Web Services for Dynamic, Weather Communities

Bill Kerr, Dave Dimitriou, and Earl Ravid – Fleet Numerical Meteorology and Oceanography Center, Monterey, California

1 Introduction

Service oriented architectures (SOA) comprised of Web services are being deployed in public and private sector environments to meet a variety of goals, including interoperability, data sharing, more user-friendly human-computer interactions, and cost savings. The U.S. Navy's Fleet Numerical Meteorology and Oceanography Center (FNMOC) is currently engaged in modernizing our weather data and product generation, assimilation, handling, and distribution infrastructure, including deploying Web services. In the course of this modernization project, we have realized that a well-designed Web service oriented architecture also allows us to better serve a user base that is becoming increasingly dynamic.

Both military and civilian missions around the world are being addressed by diverse coalitions of personnel and resources. Missions as widely varying as combat, providing humanitarian aid, and safe daily operations are characterized by changing support teams co-located in command and control centers and sharing or trying to share a military or civilian operational picture that includes meteorological information.

Providing support to these coalitions involves addressing releasability issues, dealing with groups with rapidly changing memberships, modifying the information available quickly, and doing so in an environment of declining personnel resources. FNMOC is building an architecture capable of complex and quickly-reconfigurable access control.

2 Background

FNMOC's primary users are military personnel with meteorology and oceanography training, supporting a spectrum of military missions. Established agencies like the North Atlantic Treaty Organization are supported through standing methodologies. FNMOC also supports a number of applications making machine-to-machine interface calls to FNMOC databases and applications, and this area of our business is increasing.

Preparing for varying coalitions of international, interagency, or interservice personnel requires significant effort. Special modifications to existing processes are required, or the establishment of redundant data and applications access crafted to the particular grouping involved. FNMOC then has to track membership in the coalition, which can be difficult, and any changes in the coalition require an analysis of what we are making available, potentially with another special effort required to modify the already-modified access.

In 2001, FNMOC began a major project to modernize products and applications that we make available, and the means by which we make data, products, and applications available. The entire project, called Applications, Transactions, and Observations Subsystem Upgrade (ATOS2), is beyond the scope of this paper, but one of the elements is an analysis of our data, products and applications from the perspective of deciding what capabilities to implement as Web services. This analysis of which functions to make available as discrete services is critical to handling highly-specific or tightly defined access. The structure of ATOS2 Web services will be discussed last as the other two elements are intended to support the SOA.

To support the software component design, the ATOS2 project is based on commodity hardware, including clusters of dual CPU server nodes, large storage area network (SAN) systems, load-balancing network switches, and a small number of medium-sized multi-CPU servers. This is a significant change from the existing architecture, which was based on Sun E-class servers, up to an Enterprise 10K.

The third tool FNMOC will use to provide flexible support is service access control. FNMOC has developed a Web portal based on Apache, Tomcat, Jetspeed, and JBoss. We augmented these open source technologies with an authentication service called Navy Enterprise Single Sign-On (NESSO) that implements the Security Assertion Markup Language (SAML) standard and works with Lightweight Directory Access Protocol (LDAP). NESSO and apache capabilities in the portal will be used to deploy a simple role-based access control (RBAC) for the short term. The long-term solution for access control will be a policy decision service capable of interpreting a rich rule language, and providing binary responses to service invocations. Security, logging, and access management will be facilitated by treating all requests and invocations as being on behalf of a unique user, including machine-to-machine transactions.

3 Approach

By building and deploying the capabilities referred to above, FNMOC will be able to respond rapidly and efficiently to the formation or modification of user communities. The services necessary for mission support can be composed by system and application managers in a short period of time, deployed to computers with capacity to run them, with user access controlled through a generic mechanism.

Today, if a new requirement is identified by our Warfighter Support personnel we are not likely to have an operational solution running for four to six months or more. We expect that, with the ATOS2 architecture in place, we could compose a solution from available services, add components related to the specific requirement and deploy the new capability operationally in one to two months.

3.1 Hardware Architecture

Important design elements of the hardware architecture are that there are a large number of relatively small commodity systems, supported by a large-capacity SAN, and a few larger server machines for applications that don't lend themselves to loose coupling (like relational databases). Internal and external networking systems (switches, routers, private networks) make it possible to scale an environment without a lot of physical equipment adjusting.

With software already broken down into loosely coupled components designed to provide a single service or function for the most part, a hardware suite that applies physical control becomes more powerful. The use of a SAN allows services to be quickly loaded to a computer that will run them. A service in demand can have additional copies installed on idle nodes and the load-balancing systems can start sending invocations to those systems. The combination of hardware and software access control allows FNMOC to provide a level of service specific to the users and/or missions being supported. If there is sufficient user load, software controls to specific services can be replaced by hardware controls to systems.

The hardware architecture can also be used to deploy two SOA configurations of interest in the context of supporting coalitions of users. One of these is essentially an entirely separate instance of the entire architecture, physically isolated from other FNMOC systems. To support the rapidly-developing emergency situation in the Gulf of Mexico when Hurricane Katrina hit the United States, FNMOC made the decision to accept a higher level of risk on our legacy systems to make support available to military and civilian units entering the area to provide relief. Had the ATOS2 system been available, it would have been easier to setup separate network and computer resources for Katrina operations and the security risk would have been greatly reduced.

The second configuration is similar in concept, but is not a complete separate capability. Network domain controls, static routing and some other features are used to create what appears to a user to be a separate environment, but it is calling the same services. Because of Department of Defense (DOD) regulations regarding public Web sites, FNMOC may stand up a separate network namespace that provides services and information to the public using the same services. The first deployment might be one or more separate Web servers with reduced network and hardware connectivity to limit potential penetration to those units alone.

Finally, a flexible hardware environment allows FNMOC to continue efficiently serving a class of customers we call "bulk data" customers. There are a few organizations that we send entire global model runs to, organizations that perform archiving, climatological analysis, and/or additional modeling. These customers remain relatively static in their data requirements, needing the same distribution for long periods of time. For now, it appears worthwhile to maintain separate network communications and processing paths to support these customers. Network and processing hardware can be dedicated to supporting this customer base, but within the broader ATOS2 maintenance framework.

3.2 Security Control

Security controls include authentication (positively identifying a unique user) and authorization (allowing the unique user to access specific combinations of services, data, and processing), as well as load controls on processes and resources. Authentication and authorization will be discussed below. ATOS2 process and resource controls are currently only in early stages of design. FNMOC has the capability to run mesoscale models in the supercomputer environment or in the ATOS2 environment. We are working toward remote operations to take advantage of supercomputers located at other government sites. We are developing "on demand" mesoscale modeling capability through the Centralized Atmospheric Analysis and Prediction System (CAAPS) program. The notion of allowing a remote user to initiate a modeling process at our facility is new and will require definite process controls. We also foresee a need to develop mechanisms to prioritize users and user requests to ensure adequate resources to the most important missions. Protection of resources is an important part of our security requirements, in addition to information security.

3.2.1 User Authentication

The U.S. DOD has mandated that all military personnel and civilian employees have a Common Access Card, a smart card that has public key infrastructure (PKI) certificates unique to the user installed on the card, and that requires the user to enter a personal identification number when they use it. This provides a relatively secure mechanism for us to uniquely identify users. On networks where CAC cards are not yet in use, we still rely on a password-based process. FNMOC is working with other DOD agencies to develop secure biometric (fingerprint, at this time) identification to provide stronger three-factor authentication.

When a user attempts to access a protected FNMOC Web portal, site, or service, the request is intercepted by the FNMOC NESSO plug-in. The user's credentials are forwarded to an identity server that uses an LDAP database. A SAML assertion is returned. This assertion currently includes a simple string of role names assigned by FNMOC's Customer Support Team, but this role-based access control (RBAC) mechanism for authorization will be replaced by the Soutei solution described below.

The Security Assertion Markup Language (SAML) assertion is forwarded in the Hyper Text Transfer Protocol (HTTP) header for the duration of the user's session and the roles are used to make access decisions. Sessions have time limits and users may be required to re-authenticate. FNMOC has built trusted networks wherein a second NESSO identity server will recognize and accept SAML assertions from a different NESSO identity server and issue an assertion for the second domain. We are building in security features like digital signatures on the SAML assertion so that when we establish a trusted server relationship with remote facilities, we can securely send them over open networks.

3.2.2 Role Based Access Control

FNMOC is implementing a role based mechanism for Initial Operational Capability of ATOS2 in order to support legacy applications still in use. The SAML assertion created by the NESSO Identity Server will include a string that applications can parse and use to map the user to existing roles. The NESSO system will not parse or act on the strings. This interim solution allows FNMOC to move forward into the ATOS2 environment while security components are developed and tested.

FNMOC has concluded that RBAC is not a sustainable approach for us, in the long term. An RBAC solution must map services, applications and/or products to roles and to users. FNMOC has identified additional access characterizations, such as quantity of products allowed. These can be incorporated into the roles available, but add considerable complexity to management of roles. A feature of use of the CAC is that we no longer allow group accounts such as "watchstander" or "administrator," driving the number of discrete users much higher than in the past. If the number of services is minimized to ease maintenance of an RBAC security model, then other benefits of functional modeling of the service oriented architecture are lost. Recent operational experience and identified role requirements have already shown that we cannot use a limited number of roles to keep the RABC system manageable.

3.2.3 Policy Decision Point

FNMOC is developing a policy decision engine, an invocable Web service, that will act in isolation from the user and the user's interface, as well as in isolation from services included in its rule set. This mechanism is called "Soutei" by its inventor. Soutei includes a rich, but sensible language to describe rules ("sensible" in that the rules are expressed in English and make sense to an experienced reader). Soutei's rules are provided by the developer of the application or service, so access control is still owned by the owner of the service and data, but the rules are interpreted and acted upon by Soutei in isolation from the service. In addition, Soutei provides the ability for users with appropriate permissions to grant subsets of their access to other users, as allowed by the application, so users can add rules to a service's rule set without interacting with the service at all, provided the service allows this sort of delegation, which is also expressed as a Soutei rule.

Soutei is based on Binder and Keynote, and is designed to be functionally provable, so it can be the Policy Decision Point (PDP) for a high Common Criteria Evaluation Assurance Level system. For ease of use by application and service developers, and users, FNMOC is developing user facing services to allow rules to be entered or modified via the Web.

The use of a single policy decision point supports highly dynamic situations more efficiently than an RBAC system: one rule change at the PDP can take the place of changes to the RBAC system, modifications to applications and services, and much time-consuming work by support staff. Access can be granted or taken away manually or automatically based on time, resource load, user prioritization, services previously used, IP address, network domain, or many other criteria that are not limited to characteristics of the user or the service being accessed.

The use of Soutei, or another PDP, also provides the benefits of decoupled development. Applications and services, as well as security components, can be modified as needed without impacting one another or requiring dependent development. The user's access can be determined without disclosing even the existence of disallowed services. Interacting with the portal, a user can get a customized view of, and catalogue of, allowed portlet services and applications, again hiding the existence of disallowed capabilities. The use of a separate, decoupled, security architecture makes it easier for FNMOC to prepare for and pass external security evaluations and certifications and, in the event of failure to pass, easier to make necessary modifications to the correct software components and configurations to pass.

3.3 Service Deployment

Decisions about the specific services to be deployed and the timetable of deploying services are very complex, and a simplified view will be presented here. FNMOC currently has well over five million lines of code under configuration management. The ATOS2 environment is an n-tier Web SOA architecture with a presentation layer, business logic (application) layer, and data layer. FNMOC also has an infrastructure of supercomputing resources that is further separated, and will not be a part of ATOS2.

Final decisions about access to the supercomputer node have not been made as of this writing. Bulk data customers, referred to in paragraph 3.1 above, sometimes have tight timing requirements. Benchmarks run to date have indicated that Web services, with the inherent drawbacks of multiple transaction interfaces and payload bloating, might not meet delivery requirements. It may be necessary to allow machines dedicated to bulk data transfer in the ATOS2 clusters to connect directly to the supercomputer nodes and use non-Web based communication protocols.

Other than this exception, FNMOC intends to distribute all data and products via the ATOS2 architecture. The foundation data repository used by FNMOC is Integrated Storage Information System (ISIS) and the ISIS database will be replicated on the ATOS2 data layer so that transactions within ATOS2 do not have to pass to and from the supercomputer node.

There are software modernization efforts underway that effect the transition of existing functions to the ATOS2 environment. Several legacy applications have been restructured to a decoupled software architecture that better matches the n-tiers. Some helper applications have been coded as services. Some applications have been restructured to call data services instead of legacy database application programmer interfaces (APIs). Some new services have been deployed; the easiest decision to make since there was no existing code investment.

Not every function needs to be a separate service. Making functions available as Web services carries some bandwidth and processing overhead since the standards FNMOC adheres to are HTTP with Secure Socket Layer (SSL) encryption (HTTPS), Extensible Markup Language (XML) and dialects of XML, SOAP, and others that add to the amount of bytes being shipped around. As well as deciding which functions should be separate, years of deploying capability as standalone applications has led to having functions that should be deployed as individual services contained within the code base of a larger project. A very simple initial gauge is that a function used by more than one application should be deployed as a service and invoked by the applications. A service used by only one application does not add value by being deployed as a service.

In some cases, services will be deployed in more than one place in the architecture because it's better to deploy the service again than to take security risks by opening ports through firewalls. In some cases, even where a service may exist for a function, performance requirements are such that legacy interfaces will be used within the network enclave. An example of this is that it is not necessary to make every network transaction deep within FNMOC's infrastructure use SSL. In a protected enclave, a lighter protocol like HTTP can be used. A non-Web service interface is being implemented to meet performance requirements in moving output model data from supercomputer systems to the ATOS2 system. An example of a mixed implementation is the Come And Get It Product Store (CAGIPS) which will become the primary data distribution mechanism for FNMOC. CAGIPS has non-Web Service cache and database API interfaces within its structure, but will also provide Web service interfaces for internal and external applications and other services to call, including other CAGIPS components where desirable. This does add to the cost of maintaining more than one interface for a given data type or service, and that cost should be considered when performing the analysis of which capabilities to deploy as services.

One of the lessons learned in the ATOS2 project has been that migration of capability does not occur at all until a threshold of infrastructure is achieved. In the older ATOS environment, the infrastructure was fairly simple: a machine needed an operating system and some configuration, possibly some support software like apache, then a developer could load and run applications. In the ATOS2 environment, FNMOC has had to deal with a storage area network, system controls for hundreds of systems, system administration in a clustered environment, mass deployment of firmware and software upgrades, and a multitude of other differences that have to be worked out before the first developer can load the first service on one of the nodes. The issues created by this threshold are not just technical, either. Tremendous pressure was placed on the ATOS2 team when the first cluster (the alpha environment) was still being built and developers had no access to it. Synchronizing development and system administration activities has continued to be challenging at times.

The result of doing this breakdown of functions into services is a collection of components that are designed to work through standard interfaces, and that can be selected by knowledgeable individuals to form a set of capabilities for a user. An example of this is our High Winds and Seas application. Model output fields are ingested by a calculation engine using the same service interface that other applications use to get the data. The analysis software identifies portions of the earth's surface where waves and/or wind speed will exceed set parameters. The analysis engine then calls a mapping service that creates map renderings, the same mapping service used by other applications for simple map views. The next step in the development of the application will be to use a Web service interface to populate an Open Geospatial Information System Consortium Web Feature Server (WFS) with warning area polygons. The mapping service, the WFS, and the data request are all services used by other services and applications.

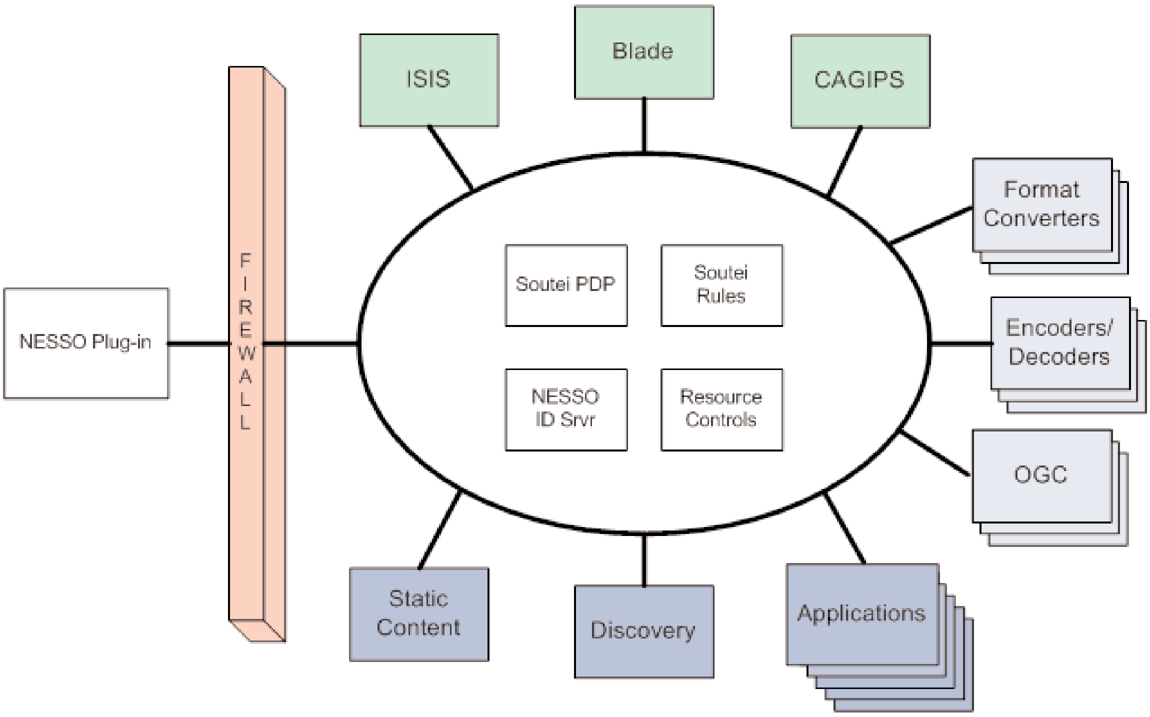

Fig. 1

Figure 1 shows a number of the features discussed in section 3, including example services, capabilities, and applications. The design decision implied by having security services in the center of the architecture has not been finalized yet. We do not have enough data to determine the impact of making all service-to-service transactions go through a security filter. This is the most secure implementation, but if FNMOC cannot meet operational timeliness requirements, security services will be invoked when making initial requests only, and internal service-to-service transactions can proceed without rechecking user access. Given that any of the services shown could, in theory, be called by any other, trying to show a block diagram would be impossible.

In Figure 1, data services are shown in green. ISIS and CAGIPS are described above. The "blade" is used for subgridding and answering more complex, non-grid based queries. Applications are shown in darker blue; other services, some of which would be invoked by applications, are shown in lighter blue and show support for industry standards, like the Open GIS Consortium standard server interfaces Web Map Server, Web Feature Server, and Web Coverage Server. Security services are shown in black. Figure 1 is not exhaustive.

4. Conclusion

The number and variety of missions supported by FNMOC has increased in recent years. The operating environment at command centers supported by the U.S. Department of Defense has become more complex as coalition warfare has become the usual practice. Increased levels of cooperation with non-DOD U.S. agencies have added complexity to FNMOC's mission. Two areas of requirements that have existed but have not been satisfactorily addressed are the short reaction times available to respond to natural phenomena like hurricanes and tsunamis; and the difficulty of effectively managing access to centralized resources, like FNMOC, for differing and changing user communities. Improvements in technology enable FNMOC to respond to all of these requirements by deploying a well-architected set of Web services and applications on a flexible hardware suite, protected by decoupled, robust security components. Deploying these capabilities has been challenging, but this architecture provides a solid foundation for hosting of future capabilities, and supports existing users and customers.