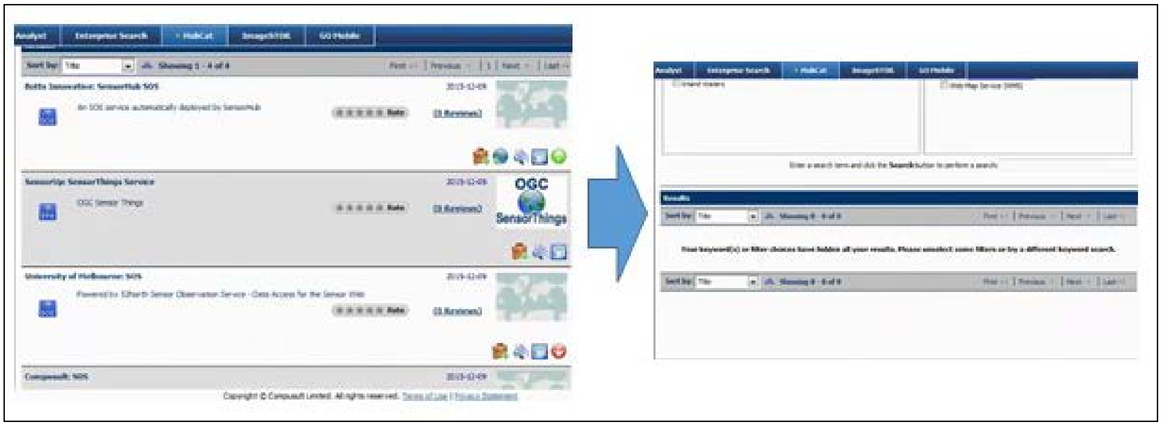

Analyst
Enterprise Search
HubCat
ImageSTOR
GO Mobile
Sort by: Title
Showing 1 - 4 of 4
First | Previous | 1 | Next | Last
Botts Innovation: SensorHub SOS
2015-12-09
An SOS service automatically deployed by SensorHub
Rate
(0 Reviews)
SensorUp: SensorThings Service
OGC Sensor Things
OGC
SensorThings
University of Melbourne: SOS
Powered by 52North Sensor Observation Service - Data Access for the Sensor Web
Compusult: SOS
Copyright © Compusult Limited. All rights reserved. Terms of Use | Privacy Statement
Results
Showing 0 - 0 of 0
Your keyword(s) or filter choices have hidden all your results. Please unselect some filters or try a different keyword search.

**Figure 5-30: Compusult HubCat Shows S-Hubs Automatically Re-register from the Catalog**

### 5.5.3 After-Action Analysis

After action analysis refers to the activities undertaken after closing the incident such as mission/task debrief and incident review. As a related task to IoT, the incident responders and analysts might want to review the incident through the analysis of historical data obtained by IoT sensors. The below subsection outlines the activities undertaken in the IMIS IoT Pilot for incident review.

Incident Review

Within the Station 5 scenario, UM and 52$^{0}$North developed functionality in their software components to replay the historical observations. In this regard, upon selecting a sensor on the map, the 52$^{0}$North client shows the historical SOS observations in a viewer diagram. It then allows for scrolling around the time series. The UM client allows replay and review of historical SOS observations and generated alert notifications (Figure 5-31).

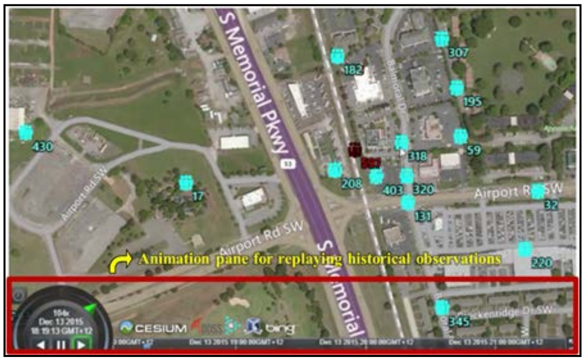


**Figure 5-31: UM Client Enables Map and Chart-based Replay of Observations and Alert Notifications**

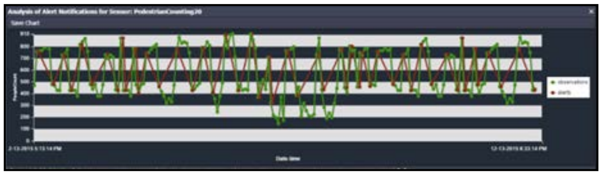
This pilot activity, as a pilot implementation of current information standards and technologies, did not aim to result in deployment-ready products. It did result in many lessons learned and issues to be addressed in reaching the goal of putting IoT devices in useful service for first responders, incident managers, and other stakeholders in emergency response.

## 6.1 Things Matter More Than Devices

The principles that the Pilot sought to implement are termed "Internet of Things" because sensor devices are in themselves not of any great value without an understanding of the parts of the real world, whether natural or manmade, that they sense. A heartrate monitor is useless noise unless it is known "and" communicated <u>whose</u> heart rate is being monitored. If that person switches monitors, the data trend needs to continue to link to the person, not just to the devices. That information transformation from a stand-alone sensor signal to a property of a real-world "Thing" is at the core of the value that IoT can provide.

## 6.2 Collaboration with and on Technology

Initial designs for incident management information sharing can reflect the hierarchical command structure of response organizations and focus on one-way information flows: status flows up from responders and instructions flow down from commanders. Effective response clearly is a much more organic collaboration among response personnel of forming and updating a shared understanding of the situation from which to take response actions. Standards-based technologies applied in this Pilot are particularly effective for supporting this type of information sharing, but sharing strategies need to be prioritized in technology choice and implementation in order to realize this value.


Analysis of Alert Notifications for Sensor: PedestrianCounting20

### 6.3 Too Little Versus Too Much Information

The benefit and the disadvantage of an IoT approach to incident management is the volume and diversity of data that can result from ubiquitous inexpensive sensors. There is a very narrow tipping point between not enough information and an information overload that impedes instead of facilitates action. It is clear from the Pilot activity that ubiquitous data needs to be processed and transformed into selected information presented in ways that help responders take appropriate actions. The danger with selectivity is important information may not make it through the filters set up to do the selecting. The Pilot examined some initial means by which context documents and complex event processing could pull out and deliver the information essential for each response role. In order not to lose information, it is important that filtering and processing of data into actionable information take place at multiple levels and positions within the information sharing system, but the responsibility for what happens to each piece of data needs to be clear.

### 6.4 Common Observation Model

One of the ways in which IoT principles enhance information interoperability is by providing a common model for observations of the world. The rigorous basis for this has roots in the Sensor Web Enhancement (SWE) and related standards / models developed over the last decade. The Pilot application of these principles emphasized that the “Observation” model of data collection can be applied to a wide range of data about the world. “Sensors” can be mechanical devices, but they can also be descriptions of unfolding events that someone posts on Twitter or transmits over their LMR. They can also be predictive models that output “future observations”. As observations with clear targets, measurements, and time/place of collection, these data can be freely shared and interchanged across a unified information sharing infrastructure without the need to create new, isolated conduits every time a new source of data becomes available.

## 7 Next Steps

There are several areas in which advances made during the Pilot showed that further useful work could be done to prove and expand the potential of IoT for incident management.

### 7.1 Experience on the Ground

A frequently asked question for IoT technologies is whether they can actually be useful for first responders. A more useful question the Pilot posed, is “how” and where can IoT be useful. The logistics of working with IoT devices and infrastructure will be as influential as the types of data made available in affecting uptake and adoption. The only way to arrive at IoT operating procedures that work for responders will be to try them and see, under conditions that iteratively approach more and more closely to current, actual incident response procedures and allow responders to adapt procedures to new technologies in a way that works for them. Neither we technologists nor they will be able to do this alone or in the abstract, but together on the ground in well-designed prototyping activities it should be possible.

## 7.2 Security

The collection and provision of sensor observations during incident response raises a number of issues of security, including controlled access to sensitive, sometimes personal information, privacy / physical security of responders and other citizens, integrity of data used for response decisions, physical integrity of ad hoc sensor devices, and a number of others. The Pilot focused on optimizing the collection and sharing of information, but later activities should address security in manageable increments that maintain a reasonable balance between security and accessibility during emergency situations.

## 7.3 Hub Hierarchy

An essential element of the information system design implemented by the Pilot is the S-Hub. It is a system component, described in more detail in the accompanying Protocol Mapping ER, that plays several mediation roles in connecting locally networked, diverse, possible proprietary, likely resource-limited device into an Internet and standards-based network of real-world information. Just as the real network shared by multiple incident management organizations and teams will be diverse and distributed, it makes sense that a hierarchy or constellation of S-Hubs will be most effective at accomplishing their data transformation, processing, caching, and exchange tasks where and when they are needed for seamless sharing of information.  Future activities should expand and detail the roles and capabilities to be assumed by multiple Hub’s acting in concert to get IoT information where it is needed.

## 7.4 Event Architecture (Data and Metadata)

Within the IMIS IoT Pilot several partners developed new approaches on how to implement event-based observation data delivery and notification methods. One of the results was the Web Processing Service for Event Detection (Web Event Processing Service, WEPS) which has been implemented as a specialized WPS server. This approach showed its potential to close the gap of a missing event processing standard in the SWE framework. While existing solutions such as the Sensor Alert Service (SAS) and the Sensor Event Service (SES) are either limited in their function, flexibility, or scope the WEPS relies fully on existing OGC standards and at the same time offers a richer functionality compared to the SES standard. To advance the WEPS further and to increase its suitability for further practical usage scenarios, there are several challenges or open work items which need to be addressed.

The first area of work comprises the way how subscriptions shall be submitted to the WEPS. More specifically, the question is how to describe the event patterns/rules that shall be used for filtering incoming data streams in order to generate notifications. In the IMIS IoT pilot the approach described by the OGC Event Pattern Markup Language (EML) Discussion Paper was successfully used. However, EML in not broadly used in practice there are other event pattern description languages which might be worth to consider. As part of future work it would be helpful to analyze which approaches are used in practice, how these approaches are suited to handle geospatially referenced data, and how event patterns could be best described in environments that are based on OGC standards.

The next topic concerns the management of subscriptions. In the IMIS pilot a rather simple approach was used (mainly due to the limited frame of time and resources). However, as one of the next steps, this approach should be extended and optimized. Also the work of the OGC Pub/Sub SWG might offer a good interface concept which could also be applied to event pattern-based notification use cases.

The question how to handle input data streams in a WEPS is important, as well. Within the IMIS pilot the feeding of observation data was realized based on regularly polling an SOS endpoint. However, there are further approaches which should be considered for future work. These comprise for example:

- Linking MQTT streams as inputs to the WEPS
- Linking of Sensor Things API instances to the WEPS
- Developing a Pub/Sub SOS that delivers automatically all new incoming observations to a WEPS (52°North already has a Pub/Sub SOS implementation but this is based on an earlier version of the specification and it would need some further work to used it in a pilot). Furthermore, a module for consuming data from such a Pub/Sub SOS would have to be added to the WEPS.

Considering these different ways how to deliver data to a WEPS, it would also be good to have a closer look, how information about these input data sources can be transmitted in a subscription request to the WEPS.

Another question is the delivery of event notifications/alerts, if the WEPS has detected an event that fits to a rule. Within the IMIS IoT Pilot RSS feeds were used as a pragmatic solution. But there are more sophisticated approaches which would be worth investigating. These include for example:

- OGC Pub/Substandard which is currently in the voting process
- MQTT based delivery
- WS-N
- OGC Best Practices such as the Web Notification Service
- SOS servers which can be polled for detected events
- …

An output of a follow-up activity could be an ER or even OGC Best Practice describing a WPS Profile for the Processing and Analysis of Event Data Streams.

For certain applications it might also be worth to investigate how to handle security aspects in a WEPS based system (e.g. how to ensure that data is reliably delivered, how to ensure that incoming data streams are not altered by third parties, etc.).

### 7.5 Device Scaling

The Pilot provided experience with deploying and managing limited numbers and types of sensing devices, hub components, and application platforms. Future activities should scale up the number and variety of both components and data types so that issues of enterprise manageability and sustainability critical to effective uptake can also be explored.

## 8 Summary and Conclusions

The IMIS IoT Pilot showed the “Art of the Possible” with respect to using Internet of Things (IoT) enabled technologies as a means to provide First Responders with better Situational Awareness and communications.  The Pilot validated the usefulness of networking IoT sensor testing with in situ environmental sensors, wearable sensors, and imaging sensors on mobile platforms such as UAV’s and autonomous vehicles.

The IMIS IoT Pilot met the stated key objectives:

- The Pilot proved the feasibility of using Internet of Things (IoT) principles to sensing capabilities for incident management

- It demonstrated the feasibility of ad hoc sensor deployment and exploitation by first responder groups within the context of a realistic emergency scenario.

- The Pilot validated the need for a standards-based architecture to share sensor-derived situational awareness across multiple responder organizations.

- It demonstrated the need for the establishment of IoT specifications and incident management best practices to further first responder capabilities to new and more complex emergencies.

The IoT Pilot success has generated a requirement for a contract extension to current IMIS IoT Pilot participants to further integrate already developed IoT Pilot capabilities with selected Next Generation First Responder (NGFR) Apex Program capabilities.  As part of the NGFR Apex Program, IoT data from on-scene S-Hubs will be uploaded to a cloud S-Hub operating in the Public Safety Cloud (PSCloud).  This will allow all agencies involved to retrieve incident information from one location if so desired and will provide a persistent source of observation data even when field S-Hubs are no longer online.