Using Ontological Information to Enhance Responder Availability in Emergency Response

Paul Ngo[1] and Duminda Wijesekera[1],

[1] Department of Computer Science, George Mason University,
Fairfax, VA 22030
{pngo1, dwijesek}@gmu.edu

Abstract. Ensuring effective communications during emergencies is an important issue for any functional government. One way to address this issue is to ensure the availability of the key personnel capable of making the appropriate decisions and taking timely actions with sufficient resources. Many XML-based languages such as the Emergency Data Exchange Language (EDXL) and associated Common Alert Protocol (CAP) have been designed to provide a basis for such communications. To ensure that messages are delivered in a timely manner, we propose some role and task based ontological enhancements for these languages. We show by example how the ontological enhancements can be used to enhance availability of emergency personnel in case of a need.

Keywords: Emergency Availability, Emergency Ontology, Emergency Response.

1 Introduction

Multiple mega-scale emergencies highlight the need for better global emergency response. The September 11th 2001 terrorist attacks in New York, Indonesian Tsunami in 2004, Hurricane Katrina in 2005, Sichaun earthquake in 2008, and the Haiti earthquake and Pakistani floods in 2010 are examples of a few. During these emergencies, urgent task-related communications must reach key officials in a timely manner. Emergency responders must know how to contact the person in charge of a specific task, which is sometimes difficult due to not being able to locate a telephone number, or when reached using directory information, the person may not be available or may have been reassigned to a different job/task. There is no automated method of redirecting the call to the current person who should be attending to that task and is on-duty at the time of the call. It's preferable to have a subject, task specific, 911-like calling number for each task, time and locality. The objective of this research is to reach such a capability for the real-time needs of emergency responders.

The basic 911 services provided in the USA serve as a pseudo name that is available to the general public at every time and every location, but is mapped to a collection of numbers belonging to an emergency call center based on the call originator's location. Although we take it for granted, the public switched telephone network (PSTN) has been designed to translate the pseudo name 911 to a location specific telephone number. Thus this address translation depends only on a single parameter, *the caller's location*. Our objective is to extend this capability in order to facilitate the communication beyond the first call from the public. The issue of extending this paradigm for emergency responders to contact each other depends on a plethora of parameters, nature of the emergency, priority of immediate needs and resources to fulfill them. We agree that if a person is not available to receive the request, the communication breaks down. But often, locating this person takes multiple calls/SMS and email messages before the correct person can be reached. It is this gap that we propose to fill by developing an ontology (hence the lexicons) as the need to parameterize the basic 911 service.

With support from the Department of Homeland Security Disaster Management eGov Initiative, the Organization for the Advancement of Structured Information Standards (OASIS) technical committee on emergency management developed a set of standards for the interagency exchange of emergency management data and messaging [1,2,3]. Standards [1] and [2] developed the Emergency Data Exchange Language (EDXL) that provides a set of XML based tags to exchange the information needed to handle an emergency. To route, receive and respond to these messages, the responder anticipating an emergency duty related request must be identifiable by other collaborators that will need his services. Consequently, our development enhances the EDXL entities to ensure that the calling party is able to reach the *best called party* based on the *latter's availability*. To do so, we propose that all potential responders expose their *capabilities* and fallback options in case they cannot be reached during emergencies. These capabilities of responders include the role played in an organization, the tasks the actor can execute, estimated time to respond to a request (perhaps due to many emergency calls) or execute these tasks, available resources, direct contact information, and an alternative contact chain in case of unavailability of the best contact and the sensitivity of the information authorized to receive, and a contact to report complains about the quality of service, including contacting difficulties.

The rest of the paper is written as follows. Section 2 describes related work. Section 3 describes the linguistic abstractions proposed in EDXL and its messaging language. Section 4 proposes our enhancements to ensure the availability aspect of the actors. Finally, section 5 describes our concluding comments. Further details are available in a Technical Report at [11].

2 Related Works

In recent years, there have been a number of publications on building ontologies to solve different aspects of emergency handling. We discuss a few that are considered to be relevant to our work. Li et al. [5] proposes an ontology for crisis management. Although they defines a common set of vocabularies that can be used to facilitate an effective communication, they do not address failure scenarios in reaching key responders in a time of crisis.

Yu et al. [8] illustrates a good use of Activity-First Method (AFM) proposed by Mizoguchi [10] to construct an emergency ontology for creating a decision support system from existing emergency documents and use cases. This methodology is aimed at decomposing the emergency documents into data components for further integration based on emergent incidents. Although this emergency ontology helps decision makers sort out existing knowledge and reach critical decisions faster and more efficiently, it does not address how to ensure the availability of decision makers during an emergency.

Malizia et al. [6] constructs an emergency ontology for event notification and system accessibility. Using the knowledge that reflects users' needs, ways to present their needs, the nature of the emergency and available technologies makes it possible to reach more people. To build such a complex ontology, the authors use three domain concepts: accessibility, user profiles and devices and verification of the validity and integrity of knowledge by using first order logic. Although the proposed ontology may address the information needs for sharing and integrating emergency notification messages and provide the accessibility for different kinds of users under different conditions, it does not address the information needs for ensuring the responder's availability at the time of the need.

The open ontology approach [9] provides great flexibility to extend into a mission-oriented ontology. In order to do so, an open ontology provides multiple spaces and views that must be taken into account during the design phase. It also provides a theoretical approach to build such an ontology rather than providing a practical open ontology for emergency response. To the best of our knowledge, no one has extended this concept and developed it into a practical open ontology yet.

To facilitate sharing of information across all levels of government, the Federal Government has initiated the Universal Lexical Exchange (ULEX), which helps define the top sharable objects that can be formed into a coherent message that can be validated via the XML schema [17]. Although ULEX defines sharable contact information, the objective is to provide the contact information for deployable systems and services, and not the availability of the contact person during an emergency based on the person's job description. Universal Core (UCore) is another Federal information sharing initiative that supports the national information sharing strategy among all federal departments and agencies. UCore defines an implementable specification in XML schema that enables the information sharing of well-known and comprehensible concepts of *who, what, when and where* [18]. Although these concepts can address some aspects of information sharing for emergencies, they do not addressing how the contact would be used to locate the person during an emergency.

The US Federal Government has established a Government Emergency Telecommunications Service (GETS) program [14], which ensures a high probability of call establishment during a crisis when the PSTN is congested. This program provides a specific and recognizable phone number to obtain a higher priority for establishing a call. In recent years, with the increased prevalence of wireless phones, the Federal Government established a Wireless Priority Service (WPS) [15] program, where subscription information is used to identify high priority callers. However, both GETS and WPS services do not guarantee call establishment but rather provide best effort due to the network bandwidth availability. These services are considered complementary to our work on ensuring updated status is maintained regarding the availability of the responder or his alternate.

Many standards have been developed by OASIS that have been widely adapted in data communication for emergency handling. One of the recent standard releases is the Common Alerting Protocol (CAP) [12, 13], which is the primary communications protocol for exchanging emergency alert messages between different parties. CAP has been used, implemented and deployed by a number of agencies and firms [16]. In this paper, we enhance CAP by adding necessary elements into the CAP schema to enhance reaching the responders in an emergency. We also illustrate the use of these elements in a real life emergency scenario.

Last but not least is the EDXL language, which was developed by OASIS and became a standard in 2006 [1]. We strengthened the EDXL language by adding syntax that can be used to attempt to deliver messages to emergency personnel when the existing mechanisms fail.

3 EDXL

EDXL is a language designed for sharing information and exchanging data among local, state, tribal, national and non-governmental organizations to facilitate emergency response [1]. Figure 1, taken from Page 10 of [1], shows the entities used in creating the EDXL syntax in the form of an Entity Relationship (ER) model, where the entity in red is our enhancement that will be described in Section 4.

As Figure 1 shows, at the highest level, each EDXL distribution element (i.e. message) has six required attributes and six optional attributes. In addition, every message has a target area identifying a geographical region and a content object describing the incident, confidentiality levels and roles for the originator and consumer of the message.

Required attributes of the distribution element consist of a distribution ID, sender ID, date and time the message was sent, distribution status (consists of one of the four values: Actual, Exercise, System and Test), a distribution type consisting of value such as Report, Update, Request, Sensor Status, etc., and Combined Confidentiality having the most restrictive level of confidentiality sought for the combined payload.

The optional attributes consist of the language used in the message and (possibly multiple instances of) the sender's role, recipient role, keywords, distribution references (indicating distribution constraints) and possibly an explicit address for delivery. The explicit address is an XML schema.

EDXL messages can have four kinds of *optional* roles. They are sender's role, recipient's role, originators' role and consumers' role. These roles are supposed to be used for two purposes: (1) identifying potential recipients and (2) message distribution. In addition, explicit addresses can also be used for the latter task. The recommended usage syntax for the sender ID is *actor@domain-name* (such as dispatcher@example.gov) where the domain-name is guaranteed using the Internet Domain Name System.

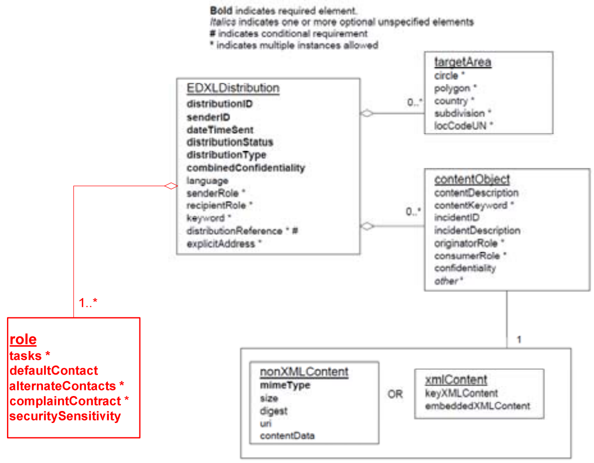

Figure 1: EDXL-DE Entity Relationship Diagram

4 Enhancing EDXL for Responder Availability

Before we explain our enhancements, several comments are worth mentioning. First, EDXL and CAP messages were designed for multiple purposes such as human-to-human, machine-to-human and machine-to-machine communications, etc., as shown by the fact that distribution element consists of optional fields such as *Sensor Status*, etc. For sensors, attributes like *roles* do not apply, but they do for human responders to emergencies. For example, we want to identify the Paramedic in an emergency response team (the role, but not the person) and his capabilities (such as is he authorized or trained to execute a certain type of medical routine like cardiac resuscitation, etc.?). Thus, for human responders, the role is more central than the recipient address, and the tasks that he is able to execute in that role. Therefore, in our enhancement the role is a mandatory attribute (marked by 1-* in Figure 1).

Because our objective is to enhance reaching the human responders with most suitable capabilities, we need to consider failure modes. One of the most important issues of recipient-address based emergency messages is that if that recipient is unreachable then it becomes the sender's responsibility to find the next available responder. Also any delivery system, such as an automated phone dispatcher, pager, SMS or email system should have an inbuilt mechanism to redirect the message automatically to the next appropriate responder. In order to facilitate this capability, either using an automated redirecting algorithm or in a sender initiated system, we propose creating a lexicon/ontology that has a list of alternative roles (where the role to person/phone number/IP address will be automated). In order to address the failure of these alternatives, we specify a *complain* role that should deliver the message to the higher authoritative personnel.

The redirecting algorithm can be easily implemented in the Private Branch Exchange (PBX) of the caller. [19] describes three common failure scenarios, Callee

Busy, Callee Unanswered or Global Errors. In all cases, when the call cannot be connected as dialed, the caller Sessions Initiation Protocol (SIP) gateway sends a *disconnect* message with the appropriate error code to the caller's PBX. Before this message is sent to the caller, we can inject the redirection mechanism by providing the PBX with a list of the default, the alternative and the complaint numbers, as will be shown in the algorithm depicted in Figure 2. For this to work properly, we made two assumptions. First, we assume that the local PBX has an Emergency Address book that is capable of translating the list of tasks to the local numbers based on their relevancy. Figure 7 illustrates an example with the <role> and <tasks> tags. Second, we assume that the order of relevancy can be selected by the local PBX. For example, in Louisiana, floods have more priority than earthquake. However, in California, the order must be reversed. This way, the selection algorithm can be regionalized, For now, we assume that our sorting algorithms addresses this based on its locality although we are working on separating these concerns. The PBX first makes a call to the *defaultContact*. If the PBX receives the Disconnect message from the local SIP gateway, the PBX will redirect the call to numbers on the alternative list. If there are no more alternatives, the PBX will redirect the call to the complaint number. Figure 2 depicts the pseudo-code for the algorithm that can run as an application at the PBX and make repeated attempts to facilitate availability of responders.

Roles and Tasks	Other Contacts	Contacts
Role: Emergency Gas technician **Tasks:** (1) Licensed to shut down main valves, (2) (dis)connect household lines, (3) Repair valves **zip codes** 22222, 22221	Email: emergency@gasexpert.com SMS: 7031111111 Response Window: 24 hrs/day Estimated Response Delay: 20 seconds	Default: 7031111111 Alternatives: 7032222222 7033333333 Complaint: 7039999999
Role: Emergency Gas technician **Tasks:** (1) Licensed to shut down main valves, (2) (dis)connect household lines, (3) Repair valves **zip codes** 22222, 22221, 22204, 22223	Email: emergency@gassol.com SMS: 7031110001 Response Window: 7AM to 10PM EDT, weekdays 9AM – 6PM EDT, weekends Estimated response Delay: 15 minutes	Default: 7031110001 Alternatives: 7031110002 7031110003 Complaint: 7031110005
Role: Emergency Gas technician **Tasks:** (1) Licensed to shut down main valves, (2) (dis)connect household lines, (3) Repair valves **zip codes** 22222, 22201, 22204, 222205	Email: emergency@gaspro.com SMS: 7032220001 Response window: 6AM – 11PM EDT, weekdays 8AM – 10PM EDT, weekends Estimated Response Delay 10 minutes	Default: 7032220001 Alternatives: 7032220002 7032220003 Complaint: 7032220009

Table 1: Key Words Translation

Figure 2 illustrates a pseudo code redirection algorithm at the local PBX. The *makeEmergencyCall* method accepts one parameter of the role node, which has been populated with the tasks that are relevant to the emergency. The *getTableFromRoleAttrs* method is then called to retrieve the table of contacts by searching the Emergency Address book for the contacts that are associated with the tasks. The table is then sorted based on time and the relevancy. The best matched entry in the table is then added to the role in three separate tags: defaultContact, alternateContact and complaintContact. The defaultContact is then called. If the disconnect is received from the local SIP gateway, each of alternateContacts is then called. If every call to the alternateContacts is failed, the complaintContact is called.

```
public int makeEmergencyCall (Node role)
begin
  table = getTableFromRoleAttrs(role.getTasks())
  role = sort(table, getCurrentTime(), role)
  defaultContact = getDefaultContact (role)
  returnCode = dial(defaultContact)
  if (returnCode == Disconnect)
  begin
    listAlts = getAltContact(role)
    while (listAlts is not empty)
    begin
      altContact= getNextAlt(listAlts)
      returnCode = dial(altContact)
      if (returnCode == OK)
        break
    end
    if (returnCode == Disconnect)
    begin
      compContact = getComplaintContact(role)
      returnCode = dial(compContact)
    end
  end
   return returnCode
end
```

Figure 2: Local PBX Redirection Algorithm

```
<role>
<tasks>
<valueListUrn>valueListURN</valueListUrn>
<task>value<task>
<estimatedTimeToFinish>time</estimatedTimeToFinish>
<currentResponseDelay>time</currentresponseDelay>
</tasks>
<defaultContact>contactURN</defaultContact>
<alternateContacts>
<valueListUrn>valueListURN</valueListUrn>
<contact>value</contact>
</alternateContacts>
<complaintContact>valueListURN</complaintURN>
<securitySensitivity>Security classification level
</securitySensitivity>
</role>
```

Figure 3: EDXL Enhancement

4.1 Ontological Enhancements or Roles

In the current EXDL-DE specification, a *mandatory* recipient role is given as a list of structures where each element is a potential recipient.

```
<recipientRole>
      <valueListUrn>valueListUrn</valueListUrn>
      <value>value</value>
<recipientRole>
```

Here the content of <valueListUrn> is the Uniform Resource Name of a published list of values and definitions, and the content of <value> is a string (which may represent a number) denoting the value itself. Multiple instances of the <value> may occur with a single <valueListUrn> within the <recipientRole> container. In addition, the <recipientRole> is *not* a required element. Our enhancements propose the following additions to a role as depicted in Figure 3.

5 Conclusion

We have taken a collection of standards for emergency management messages and proposed enhancements that would ensure that the messages are delivered to a set of recipients that are capable of responding to the needs at hand. Our proposal is based on a set of attributes that characterize the tasks that are needed of an external emergency handling entity. We have expressed these attributes by extending the proposed EDXL language. Our objective in doing so was to provide a 911 like pseudo name that is parameterized based on the organization, required responder's role and tasks he is expected to perform in order to satisfy the needs of the call. Our ongoing work addresses translating these pseudo names to addresses available on the telephone, email and pager services so that they can take advantage of PSTN based and wireless based priority calling services provided for specified actors of federal, state, local and tribal agencies.

Reference

1. Emergency Data Exchange Language (EDXL) Distribution Element, v. 1.0 OASIS Standard EDXL-DE v1.0, 1 May 2006.
2. Emergency Data Exchange Laguage Resource Messaging (EDXL-RM) 1.0 OASIS Standard incorporating Approved Errata 22 December 2009.
3. OASIS. www.oasis-open.org/committees/emergency.
4. Jacqueline Yang, Duminda Wijesekera and Sushil Jajodia, *Subject Switching Algorithms for Access Control in Federated Databases*, in the proceedings of the 15th Annual IFIP Conference on Database Security, 2002. Pages 61-74.
5. Xiang Li, Gang Liu, Anhong Ling, Jian Zhan, Ning An, Lian Li, and Yongzhong Sha, "Building a Practical Ontology for Emergency Response Systems," *Computer Science and Software Engineering, 2008 International Conference on*, 2008, pp. 222-225.
6. A. Malizia, T. Onorati, P. Diaz, I. Aedo, and F. Astorga-Paliza, "SEMA4A: An ontology for emergency notification systems accessibility," *Expert Systems with Applications*, vol. 37, Apr. 2010, pp. 3380-3391.
7. W. Xu and S. Zlatanova, "Ontologies for Disaster Management Response," *Geomatics Solutions for Disaster Management*, 2007.
8. Kai Yu, Qingquan Wang and Lili Rong; , "Emergency Ontology construction in emergency decision support system," *Service Operations and Logistics, and Informatics, 2008. IEEE/SOLI 2008. IEEE International Conference on* , vol.1, no., pp.801-805, 12-15 Oct. 2008.
9. P. Di Maio, "An Open Ontology for Open Source Emergency Response System", http://opensource.mit.edu/papers/TOWARDS_AN_OPEN_ONTOLOGY_FOR_ER.pdf
10. R.Mizoguchi, M.Ikeda, K. Seta, J.Vanwelkenhuysen, "Ontology for Modeling the World from Problem Solving Perspectives," Proc. of IJCAI-95 Workshop on Basic Ontological Issues in Knowledge Sharing,1995, pp.1-12
11. Technical Reports. http://cs.gmu.edu/~tr-admin/.
12. Common Alerting Protocol, v.1.1, OASIS Standard CAP-V1.1, October 2005.
13. Common Alerting Protocol Version 1.2, OASIS Standard, 01 July 2010.
14. Government Emergency Telecommunications Service. http://gets.ncs.gov/.
15. Wireless Priority Service. http://wps.ncs.gov/.
16. Common Alert Protocol (CAP). http://www.oasis-emergency.org/cap.
17. ULEX. http://www.lexs.gov/content/ulex.
18. Universal Core (UCore). https://ucore.gov/ucore/.
19. Call Flow Scenarios for Calls Failed. http://www.cisco.com/en/US/products/sw/iosswrel/ps1831/products_programming_reference_guide_chapter09186a0080087348.html.