I2MTC 2009 - International Instrumentation and Measurement
Technology Conference
Singapore, 5-7 May 2009

Sensor Alert Web Service for IEEE 1451-Based Sensor Networks

Kang B. Lee
National Institute of Standards and Technology
100 Bureau Drive, MS# 8220
Gaithersburg, Maryland USA 20899-8220
Phone: (301) 975-6604
E-Mail: kang.lee@nist.gov

Eugene Y. Song
National Institute of Standards and Technology
100 Bureau Drive, MS# 8220
Gaithersburg, Maryland USA 20899-8220
Phone: (301) 975-6542
E-Mail: ysong@nist.gov

***Abstract* - This paper describes the Sensor Alert Web Service (SAWS) for the Institute of Electrical and Electronics Engineers (IEEE) 1451-based sensor networks developed at the National Institute of Standards and Technology (NIST). The SAWS consists of the Sensor Alert Notification Producer Service (SANPS) and Sensor Alert Notification Consumer Service (SANCS). The SANPS and SANCS are described in the Web Service Description Language (WSDL) based on Web Services Notification (WS-Notification). A SAWS prototype system consisting of a SAWS consumer, a SAWS provider, and an IEEE 1451.5-802.11 wireless sensor node is presented in the paper. When a sensor reading exceeds the preset sensor thresholds, the SAWS provider will automatically send to the SAWS consumer a sensor alert message described in the Common Alert Protocol (CAP). A case study of sensor alerts is presented to validate the SAWS system.**

Keywords: *CAP, IEEE 1451.0, IEEE 1451.5, NCAP, Sensor Alert Notification Consumer Service, Sensor Alert Notification Producer Service, Sensor Alert Web Service, SOAP, TEDS, TIM, WSDL*

I. INTRODUCTION

Sensors and networks are key components in building distributed sensor networks nationwide for detecting weapons of mass destruction and monitoring critical infrastructure, such as airports, bridges, buildings, railways, utility, and water supplies [1]. An alert is a warning service to notify people of impending danger or emergency. An alert message identifies an imminent problem. A conceptual sensor alert system shown in Figure 1 includes various regional sensor networks deployed at critical locations, such as seaports, airports, bridges, prominent buildings, and train stations across the country. When an emergency is detected, a sensor alert is broadcast to regional coordinators and various agencies of the federal government. If these sensor data and sensor alert messages use standardized communication protocols, it is much easier for the recipients to understand, process the alert messages, and be able to make appropriate decisions in a timely manner.

To achieve such a conceptual sensor alert system, the standardization of sensor data format, sensor communication protocol, and sensor alert message format is essential. Due to recent emergencies, such as hurricanes, tsunami, global terrorist attacks, government leaders have been reminded of the urgent need for emergency communication and notification solutions (alert systems), which can help government agencies to communicate quickly and effectively during public safety incidents [2]. Hence, Sensor Alert Web Service (SAWS) for sensor networks could play a significant role in homeland security applications and other situations where timely information is needed.

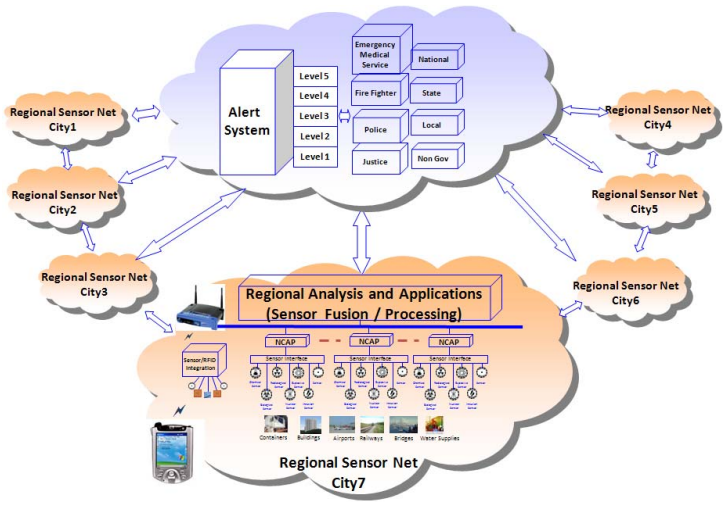

Fig. 1. A conceptual sensor alert system.

This paper mainly focuses on the SAWS for the Institute of Electrical and Electronics Engineers (IEEE) 1451-based sensor networks. Section II describes related work. The SAWS architecture is proposed in section III. A case study for validating the SAWS is illustrated in section IV. A conclusion is provided in section V.

II. RELATED WORK

An alert notification is the detection of events that occur and the automatic generation and delivery of alert messages relating to those events. The publish/subscribe paradigm is a practical pattern for event dissemination in distributed systems [3]. Web service technologies provide a platform-, operating-system-, and language-independent interoperability among different applications; therefore, Web services can integrate heterogeneous applications through the Internet. Web event notification systems enable asynchronous communications among different entities in distributed systems [4]. Web service-based event notification systems utilize the Web service technology to deliver event notifications. Hence, Web service-based event notification systems have the capabilities of combining event notifications and Web services technologies [5]. Multiple related specifications have been proposed to define standard operations and message formats for Web service-based systems. Two major competing specifications are the Web Services Eventing (WS-Eventing) [6] and Web Services Notification (WS-Notification) [7-9]. The WS-Notification includes three specifications: the Web Services Base Notification (WS-BaseNotification) [7], the Web Services Brokered Notification (WS-BrokeredNotification) [8], and the Web Services Topics (WS-Topics) [9]. The family of WS-Notifications was submitted to the Organization for the Advancement of Structured Information Standards (OASIS), in April 2004. The Globus Toolkit 4.0[1] implements WS-BaseNotification [10]. The Apache PubScribe is a Java implementation of the Web Services Notification (WSN) specifications [11]. In the National Oceanic and Atmospheric Administration (NOAA) National Weather Services (NWS) dissemination system, the weather data play a critical role in predicting the path of natural disasters, such as hurricanes and tornadoes, fires, flood, earthquakes, accidents, terrorism, and amber alerts. NWS discovered that both Really Simple Syndication (RSS) and Common Alert Protocol (CAP) appeared to be a natural fit for the dissemination of watches, warnings, and advisories [12]. The Open Geospatial Consortium (OGC)-Web Notification Service (WNS) is a service by which a client may conduct asynchronous dialogues (message interchanges) with one or more other services [13]. The OGC-Sensor Alert Service (SAS) defines an alert as a special kind of notification [14]. The SensorNet is a framework to tie together sensor data from all over the country to create a real-time detection and alert system for various threats, whether they are chemical, radiological, biological, nuclear, or explosive (CBRNE) [15]. This paper primarily focuses on the definition and development of the SAWS for IEEE 1451-based sensor networks based on WS-BaseNotification specification.

III. ARCHITECTURE OF SENSOR ALERT WEB SERVICE FOR IEEE 1451-BASED SENSOR NETWORKS

The family of IEEE 1451 standards defines a set of common communication interfaces for connecting transducers (sensors or actuators) to microprocessor-based systems, instruments, and field networks in a network-independent environment. These standards provide a set of protocols for wired and wireless distributed applications [16]. The IEEE 1451.0 standard provides a common set of commands, functions, electronic data sheet format, and communication protocols for the family of IEEE 1451 standards [17]. The Smart Transducer Web Services (STWS) consists of a common set of web services for IEEE 1451 smart transducers [18, 19]. Figure 2 shows the architecture of the Sensor Alert Web Service (SAWS) for IEEE 1451-based sensor networks. This architecture consists of SAWS consumers, SAWS providers, and sensor nodes. The SAWS provider is an IEEE Network Capable Application Processor (NCAP). Sensor nodes are the IEEE 1451 Transducer Interface Modules (TIMs). The SAWS consists of the Sensor Alert Notification Producer Service (SANPS) resided at the SAWS provider and Sensor Alert Notification Consumer Service (SANCS) resided at the SAWS consumer. The SAWS provides web services for publishing and subscribing to sensor alerts from sensors. It gets sensor data from sensor networks through the STWS.

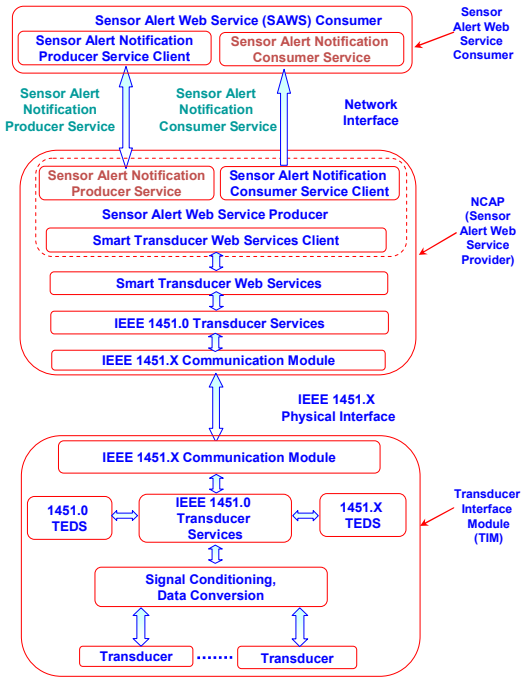

Fig. 2. SAWS architecture for IEEE 1451-based sensor networks.

[1] Commercial equipment and software, many of which are either registered or trademarked, are identified in order to adequately specify certain procedures. In no case does such identification imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.

A. Sensor Alert Web Service (SAWS) Consumer

The SAWS consumer consists of the SANPS client and SANCS service. The SAWS consumer can use the SANPS client to call the subscribe service of the SANPS to register or subscribe for a sensor alert topic to the SAWS provider. The SAWS consumer will consequently receive sensor alert notifications from a SAWS provider through the SANPS.

B. Sensor Alert Web Service (SAWS) Provider

As shown in Figure 2, the SANPS of the SAWS provider can accept incoming subscription requests from the SAWS consumer and send a response back to the SAWS consumer. Each subscription request identifies one or more topics of interest and a reference to a SAWS consumer. The STWS client of the SAWS provider can communicate with the STWS for sensor data. Then the request is passed on to the NCAP 1451.X communication module through the IEEE 1451.0 transducer service. Finally, the sensor data request is conveyed to the TIM 1451.X communication module in a wireless sensor node. The SANCS client (here defined as an alert notifier) of the SAWS provider creates a sensor alert notification in a CAP message based on the sensor threshold analysis. It then sends the alert notification to the SAWS consumer by calling the notify service of the SANCS.

C. Sensor Nodes

Sensor nodes, the IEEE 1451 TIMs, consist of physical sensors and actuators, sensor signal conditioning, data conversion, IEEE 1451.0 TEDS, IEEE 1451.X PHY TEDS, IEEE 1451.0 transducer services, and IEEE 1451.X communication module. The TIMs provide sensor data for the NCAP through IEEE 1451.X communication module.

IV. SENSOR ALERT WEB SERVICE (SAWS) DESCRIPTION

The Web Services Description Language (WSDL) is a XML-based language used to define Web services and describe how to access them. The SANPS and SANCS of SAWS are described in WSDL. The WSDL specification consists of six major elements: definitions, types, message, port type, binding, and service. The definition element defines the name of the Web service, declares multiple namespaces used in this paper, and contains all the service elements described. The type element describes all the data types for messages exchanged between the service provider and consumer. All data types are defined by XML Schemas. The message element defines the name of the message and contains one or more message part elements. The port type element describes a Web service with the operations that can be performed, and the messages involved. A port type is a named set of abstract operations and the abstract messages involved. The binding element describes the concrete specifications of how the services can be implemented on the network. The WSDL specifies the style of the binding as either a Remote Procedure Call (RPC) or document. For document style in the SAWS, the content is specified by the XML Schema defined in the WSDL type section. It does not need to follow a specific SOAP convention. In our implementation, we use the SOAP document/literal style. The service element defines the address for invoking the specified service. Most commonly, this includes a uniform resource locator (URL) for invoking the SOAP service.

A. Sensor Alert Notification Consumer Service

The Sensor Alert Notification Consumer Service (SANCS) of SAWS consumer includes a notify service. The SAWS consumer can receives sensor alert notifications from the SAWS provider. When emergency situations are detected, the SANCS client of the SAWS provider is responsible for sending a sensor alert notification message to the SAWS consumer by calling the notify service of the SANCS. The notify service is a one-way operation. The SANCS is described below in WSDL based on the WS-BaseNotification and CAP message, which provides an open, non-proprietary digital message format for all types of alerts and notifications [20].

Type definitions:

```
<types>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified"
targetNamespace="http://sensor10.MEL.NIST.GOV/SensorAlertNotification
ConsumerService">

<CAP message here>
……
<xsd:complexType name="NotifyServiceRequestType">
<xsd:sequence>
          <xsd:element name="topic" type="xsd:string"/>
          <xsd:element name="topicExpression" type="xsd:string"/>
          <xsd:element name="sensorLocation" type="xsd:string"/>
          <xsd:element name="producerReference" type="xsd:string"/>
          <xsd:element name="alertMessage" type="ncs:alertType"/>
</xsd:sequence>
</xsd:complexType>
<xsd:element name="NotifyServiceRequest"
type="ncs:NotifyServiceRequestType"/>
</schema>
</types>
```

Message Definitions:

```
<message name="NotifyRequest">
<part name="requestParameters" element="ncs:NotifyServiceRequest"/>
</message>

<portType name="SensorAlertNotificationConsumerServicePortType">
          <operation name="Notify">
                    <input message="ncs:NotifyRequest"/>
          </operation>
</portType>
```

Service binding:

```
<binding name="SensorAlertNotificationConsumerServiceSoapHttpBinding"
type="ncs:SensorAlertNotificationConsumerServicePortType">
<soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
<operation name="Notify">
          <soap:operation soapAction="urn:#Notify"/>
                    <input>
                              <soap:body use="literal"/>
                    </input>
</operation>
</binding>
```

Service location:

```
<wsdl:service name="SensorAlertNotificationConsumerService"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
<wsdl:port name="SensorAlertNotificationConsumerServicePortTypePort"
binding="ncs:SensorAlertNotificationConsumerServiceSoapHttpBinding">
<soap:address
location="http://localhost:8080/SensorAlertNotificationConsumerServic
e/SensorAlertNotificationConsumerService"/>
</wsdl:port>
```

```
</wsdl:service>
```

B. Sensor Alert Notification Producer Service

The Sensor Alert Notification Producer Service (SANPS) of the SAWS provider includes a subscribe service. The SAWS consumer can register or subscribe for sensor alert topics by calling the subscribe service of SANPS. The SAWS provider accepts incoming subscription requests from the SAWS consumer. Each subscription request identifies one or more topics of interest and a reference to the SAWS consumer. The subscribe service is a two-way operation. The SANPS is described below in WSDL based on the WS-BaseNotification.

Type definition:

```
<types>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" targetNamespace="http://
localhost/SensorAlertNotificationProducerService">

<xsd:element name="SubscribeServiceRequest"
type="nps:SubscribeServiceRequestType"/>

<xsd:complexType name="SubscribeServiceRequestType">
<xsd:sequence>
<xsd:element name="consumerEndpointReference" type="xsd:string"/>
<xsd:element name="topicExpression" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>

<xsd:element name="SubscribeServiceResponse"
type="nps:SubscribeServiceResponseType"/>

<xsd:complexType name="SubscribeServiceResponseType">
<xsd:sequence>
<xsd:element name="subscriptionReference" type="xsd:string"/>
<xsd:element name="status" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>

</schema>
</types>
```

Message Definition:

```
<message name="SubscribeRequest">
<part name="requestParameters"
element="nps:SubscribeServiceRequest"/>
</message>

<message name="SubscribeResponse">
<part name="requestParameters"
element="nps:SubscribeServiceResponse"/>
</message>
```

PortType Definition:

```
<portType name="SensorAlertNotificationProducerServicePortType">
        <operation name="Subscribe">
                <input message="nps:SubscribeRequest"/>
                <output message="nps:SubscribeResponse"/>
        </operation>
</portType>
```

Service Binding:

```
<binding name="SensorAlertNotificationProducerServiceSoapBinding"
type="nps:SensorAlertNotificationProducerServicePortType">
<soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
        <operation name="Subscribe">
                <soap:operation soapAction="urn:#Subscribe"/>
                <input>
                        <soap:body use="literal"/>
                </input>
                <output>
                        <soap:body use="literal"/>
                </output>
        </operation>
</binding>
```

Service Location:

```
<wsdl:service name="SensorAlertNotificationProducerService"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
<wsdl:port name="SensorAlertNotificationProducerServicePortTypePort"
binding="nps:SensorAlertNotificationProducerServiceSoapBinding">
<soap:address location="PUT ACTUAL ADDRESS HERE"/>
</wsdl:port>
</wsdl:service>
```

V. PROTOTYPE SYSTEM OF SENSOR ALERT WEB SERVICE FOR IEEE 1451.5-802.11 WIRELESS SENSOR NETWORK

A. Prototype Sensor Alert Web Service System

Figure 3 shows a prototype SAWS system for an IEEE 1451.5-802.11 wireless sensor network. The system consists of three parts: a SAWS consumer, a SAWS provider (IEEE 1451 NCAP), and a wireless sensor node (IEEE 1451.5-802.11 WTIM (wireless TIM)).

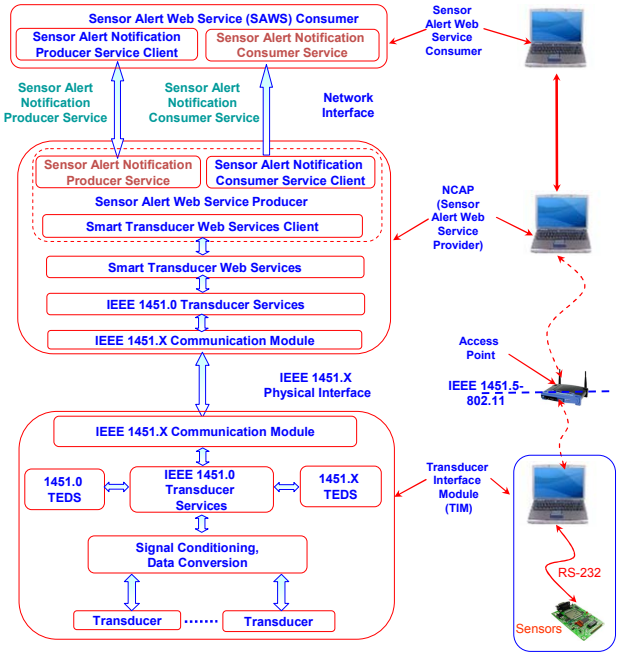

Fig. 3. Prototype SAWS system for IEEE 1451.5-802.11 wireless sensor network.

Figure 4 shows the sequential processes of the Sensor Alert Web Service. The processes are defined in five steps as follows:

- Step 1: The SANPS client of the SAWS consumer subscribes for a sensor alert topic to the SAWS provider by calling the subscribe service of the SANPS of the SAWS provider.
- Step 2: The SAWS provider sends the responses to the SAWS consumer.
- Step 3: The STWS client of the SAWS provider sends sensor data request to the STWS by calling the read transducer data service of the STWS, which communicates to the IEEE 1451.0 transducer service by calling read data service of the IEEE 1451.0 transducer service, which communicates with the

WTIM via the IEEE 1451.5-802.11 communication module [21].

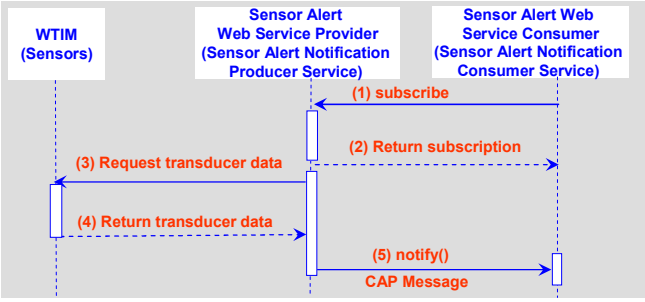

Fig. 4. Sequential processes of Sensor Alert Web Service.

- Step 4: WTIM receives the sensor data request, reads data from a sensor, and then sends the sensor data back to the SAWS provider through the IEEE 1451.5-802.11 communication module.
- Step 5: The SAWS provider gets the sensor data through the STWS and analyzes the data by comparing them with the preset thresholds. If the sensor data go beyond the sensor thresholds, the SANCS client of the SAWS provider automatically sends an alert notification (CAP message) to the SAWS consumer by calling the notify service of SANCS. The SAWS consumer, a decision maker, can make a decision based on the sensor alert information (CAP message).

VI. CASE STUDY OF SENSOR ALERT WEB SERVICE

This section discusses a case study of a SAWS integrated with an IEEE 1451.5-802.11 wireless sensor network.

Fig. 5. Screenshot of a sensor alert subscription.

A. Register a Sensor Alert to Sensor Alert Web Service

Figure 5 shows a screen shot of a SAWS consumer subscribing for a "SensorAlert" topic to the SAWS provider. The SANPS client of the SAWS consumer registers or subscribes for a sensor alert topic to the SAWS provider by calling the subscribe service of the SANPS. The subscription request shown in Figure 5 includes a reference of a consumer and a sensor alert topic. The SANPS of the SAWS provider sends the response to the SAWS consumer. The response shown in Figure 5 includes references of the consumer and provider, and the status of the subscription.

B. Notify a Sensor Alert CAP Message

When the SAWS provider gets a sensor alert subscription from the SAWS consumer, it sends a request to the WTIM for sensor data. The request includes parameters, such as sensor thresholds (e.g., minThreshold=5 and maxThreshold=20). The WTIM gets the sensor data request from the SAWS provider. Then it reads the sensor data and sends it back to the SAWS provider. Figure 6 shows a sensor reading of 3 from the WTIM.

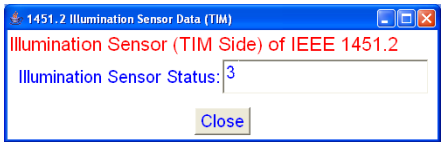

Fig. 6. Screenshot of a sensor reading from a WTIM.

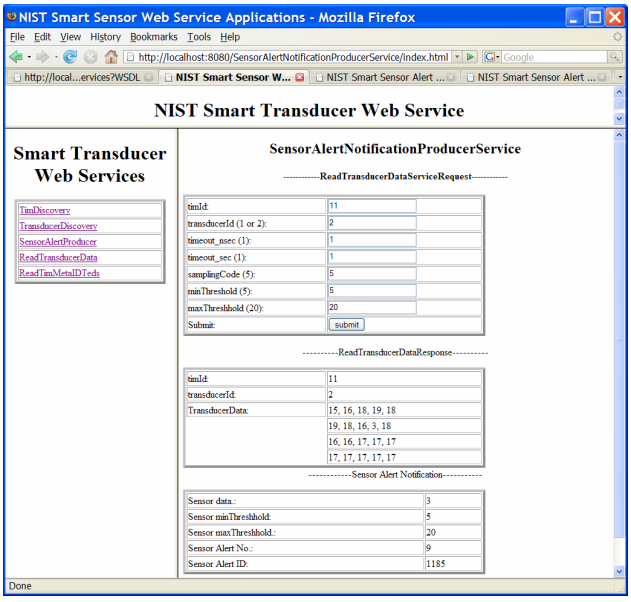

Fig. 7. Screenshot of a sensor alert Web service.

Figure 7 shows some sensor data delivered to the SAWS provider. The SAWS provider analyzes them based on the preset sensor thresholds. If the sensor reading is less than and equal to 5, or if the sensor reading is great than and equal to 20, then a sensor alert notification (CAP message) will automatically be sent to the SAWS consumer by calling the notify service of the sensor alert notification consumer

service. In Figure 7, a screenshot shows that the sensor alert ID is 1185 with a sensor reading of 3, therefore an alert message is automatically send out to SANCS. Figure 8 shows this sensor alert notification message in the CAP format. The decision maker can thus make an appropriate decision based on the sensor alert CAP message.

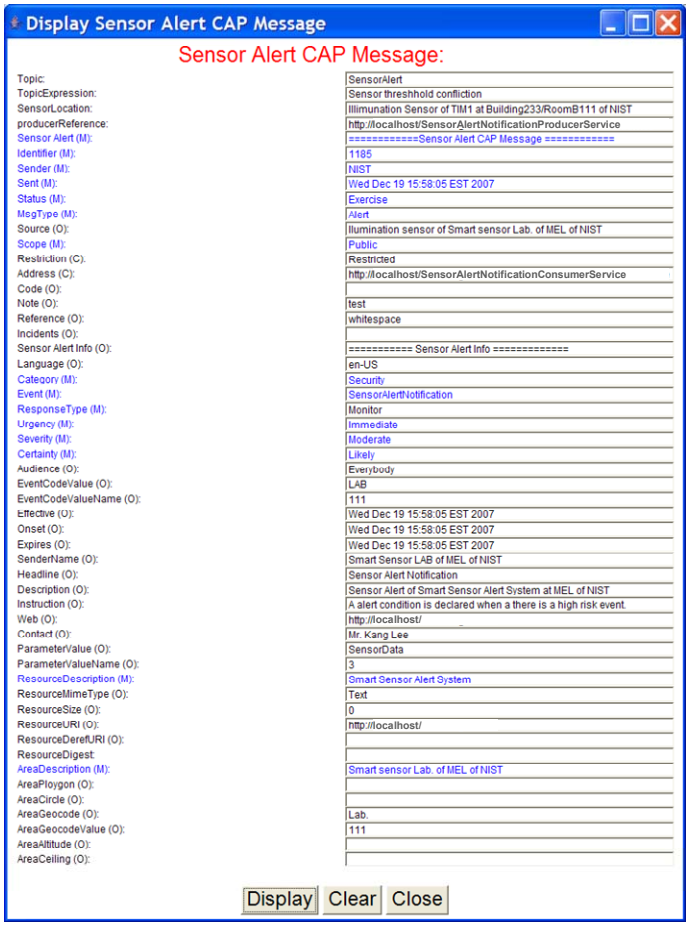

Fig. 8. Screenshot of a sensor alert CAP Message.

VII. CONCLUSION

The Sensor Alert Web Service (SAWS) for IEEE 1451-based sensor networks was presented. A SAWS prototype system was described and a case study was presented that illustrated the Sensor Alert Web Service was functionally integrated with the IEEE 1451.5-802.11 wireless sensor network. This work provides a good foundation for the standardization of sensor alert Web services.

REFERENCES

[1] Kang. B. Lee, Mark. E. Reichardt, "Open Standards for Homeland Security Sensor Networks", IEEE Instrumentation & Measurement Magazine, Dec. 2005, Vol. 8, No. 5, p.14-21.

[2] Emergency Communication and Notification Solutions for Government and Business Conference. (2007, May 20) [Online] Available: http://homelanddefensejournal.com/hdl/Emergency-Communication-notification.htm.

[3] P. T. Eugster, et al., "The Many Faces of Publish/Subscribe", ACM Computing Surveys, Vol. 35, No. 2, June 2003, pp.141-131.

[4] Yi Huang and Dennis Gannon, A Comparative Study of Web Services-based Event Notification Specifications, (2008, October 15) [Online] Available: http://www.cs.indiana.edu/~yihuan/research/yhuang-comparativeStudy.pdf.

[5] Yi Huang and Dennis Gannon, A Flexible and Efficient Approach to Reconcile Different Web Services-based Event Notification Specifications, (2008, Sept. 20) [Online] Available: http://www.cs.indiana.edu/~yihuan/research/huang-reconcile.pdf.

[6] D. Box, L. F. Cabrera, et al., "Web Services Eventing", (2008, July 10) [Online] Available: http://ftpna2.bea.com/pub/downloads/WS-Eventing.pdf.

[7] Web Services Base Notification 1.3, (WS-BaseNotification), OASIS Standard, 1 October 2006. [Online] Available: http://docs.oasis-open.org/wsn/wsn-ws_base_notification-1.3-spec-os.pdf.

[8] Web Services Brokered Notification 1.3 (WS-BrokeredNotification), OASIS Standard, 1 October 2006, [Online] Available: http://docs.oasis-open.org/wsn/wsn-ws_brokered_notification-1.3-spec-os.pdf.

[9] Web Services Topics 1.3 (WS-Topics), OASIS Standard, 1 October 2006, [Online] Available: http://docs.oasis-open.org/wsn/wsn-ws_topics-1.3-spec-os.pdf.

[10] B. Sundaram, "WS-Notification and the Globus Toolkit 4 WS-Java Core", (2008, 10 25) [Online] Available: http://www-128.ibm.com/developerworks/grid/library/gr-wsngt4/

[11] Pubscribe, (2005, Oct. 25,) [Online] Available: http://today.java.net/pub/n/Pubscribe1.1

[12] Karen L. Jones, Hung Nguyen, An end-to-end architecture for distributing weather alerts to wireless handsets, Proceedings of SPIE, the International Society for Optical Engineering, June 2005, Vol. 5819, pp. 427-435,

[13] OGC Web Notification Service, Open GIS Consortium Inc. Date: 2003-04-21, Reference number of this OpenGIS, Project Document: OGC 03-008r2, Version: 0.1.0.

[14] George Percivall, OGC® Sensor Web Enablement Standards, Sensors & Transducers Journal, Vol.71, Issue 9, September 2006, pp.698-706.

[15] Bryan L. Gorman, Mallikarjun Shankar, and Cyrus M. Smith, Advancing Sensor Web Interoperability, April 1, 2005, [Online] Available: http://www.sensornet.gov/sen4_45_05e.pdf.

[16] K. Lee, "IEEE 1451: A Standard in Support of Smart Transducer Networking", Proceedings of the 17th IEEE Instrumentation and Measurement Technology Conference, Baltimore, MD, May 1-4, 2000, Vol. 2, pp. 525-528.IEEE 1451

[17] IEEE 1451.0, Standard for a Smart Transducer Interface for Sensors and Actuators– Common Functions, Communication Protocols, and Transducer Electronic Data Sheet (TEDS) Formats, IEEE Instrumentation and Measurement Society, TC-9, The Institute of Electrical and Electronics Engineers, Inc., New York, N.Y. 10016.

[18] Eugene Y. Song, Kang Lee, Smart transducer web service based on IEEE 1451.0 standard. IMTC 2007 - Instrumentation and Measurement Technology Conference, WARSAW, POLAND, MAY 1-3, 2007

[19] Eugene Y. Song, Kang Lee, "STWS: A unified Web service for IEEE 1451 smart transducers", IEEE Transaction on Instrumentation and measurement, Vol.57, No.8, Aug. 2008, pp.1749-1756

[20] Common Alert Protocol, (2005, October 1) [Online] Available: http://www.oasis-open.org/committees/download.php/14759/emergency-CAPv1.1.pdf.

[21] IEEE 1451.5, Standard for a Smart Transducer Interface for Sensors and Actuators– Wireless Communication and Transducer Electronic Data Sheet (TEDS) Formats, IEEE Instrumentation and Measurement Society, TC-9, The Institute of Electrical and Electronics Engineers, Inc., New York, N.Y. 10016.