# The virtual solar observatory

**C. A. Young[1], J. B. Gurman[1], G. Dimitoglou[1], J. Hourcle[1], R. S. Bogart[2], K. Tian[2], F. Hill[3], I. Suarz-Sola[3], S. Wampler[3], P. Martens[4], S. Yoshimura[4] and A. Davey[5]**

[1] NASA Goddard Space Flight Center
[2] Stanford University
[3] National Solar Observatory
[4] Montana State University
[5] Southwest Research Institute (Boulder, CO)

**Abstract**. The recent explosion of data quantity and complexity has lead to the need for a new way to make data available, the virtual observatory. The Virtual Solar Observatory (VSO) provides a versatile means for solar physicists to discover and share the growing sources of data. We present the development of the VSO, show the system in action and discuss how data users and data providers can benefit from it.

**Index Terms.** Data analysis, data sources, virtual observatories.

## 1. Introduction

The VSO is a distributed system that provides users with uniform query interface in order to access diverse solar sources around the world. It is virtual because it has no physical structure only existing on the Internet. The VSO was initially proposed in order to facilitate large statistical studies in solar physics spanning many data types and data sets.

The concept of a virtual observatory grew out of the astronomical community in the 1990s. With the birth of the World Wide Web and the fast growth of data sources it was recognized that virtual data set linked through the web was a practical solution to the data access problem. Sanchez-Duarte et al. (1997) first proposed combining solar data archives into a unified system. This idea was discussed further for the “Whole Sun Catalog” by Dimitoglou et al. (1998). This was then formulated as the VSO by Hill (2000) where he outlined a set of solar analysis topics that would be accelerated by a VSO. Hill pointed out that most large studies only use 3-4 data sources and that larger studies were not very feasible due to the diffulty in accessing and even finding what data sources exist. At the 2000 Solar Physics Division (SPD) meeting of the American Astronomical Society (AAS) as session called the “Birds of a Feather” (BoF) session was held to discuss and coordinate the current efforts for the VSO. The 2001 Senior Review started a 1-year study of the VSO and development began over the following two years (2003 June - 2005 May). This effort was undertaken by a consortium consisting of the National Solar Observatory (NSO), Stanford University, Montana State University and the Solar Data Analysis Center (SDAC) at the Goddard Space Flight Center.

The first beta of the VSO was released at the 2003 Fall American Geophysical Union meeting. The design for the released system remained virtually unchained from initial designs released the year before. The version 0.5 beta was released at the 2004 Spring AAS/SPD meeting and version 1.0 was released at the 2004 Fall AGU meeting. The VSO is currently running version 1.2, released in 2005.

## 2. Design

The VSO has several basic design characteristics (Bogart et al., 2004). These are utility, expandability, robustness and simplicity. The utility is a uniform interface for queries and data requests. This also includes the unification of distributed data sources. The expandability and robustness mean the use of multiple data servers and providers along with multiple user and application interfaces. This also means minimal centralization of metadata, minimal effort for provider participation, minimal information in a data registry and no assumed data server database architecture. The simplicity is met by having no data delivery, no centralized data catalog and the simplest possible data model. The VSO simply acts as a broker between the user and the data providers. This follows the “small box” (Gurman et al., 2005) idea for the VSO system. These characterics mean that the VSO is not designed for data mining, it has no intelligent agents (AI/neural nets) and there is no access control.

The basic parts of the VSO are show in Fig. 1. The VSO consists of a user interface, the VSO instances or query engine and the data service (Davey et al., 2004). The primary user interface is a web browser but there are other

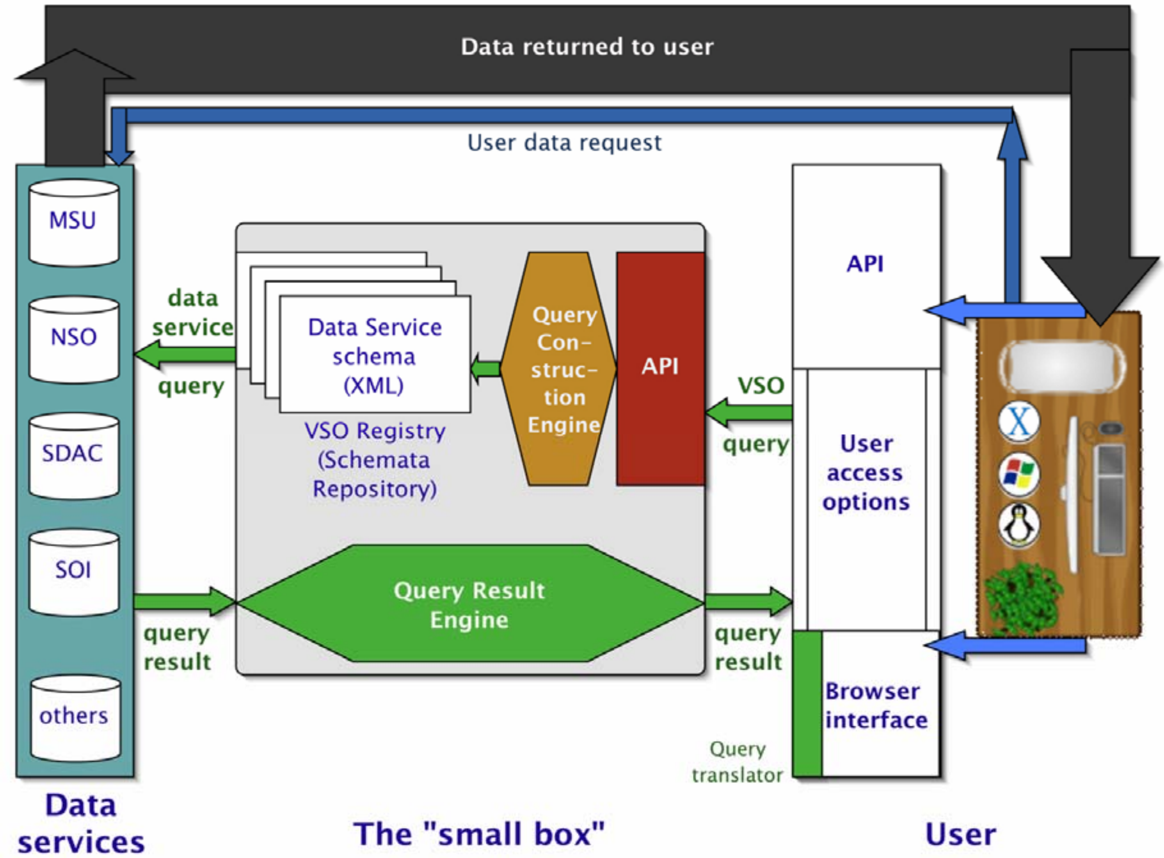


**Fig. 1.** This is the general "small box" design layout for the VSO. The user accesses the VSO using an API (web browser is the standard interface). A query is constructed by the VSO instance. Data sources are query for the data. The query results are returned to the users through the VSO instance. The user makes their selection and obtains data directly from the data sources (Gurman et al., 2005).

access methods (see section 4). This web interface allows the user to select query options based on the data model. The data model was developed to be as simple as possible. It describes the datasets held at a particular data site. This model determines which sites are contacted for a particular query. The data model also uses common names used by solar physicists to describe different datasets. This helps to make the data descriptions and interfaces more intuitive. The web interface works via cgi scripts that construct a query that then accesses the VSO instance. The instance is the VSO core that handles use queries, determines where to query for data and returns results to the user. The VSO core is implemented in Perl scripting language. This language was chosen because of the large number of packages already written that handle the needs for the instance such as networking and database interaction. Platform and operating system independent access to solar data is provided in the system using the Simple Object Access Protocol (SOAP) by way of the Perl SOAP::Lite module. The VSO has a registry for each provider. This consists of XML data service schema document that holds the configuration parameters. The data services are the actual providers of the data. Once a query has been matched data is retrieved directly from the providers. Table 1 shows the current list of VSO data providers and the data sources they serve.

**Table 1.** VSO Data Providers and Sources

| Data Provider | Data Source |
|---|---|
| HANET | BBSO |
| | KANZ |
| | OACT |
| | OBSPM |
| | YNAO |
| HAO | MLSO |
| LSSP | RHESSI |
| MSU | Yohkoh |
| MWSPADP | Mt. Wilson |
| NGDC | GOES-12 |
| OBSPM | Nancay |
| | OBSPM |
| | Pic du Midi |
| OVRO | OVRO |
| SDAC | SOHO |
| | TRACE |
| SFO | SFO |
| SHA | GONG |
| | JSPO |
| | MtWilson |
| | SOHO (MDI) |
| | TON |

This table contains VSO data providers in the first column and their corresponding data sources in the second column.

The VSO has several currently available searches. The first is a search of all by time. This is a very simple form that automatically searches all datasets registered by all providers in the selected time interval. A search can be made with selected instruments by time. This performs a joint time-based query on data from selected instruments or archives only. A sample VSO query interface of this type of search is given in Fig. 2.

A user can search selected observables by time. This searches all data sets for data in the selected time interval matching selected criteria for physical observables. A user can select observables, instruments and time. This searches selected instrument or archive data sets for data in the selected time interval matching selected criteria for physical observable. A user can select observable spectral range by time. This searches all data sets for data in the selected time interval matching selected criteria for physical observable and/or spectral range. Fig. 3 shows a sample VSO query interface for this type of search. Lastly, a user can make a search of nicknames. This searches all data sets for data in a selected time interval matching selected criteria for nicknames. Nicknames are common terms used by solar physicists for certain types of observable/spectral range images. Examples include *magnetogram, UV image* or *coronagraph image*. Fig. 4 shows a sample VSO query interface of this type of search.

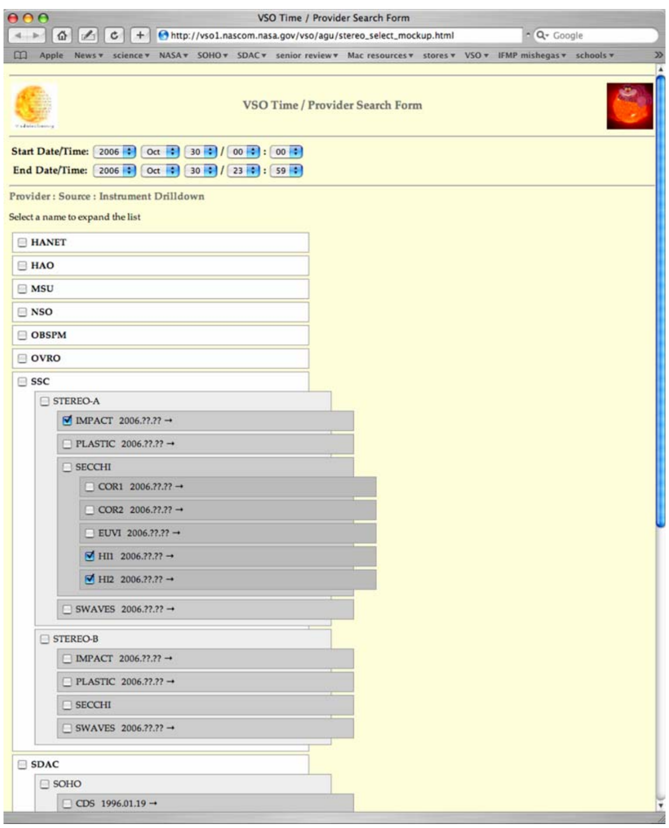


**Fig. 2.** A sample VSO query interface that combines a search in time with a one over data providers and data sources.

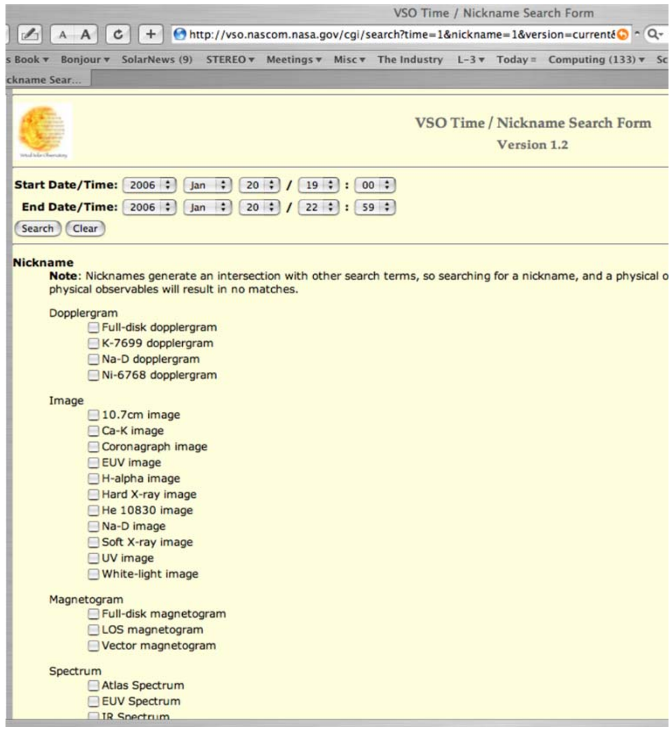


**Fig. 3.** A sample VSO query interface that combines a search in time with a one over data nicknames.

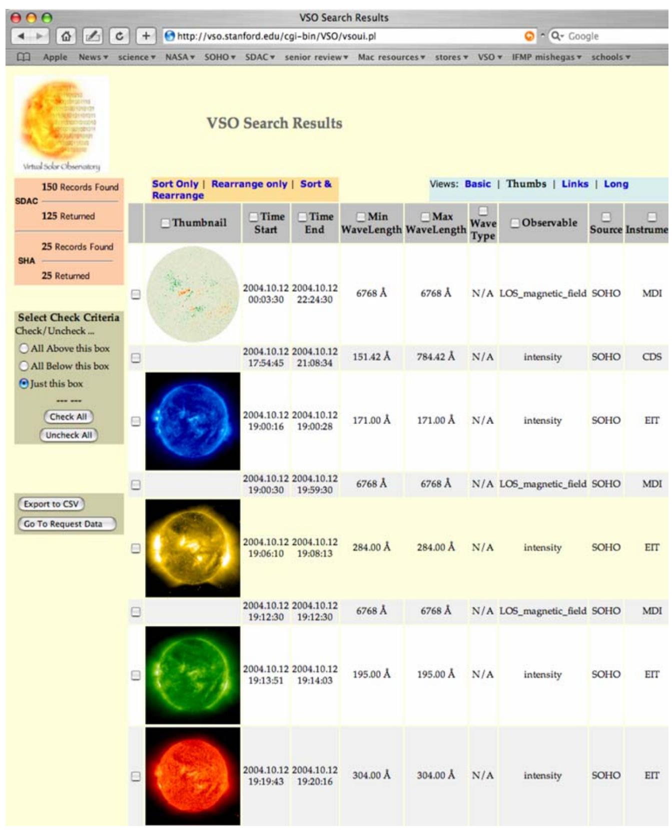

**Fig. 4.** Shown here is a sample VSO query result page. Thumbnails of the displayed data are shown when they are available.

After a search is made the user is presented with a VSO search results page. This page contains rows of all data matching the selected criteria. Next to each individual dataset are columns containing all the criteria under which the data fall, such as time range, observables and so on. When images are displayed, a thumbnail of the image is shown if it is available. Fig. 5 shows a sample VSO query results page.

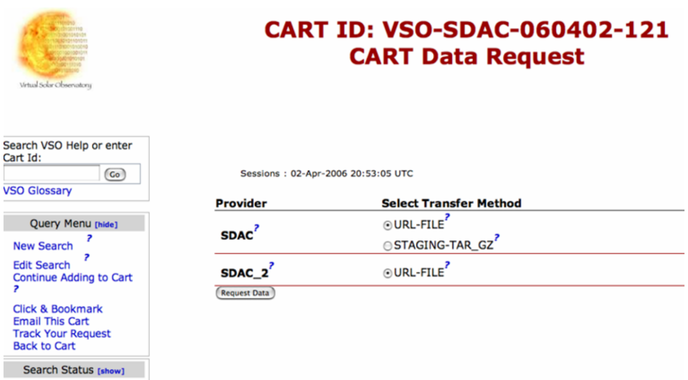


**Fig. 5.** A sample VSO cart data request. It contains options to select the data transfer method. These options are either a URL to down the data from or file to be transferred via electronic mail.

## 3. Adding providers and interfaces

In order to serve as a data provider to the VSO a data registry must be constructed in order to describe the data at in the provider's archive. This is an XML document, the Data Service schema, provides the translation between the data provider and the VSO instance. Nominally, the VSO is designed to work with providers that maintain their data with a relational database. However, if a provider only has there data organized in ftp or http directory trees there is a solution. A proxy server is offered for the provider. This contains the metadata to access the archive and has its metadata cache updated periodically.

The architecture of the VSO allows for any number of interfaces to search data served by the individual data providers. By using the VSO API, programmers can write their own custom interfaces, or searching to their existing search utilities. Although VSO API uses Web Services, it does not require the front end to be web-based. The front end may be a desktop application, web-based service, a command line program, or accessed through functions in other computer languages such as IDL.

Common tasks can be automated to free up researchers to focus on science, instead of trying to find and download new data. Three paths currently available: Perl with SOAP:Lite, Java using Axis, C or C++ using gSOAP. There is a WSDL (Web Service Description Language) document describing the API. Currently, the VSO interface has been implemented in the standard web browser form, in a JAVA GUI and with IDL search tools in the SolarSoft package. Also, a web interface just for the STEREO mission's science center (SSC) is being developed. A mock up of this web interface is shown in Fig. 5.

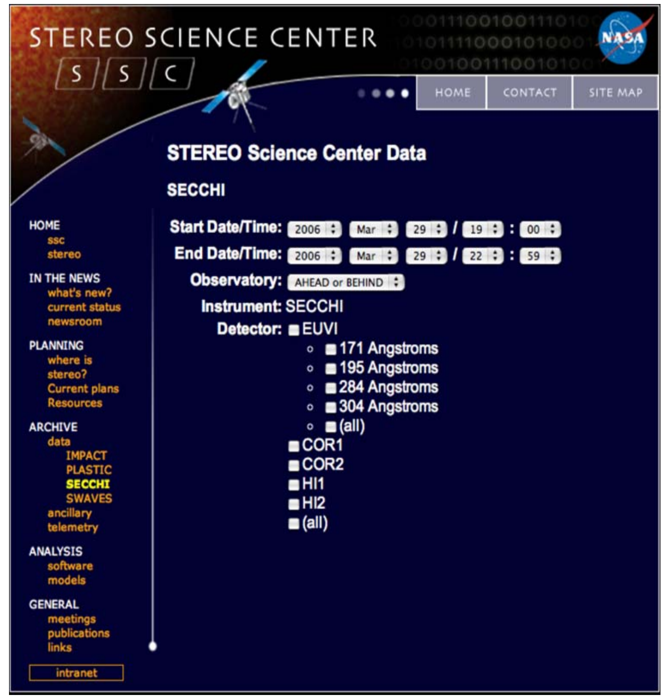


**Fig. 6.** This is a mock-up of the VSO web interface for the STEREO Science Center Data (Hourcle et al., 2004).

## 4. The VSO future

The VSO is continually evolving. New data sources will continue to be added. There are currently plans to include data from the upcoming missions STEREO, Solar-B, and SDO. In addition to data, more data catalogs (catalogs are meaningful groupings of data, e.g. CME list) will be added as they become available. Related to this is this interaction with EGSO, CoSEC and other virtual observatories. The VSO is currently working to have other observatories available through the VSO and make the VSO available to them.

The VSO has plans to incorporate a central service to easily facilitate logging to provide an accounting of the systems activities. To reduce overhead, moving from RPC encoded to document literal in the WSDL is being considered (Bogart et al., 2005). Also, a Data Provider's Implementation Kit is being developed.

## References

R. S. Bogart et al., "Building a Virtual Solar Observatory: Lessons Learned", American Geophysical Union, Spring Meeting 2005, abstract #SH51B-03.

R. S. Bogart et al., "The Virtual Solar Observatory", 2004, American Astronomical Society Meeting 204, #52.04.

A. R. Davey et al., "Implementation of the Virtual Solar Observatory", 2004, American Astronomical Society Meeting 204, #70.02.

G. Dimitoglou, C. Mendiboure, K. Reardon and L. Sanchez-Duarte, "Whole Sun Catalog: Design and Implementation", in ASP Conf. Ser. vol. 155, *2nd Advances in Solar Physics Euroconference: Three-Dimensional Structure of Solar Active Regions*, C. E. Alissandrakis and B. Schmieder, Eds. San Francisco: ASP, 1998, pp. 297-301.

J. B. Gurman et al., "The Virtual Solar Observatory: Still a Small Box", American Geophysical Union, Spring Meeting 2005, abstract #SH43B-03.

F. Hill, "Virtual Observatories of the Future", in *Proceedings of the 1st Solar and Space Weather Euroconference*, vol. 463, A. Wilson, Ed. Noordwijk, Netherlands: ESA Publications Division, 2000, pp. 569-574.

J. A. Hourcle et al., "STEREO in the Virtual Solar Observatory Context", American Geophysical Union, Fall Meeting 2004, abstract #SH21B-0413.

L. Sanchez-Duarte, B. Fleck and R. Bentley, "Whole Sun Catalog: Design and Implementation", in ASP Conf. Ser. vol. 118, *1st Advances in Solar Physics Euroconference: Advances in Physics of Sunspots*, B. Schmieder, J. C. del Toro Iniesta and M. Vazquez, Eds. San Francisco: ASP, 1997, pp. 297-301.