

# A New Cloud-based Deployment of Image Analysis Functionality

## The ENVI Services Engine

Thomas Bahr

Exelis Visual Information Solutions GmbH  
 Gilching, Germany  
 e-mail: thomas.bahr@exelisvis.com

Bill Okubo

Exelis Visual Information Solutions  
 Boulder, U.S.A.  
 e-mail: bill.okubo@exelisvis.com

**Abstract**—For the past 30 years Exelis Visual Information Solutions has been delivering image science tools and solutions to both research scientists and image analysts within the Environment & Natural Resources community. In this paper we present a new capability, the ENVI Services Engine, which exposes ENVI processing functionality as RESTful web services providing online and on-demand image science Apps to the geospatial community. The ENVI Services Engine is interoperable with other middleware and servers, adheres to open standards, can support multiple clients (web, mobile, desktop) and is configurable to thin-client environments, such as the JavaScript. Thus we solved the requirements of today’s geospatial users expecting online, on-demand access to image processing in a client-server enterprise or cloud architecture.

**Keywords**-SDI; Geospatial Web Services; Cloud; Satellite Imagery; Image Processing; Mapping.

### I. INTRODUCTION

Geospatial software users are looking for tools to get answers from imagery anywhere, anytime. As organizations grow and more users employ image analysis in their decision making process, these organizations have a need to centrally deploy and manage applications, tools and data. Additionally, as users are located in a variety of environments, results need to be available on thin clients such as a web browser or native mobile device [1].

Desktop software only solutions no longer meet these needs for geospatial processing. Therefore, we solved the above in an interoperable, standards-based REST (Representational State Transfer) solution.

The ENVI Services Engine (ESE) is a cloud-based deployment of image analysis functionality that integrates into geospatial platforms. The ESE is comprised of a RESTful API that allows users access to ENVI’s imagery exploitation capabilities, as well as the necessary components to publish and deploy these consumable web services into any enterprise environment. By offering on-line, on-demand image analysis, geospatial users have access to information derived from remotely sensed data from any location at any time.

In the current state of development, we do not provide any off-the-shelf client application or Software development kit (SDK), leaving it free to work with any type of client.

Section II gives a short summary on ENVI, and introduces the ESE, which accesses ENVI’s tools for image processing and analysis. Section III covers the development of Apps. Section IV describes a reference implementation. Finally, section V presents the conclusions.

### II. ENVI-BASED SERVICES IN THE ENTERPRISE

#### A. ENVI - Environment for Visualizing Images

ENVI solutions combine spectral image processing and image analysis technology to get detailed information from geospatial imagery. Data gathered from today’s satellite and airborne sensors, including panchromatic, multispectral, hyperspectral, radar, thermal, and LiDAR, is supported. ENVI ingests, reads, and extracts information from these various sources, and can fuse multiple data modalities to exploit the strengths of each data type. It works with any size data set and has automated tools to prepare big and small geospatial imagery for viewing or further analysis. Data preprocessing and analysis includes a comprehensive suite of tools for, e.g., atmospheric correction, registration, orthorectification, filtering, geospatial statistics, topographic modeling, and object based feature extraction, etc. [2].

ENVI is written in the IDL programming language. This allows users to extend or customize ENVI features and functionality to fit both image analysis requirements and specific project needs [3].

#### B. ENVI Services Engine Overview

ESE incorporates open source standards, and, as a result, supports integration into many different types of systems. By running ENVI or IDL algorithms and routines as enterprise services, users can access any of the image analysis components they need for data exploitation. Once an ENVI or IDL routine is developed, it is quite simple to wrap it as a service and deploy it in the engine for consumption.

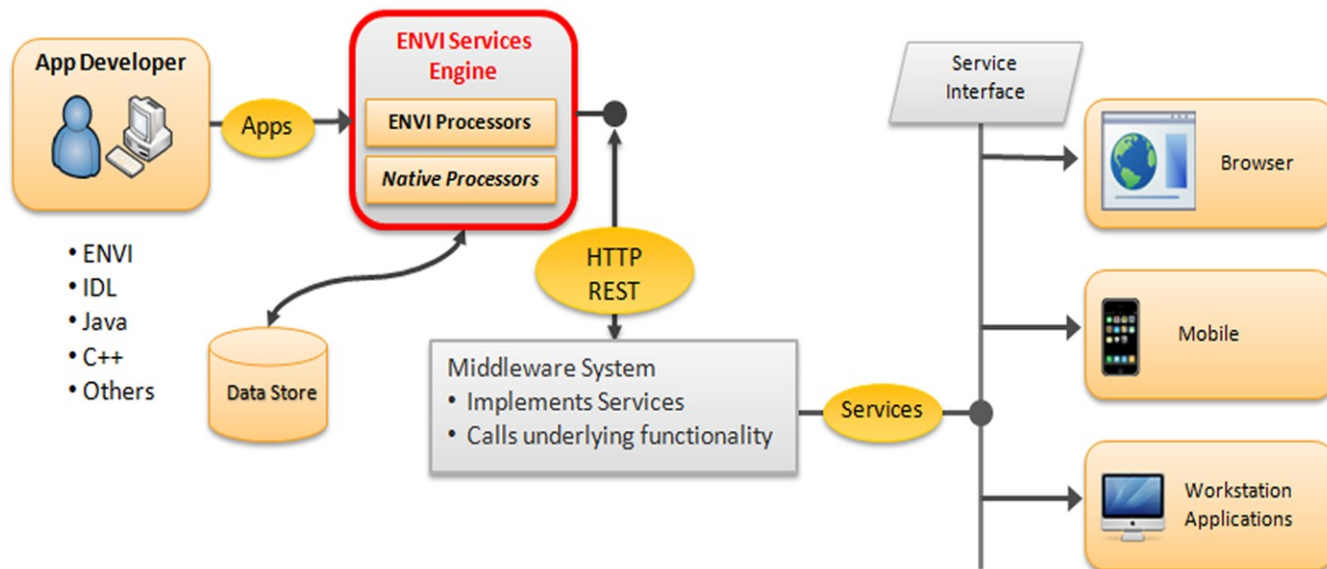


Figure 1. ENVI Services Engine conceptual rendering. ESE runs e.g. ENVI and IDL routines through a standard HTTP RESTful interface. Data requests are handled via WCS or WPS calls. ESE is called by a web client or mobile app via a middleware component.

ESE runs these algorithms through a standard HTTP RESTful interface, allowing users to provide basic user interfaces to complex analysis tasks via lightweight client. (Fig. 1). In particular, it is based on ESRI's GeoServices REST specification [4]. The ESE implements a superset of ESRI's specification wherever it makes sense for IDL and ENVI processing [5].

Results are returned to the client via standard HTTP mechanisms. They can be saved and displayed in a variety of clients, including online, desktop, and mobile clients, depending on the user's implementation. These image analysis applications and components can be integrated into a cloud environment, independent of any existing middleware configuration (Fig. 1). The ESE is therefore a flexible and easy to use framework that functions within the user's existing infrastructure.

As well as offering on-demand, online access to ENVI functionality, the ESE can be used to develop and expose unique analytics, allowing organizations to create their own enterprise image analysis workflows. Using existing ENVI functionality in a cloud environment will help to save time and resources, as the user can develop an algorithm once and deploy it to multiple locations. Additionally, the user can release new applications and functionality as mission needs change and evolve.

One of the largest benefits of deploying image analytics in the cloud is the ability to run complex, resource-intensive analysis on extremely large datasets from thin or mobile clients. By having both the data and analysis components on the web, lightweight applications can be used to call the analysis functions, making the analysis workflow more efficient for users and moving image exploitation closer to the end user.

### C. Architecture

The ESE application consists of one "master" process and one or more "worker" processes (Fig. 2).

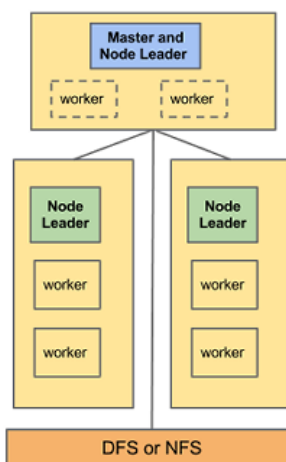


Figure 2. ENVI Services Engine architecture: "master" process and associated "worker" processes.

Processing requests are passed to the Services Engine via HTTP REST calls. The master contains an embedded web server. The master's job is to manage requests and jobs for worker processes. Worker processes contain an IDL interpreter and do the actual IDL and ENVI processing. When workers complete a job, they notify the master of the results and the master returns those results to the client [5].

ESE runs on 64-bit Linux and Windows as either a daemon or a standard application (launched manually). Security must be provided by the user by way of file and execution permissions, proxies, firewalls, etc. [5].

An ESE process assumes the role of master if it is the first instance of the process that is started up on the machine that is designated the master machine. On any other machine in the cluster, the first instance is run as a node leader. A node leader will launch and monitor a configurable number of worker processes. This number can be specified by a configuration file but typically defaults to the number of CPUs on the system. Node leaders, including the master, will relaunch workers that have exited [5].

D. Scalability and Load Balancing

The important need of scalability is supported by the ability to scale and load balance because it implements a REST architecture and is designed to run on a cluster. As requests arrive from any number of clients, the master process' web server will accept and route the request. If the request is to perform a task, then the master will give the request to one of its worker processes. Worker processes can run on any machine in the ESE's cluster. The cluster is a collection of machines tied to one particular master machine. Namely, a machine is part of the cluster if it is running an instance of the ESE executable and that executable is wired to communicate with the same master as other machines on the cluster. The master machine is specified by a configuration file [5].

III. DEVELOPING APPS

A. Access image analysis functionality through Apps

ESE is designed for such enterprise web services and can be configured to work with multiple types of thin-and-thick clients including web browsers, mobile devices (Android, iOS) and workstation applications such as ENVI or ArcGIS (Fig. 3) [6].



Figure 3. An interactive Line-Of-Sight App, launched from Native Android Client.

- (a) The user locates an AOI, (b) draws an interactive circle to indicate parameters, (c) the middleware passes the location and AOI to the service, (d) the high-level service finds the appropriate data in the catalogue and constructs the ESE call, (e) the ESE runs and returns the vector result showing the line of sight, (f) the vector result is passed through the middleware to the client, (g) the client renders the result in the appropriate geographic location in the interface.

The advantage of this approach is that analysts can access specific image science algorithms, tools and configured workflows through discrete Apps that are easy to update and maintain.

B. Benefits for developers creating Apps

The benefits for the developers creating Apps to use with the ESE include [6]:

- Access to 100+ different formats and modalities of imaging data.
- Ability to exploit multi-source image data using proven analytic methods such as Target Detection, Object-Based Feature Extraction, Change Detection.
- Ability to deliver derived products in easily consumable formats (.shp, kml, NITF, etc.).
- Access to analytic methods that have been widely used in the scientific community for decades.
- Ability to fuse multiple sources of information, resulting in higher confidence in derived products.
- Extensibility allows for quick insertion of capabilities to support new and emerging modalities and/or analytic approaches.

IV. A REFERENCE IMPLEMENTATION

In an effort to provide examples of how the ESE can be used, Exelis VIS created a reference implementation that redeploys several pieces of existing ENVI code as 'apps'. These include [1]:

- Anomaly Detection for detection of spurious material in a large image.
- Pan Sharpening for enhancing multispectral data with high resolution panchromatic imagery.
- Vegetation Delineation for identifying vegetation presence and level of vigor.
- Line of Sight to create a viewshed using terrain data (Fig. 3).
- Spectral Identification to show using a spectral library to classify materials in a hyperspectral dataset.

This implementation envisions a web client or mobile app calling the ESE via a middleware component such as MapServer, GeoServer, or some other development environment and employs a web based GUI and/or a mobile app for Android or iPhone as an access mechanism [1].

Thin and mobile clients are used to discover data using a cataloguing specification such as Web Catalog Service (WCS) and to make Open Geospatial Consortium (OGC)-compliant analysis requests via Web Processing Service (WPS) calls to the ENVI Services Engine via the middleware (Fig. 4) [1].

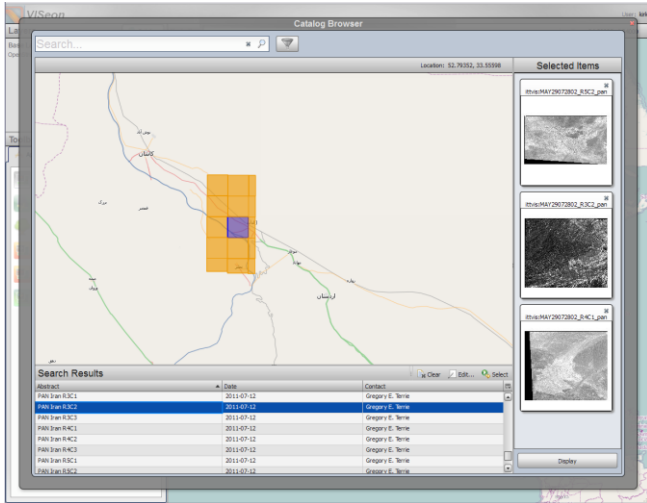


Figure 4. Accessing a data catalogue with a JavaScript interface.

One use case example could be looking at the Landsat archival data with the vegetation delineation tool. For a region or a specific scene, this ESE implementation would facilitate in discovering imagery, identifying vegetation and analyzing change over time. Results could include an animation, a profile of vegetation change over time, or specific points correlated to ground truth measurements. These results could be delivered to web, mobile or a desktop clients [1].

### V. CONCLUSION

The ENVI Services Engine (ESE) provides the community with a flexible and easy-to-use framework for creating and using image science Apps. Tasks such as detecting target signatures from hyperspectral imagery or calculating a Line of Sight (LOS) from LIDAR data are simple to execute and easy to share. The power comes from exposing all of ENVI's image processing algorithms and tools as RESTful web services and using ENVI/IDL, Java or other development languages/tools to create Apps for image science tasks such as orthorectification, classification, etc.. Benefits of the ESE approach include [6]:

- The flexibility to integrate into any architecture that can make http calls.
- Delivers an online, on-demand approach to data analytics.
- Takes advantage of open standards allowing for integration with other Apps.

- Allows experts to compile and expose their workflows and tradecraft, enabling end users to create their own tailored GEOProcessing products.
- Minimizes desktop software licensing costs.
- Emphasizes the use of server side hardware by offloading processing tasks previously performed at the desktop level.

The way imagery data is being stored and analyzed is changing. ENVI and IDL have long been tools used in Earth science communities. The ESE helps scientific data users transition to the cloud while still being able to use legacy code and algorithms [1].

The ability to host and disseminate data and complex functionality within the cloud will lead to wider enterprise-wide hosting and dissemination of earth science data in the future. The work behind ESE is aimed at making this concept a reality. More people being able to access and process scientific data means more discoveries, more collaboration, and more progress, e. g. toward future environmental challenges [1].

### ACKNOWLEDGMENT

The present contribution has been developed by Exelis Visual Information Systems in Boulder, Colorado, U.S.A.

### REFERENCES

- [1] A. O'Connor, K. Lausten, B. Okubo, and T. Harris, "ENVI Services Engine: Earth and planetary image processing for the cloud", American Geophysical Union, Poster IN21C-1490, 2012.
- [2] Exelis Visual Information Solutions [Ed.], "ENVI Capabilities", <http://www.exelisvis.com/ProductsServices/ENVI/Capabilities.aspx>, 23.01.2013.
- [3] Exelis Visual Information Solutions [Ed.], "ENVI Capabilities", <http://www.exelisvis.com/ProductsServices/ENVI/Capabilities/CustomizeCapabilities.aspx>, 23.01.2013.
- [4] ESRI [Ed.], "GeoServices REST Specification Version 1.0." An ESRI White Paper, 2010.
- [5] Exelis Visual Information Solutions [Ed.], "ENVI Services Engine Documentation", ENVI Services Engine v1.0 beta, 2012.
- [6] Exelis Visual Information Solutions [Ed.], "Image Science Apps for the NGA's Online GEOINT Services (OGS)." Exelis Visual Information Systems White Paper, unpublished, 2012.