Joint Space Operations Center Mission System Application Development Environment

Major Rick Luce[1], Captain Paula Reele[1], ChrisSabol[2], Paul Zetocha[3], Juan Echeverry[4], Richard Kim[4], Barbara Golf[5]

[1]Space & Missile Systems Center, Space Superiority Systems Directorate
[2]AFRL/RDSMA, [3]AFRL/RVSVC, [4]Scitor Corporation, [5]Integrity Applications Incorporated

CONFERENCE PAPER

The Joint Space Operations Center (JSpOC) Mission System (JMS) is the program of record tasked with replacing the legacy Space Defense Operations Center (SPADOC) and Astrodynamics Support Workstation (ASW) capabilities by the end of FY2015 as well as providing additional Space Situational Awareness (SSA) and Command and Control (C2) capabilities post-FY2015. To meet the legacy replacement goal, the JMS program is maturing a government Service Oriented Architecture (SOA) infrastructure that supports the integration of mission applications while acquiring mature industry and government mission applications. Future capabilities required by the JSpOC after 2015 will require development of new applications and procedures as well as the exploitation of new SSA data sources.

To support the post FY2015 efforts, the JMS program is partnering with the Air Force Research Laboratory (AFRL) to build a JMS application development environment. The purpose of this environment is to: 1) empower the research & development community, through access to relevant tools and data, to accelerate technology development, 2) allow the JMS program to communicate user capability priorities and requirements to the developer community, 3) provide the JMS program with access to state-of-the-art research, development, and computing capabilities, and 4) support market research efforts by identifying outstanding performers that are available to shepherd into the formal transition process.

The application development environment will consist of both unclassified and classified environments that can be accessed over common networks (including the Internet) to provide software developers, scientists, and engineers everything they need (e.g., building block JMS services, modeling and simulation tools, relevant test scenarios, documentation, data sources, user priorities/requirements, and SOA integration tools) to develop and test mission applications. The developed applications will be exercised in these relevant environments with representative data sets to help bridge the gap between development and integration into the operational JMS enterprise.

1. JMS ACQUISITION IMPERATIVES

The JMS program is working to bring new SSA tools to the user in a much more agile manner than the current acquisition process allows. This will enable timely deployment of capabilities to the user in order to be more responsive to space events. The current commander of Air Force Space Command, General William L. Shelton, stated [1]:

"The JMS program will have a huge impact on just about everything we do in space. Acting as the hub, JMS will revolutionize Space Situational Awareness capabilities, taking inputs from a huge variety of radar and optical, ground- and space-based, space weather, and many other types of sensors. JMS is a great example of how an industrial age acquisition system just isn't agile enough for an information age program. The system is too slow, too stodgy, and the requirements it places on program developers are too cumbersome. Streamlined acquisition requires everyone to streamline their expectations and process."

Colonel Mike Wasson, Chief, Combat Operations Division, 614th Air and Space Operations Center, stated [2]:

"The JSpOC requires highly responsive SSA capabilities that rapidly detect, track, and characterize objects in space. As such, new developments in SSA tools and capability to assess and respond to events in space are imperatives for the future."

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
SEP 2012		**00-00-2012 to 00-00-2012**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Joint Space Operations Center Mission System Application Development Environment	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Space & Missile Systems Center,Space Superiority Systems Directorate,Los Angeles AFB,El Segundo,CA,90245**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
In Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS), 11-14 Sep 2012, Maui, HI.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**Same as Report (SAR)**	**10**	

To address both of these statements, AFRL and the JMS program are proposing the development of an application development environment that would give future capability developers access to a common environment as a step in achieving a streamlined acquisition process for new SSA tools. The environment will help facilitate the development of high-impact applications in a timely manner as well as help bridge the gap between development and integration into the operational JMS enterprise by exercising the developed applications in a relevant environment with representative data sets.

2. HIGH-LEVEL JMS ACQUISITION APPROACH

The JMS is designed to deliver an integrated, net-centric SSA and C2 capability, with a space User Defined Operational Picture (UDOP) and mission services which support and enable the missions of the Commander Joint Forces Component Command for Space (CDR JFCC SPACE). CDR JFCC SPACE requires the ability to protect space capabilities supporting US, allied, and coalition operations. This requires operationally-relevant SSA and integrated C2 of Space Control capabilities.

The JMS program is being executed by the Space and Missile Systems Center Space Superiority Systems Directorate (SMC/SY) for the Program Executive Officer for Space (PEO Space). The JMS program is leveraging investments in existing government prototype efforts and existing industry applications to rapidly deliver needed capabilities. It will deliver capabilities to get off legacy SPADOC and ASW through two increments which leverage these mature capabilities by the end of FY14. Once this is accomplished, the JMS system will have the necessary capabilities and framework upon which additional SOA-compliant capabilities can be easily added in the Post-Increment 2 time period. The JMS program office will act as the system architect and as the integrator and maintainer of the JMS system baseline.

Increment 1 (2011-2012): Increment 1 leverages capabilities that have already been developed. It builds on the AFRL JMS prototype known as Capability Package 0 (CP0). CP0 was developed, tested, and deployed in close cooperation with the JSpOC user community. It provides automated links to existing data sources and includes a UDOP to integrate and display the information. CP0 also provides the baseline JMS SOA. The CP0 prototype has been incrementally improved through Service Packs (SPs). SPs aggregate applications of capability developed, integrated, tested, and deployed in approximately 3-month cycles with odd numbered SP's delivering new capability and even numbered SP's largely containing only IA updates. CP0 plus SPs 1 through 5 complete Increment 1, which will satisfy two of the five JMS Capability Description Document (CDD) Key Performance Parameters (KPP), namely UDOP and Net-centricity.

Increment 2 (2012-2015): Increment 2 will build upon the Increment 1 baseline, satisfy the remaining 3 JMS CDD KPPs (Space Catalog, Orbital Conjunction, and Ops Availability) and allow decommissioning of the legacy SPADOC hardware by the end of FY 2015. Increment 2 integration, test, and sustainment will be managed by the Space and Naval Warfare Systems Command (SPAWAR) San Diego as part of the JMS program office. The content of Increment 2, number and type of contracts used to acquire Increment 2 services for integration by SPAWAR, and timing of Increment 2 SP releases will be determined based on Increment 1 capability, program lessons learned, and maturity of industry and government applications.

Post-Increment 2 (2015+): At the completion of Increment 2, JMS will have delivered a flexible, extensible SOA framework with foundational services (such as catalog and conjunction services) upon which additional SSA capabilities can be added. During this phase, the JMS program office will rely on the JSpOC user and the Requirements & Planning Council (R&PC) co-chairs (CDR JFCC SPACE and PEO Space) to establish and continually refine capability priorities. The JMS program will communicate these priorities to the application development community with regular updates to focus application developments on the most important user needs. SPAWAR will continue as part of the JMS program office in the role of software integration and sustainment of the JMS baseline.

The buildup of the application development environment will begin during Increment 2. Existing projects like Ibex (a joint DARPA/AFSPC effort leveraging expertise at AFRL and MIT LL) and space weather services can take advantage of this initial buildup to help the teams build, test and collaborate from different parts of the country. The JMS program will work with AFRL on this initial infrastructure to turn the initial Ibex development environment into a worldwide asset for SSA tool development by the time JMS enters the Post-Increment 2 time period.

3. BASELINE FRAMEWORK

JMS SOA: The JMS program completed an assessment using inputs from Industry, SPAWAR, and multiple Federally Funded Research and Development Centers (FFRDCs) which concluded that the current CP0 SOA architecture provides a suitable, low-risk solution on which to build JMS. The PEO SPACE signed the JMS SOA Decision Report on 20 Jan 2012 to formalize the SOA framework for future Increments of JMS.

SOAs are built as loosely-coupled systems consisting of infrastructure components and applications that are orchestrated to provide defined capabilities. Typically, SOAs are designed using standard commercial off the shelf/free and open source software (COTS/FOSS) products and are tailored to specific mission needs. The key components of a standard SOA design are: the enterprise service bus, the service registry and repository, service orchestration, event management, data management, and security (Fig 1). Each of these components is tailored to support mission specific requirements. For example, to meet particular mission requirements a system may require unique data and service models or may have unique security needs.

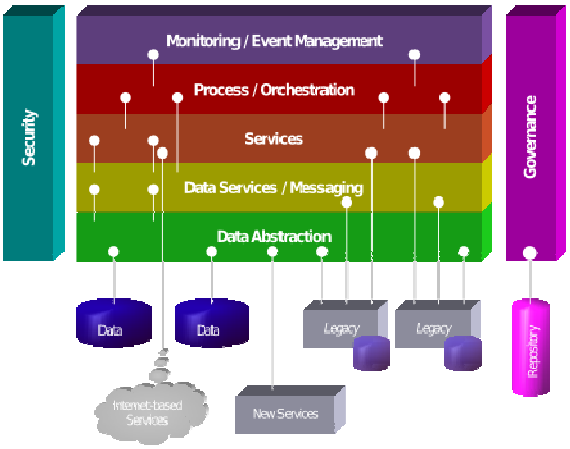

Fig. 1, SOA Meta-Model, The Linthicum Group, 2007 [3]

The JMS infrastructure provides a suite of components and implements a set of standards to support mission services. To effectively integrate into the SOA infrastructure, an application developer must understand various components of the infrastructure. The following is a list of the JMS infrastructure components:

- Security
- Messaging
- Collaboration Support
- Enterprise Service Bus & Java Enterprise Edition (JEE) Application Server
- Data Management
- UDOP Framework
- General Support and Components

JMS UDOP: The JMS UDOP is an application authoring tool that combines data with process and display components to enable the creation and operation of application views. The base implementation of the UDOP provides a set of generic components for visualization, process control, and data access.

Each JMS mission application must interface with the UDOP to display information to the user. Fig 2 shows the current welcome screen for the JMS UDOP. From here, a user can access a wealth of applications for the various JSpOC mission areas.

Fig. 2, JMS UDOP Welcome Perspective

Application developers may add to the list of UDOP components that are available by building custom visualization, process, and data access plug-ins. Detailed standards for integration and best practices regarding the development of 3rd party UDOP plug-ins are outlined within the JMS UDOP software development kit (SDK). This document will be made available to application developers through the development environment.

When integrating with the JMS UDOP, the application developer defines a UDOP Document. The UDOP Document is a definition document that contains the structure of the visualization components, the data sources available, and the process information used to tie the data to the visualization components. The document may also contain images, static data, historical data, or other resources. The visualization components within a UDOP Document are grouped into one or more perspectives. A perspective is a collection of visual components that are displayed together. The visual components contained by a perspective are views, menus, and toolbars. A view is a component used to visualize and navigate a collection of data. Multiple views may be used to provide different visualization and navigation for the same collection of data.

The UDOP Authoring Tool, shown on the left of Fig 3, is an example of a development tool that will be available to users of the development environment. The tree entry for the Space Order of Battle perspective is expanded in the figure (on the left) as an example. The child nodes under the Space Order of Battle entry identify the gadgets and views placed in the perspective. The components shown on the left of Fig 3 trace to the example of the Space Order of Battle perspective shown on the right side of the figure.

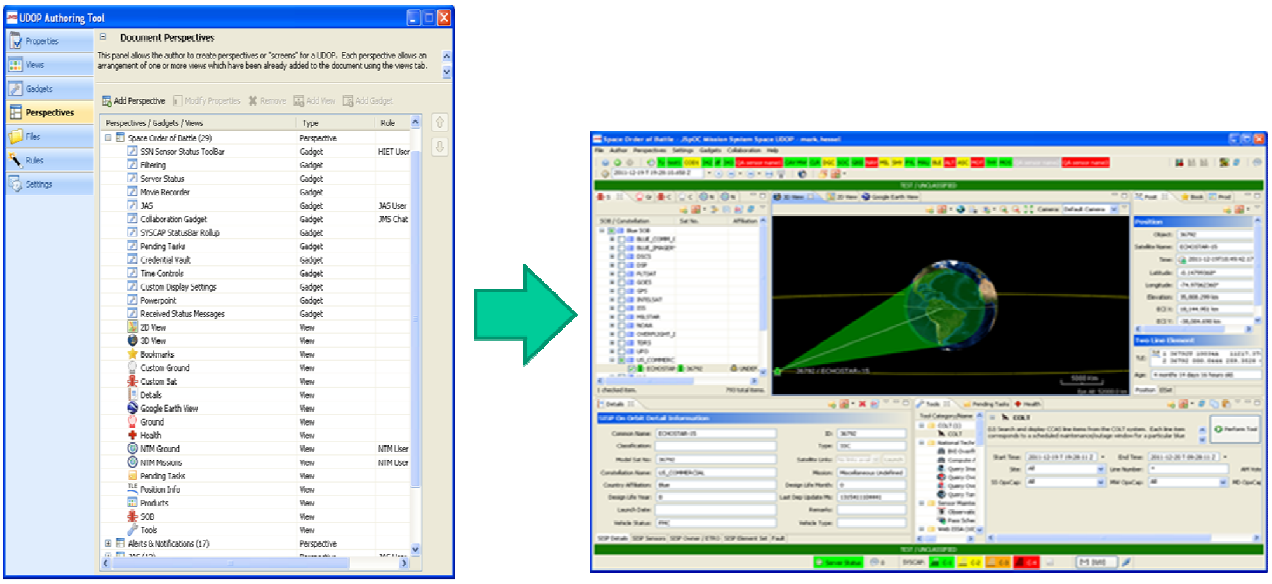

Fig. 3, JMS UDOP Authoring Tool and Space Order of Battle Perspective

4. KEY ELEMENTS OF THE APPLICATION DEVELOPMENT ENVIRONMENT

Overview: AFRL and the JMS program are partnering to develop an application development environment on both unclassified and classified networks. The end state of this environment will give various user groups (e.g., software developers, SSA experimenters, and JSpOC users) the documentation, building block JMS services, modeling and simulation tools, relevant test scenarios, data sources, JMS user requirements/priorities, and SOA integration tools required to support JMS development and integration. The environment will also provide these user groups with a framework that enables and encourages multi-organizational collaboration. This environment is envisioned as a means for the JMS program to become a more agile acquisition system that accelerates the delivery of prototypes into SPs.

Cloud Architecture Services: There is no single definition for a Cloud Architecture. Although this environment is not being implemented in a manner akin to well-known computing clouds, such as Amazon's Elastic Compute Cloud or Google's Compute Engine, the envisioned development environment will provide the four classic services usually defined in cloud architectures (as shown in Fig 4). These include:

- ***Infrastructure as a Service:*** JMS application development environment will provide computational storage and networking resources available on demand
- ***Platform as a Service:*** JMS application development environment will provide the JMS SOA and SDK; AFRL will investigate methods to optimally integrate SOA technologies with Cloud technology
- ***Software as a Service:*** JMS application development environment will provide astrodynamic modeling and simulation tools for SSA experiments
- ***Data as a Service:*** JMS application development environment will provide both canned and real data sources to application developers and SSA experimenters formatted according to data standards as defined by the JMS common data model

The JMS development environment will provide some or all of these services as applicable to each type of user. User profiles will catalog each user attribute according to the type of service required.

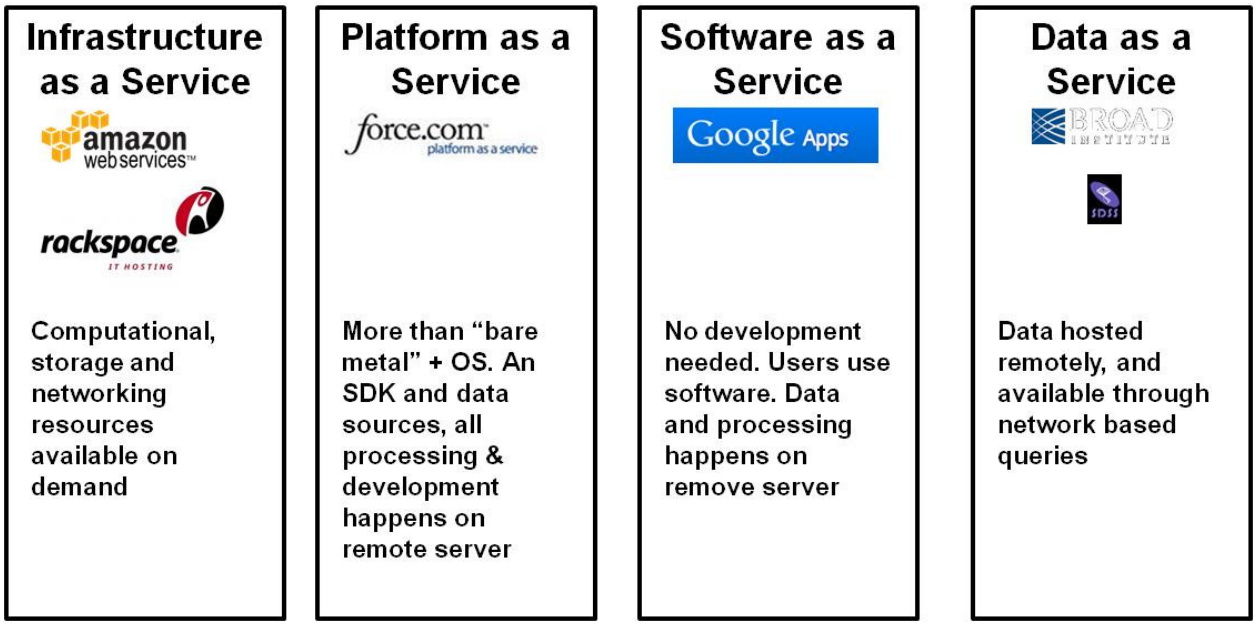

Fig. 4, Service Layers to be considered for the JMS application development environment

Unclassified Application Development Environment Foundation: AFRL is making upgrades and expansions to existing network environments as the foundation of the JMS application development environment. For the unclassified environment, which is critical to the application developer user segment, JMS will leverage the existing AFRL/RD High Performance Computing Modernization Program (HPCMP) Portal capability and JMS technologies developed within the AFRL/RV Battlespace Evaluation and Assessment SSA Testbed (BEAST). The HPCMP portal capability is easily accessible through the Internet, leverages existing hardware, account management, and user support services, and publishes a downgraded form of the existing JMS software environment. It currently uses PKI-CAC and Yubikey authentication, which allows access to a wide community of users. As shown in Fig 5, the Portal allows a user to access the Maui High Performance Computing (HPC) via the Internet and web browser with no additional user-side requirements.

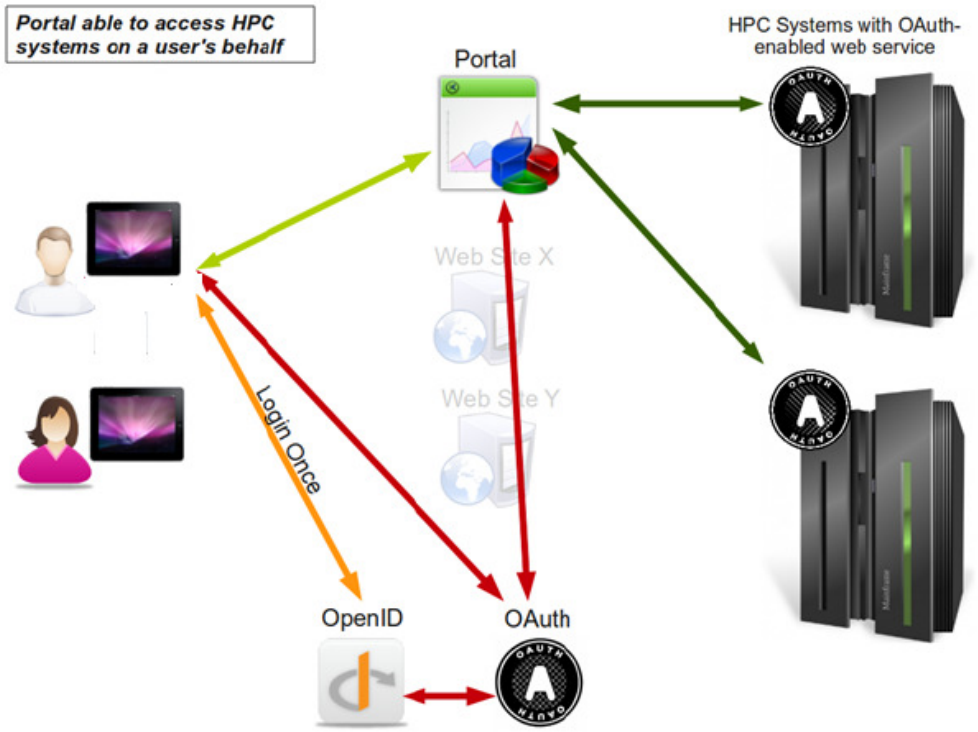

Fig. 5, HPCMP Portal Access from Internet to HPC Network

The goal of the HPCMP Portal program (separate from the JMS program) is to transform DoD High Performance Computing (HPC) from stand-alone to cloud-based. The program contains two software product areas in the initial deployment, namely computational Research & Engineering Acquisition Tools and Environments (CREATE) Kestrel and Distributed Matlab. The Portal offers the following benefits:

a) Eliminates traditional HPC application stove-pipes making application and data sharable world-wide on-demand
b) Does not require HPC knowledge or traditional hurdles to fully exploit HPC
c) Does not require installed software, thus eliminating unique user configuration and maintenance
d) Provides a simplified and enhanced security model
e) Provides immediate access to HPC resources using fieldable, low power, portable devices such as tablets

Fig. 6 shows the current application suite startup panel which includes Kestrel, CREATE Job Manager, Distributed Matlab, and Virtual Applications.

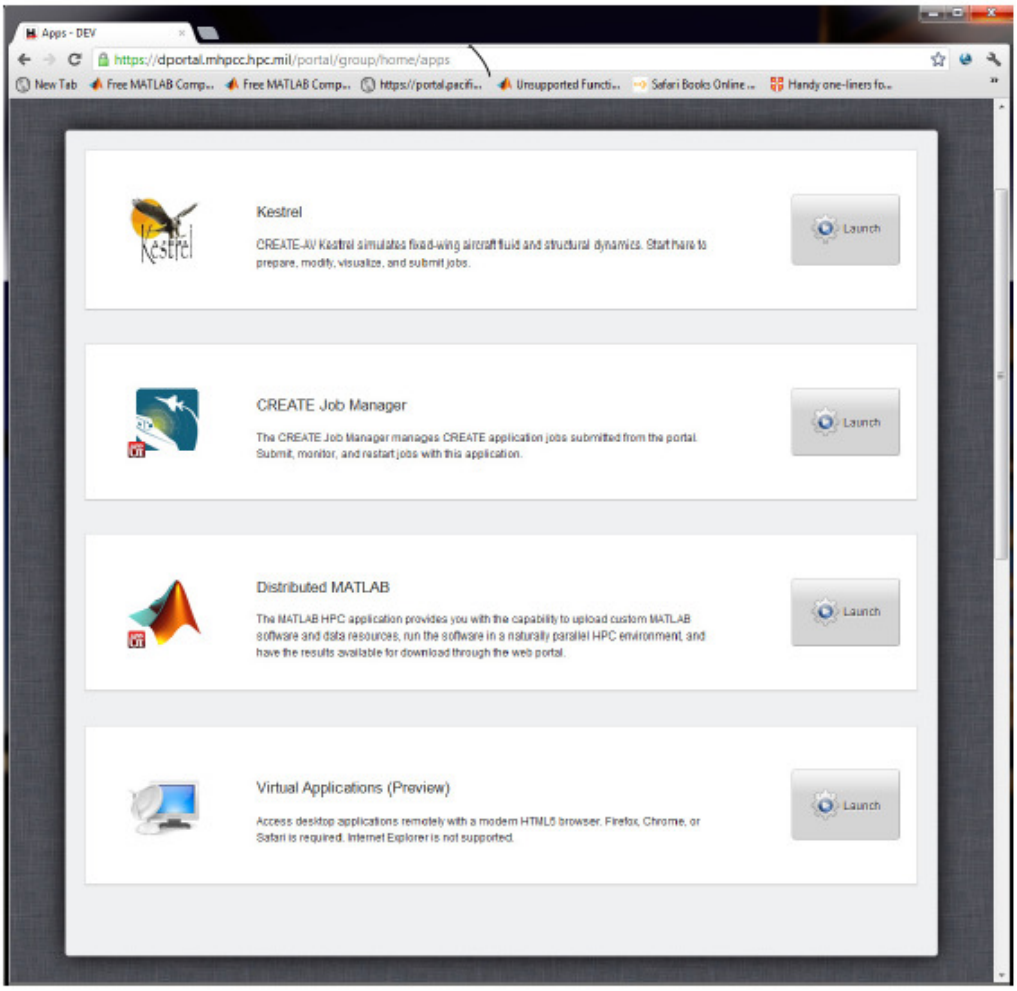

Fig. 6, Current HPCMP Portal Startup Panel

The unclassified development environment will build upon the HPCMP Portal infrastructure. A partition will be made on the HPC to allow JMS to have a unique startup panel, a SOA environment to check for application compatibility with JMS, a variety of modeling and simulation tools and data sources, etc.

JMS Application Development Environment Network: The JMS program intends to open the unclassified environment to a broad range of users by putting appropriate access controls in place. These controls will provide access to user groups that range from large industry to rapid prototyping shops in academia, government, and small business (as shown in Fig 7) and will protect the intellectual property rights of these users. The controls will also give JMS personnel the accesses necessary to review prototypes while in the development and integration phases. This review will support JMS market research efforts and allow the JSpOC operators to get early exposure to development applications as well as give government experts the opportunity to provide feedback to the developer on usability, the viability of the processing algorithms, scalability to relevant or operationally-sized data feeds, and more. Although this broadened field of participants will necessitate stringent security measures, this framework minimizes security challenges for developers in early phases which, in turn, provides the JMS program with the opportunity to access more state-of-the-art capabilities and improve market research efforts, and provides the developers with the opportunity to create strawman prototypes in a sandbox environment.

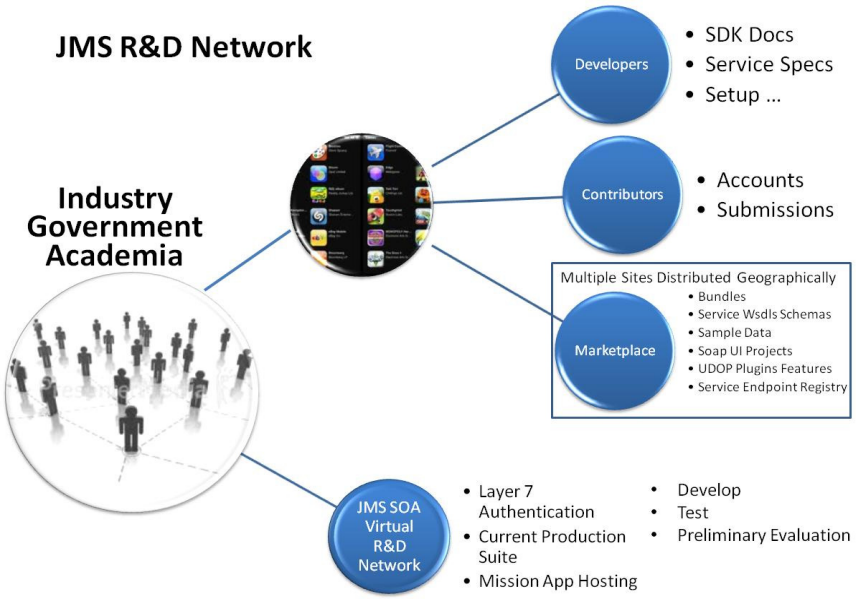

Fig. 7, Features of the JMS Application Development Environment on a Network

Transition to Operations: The mission application path to transition will incorporate a phased progression from unclassified development to operations. If a strawman application is flagged as a viable candidate for JMS inclusion, the application will be migrated (after appropriate security checks and contracting requirements are completed) to the classified application development environment (Fig 8). This enclave would provide the application developer access to canned, classified data and other test sources, eventually upgrading to live data feeds (as needed). Historical data may be provided for relevant scenarios with modeling and simulation data that mimics the cadence of existing collection systems. The classified development environment will also contain a robust image of the operational JMS SOA to allow application developers to measure improvement against the baseline.

The downside to this migration approach is that there are multiple development environments to develop, deploy, and maintain. However, the transition is believed to be the most agile means to acquire and deploy mission applications as well as the best strategy to mobilize the broadest possible community of developers and experimenters. Accessibility is a fundamental strength of this approach.

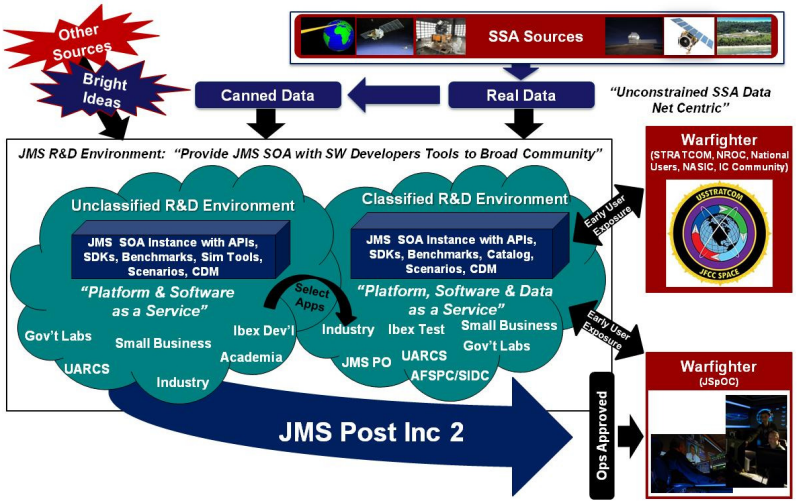

Fig. 8, Unclassified - Classified JMS Application Development Environment Enables Responsive Acquisition Process

5. ACHIEVING RESPONSIVE CAPABILITY DELIVERY TO JMS

Industry and government entities have clearly identified the need for a common, relevant development environment to support mission application development for the JMS user. Through access to relevant tools and data, JMS will empower the development community to accelerate technology development, define a new business case, lower costs, and identify development areas that support user-identified priorities/requirements. The JMS application development environment will encourage participation by niche application developers and small businesses by removing the need for developers to invest in duplicative, stand-alone infrastructure and facilitating access to test data sources. By providing JMS integration process requirements early in the development process and enabling successful integration into the SOA framework as soon as possible, the environment will flatten the application developers' learning curve.

From the perspective of the end users and SSA experimenters, the JMS application development environment will allow JSpOC operators to get early exposure to development applications via UDOP-compliant plug-ins that may address needs identified in exercises such as Global Lightning/Terminal Fury. This would enable early operator feedback to code developers for prototype applications.

Once an application progresses through the utility testing enabled by the development environment, it can finalize its path to SOA integration by entering the SPAWAR JMS integration gating processes and eventually Operational Test & Evaluation. If used properly, the environment will empower users to develop high-impact, high-TRL applications in a timely manner as well as help bridge the gap between development and integration into the operational JMS enterprise by exercising the developed applications in a relevant environment with representative data sets.

6. DEPLOYING THE JMS APPLICATION DEVELOPMENT ENVIRONMENT

AFRL/RD and AFRL/RV are moving out with the first phase of the unclassified and classified environments. The initial customer will be the Ibex 2.0 team who are maturing the Ibex program tools and framework in FY13/14. The development environments will be proved out by this initial set of scientists, engineers, and software developers. This group will help shape the user interfaces on these environments to prepare for a distributed launch to a broader user group by the end of 2015. Once the JMS application development environment is launched to the broader user group, the JMS program office will be working closely with AFRL to maintain insight into the progress of these application developments to identify applications for integration into the operational JMS.

7. REFERENCES

1. Shelton, William L., General, Air Force Space Command Commander, 2012 Armed Forces Communications and Electronics Association Cyberspace 2012 Symposium Remarks on JMS, Colorado Springs, Colorado, 7 Feb 2012.
2. Wasson, Mike, Colonel, Chief, Combat Operations Division, 614th Air and Space Operations Center, 2011 AMOS Conference paper, *Space Situational Awareness in the Joint Space Operations Center*, 12 Sep 2011.
3. Linthicum, David, *Steps to SOA Success*, 2007.