Connecting the Dots: CAP and WSRP

Rex **Brooks** <rexb@starbourne.com>

Abstract

This paper explores the theme of XML 2004, "XML -- From Syntax to Solution." The 'dots' in this case feature the OASIS open, public standards, CAP 1.0, the Common Alerting Protocol, WSRP 1.0, Web Services for Remote Portlets, ebXML Registry and Repository standards, eXtensible Access Control Markup Language, XACML, 1.0 Specification Set and the Web Services Security 1.0 set of specifications.

These dots are 'connected' within the context of multiple, sequential and simultaneous emergency incident scenarios by using these open, public, xml-based standards together to show how the value of those standards is multiplied when used in such combinations.

Table of Contents

1. Introduction: XML Needs to Start Connecting the Dots

Connecting Dots to make a picture is an apt analogy for the purposes of this paper. It was chosen to provide a slightly different focus than usual since it came into a popular use in the aftermath of 9/11. News stories coverage of stories such as FBI middle management failing to pass the Phoenix memo up to top management revealed a critical inability to connect several contemporary domestic intelligence reports with closely related foreign intelligence reports. This connection was only seen as a clearly dangerous vulnerability. However, this report focuses on the emergence of potential tools which can help make these connections

We need to start using these tools: tools like open, public standards based on the interoperability that XML provides; tools that allow making these connections easier; tools that need to be adopted now to hasten making these connections and to hasten the process of improving these tools based on real world experience using them.

First, the Dots:

As one might expect in a conference dedicated to XML, the standards whose use is described in this paper, shown in this presentation and demonstrated in action, come from the Organization for the Advancement of Structured Information Standards, OASIS.

- CAP[CAP]: OASIS, Common Alerting Protocol, v. 1.0http://www.oasis-open.org/committees/download.php/6334/oasis-200402-cap-core-1.0.pdf [http://www.oasis-open.org/committees/download.php/6334/oasis-200402-cap-core-1.0.pdf]was approved as a Standard earlier this year, in April 2004. It was developed to provide a standard message format that could be used to "...exchange all-hazard emergency alerts and public warnings over all kinds of networks.[1]

- WSRP[WSRP], OASIS Web Services for Remote Portlets v. 1.0http://www.oasis-open.org/committees/download.php/3343/oasis-200304-wsrp-specification-1.0.pdf [http://www.oasis-open.org/committees/download.php/3343/oasis-200304-wsrp-specification-1.0.pdf]

  was approved as a Standard in August, 2003. It was developed to "...enable an application designer or administrator to pick from a rich choice of compliant remote content and application providers, and integrate them with just a few mouse clicks and no programming effort."[2]

- These two standards are the main focus of this paper, presentation and demonstration, showing how CAP messages can be used within a WSRP-compliant Public Service Portal to deliver different kinds of self-contained information units, Portlets, from different sources on the same page. However, the value of this main functionality is considerably enhanced and made more useful to the audiences served by employing some additional standards. These standards provide the means to establish connections, exchange information on portlets and agree on some policy issues prior to emergency use, making it possible to respond more quickly when needed. These are:

- OASIS ebXML Registry Services Specification v2.1[EBRS]http://www.oasis-open.org/committees/regrep/documents/2.1/specs/ebrs.pdf

  was approved in April 2002. It "...defines the interface to the ebXML Registry Services as well as interaction protocols, message definitions and XML schema."[3]

- OASIS/ebXML Registry Information Model v2.1[EBRIM] http://www.oasis-open.org/committees/regrep/documents/2.1/specs/ebrim_v2.1.pdf [http://www.oasis-open.org/committees/regrep/documents/2.0/specs/ebrim.pdf]

  was also approved in April, 2002. It "...specifies the information model for the ebXML Registry."[4]

- OASIS Security Assertions Markup Language, SAML, v. 1.1[SAML]http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security

- Assertions and Protocol

- Bindings and Profiles

- Security and Privacy Considerations

- Conformance Program Specification

- Glossary

- Assertion Schema

- Protocol Schema

[1]OASIS Emergency Management Technical Committee, "Common Alerting Protocol, v. 1.0," http://www.oasis-open.org/committees/download.php/6334/oasis-200402-cap-core-1.0.pdf
[2]OASIS Web Services for Remote Portlets Technical Committee, "Web Services for Remote Portlets,"http://www.oasis-open.org/committees/download.php/3343/oasis-200304-wsrp-specification-1.0.pdf
[3]OASIS ebXML Registry Technical Commitee, "ebXML Registry Services Specification v2.1" http://www.oasis-open.org/committees/regrep/documents/2.1/specs/ebrs.pdf
[4]OASIS ebXML Registry Technical Committee, ebXML Registry Information Model v2.1[EBRIM]http://www.oasis-open.org/committees/regrep/documents/2.1/specs/ebrim.pdf [http://www.oasis-open.org/committees/regrep/documents/2.0/specs/ebrim.pdf]

"...defines a framework for exchanging security information between online business partners."[5]

- OASIS Web Services Security 1.0 set of specifications[WSS]http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss
- Web Services Security: SOAP Message Security 1.0 (WS-Security 2004)http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf
- Web Services Security Username Token Profilehttp://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0.pdf
- Web Services Security X.509 Certificate Token Profilehttp://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0.pdf
- Schema file V1.0 [1]http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd
- Schema file V1.0 [2]http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd

was adopted as a standard in March, 2004. It "...describes enhancements to SOAP messaging to provide message integrity and confidentiality. The specified mechanisms can be used to accommodate a wide variety of security models and encryption technologies."[6]

- XACML 1.0 Specification Set[XACML]
- eXtensible Access Control Markup Language (XACML) Version 1.0 http://www.oasis-open.org/committees/download.php/2406/oasis-xacml-1.0.pdf
- Policy Schema http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml
- Context Schema http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml [http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml]

was approved as a standard in March 2003, It ",,,defines an XML schema for an extensible access-control policy language."[7]

---

[5]OASIS Security Services Technical Committee, "Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V1.1, Bindings and Profiles for the OASIS Security Assertion Markup Language 3 (SAML) V1.1, Security and Privacy Considerations for the OASIS Security Assertion Markup Language (SAML) V1.1, Security and Privacy Considerations for the OASIS Security Assertion Markup Language (SAML) V1.1 ,Conformance Program Specification for the OASIS Security Assertion Markup Language (SAML) V1, Glossary for the OASIS Security Assertion Markup Language (SAML) V1.1 , Assertion Schema - sstc-saml-schema-assertion-1.1.xsd, Protocol Schema - sstc-saml-schema-protocol-1.1.xsdhttp://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security

[6]OASIS Web Services Security Technical Committee, Web Services Security SOAP Message Security 1.0 [WSS]http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf

[7] OASIS eXtensible Access Control Markup Language Technical Cimmittee, "XACML 1.0 Specification Set,"http://www.oasis-open.org/committees/download.php/2406/oasis-xacml-1.0.pdf

While this appears to be a lot of Dots, it is really only five:

- CAP;
- WSRP;
- ebXML Registry;
- Web Services Security; and,
- eXtensible Access Control.

Web Services Description Language, WSDL, and Simple Object Access Protocol, SOAP, are taken for granted because they are relatively mature and this paper is focusing on the adoption of newer standards. This is also not an endorsement of ebXML Registry over UDDI as both registry standards have great value, and can be used as complements. Neither is this approach intended to tacitly approve all particulars of current standards

Second, Connecting:

This is the objective. Promoting early adoption of these recent standards, and using them in combination starts building our experience base more quickly. It is the nature of emergencies to cut across procedural channels and jurisdictional boundaries. Thus, we are prompted to encourage adoption of multiple standards in combination to improve performance in emergencies. This requires such standards to be integrated together as quickly and as efficiently as possible. It is a challenge, but one that has been accepted in building these standards in the first place.

CAP is not restricted to a single transport mechanism, nor is WSRP limited to a particular kind of web application or category of information. The pervasiveness of electronic media, and the web in particular, argues strongly that we ought to take advantage of our most widely adaptable media. .

Only by using these standards together can we quickly learn what needs improvement and refinement. Creating the Dots, building the standards is phase 1. Connecting the dots is phase 2. Learning from experience and refining both the dots and the practices of connecting them is phase 3. This paper, presentation and demonstration is devoted to phase 2 in the hope that we can move on to phase 3 sooner rather than later.

2. Reminder: Standards are Important

We tend to take standards for granted, seldom asking ourselves what a standard actually is and is meant to do. So, while we recognize the need for standards, such as the standard for screw types and sizes, e.g., what constitutes a wood screw or machine screw, such as the number of threads per unit of measurement, we don't think much about it. It is very important if we need to fix a screen door, or attach a hinge to a door frame, but we don't usually consider the process that has gone into establishing such a standard.

With newer standards for an area of human endeavor as abstract and complex as electronic Information Technology, we usually recognize the importance of such standards in the breach. That is, we notice it in its absence, when it inconveniences us. As a consequence, many standards efforts come into being for that reason. We need something that is missing, so we look to see where the work for supplying that lack is being done. We look to see if it is being done in a way that will satisfy our needs or at all, and only then do we think of taking on the work ourselves. This is not a lament, just an observation.

However, when there is a burgeoning of such efforts as there is now with respect to web services, it can seem both confusing and frustrating. A question heard fairly often asks, "If there are so many standards efforts, aren't many overlapping? So, if that's the case, isn't the practice self-defeating? If some of these standards are bound to conflict with or contradict other standards, doesn't that make the effort useless?"

As a matter of fact, there is fairly little overlap, though there is a long list of bewilderingly named web services standards specifications efforts. Certainly there will be some such overlap, but this requires more diligence rather than forsaking the effort.

Still, the points are well taken. We can't guarantee compatibility amongst these standards. It is wise, therefore, to remember that a standard, may have to compete in the public marketplace even if it has been approved through a rigorous process.

A standard is only as valid as its use. If it isn't used, it isn't much of a standard, while a de facto practice, regardless of whether or not it has been annointed by a standards body is more of a standard if it is used by a vast majority of users than the most painstakingly crafted standard ever written. The web browser wars showed us this.

---

because it appears to ignore existing issues with older standards. Nor is it meant to imply that the current configuration of internet protocols or web services architecture and infrastructure is set in stone, since we are in the midst of the transition to IPv6 and Internet 2.

This set of standards was chosen to use the value added by each to the overall purpose of helping deliver the correct kind of information to the first responder community in the midst of emergencies

So it is wise, also, to remember that standards tend to arise out of a set of negative contexts. We tend to seek standardization where a lack of standards proves inconvenient or harmful.

Therefore, we should, perhaps remember what standards are not. A standard is not a given document. A standard is not a filesystem. A standard is not a namespace. A standard is not a particular piece of legislation. A standard is only a standard if it is widely used. So, the following lists offers one set of criteria by which a potential standard can be thought likely to succeed:

- **It fills a real need.** This is the bottom line, and it often directly reflects a business bottomline. This is often not a question, since one is usually prompted to create a standard when trying to do something that isn't done the same way by anyone else with the same need. If one is not working on a standard due to such a need, it probably doesn't make much sense to be engaged in the activity.

- **It meets public requirements.** This means there is a public which can use it. If you write a specification for a public need, but you write it in a language that particular public can't read, it won't be used. Using the appropriate language is such a public requirement.

- **It prevents less expert organizations, such as government agencies, from creating the particular standard or standards**.

  Some standards can only be agreed to by governments, but there are many standards that government is not well-suited to creating, such as those we are examining.

Standards work is also very demanding. It requires patience and more than a small dollop of good luck to succeed. Often, this kind of luck is the luck of the draw. The element of group chemistry plays a very large role in the probability of success for any standard-writing initiative.

3. Web Services from Hyperbole to Consensus

In August of 2001 a 'call for participation' announcement went out to the general membership of OASIS for a new Technical Committee, TC. This early stage of discussions was initiated by OASIS members led by IBM and tentatively named "Web Services Component Model." This was in the period when Web Services were being hyped as the "Next Big Thing." This context is important because it points out some of the foibles inherent in the process of developing new web technologies following the implosion of the so-called "dot-com bubble."

At its first meeting in January 2002, the newly formed Technical Committee changed its name to **Web Services for Interactive Applications, WSIA**, in order to establish clearly that the focus of the group was to supply the "last mile" of delivering aggregatable units of web content from multiple sources on the same web page. This objective gradually evolved into a general-purpose description as "user-facing" web services. It also became clear in that formative week of meetings at IBM's Watson Research Center north of New York City, that these self-contained information content fragments would be hosted in Portals, for the most part, as the vehicle for delivering this rather vaguely defined category of so-called web services.

This dovetailed well with a second TC that was also in the planning stages at that time, albeit with a far more structured approach and focus, the **Web Services for Remote Portlets, WSRP**, TC. As the passing year showed, the intersection between these two TCs, the "Portlet," would become the chief defining feature, with an initial commitment to providing for "user-facing" interaction capability.

Growing out of a joint interfaces subcommittee between the two groups, the overall group consolidated under the WSRP name. The WSIA TC dissolved itself following the approval of WSRP v1.0 by a vote of OASIS member companies in August 2003, about a year and a half later.

A logical consequence of this work was tackled in those first meetings; This consequence led to developing the working concepts of a **WSRP Producer role** for supplying portlets, a **WSRP Consumer role** for aggregating and presenting some selection of portlets from one or more producers and an **End-user role** accessing the Consumer.

Additionally, in order to maximize the audience and begin the work of bringing standards into greater alignment across platforms, programming languages and transport protocols, it was also required from the start of this effort to make **WSRP** and the Java Portlet Specification, **JSR 168**, interoperable or easily convertible.

Reaching Consensus

This bit of history is instructive for a couple of reasons that highlight the main point of this paper, presentation and demonstration:

- **Flexibility** in the conception of standards-writing efforts is essential. Early formulations can, and often do, evolve as more participants join an effort, and new possibilities for applying a technology under development arise that merit further exploration.

- **Business Scenarios to Basic Use-Cases to Formal Requirements** describes a formula that is evolving as a common, 'best' practice in this work. The standard-writing group starts with basic business scenarios that model the service specification by determining requirements and forming a checklist against which to gauge fulfillment of building the specification. For WSRP, one primary use-case was a somewhat generic end-user's personalized page which would contain that end-user's local weather portlet alongside that user's stock portfolio portlet. Additionally, the stock portlet needed to be capable of the simple interaction of adding a new stock symbol to that user's usual portal page and the weather portlet needed to be capable of delivering the weather for a new geographic location as required..

  This formula also produced a corollary. Although more complex scenarios may be required, it is best to start with the most simple and basic scenarios and use-cases to specify the fundamental principles at work, and then move on to the more complex processes. In this case, the simplest use-cases were 'stateless.' That is, though it was given that statefulness would be required, we began at the most basic level.

  Before specifying how to change state and keep track of the information about state changes, we needed to establish the process of providing the stateless portlet first, and only then explore what was needed to accomplish state changes reliably and efficiently.

Producing the **Common Alerting Protocol, CAP**, through OASIS was a much different task, so similarities in the process reinforce the value of adopting best practices for those similarities. Following the horrific events of 9/11, the Partnership for Public Warning, PPW, a non-profit, public-private organization was formed http://www.partnershipforpublicwarning.org/ppw/

PPW began work on the Common Alerting Protocol in 2001 following the events of 9/11, then, in February of 2003, brought that work into OASIS at the founding of the **OASIS Emergency Management Technical Committee**.

However, though the process had been somewhat different than the norm for OASIS, it was still determined that developing a Requirements document in the TC , was valuable. So, the needs underpinning the CAP effort became a fundamental part of the process, even though the work on the standard had been underway long enough for it to have been reasonable to dispense with that step.

Some significant differences between CAP and WSRP include the fact that CAP is aimed at a much wider variety of communication media, including the Emergency Alert System , EAS, the broadcast media of television and radio, as well automated telephone emergency networks. CAP, therefore, is scoped for broad use rather than transport-specific channels. However, CAP is also framed using XML, and is meant to be flexible enough to allow for use via the same web services framework as WSRP.

By and large it is easier to consider CAP a message format, whether that message is transmitted via EAS in a broadcast television context or it is contained in a signed and encrypted SOAP message payload.

CAP was approved by OASIS as a standard in April 2004.

4. CAP and WSRP: A Service-Oriented Architecture Spanning Public-Private Boundaries

The collaboration underlying this demonstration of using CAP within WSRP makes a statement. Traditional boundaries between the public and private sectors are being bridged. It can be expected that further resistance or competition will be encountered from entrenched interests promoting proprietary solutions in spite of the drawbacks of dependence. However, as we move into new territory based on the use of open standards that span the separation of governmental agencies from the citizenry at large, we should remember that a key point is that this bridging improves our ability to respond effectively to emergencies of all kinds.

In addition to spanning the public and private sectors, this demonstration also shows a practical example of a **Service-Oriented Architecture**, SOA., This is a term that can be better understood through an example and a graphic diagram than through a simplified explanation of a complex technical discussion. It should be noted, therefore, that the concept of SOA is greatly simplified here.

For our purposes, we can think of SOA as a connecting structure, like a bridge, or channel or pipe that allows for reusable functional IT service components to be implemented across divisions within an organization. In this sense it owes this concept of a common structure for using information to the "Framework for Enterprise Architecture developed by Zachman, et al.[EA]

Our example showcases WSRP-compliant web services in the form of specialized portlets. In our cases, selecting a particular CAP incident triggers the portlets to display information related to that incident. Of course, in other contexts, WSRP-compliant portlets can share applications as well as data and the presentation of that data among departmental or divisional islands, such as Product Line Management, Call Center Operations, Human Resources and Financial Services within an organization or enterprise.

A Service-Oriented Architecture is a plan for connecting the infrastructure of hardware and software components across an enterprise or organization through reusable service components.

Specifically, "SOA is an architectural style whose goal is to achieve loose coupling among interacting software agents. A service is a unit of work done by a service provider to achieve desired end results for a service consumer. Both provider and consumer are roles played by software agents on behalf of their owners."[8]

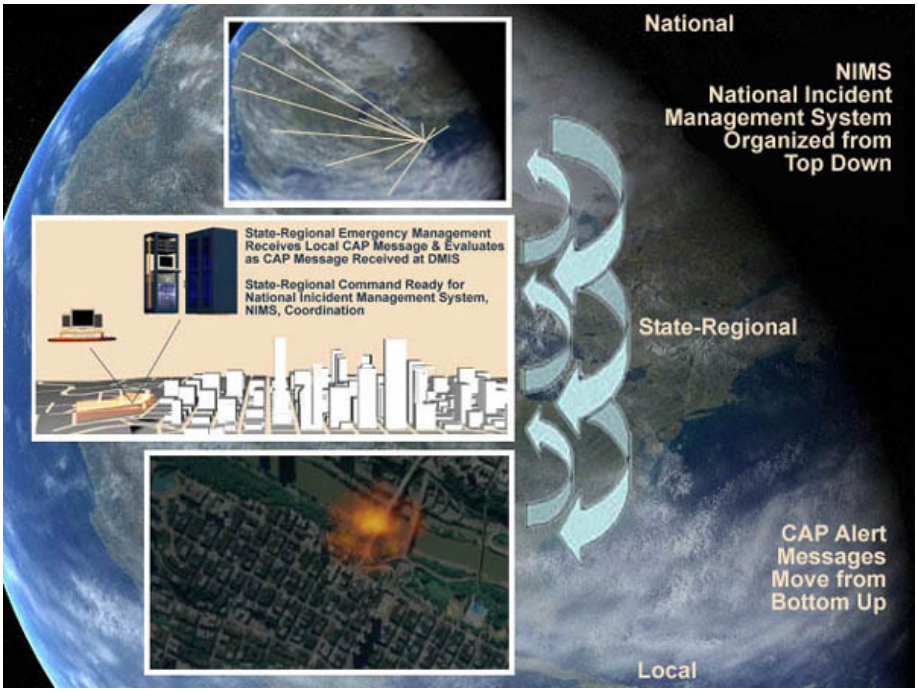
The follow graphic diagram shows this loosely coupled structure.

[8]Dr. Hao He,, "What is Service-Oriented Architecture?"http://webservices.xml.com/pub/a/ws/2003/09/30/soa.html

One particular set of structural principles makes this architecture noteworthy. For CAP, the way in which messages are generated and distributed can best be described as bottom-up. An emergency incident occurs locally, for the most part. It is the end point at which the chain of distribution of a CAP message starts, such as an oil refinery, a county fire station, a municipal police department, etc. So, an incident message comes into the system with the most complete local information included due to the proximity between the reporting agency and the incident itself. Such a bottom-up structure tends to more accurately reflect both the incident and the local agencies on scene, allowing the system to respond more quickly to the incident.

At the same time, the web-based system that governs the national, regional, statewide and local distribution of CAP messages is organized in a federated model, like the overall structure of the United States Government, from the top-down, reflecting the chain of command and responsibility within the first response community. Bear in mind that the organizational procedures are still under development, so the sequence described is best viewed as this observer's opinion of the most likely sequence that will develop with time and practice into actual policy

The following graphic offers a simplifed view of how this system is designed to work at this time.



Once a valid CAP alert message has been generated and security authenticated, there is or will be a threshold for urgency, severity and certainty and, perhaps, a threshold for size or population density of the impacted incident area. The CAP message will then be propagated to the Disaster Management Interoperability Services organization of FEMA as recommended or required by the National Incidencent Management System, NIMS, developed and adopted by the Department of Homeland Security where the current "official" CAP message list is or will be kept. The word official is put in quotes here because this is conjecture at this point. Whether, in fact, there will be some official authorization of the current set of such CAP alert messages that appear on the list returned from the DMIS server has not yet been determined.

Likewise, the exact methodology of push versus, or in addition to, pull will then be available for a trusted network of authenticated and authorized agencies to poll. The term poll here simply means retrieving the current list at any given time. Based on accordance with the NIMS policy guidelines in effect, messages will be accessed and responses, beyond those local responses already underway, will be triggered.

The following diagram shows a simplifed depiction of this basic architecture. Notably important concepts at work here show that DMIS Infrastructure and the disparate Incident Command Systems become a connected service architecture, the key aspect of SOA. By working to a common interface implementing CAP, the benefit of a limited vocabulary and restricted message structure becomes apparent because it allows a focused range of operations that can be shared by heterogenous systems.

Demonstration

Our demonstration starts at this point in the process. A WSRP-compliant Portal polls the DMIS CAP server to retrieve the current list of CAP alert messages.

First we look at this Portal, a Public Service Portal, hosted by Humanmarkup.org, Inc., the 501 (c)(3) Non-Profit Corporation established to provide assistance for the effort to create the Human Markup Language in the OASIS Human-Markup Technical Committee. Such a portal can serve the public as well as the first responder community, and can serve as a convenient cyberspace station where a range of follow-on web services that provide information and applications related to CAP alert messages can be accessed.

Public Service WSRP Portal

This portal takes a step beyond the proof-of-concept stage, serving as an example that is capable of further implementation to create an ongoing public service resource and serve as a testbed for moving more fully into the operational, or full production phase of adopting the standards demonstrated with the functional extensions supported by many portal vendors, such as collaboration environments, project and content management and customized productivity applications.

For our purposes, we hypothesize a set of scenarios involving various types of emergencies. Some occur in sequences. Some are simultaneous. The object is to simulate the kinds of coincidences that we think are likely based on our experience.

We wish to express our thanks to Plumtree Software, Inc., for their donation of the Plumtree Corporate Portal toolkit and their assistance in developing this example Portal as well as their donation of associated components of the Plumtree Enterprise Web Development Kit, (EDK), used to build the WSRP-compliant Public Service Portal.http://plumtree.com

WSRP-compliant Portlets

However, a portal is only as valuable or effective as the portlets it hosts for the end user audience.

Accordingly, we also wish to express our gratitude to Oracle Corporation for working with us to design, develop and produce the portlets and the backend databases deployed in this demonstration using the Oracle Portal Developer Kit.http://www.oracle.com/technology/products/ias/portal/index.html

Likewise, we would like to thank all those who have helped develop and implement the range of technologies demonstrated. The key point we make in using components from a variety of providers created in a variety of develop-

ment products on a variety of platforms is that by using open, public standards in such a set of divergent resources, a heterogenous web environment can be supported. The advantages are numerous in using such an approach, but the interoperability of terminology and data and the cost savings allowed by the ability to use products from different vendors in the same environment together rank highest.

The WSRP-compliant Public Service Portal aggregates portlets from different producers or separate portlets from the same producer on the same portal page. Portlets can contain a range of content from applications such as email clients to condensed fragments of a single web page, exemplified by the classic examples of weather and stock portlets.

Future versions of WSRP and its close cousin, JSR168 in the J2EE realm, will allow greater flexibility, personalization and coordination between portlets including greater incorporation of applications within WSRP-compliant Portals and Portlets, as the standard evolves from the relatively simple set of properties accommodated today, to the more complex possibilities envisioned for the future.

For now, we can provide some very basic, but fairly impressive resources in the **Emergency Medical Symptomology and Treatment Portlet** and the GIS-based **Common Operational Environment Portlet**. However, this work revealed some areas where improvements are needed in the overall and specialized IT domains explored for this project.

Emergency Medical Symptomology and Treatment Portlet

This portlet depends on a backend database configured to search for medical resources for injuries, illnesses and conditions based on the type of emergency that the CAP message describes in relation to the list of basic emergency types:

- Chemical(C) Agent
- Nerve(N) Agent
- Radiological(R) Agent
- Aviation (Airway) Transportation Incident
- Motor Vehicle (Roadway)\ Transportation Incident
- Marine (Waterway) Transportation Incident
- Fire
- Flood/Storm
- Earthquake

It would be incorrect to say that the field of medical informatics was poorly organized since, until recent efforts epitomized by the Stanford Medical Informatics Group in the development of its ontological processing tool, Protégé, this field simply wasn't organized in any one place as a whole rather than a collection of specialized parts. The Unified Medical Language System of the National Library of Medicine of the National Institutes of Health serves as a useful starting point, but one must obtain a license to use the resources and these resources are not organized to be useful in fields related to but not engaged in medical practice, such as insurance, law enforcement, forensic anthropology, etc. However, as it turns out, this lack of an overall medical or healthcare registry and repository system can be a positive circumstance since a preponderance of these resources fall into those special interest categories under which those resources were conceived and produced. This makes classification and categorization a less difficult and complex task.

Of course, Medical Education and Research, particularly in Pharmaceuticals and high-profile technologies such as CAT, PET and MRI scanning, were well organized, and are, therefore, organized within those narrow purposes.

To a surprising extent, the search for medical resources by types of injury or condition still provides an exercise in frustration. However, the Stanford Medical Informatics Group at Stanford University continues developing a promising

ontological processing tool,Protégé. In this tool and in the development of the Unified Medical Language System, UMLS, this lack of organization is being corrected. In fact, RDF, the Resource Description Language, close sibling of XML, is used in the newly approved W3C Web Ontology Language, OWL. Protégé writes ontologies in OWL using RDF which can then be used to produce viable application code.

This, in turn, allows for programmatically connecting appropriate resources to applications. In an irony, this shortcoming in medical informatics required the very improvement which makes it one of the first domains of IT to start integrating with the SemanticWeb.

Geospatial Information Systems (GIS) Provides Basis for Common Operational Environment (COE) Portlet

In a similar vein, integrating the recently developed Geospatial Markup Language, GML, http://www.opengeospatial.org/specs/?page=specs [http://www.opengeospatial.org/specs/?page=specs]created by the Open Geospatial Consortium,http://www.opengeospatial.org/ [http://www.opengeospatial.org/]further demonstrates employing open standards where feasible.

Taking this a step further, we use the current version of the Federal Geographic Data Committee Homeland Security Working Group Symbology Reference, which is still under development and review, http://www.fgdc.gov/HSWG/ [http://www.fgdc.gov/HSWG/]

The concept of a "Common Operational Environment" is not new, nor is the term itself, since the Department of Defense uses a very similar concept it terms a "Common Operating Environment." Our COE refers to a customizable set of maps centered on the geocode used in a CAP message in the "area" element. Unless you have an interest in the exact usages of various governmental agencies related to GIS information, all you need to know is that an end user of the portlet can switch among any combination of various common views available to the portlet such as local, regional, political, topographical, etc. The emergency symbols mentioned above can serve as links to indexed information such as locations for hospitals with trauma treatment facilities, main sewer valve locations and what current traffic conditions need to be avoided, particularly while first responders are en route to an incident scene.

A host of other kinds of information can be conveniently indexed by map symbols. The methodology for this kind of organizing principle is gazeteering. It can be found in such divergent disciplines as archeology, anthropology, and oceanography. Gazeteering illustrates another of the main points that needs to be made in regard to using open, public standards, particularly in combinations, discovering new uses for methods and techniques.

When multiple standards are considered for use together, unexpected uses for techniques typically associated with the disciplines involved in the standards but not typically used together can be suggested. Thus we discover similar uses for geospatial information systems among archeologists, emergency managers, meterologists and battlefield tacticians.

Likewise, CAP serves multiple functions in our demonstrations.

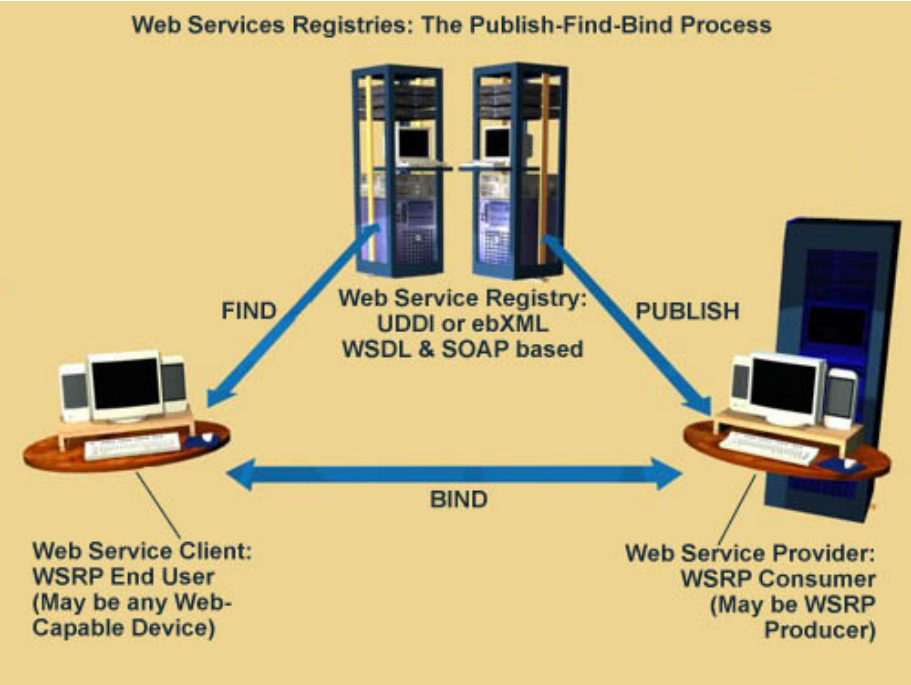
In addition to serving its warning function as an alert message, CAP can serve as the gateway to specific kinds of information for specific kinds of first-responder communities in Public Health and Safety. There are multiple levels of security possible with WSRP-compliant portals. Access control is one such security provision we demonstrate through a Registration process.

5. Publish, Find, Bind: Registries Provide Vital Link

Before the value of open standards began to be recognized in the general acceptance of XML, there were industry-based programs such as EDI which had already delivered substantial ROI through adoption of common data interchange systems. So it was a logical extension of open standards based on XML to employ the principles of metadata in standards aimed to create standard data interchange through registries where organizations of all kinds can locate other organizations that can fill their needs or to whom they can supply goods and services. With open, public registry standards using XML, such as the Universal Description, Discovery and Integration Specification (UDDI) and Elec-

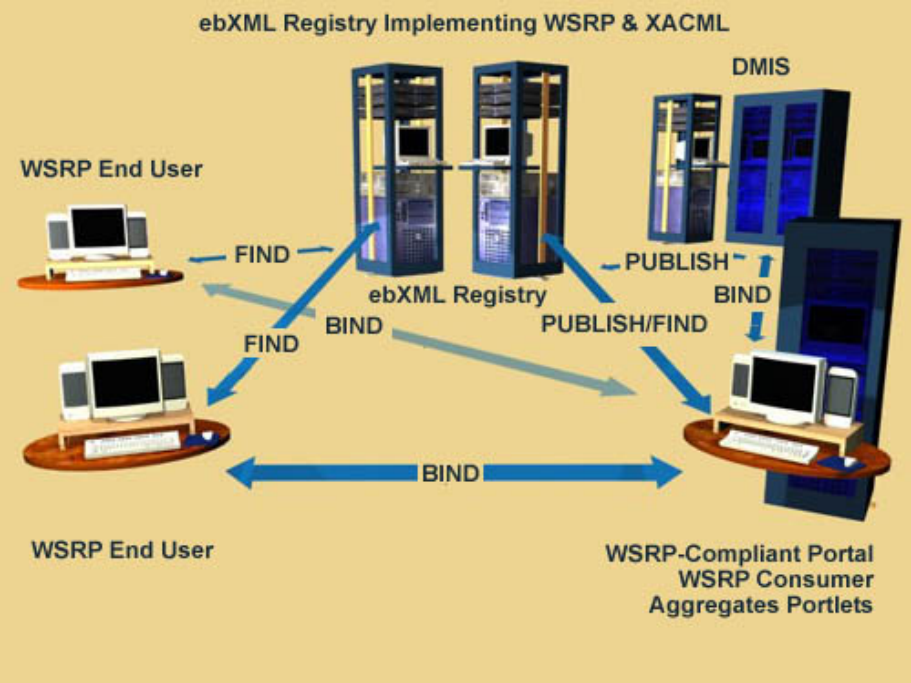
tronic Business XML (ebXML) Registry Information Model and Repository specifications, organizations can do much more than simply locating compatible resources.

The basic model for this process is shown in the following graphic diagram.



For this project the NIST ebXML Registry Pilot Program was chosen to demonstrate the WSRP Registration process and also to show one way in which to implement the eXtensible Access Control Markup Language, (XACML) in order to control access to the Common Operational Environment Portlet and the Medical Symptomology and Treatment Portlet.

For our purposes we chose to use DMIS as the logical point in the chain where registration information could be required based on XACML and provide this kind of security. The following graphic diagram illustrates this.



Because these portlets may contain sensitive information, as well as operational details of how the emergency incident is being handled as well as the most current updates, it would not be appropriate to allow the general public to access those portlets at the same level as authorized first responders and emergency managers, if at all. For the general public it would be appropriate to present basic information correlated to the instructions in a CAP message, such as instructions to evacuate or shelter in place.

There are subspecifications, known as Technical Notes which are recommendations for using UDDI and ebXML registries with WSRP. Current drafts are available for each on the OASIS WSRP TC document repository. http://www.oasis-open.org/apps/org/workgroup/wsrp/documents.php

6. Our Scenarios

The set of scenarios used in our demonstration was developed over a period of months amongst a group of collaborators and informal consultants whose composition changed over the course of developing this presentation, so a great deal of thought from diverse viewpoints has been given to it. These scenarios are included in this paper to illustrate what is meant by the practice of developing scenarios which can then be boiled down into specific use-cases. This set has been winnowed and refined by several sets of reviewers, keeping only those scenarios which passed the criterion of believability by more than one individual. However, as the events of 9/11 proved beyond a shadow of doubt, we all can be surprised.

Our set of scenarios follows events in a week in September. Since the presentation covers this in detail and the demonstration shows sample views into CAP and WSRP in processing and functionality in the course of these scenarios, only the purposes and conclusions are discussed here.

The single, ongoing scenario that ties the week-long series of events together is the **Hurricane Scenario** that works its unpredictable way up the east coast of the United States, bringing various kinds of emergency incidents with it. This scenario demonstrates the most common, but by no means least damaging type of emergency, a severe storm emergency, among the various types we examine. This scenario will engender a more or less continuous set of emergency incidents over the course of the week with flooding, wind damage and coastal storm surge damage.

In the conclusion of the week's events, the hurricane scenario provides a complicating factor and points up the need to prepare for a concatentation of emergency incidents which compound the stress on any emergency response system.

Early in the week, a remote sensor will automatically trigger a local CAP alert in a county in northern California This **(unconnected) Oil Refinery Scenario** highlights detection and warning systems that are in place in the county examined but lacking throughout most of the counties in the United States that have similar facilities, such as chemical plants, within their jurisdictions.

This scenario also serves to highlight the most common kind of association between emergency incidents which trigger CAP alerts, namely no association because these are unconnected incidents. However, no association does not equate to no effect. Concatenation of emergencies is the most common effect even as no association is the most common type of association. Evaluating associations among concurrent and sequential incidents needs to be addressed. This scenario, therefore, begins the process of providing a use-case for CAP v2.0.

Shortly after the Oil Refinery Scenario, a second incident occurs in a Southern California county. In this **(initial) Intercepted Terrorist Scenario** a routine traffic stop interrupts the execution of a terrorist plan to launch a catastrophic suicide car bombing of Liquified Natural Gas terminal facilities in a major Pacific Rim Port. This scenario results in a car bombing that fails to detonate a major catastrophe.

Meanwhile, the hurricane will spawn a **Hurricane-Connected Tornado Scenario** in a coastal county in Virginia. This specific incident highlights how such a storm produces sequential emergencies that force triage decisions on how to deploy resources and assets.

The culminating **(connected) Coordinated Terrorist Scenario** occurs as the now-downgraded hurricane reaches the vicinity of a Liquified Natural Gas Terminal in New Jersey Port. While the LA County attack was conducted by a car bomb, this one is delivered by boat. This attack is aimed at a tanker, rather than the onshore storage facilities. The demonstration will climax with this incident.

Events are unlikely to build as a crescendo such as this demonstration does, but the coordination that was exhibited in the 9/11 attacks argues strongly that we need to be prepared to respond to worst possible scenarios. Hence, the choice demonstrating CAP-based WSRP implementations capable of adding value to the first responder community and to the public at large.

7. Summary

Standards are meant to be used together, and multiply value when used in concert. While the web services arena is currently burgeoning with proposed new standards, and some of them may seem frivolous, it is well worth the effort to master this arena. Debateably, end-to-end integration of IT within a portal based structure may prove inevitable.

While it may be that large, private, highly secured networks will be the vehicle of choice for large organizations, the web, with increasing emphasis on improving security practices as well as adopting the appropriate security standards for web services, will serve the integration needs for the vast majority of businesses and non-business organizations.

Therefore, a modular flexible approach to working within the existing web stacks is needed. Allowance should also be made for incorporating components from different vendors based on different strengths. This will also serve to encourage vendors to emphasize interoperability over proprietary solutions to avoid losing market share. A modular approach to building appropriately scaled solutions for specific needs will save both time and costs as well as optimizing ROI.

Promoting adoption of solutions based on open standards is necessary to improve these standards and shorten the time to further adoption across the marketplace.

Early adoption also provides developer feedback needed to improve successive versions of standards, such as CAP and WSRP, based on real world experience. Of course, early versions of software and standards share a common

shortcoming. They almost always have bugs, or simplye lack features that prove to be needed. However, by actively participating in the development of open standards, and learning early what is needed, vendors and implementers gain valuable experience which in turn allows them to integrate successive versions of standards more quickly and more effectively.

We have also learned in the course of recent history, that budgeting for the long term is difficult when short term realities require postponing equipment and software purchases, regardless of the impact on plans to upgrade or redesign the structure of an organization's overall IT systems. This is actually another very good reason to modularize IT architecture development while proactively evaluating migration paths to organization-wide IT integration. Waiting will cost more in lost market share to competition willing to harvest open standards early. While that lost time is not easily equated to lost ROI, that loss is real nevertheless.

Lastly, the ability to automate planning, tracking and collaborating across the enterprise by using portals and portlets in applications implementing standards such as CAP, WSRP, ebXML Registry, XACML and WSS, can yield unexpected positive results which we are only now beginning to see. It is an exciting time to be involved in this effort.

Acknowledgements

The author would like to acknowledge the gracious advice, contributions, corrections and comments by the following individuals, in no particular order or priority: Russell Ruggiero, IT Analyst, frequent co-author of IT journal articles; Rich Thompson, IBM, chair of OASIS WSRP TC; R. Allen Wyke, IT Author, original chair of OASIS Emergency Management (EM) T; Art Botterell, co-chair OASIS EM Messages and Notification Subcommittee (SC), Author/Editor of CAP 1.0; Ranjeeth Kumar Thunga, co-chair OASIS HumanMarkup TC, co-founder Humanmarkup.org, Inc; Ross Fubini, Plumtree Software, Inc, OASIS WSRP TC Member; Mohamad Afshar, Oracle Corp; Carrie-Anne Michos, Oracle Corp; Ali Niazi, Oracle Corp; OASIS Administration; OASIS WSRP TC Members past and present; OASIS Emergency Management TC Members past and present; OASIS HumanMarkup TC Members; Gary Ham, Disaster Management Interoperability Systems, FEMA, DHS; Neil Bourgeois, Disaster Management Interoperability Systems, FEMA. DHS; Carl Reed, Open Geospatial Consortium, chair OASIS EM Geospatial Information Systems SC; Christopher Lakey, Image Matters, LLC; Roy Morgan, NIST, ebXML Registry Program; Duane Nickull, Adobe Systems, Inc, ebXML Registry TC Member; Tony Pizi, Level 8 Systems, Inc; John Sarazen, Level 8 Systems, Inc; Roland Pen, Level 8 Systems, Inc; Mark A. Musen, MD, Ph.D., Head, Stanford Medical Informatics

Bibliography

[CAP] *Common Alerting Protocol 1.0.*
[http://www.oasis-open.org/committees/download.php/6334/oasis-200402-cap-core-1.0.pdf]

[WSRP] *Web Services for Remote Portlets 1.0*
[http://www.oasis-open.org/committees/download.php/3343/oasis-200304-wsrp-specification-1.0.pdf]

[EBRS] *ebXML Registry Services 2.1 [http://www.oasis-open.org/committees/regrep/documents/2.1/specs/ebrs.pdf]*

[EBRIM] *ebXML Registry Information Model 2.1*
[http://www.oasis-open.org/committees/regrep/documents/2.1/specs/ebrim_v2.1.pdf]

[SAML] *Security Assertions Markup Language 1.1*
[http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security]

[WSS] *Web Services Security 1.03 [http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss]*

[XACML] *eXtensible Access Control Markup Language 1.0*
[http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml]

[EA] EA *Concepts of Framework for Enterprise Architecture,*
*[http://members.ozemail.com.au/~visible/papers/zachman3.htm#Article]*John Zachman,

[SOA] *What is Service-Oriented Architecture?, [http://webservices.xml.com/pub/a/ws/2003/09/30/soa.html]*Dr. Hao He Ph.D.

Biography

Rex **Brooks**
Executive Director
Humanmarkup.org, Inc.
Berkeley
California
United States of America
rexb@starbourne.com