Sensor Webs with a Service-Oriented Architecture for On-demand Science Products

Daniel Mandl[a], Rob Sohlberg[b], Chris Justice[b], Stephen Ungar[a], Troy Ames[a], Stuart Frye[c], Steve Chien[d], Daniel Tran[d], Patrice Cappelaere[e],Linda Derezinski[f], Granville Paules[g], Don Sullivan[h], Liping Di[i], Stephan Kolitz[j]

[a]NASA/GSFC, Greenbelt, MD, 20771,U.S.
[b]Univ. of Maryland, College Park, MD, 20740, U.S.
[c]Noblis, Falls Church, VA, 22042 U.S.
[d]JPL, Pasadena, CA, 91104 U.S.
[e]Vightel, Ellicott City, MD,21043 U.S.
[f]Innovative Solutions, Essex, MD, 21221 U.S.
[g]Kelly, Anderson & Assoc., Alexandria, VA, 22314 U.S.
[h]NASA/ARC, Moffet Field, CA, 94035 U.S.
[i]George Mason Univ., Fairfax, VA, 22030 U.S.
[j]Draper Lab, Cambridge, MA, 02139 U.S.

ABSTRACT

This paper describes the work being managed by the NASA Goddard Space Flight Center (GSFC) Information System Division (ISD) under a NASA Earth Science Technology Office (ESTO) Advanced Information System Technology (AIST) grant to develop a modular sensor web architecture which enables discovery of sensors and workflows that can create customized science via a high-level service-oriented architecture based on Open Geospatial Consortium (OGC) Sensor Web Enablement (SWE) web service standards. These capabilities serve as a prototype to a user-centric architecture for Global Earth Observing System of Systems (GEOSS). This work builds and extends previous sensor web efforts conducted at NASA/GSFC using the Earth Observing 1 (EO-1) satellite and other low-earth orbiting satellites.

Keywords: sensor webs, OGC, on-demand science, service-oriented architecture

1. INTRODUCTION

A sensor web is a set of disparate sensors tied together with a communications fabric that are controlled in such a way as to form a cohesive whole. Figure 1 depicts our vision for a user-centric sensor architecture that facilitates the vision for GEOSS. Note that the essential features of this architecture are the encapsulation of sensors and data processing algorithms to provide easy access to the users over the Internet via web services.

Sensors are encapsulated as sensor data nodes with Sensor Markup Language (SensorML) web accessible documents that are XML-based descriptions of the capabilities of the sensor. The capabilities may include location, resolution, spectral bands, swath and how to task the sensor. Furthermore, data processing algorithms are encapsulated in data processing nodes with SensorML or similar web accessible documents that describe what the algorithms do. These descriptions may include inputs, outputs, methods employed by the algorithm and how to invoke the algorithm for user data. In both cases, the web accessible documents are created so that information about the sensors and algorithms can be discovered over the Internet and provide information on to how to access the sensors and algorithms. The user can then assemble sensor data and selected algorithms into a customized workflow automatically, which includes automatic electronic delivery of data products to the users' computer desktop, thus enabling on-demand science products.

The key challenge to this approach is managing the workflows, which produce the science products. This can involve a variety of challenges such as; (1) discovery of an existing workflow, (2) the creation of a new workflow, (3) the modification of an existing workflow and (4) the execution of a workflow. Another challenge is finding the status of the components in the workflow, in light of the fact that the sensors and algorithms are distributed over the Internet and may or may not be operational at the time that the user attempts to invoke a particular workflow.
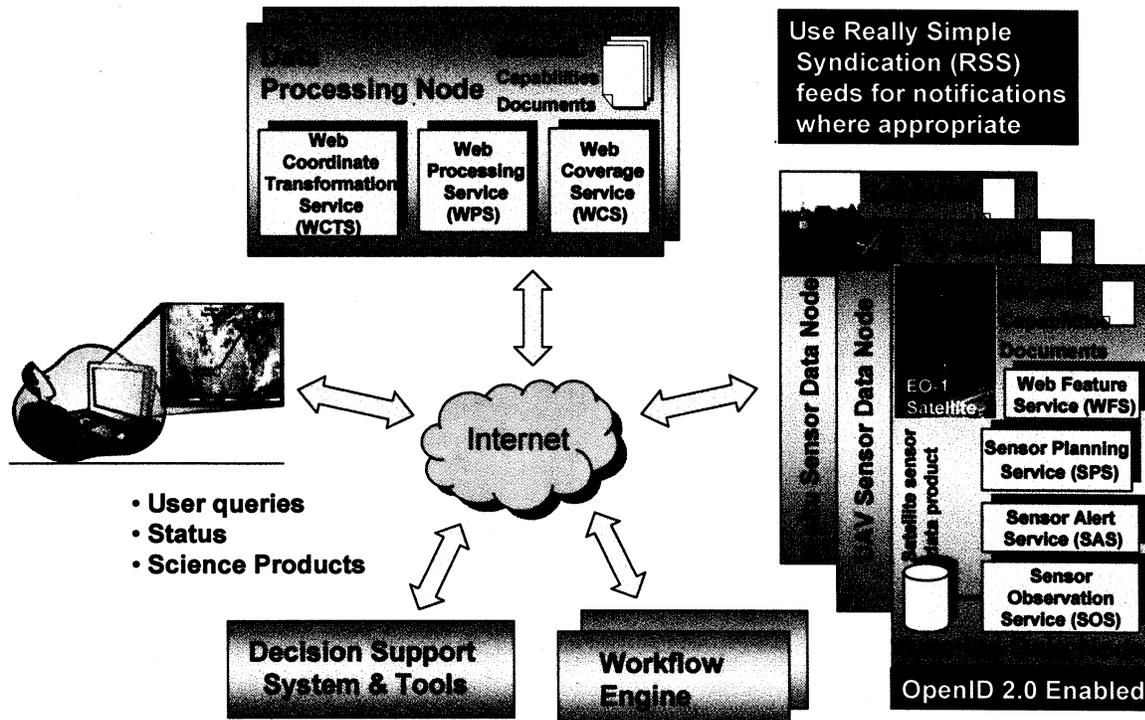
Figure 1 User-centric sensor web reference architecture

2. BACKGROUND

Our sensor web experiments began with the EO-1 fire sensor web experiment that was conducted in August 2003[1]. In this experiment, the MODIS instrument on the Terra satellite was used to survey major national wild fires as cataloged by the National Interagency Fire Center (NIFC) and then autonomously triggered a high resolution EO-1 image of the specified wild fire using a latitude/longitude of the hot spot pixels located and identified by the MODIS instrument. Later experiments included a volcano sensor web experiment[2] in which various autonomous triggers could invoke EO-1 high-resolution images, which included MODIS, insitu sensors and even triggers from science processing, occurring onboard EO-1, such as the detection of hot pixels. There were also additional experiments with floods, ice and dust detection.

Presently, we are in the process of upgrading our sensor web architecture with the web-enabled OGC Sensor Web Enablement (SWE) suite along with some Web 2.0 capabilities. Web 2.0 represents next generation technology for the Internet.

3. WEB 2.0 AND OGC STANDARDS INTEGRATION

The implementation of GEOSS will be influenced by the emergence of Web 2.0 and OGC SWE standards. These are not independent activities in that Web 2.0 capabilities will be used to implement some of the OGC SWE standards. In fact, that is one of the key goals of our activity.

Web 2.0 is commonly understood as a transition of websites from isolated information silos to sources of content and functionality, thus becoming computing platforms serving web applications to end-users. It is also a social phenomenon embracing an approach to generating and distributing Web content itself, characterized by open communication, decentralization of authority, freedom to share and re-use, and "the market as a conversation". It is also characterized by enhanced organization and categorization of web information content, emphasizing deep linking. Finally, Web 2.0 is characterized by a rise in the economic value of the Web, which in the science world, translates to more cost-effective and responsive science products. Earlier users of the phrase "Web 2.0" employed it as a synonym for "Semantic Web", and the two concepts do complement each other [3].

The key Web 2.0 capabilities that we plan to implement in our demonstrations are Really Simple Syndication (RSS), a method to publish frequently updated digital content in a news feed format and OpenID 2.0, an open, decentralized, free framework for user-centric digital identity. For the RSS capability, we are trying to use geoRSS, which is Geographically Encoded Objects for RSS feeds.

The OGC SWE standards that we are using enable the following capabilities:

- Discovery
- Standard data access
- Standard tasking
- Standard alerts

In particular we are using the following OGC SWE services:

- Sensor Planning Service (SPS) – is a standard web service interface for requesting sensor acquisitions and observations. This is an intermediary interface between a user and a sensor management system.

- Sensor Alert Service (SAS) – is a standard web service interface for publishing and subscribing to alerts from various sensors.

- Sensor Observation Service (SOS) – is a standard web service interface for requesting, subsets of data produced by selected sensors.

We are also using additional OGC standard services as follows:

- Web Feature Service (WFS) - is a standard web service to allow requests for geographical features across the Internet using platform-independent calls.

- Web Processing Service (WPS) - is a standard web service that takes a defined set of geospatially referenced inputs, applies a specific calculation, defined by its owner, and produces a defined set of outputs.

- Web Coverage Service (WCS) – is a standard web service for exchanging geospatial data. WCS provides available data together with their detailed descriptions; allows complex queries against these data; and returns data with its original semantics (instead of pictures), which can be interpreted, extrapolated, etc.

- Web Map Services (WMS) – is a standard web service, which produces a digital image file and is often used to display data produced by a WCS on a map.

Our demonstrations and experiments will use the above-mentioned capabilities and services for as many sensor assets as we are able to integrate into our experiments. Our primary sensors are the Advanced Land Imager (ALI), a multispectral imager, and the Hyperion, a hyperspectral imager, on EO-1. We have the most control over the interfaces to this satellite and its sensor since the primary author is also the EO-1 mission manager. For other sensors, we are collaborating with other groups, such as the ASTER instrument team, and therefore are often limited in how much of the above capabilities we can implement. In the next section, the target demonstration for the first year of our ESTO effort is discussed.

4. TARGET FIRST YEAR DEMONSTRATION

Figure 2 depicts a portion of our first year demonstration scenario. This figure depicts a workflow for processing EO-1 data into classified hot pixels and then combining the pixels with a map for delivery to a user. As of July 2007, the

majority of this workflow has been successfully demonstrated. Note that a Business Processing Execution Language (BPEL) engine, supplied by George Mason University, is used to drive the workflow. The workflow transforms the EO-1 downlinked Hyperion data through a number of steps to create the JPEG fire map. The steps involved are Level 0 processing, Level 1 Georeferenced (Level 1G) processing, hot pixels classification, transformation of the hot pixels into a JPEG and finally combining that JPEG with a map to create the composite fire map.
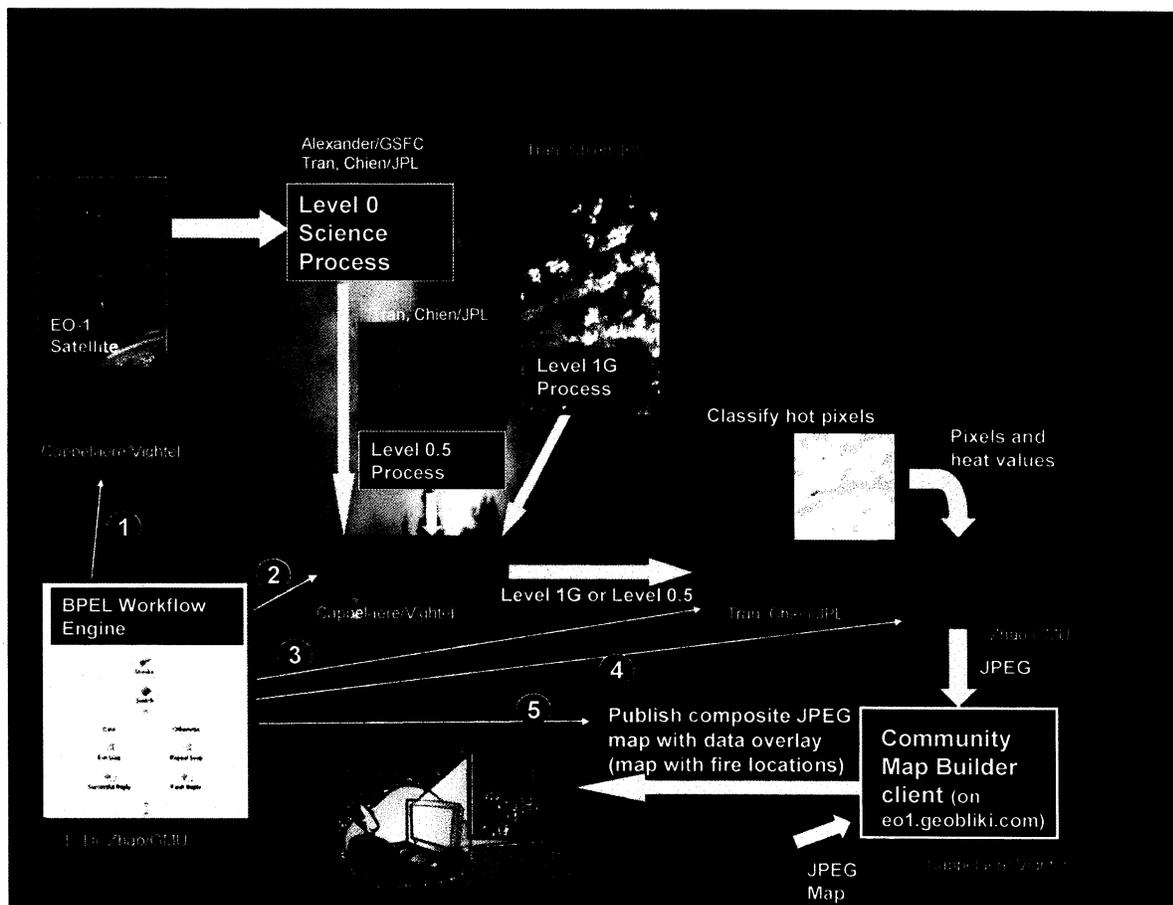
Figure 2 EO-1 workflow that automatically creates a fire map

This workflow will be integrated into the larger demonstration scenario depicted in Figure 3. To accomplish this demonstration, we are collaborating with the U.S. Forest Service, the NASA Ames Research Center (ARC), the National Interagency Fire Center (NIFC), the Air Force Weather Agency (AFWA), Draper Lab, George Mason Univ., University of Washington at St. Louis, Univ. of Utah and NASA's Terra, Aqua and EO-1 satellite teams. In this composite workflow, triggers are received from the National Interagency Fire Center (NIFC) ICS209 database of important national fires, the MODIS active fire map and the sensor on the Ikahana Unmanned Aerial System (UAS). The decision support system then examines the available sensors for the feasibility of tasking those sensors and then tasks the appropriate sensors as needed automatically. This includes the retrieval of data from ground sensors such as the Remote Automated Weather Stations (RAWS)

One of the key workflows in this composite demonstration is the workflow for the Unmanned Aerial System (UAS) that is being flow under the Wildfire Research Applications Program (WRAP) at the NASA Ames Research Center (ARC),

led by Vince Ambrosia. Figure 4 depicts the expanded UAS workflow that we are working towards. Don Sullivan, at NASA ARC built the OGC compliant interfaces that enable users to modify the flight plan of the UAS via an SPS. In the preliminary demonstrations, users can request an image acquisition, via the SPS, using the planned configuration for the flight and its current travel position plan. However, future enhancements will enable users to reconfigure the UAS and replan the acquisitions. In addition, additional enhancements will even enable a user to alter the original travel position plan via the same SPS.

In addition to the SPS interface, a set of WCS, WFS and WMS interfaces were created to retrieve and distribute acquisition data from the UAS. Note that the target tools to produce the final science products are open source. The plan is to use User friendly Desktop Geographical Information System (Udig) and Google Earth. This exemplifies how this architecture greatly reduces the cost to produce customized science products.
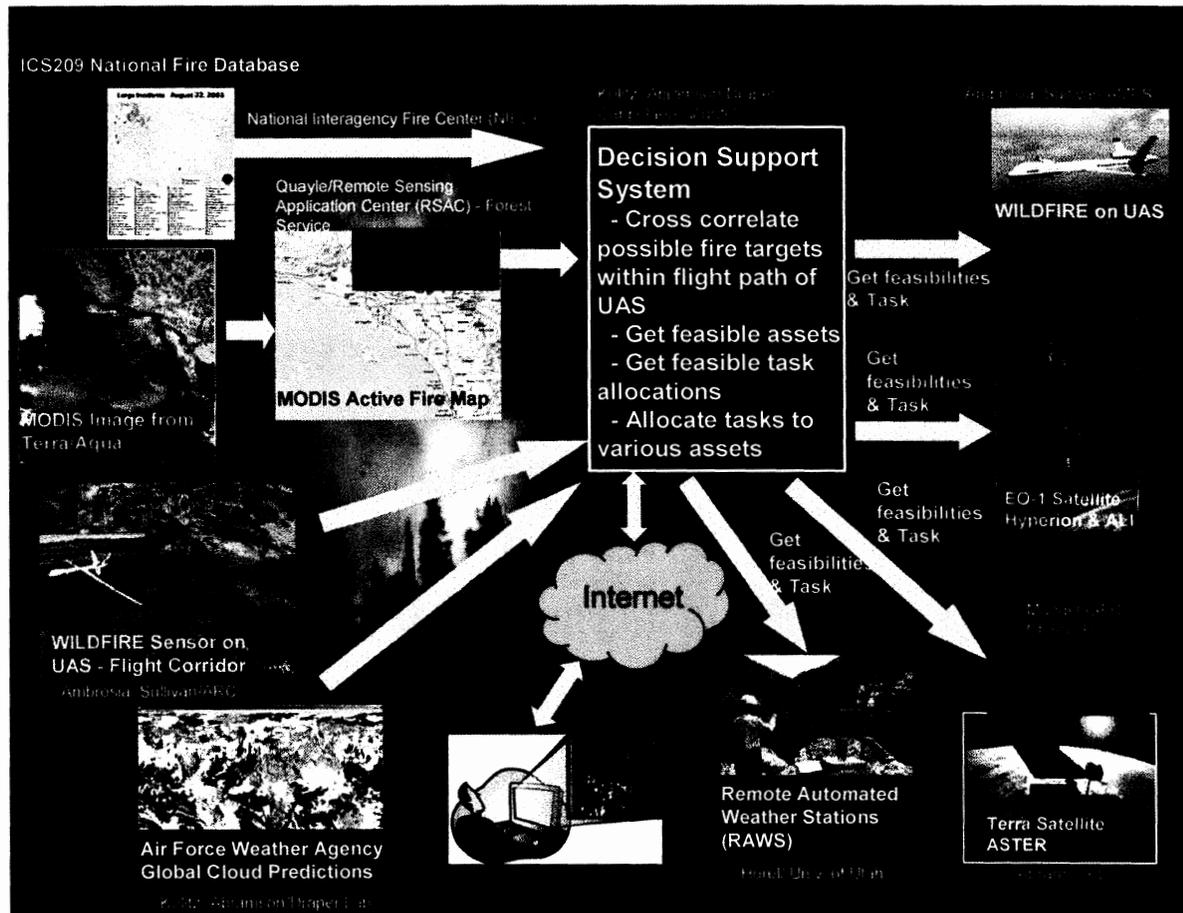

Figure 3 Composite sensor web demonstration
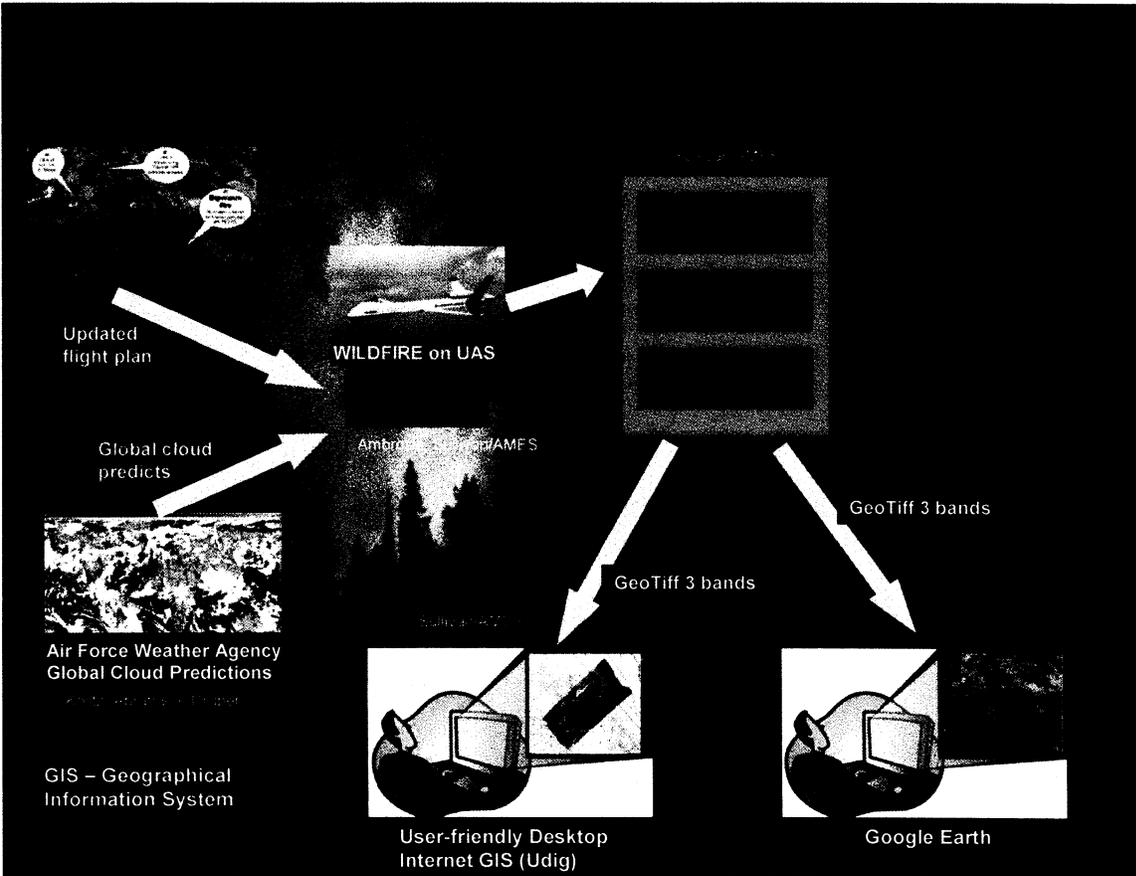
Figure 4 The Unmanned Aerial System(UAS) workflow

5. MANAGING WORKFLOWS

One of the key challenges is the management of workflows. As the number of available workflows increases, there needs to be method for cataloging these workflows so that users can find and execute them. But the biggest problem is that the workflows are not static since they trigger active assets such as satellites. A stored workflow may not be feasible anymore once requested because one of the satellites within the workflow is no longer available. Similarly, a website that provides an algorithm service may be taken off line. So a variety of functionality is needed to manage these workflows, which includes creation, editing and executing the workflows. Also, the user needs to be able to perform a "get feasibility" to understand whether the workflow will actually execute for the desired application. Finally, there has to be a method to depict workflows in a generic manner and then to activate a concrete workflow, which makes use of one of many available execution engines such as BPEL.

Figure 5 depicts a generic scenario for managing and triggering a workflow. Note that for this scenario, Business Processing Management Notation (BPMN) is used as a middleware workflow generator for the user interface. An Application Processing Interface (API) is needed to translate the XML Process Definition Language (XPDL) into the generic directives for any of the execution engines that may be selected for use. In addition, API's are also needed to create, delete, update, describe and register the workflow. There also has to be a means to get the status of the workflow to keep the user updated as to how far along the workflow has proceeded and an anticipated completion time. During this upcoming year, we will prototype portions of this workflow schema to the degree that funding allows.

In the meantime, we are beginning to implement a preliminary simplified scenario as depicted in Figure 6 for discovery and triggering some of our existing workflows. In this scenario, we will make use of the Global Master Change

Directory (GCMD) at Goddard Space Flight Center as a means to register an existing workflow and then to trigger it. Future demonstrations will also make extensive use of the Earth Science Gateway at NASA/GSFC and the GEOSS registry as appropriate.   The GCMD incorporates a registry that allows users to search for sensors, data and workflows. In addition, an RSS feed provides the relevant information to Google and other search engines so that a search can be conducted for the workflow based on key words provided by the person who registered the workflow. In addition, when the appropriate information about the workflow is returned, a link is provided to the appropriate web page to trigger the workflow. This will be a rudimentary first step to enable the team to examine the issues of managing these workflows.
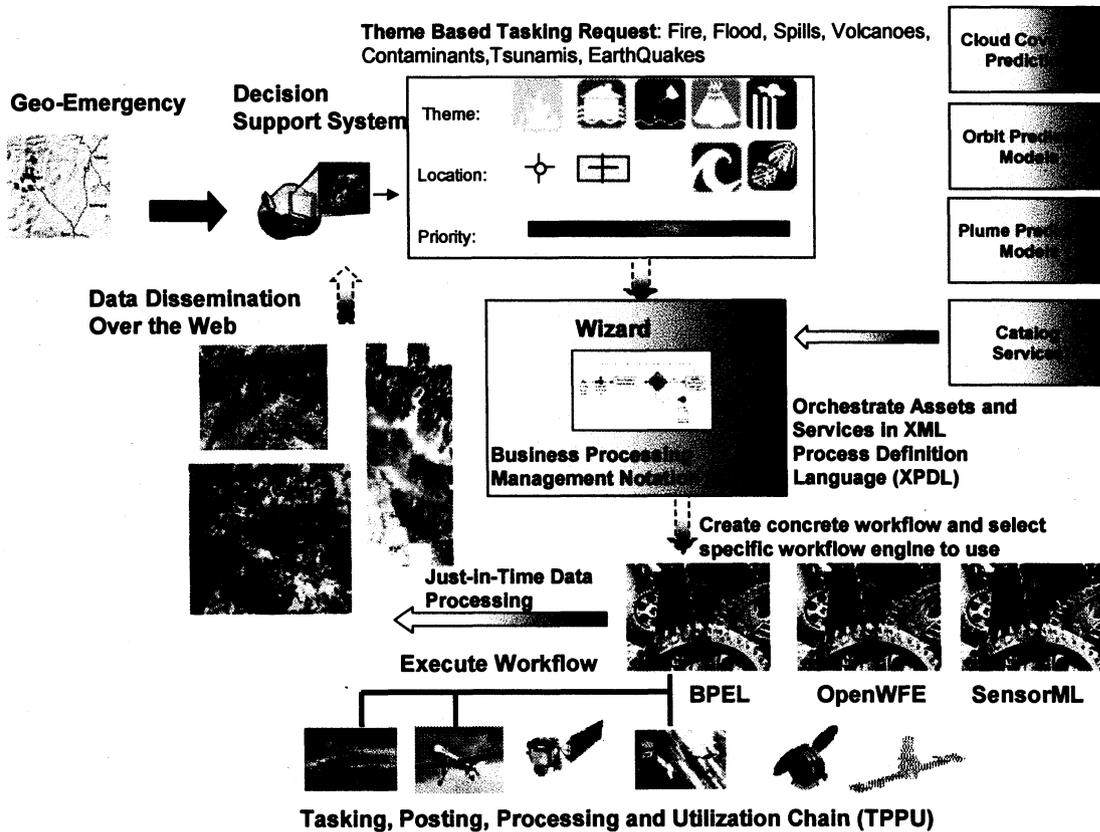

Figure 5 Generic workflow scenario
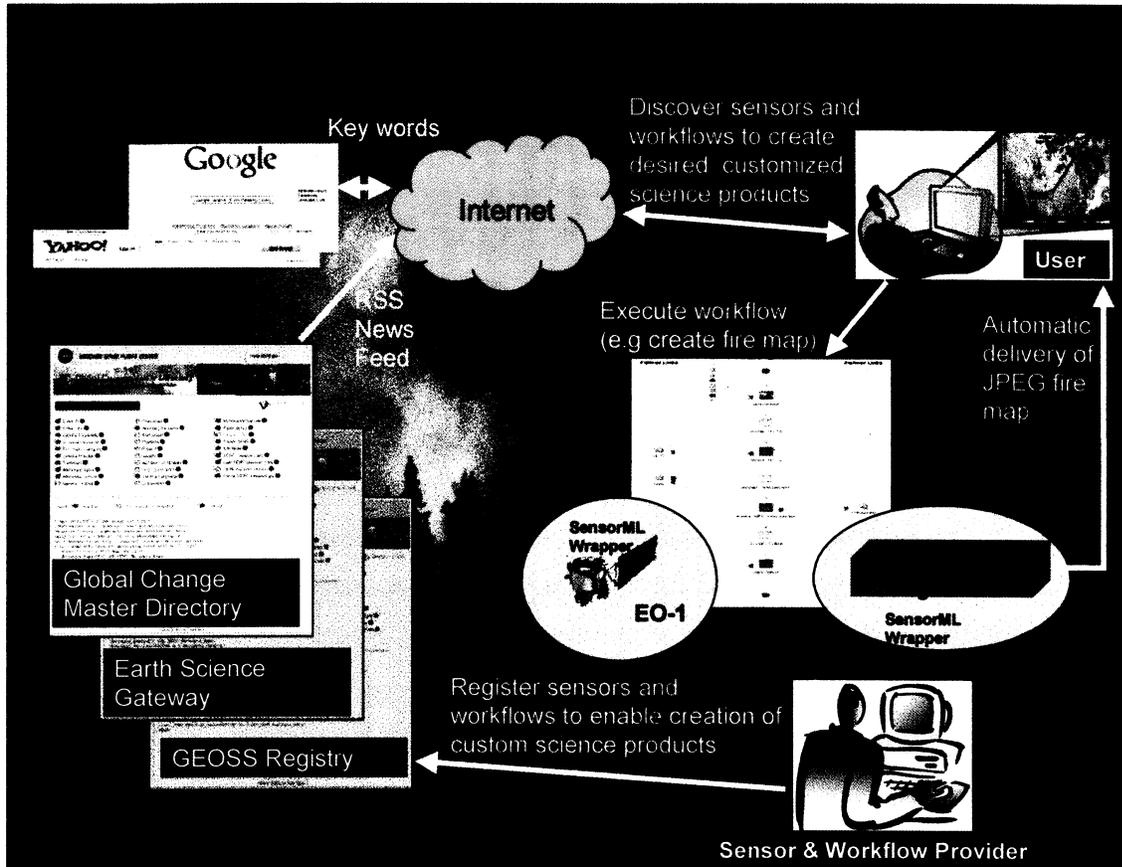
Figure 6 Rudimentary discovery and trigger scenario for a workflow

6. FUTURE ENDEAVORS

The team is extending the scenarios that have been built and are being built for more satellites and in conjunction with GEOSS architecture prototypes. To that end, the team is participating in the upcoming GEOSS Implementation Architecture Pilot demonstration and, in addition, participating in the OGC Web Services Testbed 5 (OWS-5) in the Fall of 2007. Figure 7 depicts some of the possible targeted assets to integrate depending on funding availability. In the case of SPOT, they will participate in OWS-5 and therefore by default will have some of the OGC web services implemented. This provides our team with an opportunity to interact with the SPOT satellite.

Done
EO-1 ✓

Additional Satellites to integrate into Sensor Web
AURA

TERRA/ AQUA ✓   CALIPSO   IRS   SPOT 5

DMSP ✓   CLOUDSAT   ICESAT   NPOESS
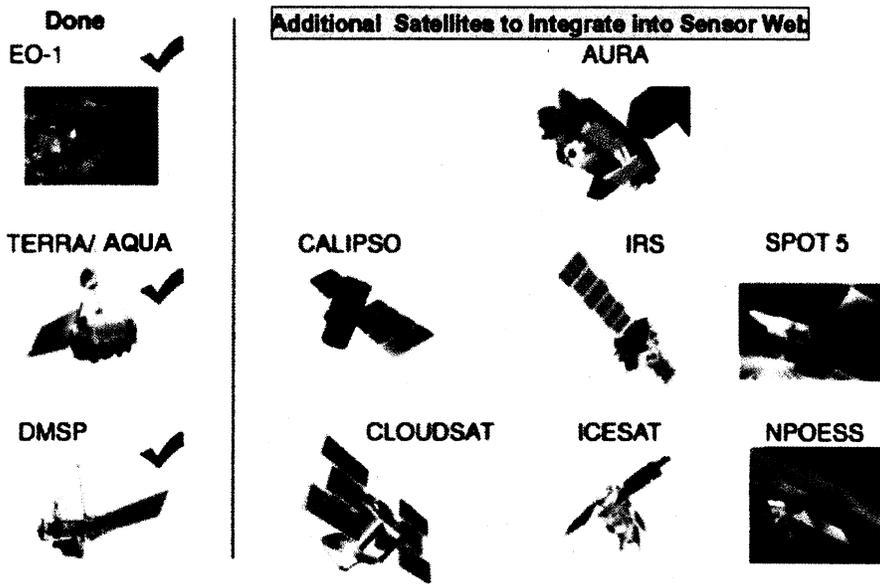
Figure 7 Additional satellite assets for possible future inclusion in our integrated demonstration

7. CONCLUSION

The sensor web experiments being developed under the NASA AIST task described in this paper serve as pathfinders for capabilities required to enable GEOSS. The real applications that are being added represent compelling stepping stones to the future GEOSS architecture. We have isolated the capabilities of discovery and managing of automated workflows as two key capabilities to make future GEOSS architectures cost-effective and achievable.

REFERENCES

[1]   D. Mandl, "Experimenting with Sensor Webs Using Earth Observing 1," IEEE Aerospace Conference Proceedings, vol. 1, pp. 183, March 2004.

[2]   A. Davies, A.., S. Chien, R. Wright, A. Miklius, P. R. Kyle, M. Welsh, J. B. Johnson, D. Tran, S. R. Schaffer, and R. Sherwood (2006), Sensor Web Enables Rapid Response to Volcanic Activity, Eos Trans. AGU, 87(1), 1. January 2006

[3]   Material from this paragraph was mainly taken from Wikipedia article on Web 2.0.

[4]   S. Chien, B. Cichy, A. Davies, D. Tran, G. Rabideau, R. Castano, R. Sherwood, R. Greeley, T. Doggett, V. Baker, J. Dohm, F. Ip, D. Mandl, S. Frye, S. Shulman, S. Ungar, T. Brakke, J. Descloitres, J. Jones, S. Grosvenor, R. Wright, L. Flynn, A. Harris, R. Brakenridge, S. Cacquard, S. Nghiem I. S. Jacobs and C. P. Bean, "An Autonomous Earth Observing Sensorweb," IEEE Conference on Systems Man and Cybernetics (IEEE-CSMC 2005). Big Island, HI. October 2005

[5]   http://eo1.gsfc.nasa.gov/new/extended/sensorWeb/sensorWeb.html

ACRONYM LIST

AFWA - Air Force Weather Agency

AIST - Advanced Information System Technology

BPEL – Business Process Execution Language

BPMN – Business Processing Management Notation

CSW – Catalog Services For the Web

DSS - Decision Support System

ESG - Earth Science Gateway

EO-1 – Earth Observing 1

ESTO – Earth Science Technology Office

GCMD – Global Change Management Directory

GEO – Group on Earth Observations

GEOSS – Global Earth Observation System of Systems

MODIS – Moderate Resolution Imaging Spectroradiometer

NIFC – National Interagency Fire Center

OGC – Open Geospatial Consortium

OpenWFE – Open Workflow Engine

RAWS – Remote Automated Weather Station

RSS – Really Simple Syndication

SAS – Sensor Alert Service (Pub/Sub)

SensorML - Sensor Modeling Language

SOS – Sensor Observation Service

SPS – Sensor Planning Service

SWE – Sensor Web Enablement

TPPU – Tasking, Posting, Processing and Utilization Chain

UAV – Unmanned Aerial Vehicle

UAS – Unmanned Aerial System

WCS – Web Coverage Service

WCTS – Web Coordinate Transformation Service

WFS – Web Feature Service

WMS – Web Map Service

WPS – Web Processing Service

WRAP – Wildfire Research Application Program

XPDL – XML Process Definition Language