Cross-Organization Service Use Management for SSA

Jeremy Witmer
The MITRE Corporation
Kelly Gerschefske
The MITRE Corporation

CONFERENCE PAPER

With the Joint Space Operations Center Mission Systems (JMS) focus to deploy a service-oriented architecture (SOA) environment, the Space Situational Awareness (SSA) community is moving rapidly toward a platform environment. Organizations will no longer rely entirely on systems within their boundaries. Instead, services and data are shared across organizational boundaries between diverse organizations. For SSA to succeed, JMS and similar efforts must employ and share resources across organizational boundaries (from AFSPC, to other US partners, to non-US partners and even universities). However, sharing services across organizational boundaries presents visibility and dependency issues. What information does an organization need to rely on these external services for mission critical needs? This paper presents an approach to dynamic service use agreement (SUA) negotiation that provides service platforms (and SOAs) the ability to dynamically negotiate the use of services across organizational boundaries. Using a small set of common service level agreement metadata (SLA metadata) parameters, service use can automatically be negotiated between SOAs. The Managing Aggregated Services (MASS) toolkit is designed to enable automated SUA across organizational boundaries by standardizing the service use parameters (the SLA metadata) as the foundation for SUA. The MASS toolkit demonstrates that this SUA can be done between systems as the SLA metadata provides the necessary visibility for the consuming organization into the provider organization services. The MASS toolkit also answers the question "who is using my service?" Service access can be limited to only SOAs which have negotiated an SUA, providing information security for service providers. In the highly dynamic SSA environment, where services will be provided by a variety of organizations, previous work has shown that SOA can shorten service integration from months to weeks. MASS demonstrates the potential to move this integration to days and hours by removing the need to negotiate the technical parameters of a service level agreement for each new service interaction by hand.

1. INTRODUCTION

The commercial world is rapidly moving toward a platform-based environment. In this environment, organizations no longer rely entirely on systems and data within their boundaries. Instead, services and data are shared between organizations. This approach has demonstrated increased flexibility, adaptation, and cost savings. It is this approach that has allowed Amazon to become the largest online retailer in the world[1], by aggressively developing all software in the company to be a service available to other internal customers. This internal service platform has now created its own business model, in the form of Amazon Web Services.

Our research shows that the SSA community will benefit from the ongoing move to a platform-based architecture, built around services. The Joint Space Operations Center Mission Systems (JMS) focus to deploy a service-oriented architecture (SOA) environment [6] [1] is moving the Air Force environment towards services. Additionally, multiple other efforts are underway to create services providing different aspects necessary to SSA. These include:

- Near-realtime conjunction analysis across 100,000 or more objects [4].
- The new USAFA Center for Space Situational Awareness, which will be implementing an array of sensors, operations center, and associated software, and analysis tools [2].
- The Sensor Exposure, Exploitation and Experimentation Environment (SE4), which provides a platform to illustrate "The Art of the Possible" showing the potential benefit of enriched sensor data collections and real-time data sharing [3].

[1] http://en.wikipedia.org/wiki/Amazon.com, July 20, 2012

This is just a short list of the efforts implementing services that are designed to be provided to the entire SSA community. Once these services are in place, the question on which we focus our research will become apparent. Fig. 1 shows this proposed platform environment, and indicates the problem inherent in this environment, which we discuss in the following section.

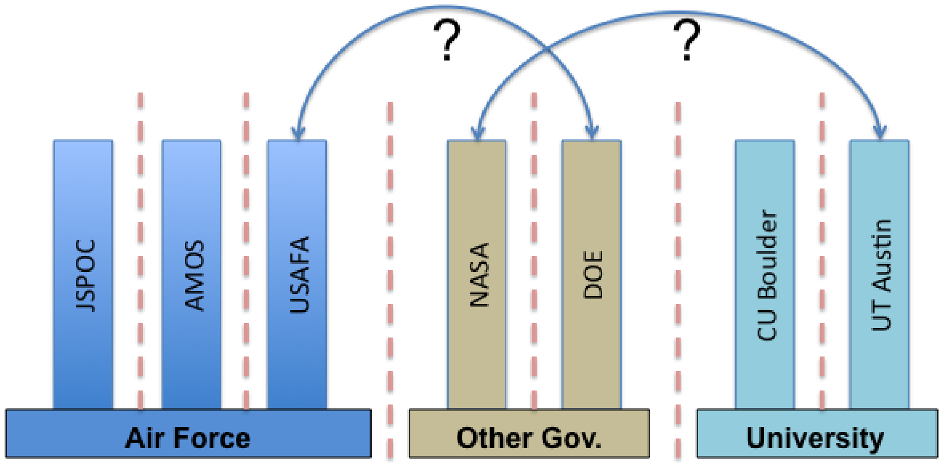

Fig. 1: The Platform-Based Environment

2. THE SSA PLATFORM PROBLEM

Sharing services across organizational boundaries presents visibility and dependency issues. As a consumer of services, what information does an organization need to rely on these external services for mission critical needs? As a service provider, what information does an organization need to manage the load on their infrastructure and control dependencies? In Fig. 1, the questions marks indicate the unmanaged dependence on external organizations.

Fig. 2 demonstrates the problem in a more distinct way. In our example service, the JSPOC is bringing in observations from multiple sensors and creating correlated two-line element (TLE) sets. A set of external organizations (both current and future, known and unknown) want to consume those TLEs to make operational decisions.

As consumer organizations move towards making operational decisions on services provided by external organizations, they must create a service level agreement (SLA). The SLA governs the availability, the load the consumer can place on the provider, and other technical and non-technical parameters. To date, these SLA agreements must be created for each service-use agreement, the technical parameters must be negotiated, and the agreement signed.

While the policy agreements must be put in place to share information between consumer and producer organizations, we demonstrate in our research that the technical parameters for a service use agreement (the technical parameters of the SLA) can be negotiated automatically. The policy agreements to share allow the automatic SUA negotiation, reducing the time to share services and increasing the situational awareness for service providers and service consumers.

3. COMPOSABLE CAPABILITY ON DEMAND® RESEARCH

This research was funded by the MITRE Innovation Program, a MITRE internal research portfolio. Specifically, this research is part of the Composable Capability on Demand® (CCOD) portfolio of projects. CCOD

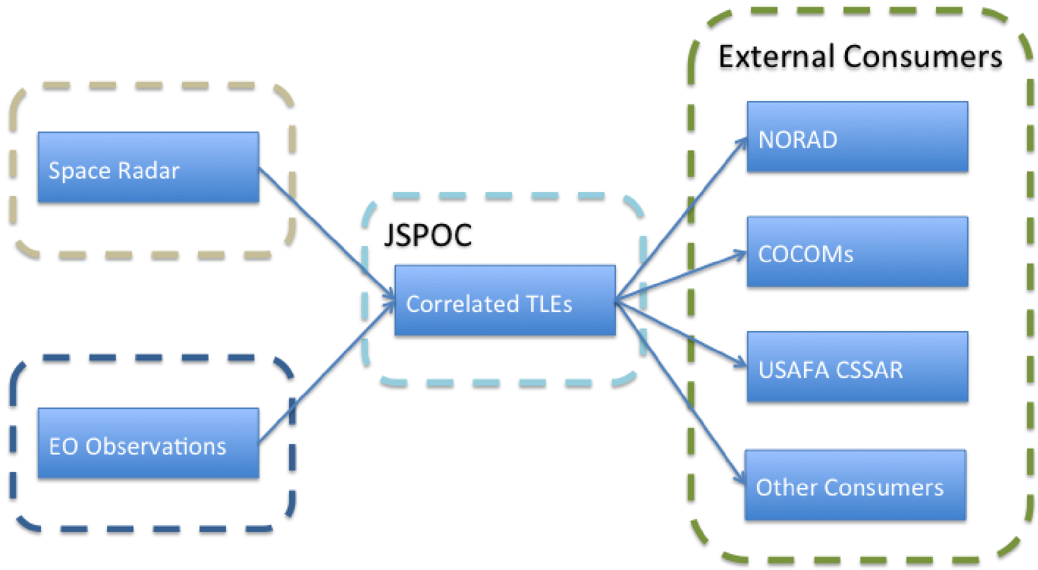

Fig. 2: Cross-Organization Service Use

presents an advancement of the net-centric paradigm, taking the advantages of networked services and sensors and adding a dynamic component to them, the "on demand" aspect. CCOD seeks to demonstrate that the entire technology stack, from the network and operating system level, through services and data sharing, to the user interface can be built in a flexible manner. Proper situational awareness in any environment is not just access to an encyclopedia of information - it is the access to the right information, at the right time, when it supports operational needs. It also speaks to access to the right tools at the right time, and the ability to adapt the tools at runtime. The research presented in this paper is focused on creating composable, flexible service-use agreements at runtime for using services across organizational boundaries, For more information on the broader CCOD portfolio, see [5] and `http://www.mitre.org/work/areas/research`.

4. MANAGING AGGREGATED SERVICES

The Managing Aggregated Services (MASS) research for creating service-use agreements consists of two major parts, the Service Level Agreement metadata (SLA Metadata), which represents the parameters on which the consumer and provider agree, and the MASS toolkit, a set of SOA platform plugins that rely on the SLA metadata.

Our research found that the technical parameters for most SLA agreements are fairly limited, and most agreements could be represented by a combination of the following parameters:

- Capacity and Resource Availability
  - Network Load
  - Server CPU
  - Server Memory Load
- Availability – average uptime of the service
- Query response time by the provider and maximum simultaneous queries from the consumer

The SLA Metadata format is an XML encapsulation of the above parameters. It provides the service provider and consumer a common representation for these parameters.

The SLA Metadata is the foundation for the MASS toolkit to negotiate the service use agreement. Fig. 3 shows the architecture of the MASS toolkit. MASS provides a plugin to the SOA for both the service provider and the service consumer, referred to as the MProvider and the MConsumer.

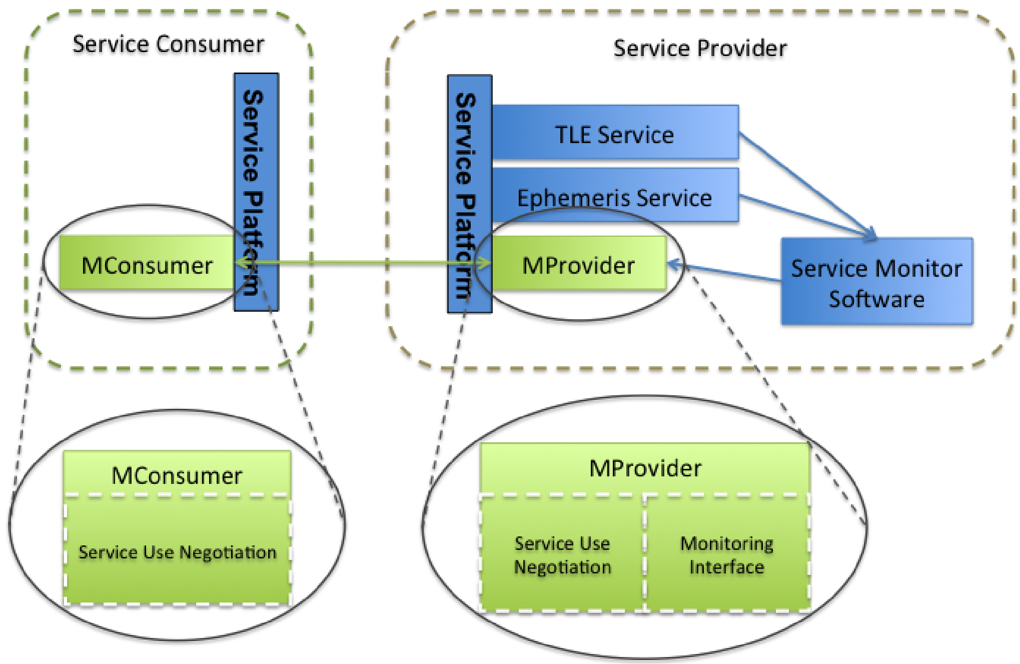

Fig. 3: MASS Toolkit Architecture

The MProvider plugin provides two functions for the service provider. First, it provides a RESTful service interface to collect monitoring information from any commercial or open-source service monitoring software. In our research, we used the open-source Nagios[2] monitoring software to monitor service activity and resources on the service platform. The service monitoring data is stored and used for the negotiation. The second function of the MProvider is to expose the negotiation service that allows a consumer to negotiate a service-use agreement (SUA) with the service provider.

The MConsumer plugin simply provides the ability to negotiate and manage SUA with any number of service providers.

Fig. 4 illustrates the message flow between the service provider and the service consumer to create the service-use agreement. The consumer requests a new SUA from the service provider. The provider returns an SLA Metadata document that represents the default set of parameters to which the provider will always agree. For instance, the provider might default to 500 Kb/second network load, 10% server CPU load, 10% server memory load, 0.999 availability, less than 2 second query response, and up to 10 simultaneous requests at a time by the consumer. These default values are set by the server admin, and will represent an agreement that the service provider is ready to create.

The consumer then sends an SLA Metadata document back to the server that represents the parameters that the consumer desires. The consumer may or may not modify the default values returned by the provider, based on the needs. This modification of the values is again configurable by the administrator of the MConsumer plugin on the consumer SOA platform. Alternately, the MConsumer could automatically modify the parameters based on historical service use monitoring data on the consumer. For instance, if the consumer SOA is collecting historical performance data that demonstrate a general average of 20

[2] http://www.nagios.org/

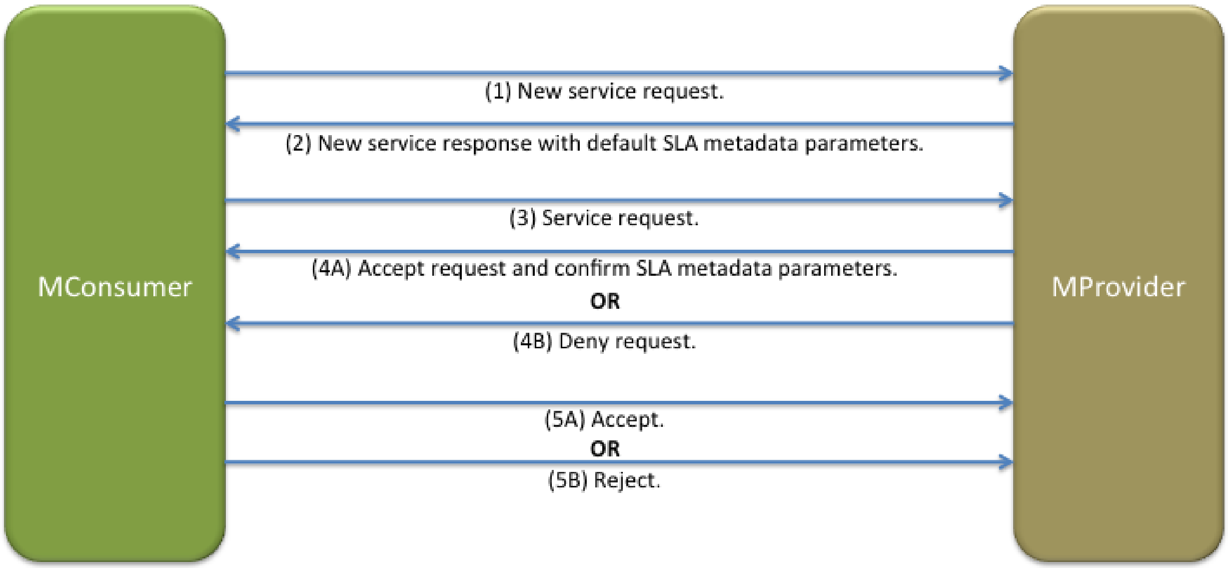

Fig. 4: MASS Service Use Negotiation

requests/second, the MConsumer can automatically request that from the MProvider when it sends the SLA metadata document back to the provider (step 3 in Fig. 4).

If the values submitted back to the MProvider plugin in step 3 are modified, the MProvider plugin on the provider calculates whether the parameters are acceptable, This calculation is performed based on the service monitoring data and admin-configured maximum and minimum values. If the parameters are acceptable, the SUA is created (step 4A), and the consumer can use the service. Alternately, if the consumer is requesting (for example) an SUA that will result in too much load on the network, the provider may reject the SUA (step 4B). On the MProvider, the calculation of the acceptable parameters is automatic, with the accept or reject decision made as a function of the current state of the server, the already negotiated server state, and the admin-set maximum values. For example, consider a new consumer request that contains a request for 10 requests/second. Within the MProvider plugin, the moving average of the service monitoring data indicates that the server is currently loaded at 30 requests/second. Additionally, the MProvider has 6 SUAs in place containing an additive agreement for 65 requests/second. The admin has determined that the server can handle at most 100 requests/second, and has set the maximum allowed value to 90% of that, or 90 requests/second. The MProvider compares the the requested 10 requests/second to the available capacity (25 requests/second), and the current server load (30 requests/second), and approves the request by the consumer. The current server load is considered because if the server is currently overloaded (e.g. over 100 requests/second), the new SUA will be rejected because the server cannot meet any additional requests. The server overload would also indicate to the admin that another consumer is using more resources than their SUA indicates.

Each request from the consumer to the provider is treated as an individual request, and rejected or accepted by the provider. However, we still refer to the whole process as a negotiation, because the consumer can simply re-submit different requests until one is accepted by the server. In every situation, the consumer can always fall back to the original response (step 2 in Fig. 4), as that represents an agreement that the server is currently prepared to make.

Note that all the variables in the above example are treated as static. Research into this area is ongoing. The two major research paths are request prioritization and dynamic use negotiation. Currently, all consumer requests are treated equally, with no priority given to one user over another. As this research continues, the MProvider will be tied into identity and access management, to provide user information to the negotiation. For example, a request for the JSPOC from operational users should be considered more highly than one from a researcher, and this should be flexible based on current events. Under normal operation, both users might be equal, but during a satellite break-up, the JSPOC requests might be made higher priority. Additionally, the MProvider will be modified to base the SUA agreements more heavily on current and historical usage of the server. For instance, if the users of a service are distributed worldwide, and all work normal hours, the load on the service will be distributed across the 24-hour period, and the server should be able to accept SUA that add up to more than 100% utilization. Of course, this presents issues if the usage patterns suddenly change. These remain topics for future research.

5. MASS AND THE SSA PLATFORM

The MASS toolkit described in the previous section provides a number of advantages to moving the SSA community toward not only a service-based architecture within organizations, but to a platform-based architecture, where services are provided across organizational boundaries.

First, MASS can aid information security for service providers. The MProvider plugin (Fig. 3) provides a simple interface to connect an access control solution to MASS, giving the service provider a significant level of control over access by service consumers. Access control for services can be tied to the SUA negotiation, with only those consumers who have an SUA allowed access to the services.

Additionally, the MASS toolkit provides significant situational awareness for the service provider. Not only does it provide details on each consumer connecting to the services, it also provides a database of metrics for each service-to-consumer pairing. This aids in infrastructure management, capacity planning, and overall management for the system administrators.

Finally, the MASS toolkit provides visibility to the service consumer to manage dependence on the external service provider. The consumer has the SUA parameters, and the MProvider plugin provides the consumer service metrics on current activity on the provider SOA. This realtime visibility provides the consumer flexibility to manage their dependence on the service provider.

6. CONCLUSION

As the SSA community moves towards a platform-based architecture, where services from any branch or organization should be available across the enterprise, the ability to communicate monitoring and management information is crucial. For any organization to make mission-critical operational decisions based on services from another organization, they must be able to have visibility into the performance and operation of that service. The MASS toolkit provides the specifications and tools that allow this communication to happen, allowing the community to act as an integrated platform, despite the fact that actual information technology resources are being provided by disparate organizations.

7. REFERENCES

[1] Diane D. Buell, L. Shayn Hawthorne, and James C. Higgins. SSA Building Blocks – Transforming Your Data and Applications into Operational Capability. Proceedings of The Advanced Maui Optical and Space Surveillance Technologies Conference, 2010.

[2] Michael E. Dearborn, Francis K. Chun, Joseph Liu, and Roger D. Tippets. USAF Academy Center for Space Situational Awareness. Proceedings of The Advanced Maui Optical and Space Surveillance Technologies Conference, 2011.

[3] Buell et. al. Sensor Exposure, Exploitation, and Experimentation Environment (SE4). Proceedings of The Advanced Maui Optical and Space Surveillance Technologies Conference, 2011.

[4] Mike Prausa and Norman Facas. An Application of Hadoop and Horizontal Scaling to Conjunction Assessment. Technical Report MTR 100411, The MITRE Corporation, 2010.

[5] J. D. Savage. Composable Capability on Demand (CCOD®): A New Paradigm for the Design, Acquisition and Employment of IT-Based C2. Technical report, ICCRTS, 2010.

[6] Mike Wasson. Space Situational Awareness in the Joint Space Operations Center. Proceedings of The Advanced Maui Optical and Space Surveillance Technologies Conference, 2011.