

Challenges in Building Scalable Network Centric Real-Time Information Dissemination Systems

Scott Arthur Moody
Boeing Phantom Works
Network Centric Operations
scott.a.moody@boeing.com

Abstract

As the DoD strives for a new Network Centric Warfare (NCW[1]) capability, a multitude of technical and organizational challenges are being faced. In large part, NCW can be classified as a collective large-scale information dissemination system that operates at multitudes of levels and will be implemented using different distributed technologies and architectural styles. This paper classifies the use of these different distributed technologies as layers of information abstractions and highlights their varied architectural uses as they are driven by latency, scalability, re-configuration and seamless programming requirements. For example, the following technologies are contrasted: Peer to Peer and mobile networks, Web Services, Content-based Routing, and Real-Time CORBA, while addressing distributed Security and Fault Tolerance issues. Unique challenges for when and how these technologies can be used, as well as their attributes for addressing scalability are highlighted. These technology challenges are classified and contrasted using the models based on the Siena project[2] which has been extended for the real-time community and termed Siena+RT. These requirements are described in context of the Boeing Phantom Works[8] XIS (eXtensible InfoSphere) Scalability Testbed Framework. XIS is a set of hierarchal distributed technologies developed to support the diverse real-time requirements for collection and dissemination of information throughout a global information grid.

1 INTRODUCTION

This paper discusses the problem domain of scalable information dissemination systems as it relates to the complex requirements of the future warfighters network centric capabilities. In particular, this domain requires a high degree of security, many levels of real-time attributes, while dealing with some clients that are highly mobile and have challenging high latency communication systems.

This paper is organized as follows: Section 2 discusses the NCW domain and unique requirements dealing with the scale of the envisioned deployed systems. This highlights the de-coupled nature of information producers and information consumers, while highlighting some unique management issues that

have traditionally been stumbling blocks (including distributed security and the ability to have pro-active information controls).

This is followed in Section 3 by an introduction of a modified Siena[2] classification scheme for event and channel based infrastructures, which has been extended with new dimensions to meet the NCW requirements for security, real-time and mobility. Using the modified Siena scheme, a number of existing and proposed technologies and strategies are then discussed as they fit the scalability characteristics. In general, as the layers of knowledge are made more abstract, or further from the raw information source, the information becomes available to a wider audience, while the information takes longer and is harder to direct to the best consumers. Each of the technologies classified is broadly defined in relationship to the real-time and security requirements, while also describing the best use for these technologies and specific challenges that must be solved for the most transparent use. For example, a channel based solution requires an ability to find the channel and the physical connection ports - while a content solution shifts the focus to describing the information and the interest profiles.

To help provide context for these technologies in a real system, Section 4 discusses Boeing's XIS (eXtensible InfoSphere) Scalability Testbed Framework, showing what has been developed and where envisioned new technologies will be integrated. Some of the issues described during the classification process are shown in the XIS testbed. Specifically, the following technologies are shown in the context of XIS: Peer to Peer networks, Mobile clients, Web Services, Content-based Routing, Real-Time CORBA, Distributed Security and Fault Tolerance issues. Section 4.2 highlights the issues involved in distributed security for these systems, and discusses the XIS solution characteristics.

As there are still many new research projects on-going in the network-centric field, Section 5 classifies a number of these and labels them as challenges for the deployment (and development) of future systems. For example, there are many different technologies that are competing for the ability to define rules that can help map information to desired recipients, while also managing secure access to the information. These are

contrasted in terms of scalability and maturity requirements. Content Based Routing is shown in section 6 to be a promising new scalability technology.

2 Network Centric Warfighting (NCW)

Network Centric Warfighting (NCW)[1] is a new classification of large scale applications applied to the very complex nature of the military. The following quote describes the fundamental shift from a platform centric warfare capability:

“NCW is an information superiority-enabled concept of operations that generates increased combat power by networking sensors, decision makers, and shooters to achieve shared awareness, increased speed of command, higher tempo of operations, greater lethality, increased survivability, and a degree of self synchronization. In essence, NCW translates information superiority into combat power by effectively linking knowledgeable entities in the battlespace.” [1]

In typical platform centric capabilities, the concerns of the problem space are limited to smaller well-defined domains. As systems transition to the large scale and need to exchange information with the entire “network”, a new paradigm is needed. Two main goals of NCW are (1) to provide for the greater interchange of information among the heterogeneous sources, and (2) to decrease the development and maintenance issues by thinking outside of the individual platforms and to structure the information exchange in an extensible manner. These two goals have consequences on different uses of technologies described in this paper.

One important component of NCW deals with an Air Force concept called the Joint Battlespace Infosphere (JBI)[4]. The JBI is a combat information management system that provides individual users with the specific information required for their functional responsibilities during crisis or conflict. This specifically deals with finding a way to get the right information to the right people at the right time. To achieve this, the JBI (and by extension NCW) will integrate data from a wide variety of sources, aggregating that information, and distributing the information in the appropriate form and level of detail to users at all echelons.

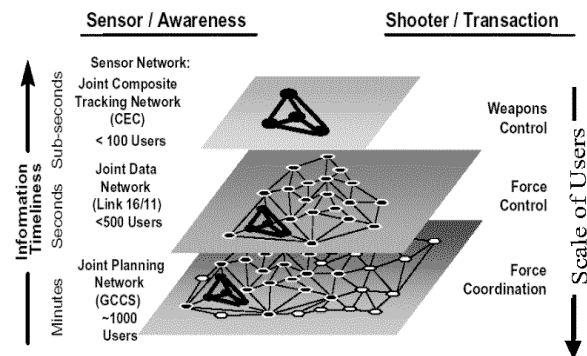


Figure 1 Variations in QoS Across Warfighters

As NCW is a very large scale undertaking, understanding the general nature of the scalability involved is useful for the rest of this paper. Figure 1 (from the NCW paper[1]) shows a hierarchy of layers representing the different users of information sources. As the number of users increases (moving down in the figure) the timeliness of information decreases while the number of users desiring the information increases. In the generalities of an NCW domain, the metrics for these systems center on the efficient use of network resources to deliver information where and when it is needed.

2.1 Example Scenario

As the NCW concept is wide spread, a brief example from a typical civilian “Homeland Security” domain is explored. This helps show the need for the goals of a scalable integrated information system, while still supporting system extensions and adaptations - and how object oriented tools are key for this maintenance and evolution.

In this type of scenario, information from all potential security related aspects will need to be collected and analyzed. This is a daunting task. Imagine classifying every manual or autonomous incident report (traffic tickets, power outages, emergencies, natural disasters, crime, fire, etc.) and correlating with the resources able to deal with the specific emergencies. What type of scalable system could handle the correlation of all potential information sources into a coherent operational picture? And then how would the disparate service providers, such as emergency response capabilities, be effectively utilized - all without hard constrained networks and pre-determined information flows?

The questions for this type of scalable system would be (1) how to integrate all this information (2) where is all the information sent - both producers and consumers (3) who performs the analysis (4) what are the schemes that are used to represent the information (5) how trustworthy and secure is the information (6) how can timeliness and priority constraints be kept (7) what happens over time as all these systems and their data evolve.

To gain perspective on the various solution techniques described in this paper, compare these questions with the typical solutions where large databases keep track of everything. In particular, question the flexibility of these solution to the NCW goals of flexibility in the producers and consumers, and evolution of systems and the information they describe.

3 Modified Siena Classification Scheme

The Siena classification scheme makes a distinction between two main architectural styles. This includes the Reactive style (Events), and the Control-flow style (Channels). Different technologies exhibit different

scalability issues along with elements of real-time and security. This classification scheme will be called Siena+RT in this paper.

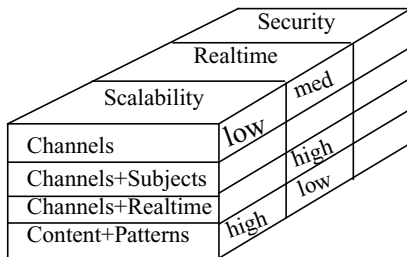


Figure 2 Modified Siena[2] Classification Scheme

Figure 2 shows the following categories graphically. Specifics of these are described in section 4.1.

- Channels - Including Direct-Connect (e.g., Sockets), IP Multicast, and brokered connections (e.g., CORBA, XMLRPC, etc.)
- Channels + Subjects - The ability to filter out just what you want from a particular channel based on subject. (e.g., Filter before sending)
- Channels + Real-Time (RT CORBA, RT Distributed Java, RT Distributed Ada)
- Content + Patterns (Interest Profiles) (e.g., anything matching a specific location)

The security column is a placeholder for highlighting the complex issues involved, including a trade-off between performance and trust. For example, what requirement differences are there between a secure facility communicating through sockets and a mobile user requesting that a standing subscription route information to a home site for later retrieval?

The goal of this ongoing research work is to develop a toolset handbook detailing the various technologies as they fit into this Siena+RT scale. This will provide developers a more concrete set of characteristics that can be applied to different technologies based on information traffic patterns and expected usage mode. It will also help in understanding and quantifying system design trade-offs of the different technologies and approaches available for large-scale real-time information dissemination.

3.1 Latency and Real-Time Attributes

When looking at the real-time characteristics of the Siena+RT model, which real-time attributes can still be achieved at the different layers? For example, when do the following attributes get provided by the underlying technology, and which ones must be implemented in a distributed QoS adaptation framework?

- Transmission and processing Speed,
- Time Management,
- Deadlines and Priorities

Research is being performed in the XIS project to determine how the various attributes of “real-time” are incorporated and then documented in production toolset handbooks. The metrics being captured include:

- Packet routing speed and routing update frequency
- XML Content description overheads
- Throughput sensitivity to number of active subscribers per router and distribution of payload size
- Efficiency of subscription changes (especially important for location aware services)

3.2 Real-Time Interaction Patterns

There is a large body of work in the real time community that could be leveraged for long lived distributed information dissemination system. These patterns include the various couplings between producers and consumers (publisher/subscriber) but with the difference that a direct connection isn't required. These patterns, taken from the real-time Ada95 Language[17], can be classified as follows:

- *Selective Service Acceptance* (publishers), where a service is provided/published selectively based on the offers it gets while juggling it's own tasks.
- *Conditional and Timed Entry Calls* (subscribers), with the ability to withdraw an offer to communicate if the system cannot support the timing or priority constraints
- *Asynchronous and Multi-Way Selection with Aborts* (subscribers) - the ability to use many services at the same time, but when one is complete (or a specified time arrives), the exiting offers are aborted.
- *Priorities and Priority inheritance* - where priorities are used to help both publishers and subscribers to determine the best service to offer or use while supporting priority inheritance to ensure processing timeliness between other needed resources.

For these patterns to be incorporated, the underlying network will have to have protocols allowing specification of these characteristics. This also shows that there should be a symmetry between the producers and consumers of information - as they both play each others role in their life-cycle (a consumer publishes their interest, and a producer consumes service interests).

4 Boeing XIS (eXtensible InfoSphere) Scalability Testbed

The XIS project has a long history in object oriented real-time distributed technology[5] which has been combined with the ability to run in many different real and simulated environment. As new tools are added to meet scalability and security requirements, XIS can test

these against real scenarios from the various projects. This feedback becomes valuable to stress the technologies. The XIS scalability testbed is exploring testing these large scale network issues through simulations and replicated capabilities through the Boeing network.

The Boeing Infosphere is a platform of protocols, processes and common core services that permit stand-alone or web-based applications to submit, discover and share information over a computer network. This is called a “publish-subscribe” information dissemination architecture. XIS supports a number of distributed clients publishing and subscribing to data objects over different transport mechanisms. A global naming capability is provided making connection of clients to XIS services transparent, and easily modified by re-allocating physical mapping of server to computing resources (without affecting the programming of the applications).

By adhering to a de-coupled information dissemination architecture, where users publish, subscribe or query based on abstract *meta* categories, a scalable system can be achieved where there are no fixed locations of information sources. There is still ongoing work determining the best representation for these ontological categories partially based on the technologies that will process this information.

4.1 XIS Toolset Concepts

In the quest for a scalable framework, a number of newer technologies are being incorporated into the XIS toolset as outlined below.

- CORBA/RT-CORBA (*e.g.*, OMG)
- Web Services (*e.g.*, SOAP, XML, UDDI)
- Context-Based Routing (*e.g.*, Siena, Semandex)
- Peer-to-Peer Networks (*e.g.*, JXTA[10])
- Mobile Computing Infrastructure

During this ongoing process, each of these technologies have different scalability and fault-tolerance attributes as they fit the Siena+RT scale shown in Figure 2

CORBA have been used for some time as a foundation capability higher in abstraction than sockets. The leverage of the IDL specification and language neutrality have been vital in large deployments. Real-Time CORBA (and tools like Real-Time Distributed Ada or Java) are also incorporated when the tighter demands on “Real-Time” are required. With the advent of the XML revolution, some think CORBA is outdated while maybe forgetting the full object-oriented facilities and processing power still useful for many applications.

The popular internet based XML web service can be characterized as software applications identified globally by a URI. Their service interfaces and binding are capable of being defined, described and discovered by XML - thus making it language independent. As

such, web services support direct interactions with other software applications using XML-based messages via internet protocols.

Content based routing is a newer technology and has many promising benefits for scalability[6][7]. Section 6 describes this technology in more detail. Unlike traditional networks, where data packets are delivered according to a low-level IP address or set of addresses, Content-Based networks route and deliver packets by examining the information content of the packet, not its IP address. These systems emphasize structuring the data into descriptions of the users interests (structured query like statements) and producers of information. The routers then communicate between themselves and pass the interest profiles around in a manner that allows efficient information matching and eventual routing to the desired users. The system effectively operates as an integrated, distributed, publish/subscribe system with reduced management and administration overheads of existing systems.

Peer-to-Peer networks are also being explored as they might be a good fit for mobile computing. There is some similarity between content routing and peer networks because of the grid of common knowledge. The big difference comes in the one-to-one connectivity to actually perform communication which goes against some of the de-coupled characteristics of a publish-subscribe system. A main challenge with peer networks is finding a resource of a desired type, and how to do so in a scalable manner. Wide area lookup services, such as Chord[15], are being explored.

The difference between the channel connection technologies and the content based routing are important because of the need for other supporting tools. For example, in channels, getting connected requires distributed discovery of the channel itself (*e.g.*, the port number, host, etc.). This is accomplished differently by the different technologies (such as using UDDI for web services, or a multicast global naming protocol useful for CORBA and others).

As the use of these tools migrates through these spectrums, the Siena+RT classification scheme will help decide when these various tools are appropriate. In particular, channel based technologies is appropriate in many domains because of its higher real-time capabilities. In contrast, Content-Based routing might be desired when the content of the information (or meta-information) is more important since it might come from an untold number of channels.

XIS is also exploring how to automate the connections between the various abstraction and tool layers. For example, some form of information digests could be generated and then handed up to the wider distribution capabilities.

4.2 XIS Distributed Security Architectures

As security is a vital aspect of deployed systems, XIS incorporates a distributed security architecture. This is based on an extensible RBAC (Role Based Access Control) capability, and incorporates a policy management language.

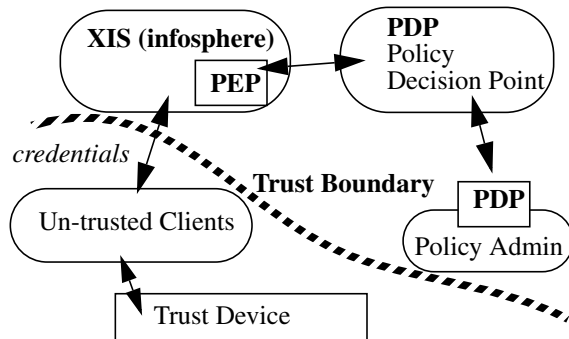


Figure 3 Trust Boundaries in RBAC Security

Figure 3 shows the broad characteristics of the RBAC distributed security. This diagram shows that a trust boundary is placed between the XIS servers and un-trusted clients. Here the Policy Enforcement Point (PEP) manages the communication to the Policy Decision Points (PDP) which evaluates policies and checks credentials. The result is used by the PEP to decide whether the action requested is allowed.

Identity and certificate management is handled through the normal public-key infrastructure (with the assumption that a military installation will have their own certificate infrastructure). Users are mapped to roles while actions (such as publish) are described using policies with possible complex characteristics. For example, a policy might describe that a user is of the correct role but doesn't have other important information (such as a secret key only available in a secure facility). The interaction between the PEP and the PDP is envisioned to follow a more *dialog* based interface as mentioned in section 5.1.

When dealing in the un-trusted computing world, many of the common practices will be replaced with some form of trust devices. Smart cards or other devices are used to encrypt any information that will be sent from a client to the secure XIS server. These devices must be trusted leading to newer trust communities (such as fledging XNS[13]) and tamper proof devices.

4.3 NCW Example Application Techniques

Using the NCW example from section 2.1, some brief contrasts can be made with the different technologies.

4.3.1 Trust and Security

For example, channels and peer-to-peer systems can deal with (5) trust and security of the parties much easier since they are communicating directly. Decoupled information systems must push the trust

boundary to the clients connection to the network, while also ensuring the node-to-node connections are all between trusted parties (or all at the same security level and trusted infrastructure).

4.3.2 Where to Send and Analyze Data

When dealing with infusing disparate information from the various incident reports, deciding, (2) where the data is sent, and (3) who performs the analysis, is a very complex problem. If a peer or channel is needed, then looking up the desired party involves mapping very complex multi-dimensional data into single-dimensional channels, requiring overlaps of channels when information has multiple fits. With a content-based solution technique, the network of routers is keeping the interests of eventual end users (grouped through router hierarchies) - and matching every packet to determine where to send the data. These will be routed differently all the time based on the frequent changes in the interests of the users (such as a location-aware service).

4.3.3 System and Information Evolution

As a system evolves over time, (7) dynamic and frequent re-binding of all the information sources will be required. What is desired is to adapt and absorb new, potentially unforeseen, data sources as they become defined. Given that information will be structured using data schemes, consisting of types made up by combining known sub-types, there are two contrasting solutions.

For example, suppose sub-types *location*, *altitude* and *vehicle speed* are the desired attributes of interest. How would a type system deal with this? The elements of multiple inheritance seem the closest fit, but are not accurate. The best offered by the channel or peer approach is to subscribe to pre-known types that have the 3 attributes (resulting in maybe many types). This can lead to inflexible and sometimes fragile solutions since one has to build in pre-knowledge of a fixed set of types. As systems and their information are re-classified, to fit a new understanding of the problem domains, existing software will need modification to re-express their interests in these new terms. (Note that filtering could still be performed on a clients behalf but that only shields the clients version of the same problem.)

Another approach would be to express an interest for any types that happen to have those three sub-type attributes, usually with further constraints (e.g., *speed* > 100mph). A system of content-based routers performs the predicate logic on behalf of the other routers and ultimately the end users. Thus an end user just receives the information of interest. Of equal importance is that by describing an interest using the sub-types, evolution of a system which adds new data types, but still uses the same sub-types in its composition (such as location) will shield existing clients from making any

modifications - while still reaping the benefits of the new information. In contrast, re-tooling for every maintenance step would be a very costly proposal.

5 Challenges of Scalable Information Dissemination Systems

As information migrates across the spectrum from tight to de-coupled connectivity, different technologies become important. In particular, what aspects of real-time characteristics can be kept as the technologies move further from the tightly integrated systems and when do these just become adaptive Quality of Service (QoS) attributes?

These challenges are highlighted in the following sections. The XIS testbed provides a framework for testing these various concepts which is still an on-going process.

5.1 Dialog Interfaces and Object Oriented Support

Dialog Interfaces (a re-mapping of object-oriented technology coined by this project) could become a useful software pattern. For example, interfacing to the security system entails exchanging credentials, while in a secure system, knowing what to pass is also a secure attribute, so knowing everything to pass might need to be negotiated through dialog. (e.g., your client is asked for more information from the server, such as secret keys they can only get from inside a secure facility.)

Designing these newer dialog interfaces is challenging even with object-oriented techniques. One of the issues is how to design interfaces so their signature is expandable. Does an interface get extended with additional arguments, or should a placeholder argument that can represent many object be used, possibly expressed in adaptable XML syntax?

5.2 Pro-Active Management in a Decoupled System

Another dichotomy of information management in a de-coupled system is the difference between information sharing (pub/sub) and proactive/directed control (someone go do something). The first is for an analyst to work with data, while the second is then to really do something (and know it was done). One example might be an analyst looking at image data, and needing a new photo from somewhere close by. Do they send a 'subscribe' request for that new location - and hope that the supplier of that type of data is subscribing to that "request", and then have the system manage that request and other requesters for the same resource?

Much of this pro-active management can be handled in an adaptable QoS framework provided the underlying protocols are defined and agreed on. As the real-time patterns shown in section 3.2 are adopted, these will help instill some of the desired "control" back into the de-coupled systems.

5.3 Mobility of Clients - Location Sensitivity

Research is being done to reduce the burden of mobility on infrastructures. For example, can IP address change be made transparent? Mobility of the warfighter also requires dealing with difficult issues including lost packets, high latency, packets out of order, intermittent communication, routing issues (proactive vs. reactive), and different networking protocols.

In general, mobility changes many of the pre-conceived notions of an existing IP network. To address some of those concerns, the XIS framework is exploring the HIP[3] protocol to help solve the C2 (Command and Control) on the move capabilities.

One of the key issues with mobility is managing the location of the end-user. Providing network services that are *Location Aware* is promising. Of particular interest for performance is how a system could efficiently keep track of all the changing location values, both for producers and consumers. For example, a mobile client will change their interest frequently based in part on their physical location - and what are the network latency costs of these changes? When looking at system evolution, these same location services can then subscribe to all events (of any type) that are in the clients area (by using a known location sub-type).

5.4 Fault Tolerance and the Problems of Existing "Connected" Clients

New scalability technologies have different fault models that must be reduced. As an example, for channel or web-service based systems, if a client is connected to a capability which dies, what is the recovery mode? For routing networks, what happens when routers die?

Other faults might be in the connections between routing systems. Does the network topology instill faults that cannot be recovered through re-routing? In an even more drastic case, what if vast numbers of nodes are invalidated. Studies in network survivability analysis, such as the Willow[12] architecture, are promising.

5.5 Languages for Describing Information Relationships (Pub/Sub, etc.)

A number of research efforts are involved in finding vocabulary and tools to classify information and provide mechanisms to make connection mapping between producers and consumers more automated. The XIS system is exploring some of these technologies itemized as follows:

- Security Policy Languages (e.g., Ponder[16])
- Ontologies and Inference Engines (e.g., FaCT)
- Agents and Agent Languages (e.g., DAML[11])
- Routing Languages (Siena[6], Semandex[9])

- Fuselets (an agent concept from the JBI[4])
- Adaptable Dynamic QoS Systems[19]
- UML Profiles (Real-Time, Workflow, etc.)

Each of these language frameworks are being addressed by different research camps. They also have different characteristics dealing with scalability in a distributed system. For example, if a central knowledge processor is required, many scalability concerns arise.

5.6 Trust in Software and Software Boundaries

Physical and organizational trust in software is an issue related to security. Issues include:

- Trust of the boundaries of the software and where the code runs (tamper proof, etc.)
- Trust-boundaries of the installed systems
- Trust in distributing rules/agents/policies/routing throughout the network (*e.g.*, no central software)

These issues also becomes more specific as their use moves closer to the real time and safety critical community. For example adding software threads to safety critical Ada software was just recently adopted under strict guidelines in the Ravenscar Profile[14], and recently being adopted to the Java Realtime language.

5.7 XML Protocols ease Abstraction Integration

Once all these capabilities become part of an integrated set of technologies, such as the XIS testbed, what are the requirements on the programmer's seamless use of these tools? This can become a difficult issue if each technology requires a brand new API. Many current systems are supporting an XML based interface to decrease some of these issues. The XIS system is exploring using XML as a mechanism to describe the entire protocol, including the actions desired (such as publish), the users credentials, and any other information. Thus the transport of this byte stream is outside the protocol and API's themselves.

The JBI Infospherics working group[4] is looking at the XML protocols for the publish subscribe paradigm. The goal is to first develop language mapping for a common set of operations (encoded in Java), and then as these become well proven, encode them into XML. Protocols like Jabber[18] are good examples to follow.

6 Scalability with Content Based Routing

Content Based Routing as described in this paper has some promising scalability and extensibility properties, both key to the NCW goals. As section 4.3.3 described various aspects of system and information evolution, this section will add two other portions of the content routing solution. First an example of a content

publication, and a subscription is shown. Then the importance of affinity grouping is shown as a key scalability tool.

6.1 Pub/Sub Languages

Using the Semandex[9] content-based toolset, the lowest level of an XML publish and subscribe is shown in the following two figures (Semandex provides easier client abstraction tools to shield this detailed XML complexity). Figure 4 represents the publishing of

```
<Rdf xmlns=http://schema.semandex.net/sp/1.0>
  <Security><Level>3</Level></Security>
  <Priority>2</Priority>
  <Content>
    <er:Event xmlns:er=http://schema.fema.org/emergency>
      <er:Address>9thNW/Pennsylvania</er:Address>
      <er:Team>Fire</er:Team>
      <er:CBA>Yes</er:CBA>
      <er:Paramedic>No</er:Paramedic>
    </er:Event>
    <er:location>
      <er:Lat>38.89</er:Lat>
      <er:Lon>77.025</er:Lon>
    </er:location>
  </Content>
</Rdf>
```

Figure 4 Sample Content Description (Publish)

information that describes two sub-types, one specifying that a fire *event* occurred where a CBA (Closed-Breathing Apparatus) is required but no paramedic is needed. The second sub-type is the location information. Clients that are interested in any events (fire or otherwise) at that or near this location could be notified. Figure 5 represents a interest subscription that specifies that any fire events where paramedics are not required will be analyzed. Here the location attribute might be used by the end user to see if they could still make the fire (or they might be on the road near the location specified). This shows the flexibility of the information characteristics.

```
<Rdf xmlns=http://schema.semandex.net/sp/1.0>
  <op:and>
    <Security><Level><op:integer-range min=6 max=1000/>
    </Level></Security>
  </Content>
  <er:Event xmlns:er=http://schema.emergency>
    <op:and>
      <er:Team><op:string-match>Fire</op...>
      <er:Paramedic><op:string-match>No</op...>
    </op:and>
  </er:Event>
</Content>
</op:and>
</Rdf>
```

Figure 5 Sample XML Interest Profile (Subscription)

6.2 Using Natural Affinity as Scalability Tool

An important characteristic of networks, as well as organizations, are their relative close grouping of common users with common interests. This is called "affinity" and is a key tool to ease network scalability solutions.

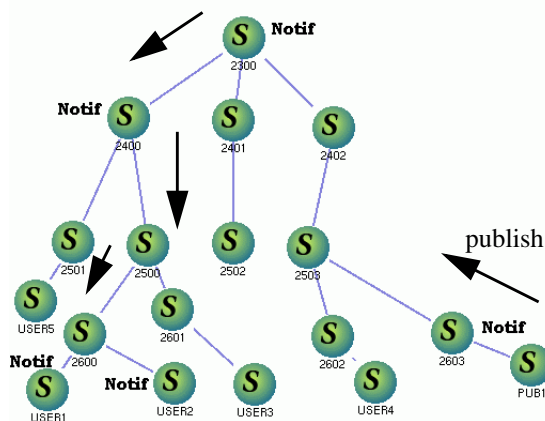


Figure 6 User Network Affinity as Scalability Tool

Figure 6 shows a simple hierarchy of Siena content based routers. The two user nodes in the bottom left have previously subscribed using similar interest profiles, possibly differentiated by a location attribute. These profiles are aggregated upwards through the routers, stopping when common information has already been routed. Later, a publisher of information, shown on the bottom right, provides content matching some of the interest. A single message is sent through the routers until it reaches the last router, and then a multi-cast might occur to replicate the message to the original subscribers. This shows three important aspects applicable to scalability: (1) publishers don't know the subscribers, (2) subscribers don't know where information of interest will come from, and (3) once information flows, it can stay as a single packet for a long portion of its travels reducing network overhead.

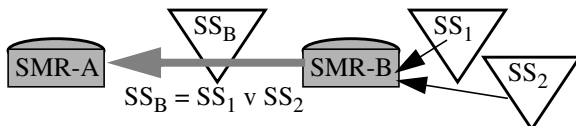


Figure 7 Aggregating Semantic Selector Clients

Figure 7 briefly shows how client Semantic Selectors (SS) are aggregated by the routers. Here, SMR-B (Semantic Multicast Router-B) presents to SMR-A the aggregated sum of interests from clients $SS_{1,2}$ while also simplifying SMR-A's processing demands.

7 Summary

Mapping the various technologies to the different layers of system development is still an on-going process and providing a testbed both for a running system and a development environment becomes crucial. This becomes more important as running metrics can be captured to test the scalability decisions against the latency requirements.

As the research work is ongoing, 2003 will see the addition of the content-based routing in a large scale environment simulation. The XIS Scalability Testbed is

a valuable framework for introducing new capabilities, while providing ongoing production capabilities to a number of customers.

Network Centric Warfare challenges the scalable of systems both in the scale of network and information resources, but also of scale through system evolution - extending and adapting to changing requirements and information sources. This paper detailed many newer internet technologies while contrasting their potential capabilities to meet these challenges.

8 References

- [1] Network Centric Warfare - Developing and Leveraging Information Superiority, 2nd Edition, David Alberts, et al CCRP. <http://www.dodccrp.org>
- [2] Carzaniga, et al. "Challenges for Distributed Event Services: Scalability vs. Expressiveness", Engineering Distributed Objects (EDO '99), ICSE 99 Workshop, Los Angeles CA. May, 1999.
- [3] R. Moscovitz, "Host Identity Payload and Protocol," Internet Draft", Nov 2001.
- [4] JBI - Joint Battlespace Infosphere, <http://www.infospherics.org/>
- [5] Moody, Scott. "Distributed Object-Oriented Real-Time Systems using a Hybrid Model of Ada95's Built-In Distributed Capability and Emerging Real-Time CORBA Capabilities". IEEE ISORC'98, April 98.
- [6] Carzaniga, Antonio, "Architectures for an Event Notification Service Scalable to Wide-area Networks", PhD Thesis
- [7] Makwana, Dhananjay. "An Introduction to Semantic Networking and its Performance Evaluation". MS Thesis, Rutgers. Jan 2002
- [8] Boeing Phantom Works, The Boeing Company <http://www.boeing.com/phantom>
- [9] Semandex Networks Inc, <http://www.semandex.net>
- [10] JXTA, <http://www.jxta.org>
- [11] DAML, <http://www.daml.org>
- [12] Knight, J., et al, "The Willow Architecture: Comprehensive Survivability for Large-Scale Distributed Applications", DSN-2002 The International Conference on Dependable Systems and Networks, (June 2002)
- [13] XNS - eXtensible Name System, <http://www.xns.org>
- [14] Ada Ravenscar Profile, <http://www.openravenscar.org/>
- [15] Stocia, et al. "Chord: A Scalable Peer-to-peer Lookup Service for Internet Application", SIGCOMM'01, Aug 2001.
- [16] N., Dulay, et al. "The Ponder Policy Specification Language". Proc. Policy 2001: International Workshop on Policies for Distributed Systems and Networks, Bristol, UK, Jan. 2001, Springer-Verlag LNCS 1995,
- [17] Burns, Wellings. *Concurrency in Ada*, Cambridge University Press, 1998
- [18] Jabber, Inc. <http://www.jabber.com>
- [19] Casimiro A., Verissimo, P., "Using the timely computing base for dependable qos adaptations." In *Proceedings of the 20th IEEE Symposium on Reliable Distributed Systems*, Oct 2001. IEEE Computing Society Press