HELIO: Discovery and Analysis of Data in Heliophysics

Robert Bentley[a], John Brooke[b,*], André Csillaghy[c], Donal Fellows[b], Anja Le Blanc[b], Mauro Messerotti[d], David Pérez-Suárez[e], Gabriele Pierantoni[f], Marco Soldati[c]

[a] *Mullard Space Science Laboratory, University College London, Holmbury St. Mary, Dorking, Surrey RH5 6NT, U.K.*
[b] *University of Manchester, Oxford Road, Manchester M13 9PL, U.K.*
[c] *Fachhochschule Nordwestschweiz, Institute of 4D Technologies, Steinackerstrasse 5, 5210 Windisch, Switzerland*
[d] *INAF-Astronomical Observatory of Trieste, Loc. Basovizza n. 302, 34012 Trieste, Italy*
[e] *Finnish Meteorological Institute, Erik Palménin Aukio 1, 00560 Helsinki, Finland*
[f] *Trinity College Dublin, College Green, Dublin 2, Ireland*

Abstract

Heliophysics is the study of highly energetic events that originate on the Sun and propagate through the solar system. Such events can cause critical and possibly fatal disruption of the electromagnetic systems on spacecraft and on ground based structures such as electric power grids, so there is a clear need to understand the events in their totality as they propagate through space and time. The eScience challenge posed is that the data was gathered by many observatories and communities that have hitherto not needed to work together. Firstly, this involves the problem of helping users to more easily find and understand the relevance of data, especially data from outside their domain. Secondly, it involves solving challenges of data integration. We describe the design of the HELIO infrastructure, based on the use of Web Services linked together by workflows and accessible via portal-based user interfaces. We also discuss current progress in the implementation of this infrastructure and the feedback from the user community.

Keywords: heliophysics, eScience, ontologies, web services, workflows

1. An e-Science infrastructure for heliophysics

Heliophysics is the study of the effects of the Sun on the Solar System; it addresses problems that span a number of existing disciplines — solar and heliospheric physics, and magnetospheric and ionospheric physics for the Earth and other planets. The discipline is closely related to the study of Space Weather (whose effects on modern technology are well documented [1, 2]) but heliophysics is more generalised, covering all parts of the Solar System rather than just the Sun-Earth connection.

In order to undertake searches that are scientifically-interesting in heliophysics, we need to understand the origins of phenomena and how they propagate through interplanetary space, i.e., the path they follow and the time scales involved. This requires the ability to track both spacecraft and different types of solar emissions in 4-dimensions, which is a key difference from other astrophysical searches based on images of the "deep-sky" which can use a two dimensional coordinate system based on the celestial sphere [3].

Virtual Observatories (VxO for short) have been a highly successful approach to issues of data sharing and re-use in astronomy [4]. A Virtual Observatory for Heliophysics (VHO) needs extra tools to extend the essentially two dimensional search space of deep sky astronomy, since even though the deep-sky astronomy community has developed standards for data models and access methods that reduce the complexity of the e-Infrastructure required for a virtual observatory, they do not address the more complex search problems of heliophysics. Within

*Corresponding author

Email addresses: b.bentley@ucl.ac.uk (Robert Bentley), john.brooke@manchester.ac.uk (John Brooke), andre.csillaghy@fhnw.ch (André Csillaghy), donal.k.fellows@manchester.ac.uk (Donal Fellows), anja.leblanc@manchester.ac.uk (Anja Le Blanc), messerot@oats.inaf.it (Mauro Messerotti), dps.helio@gmail.com (David Pérez-Suárez), gabriele.pierantoni@scss.tcd.ie (Gabriele Pierantoni), marco.soldati@fhnw.ch (Marco Soldati)

the Heliophysics Science Division of NASA the approach has been to establish a number of small virtual observatories (often called the VxOs) that address individual parts of the heliophysics system — Sun, heliosphere, magnetosphere, etc. In the HELIO (Heliophysics Integrated Observatory)[1] project we decided instead to address the complete system, since it allows a richer and more comprehensive search environment, and deal with the problems of data management separately.

The communities involved in heliophysics have evolved independently over decades, even centuries. Although the links between the effects observed in the disciplines are now evident, there have been virtually no attempts to coordinate the way the scientists collectively conduct their data analyses. As a consequence, there are considerable differences in the way the communities store, describe and think about data, and this has a consequence of encouraging scientists in the domain to focus on extremely narrow data-sets instead of looking at the much broader sweep of data available from the past 40 years of data collection; it is these challenges that the HELIO project was established to address.

In order to facilitate the study of this new discipline, HELIO needed to tackle issues in a number of areas related to two basic requirements:

- Provide integrated access to data from all the domains of heliophysics that are held in archives around the world.
- Provide the means to conduct searches across the domains to identify data-sets of interest.

We have previously [5] described the scientific challenges involved. In the present work we describe the eScience infrastructure we are creating to meet these challenges. A major research problem is to search multiple catalogues or databases to track the development of an event when the effects of that event travel at different speeds. Heliophysical events are first observed (remotely) on the Sun, and then propagate through the solar system while potentially being detected by a variety of space- and earth-based instruments. Effects caused by photon emissions require line-of-sight view of the source and any delays are related to exactly predictable light travel times; those that are caused by particles occur with much longer delays. These delays are not exactly predictable due to the interaction of the particles with the interplanetary magnetic field, and in most cases the effects are only experienced if the propagating phenomena directly passes the observer (see Figure 1).

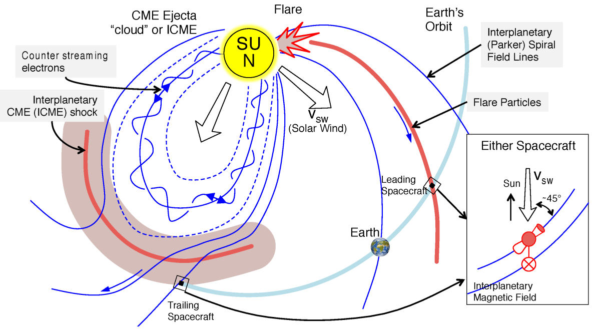


Figure 1: Illustration of how the location of an instrument (using STEREO mission as exemplar) is a vital consideration for whether it is able to provide an observation relevant to the study of an event. The leading spacecraft will detect particles from a flare issuing from one side of the Sun, whereas the trailing spacecraft will detect a CME shock wave on the other side.

We use previous work on virtual observatories as much as possible and adopted a number of techniques and standards developed within the International Virtual Observatory Alliance (IVOA) [23]. However, the dynamic nature of heliophysics (in particular, its strong dependency on time series) has meant that we have had to borrow Web Services approaches [6, 7] from other fields where data sources change more rapidly than in deep-sky astronomy (e.g., biosciences). An example of this is our use of workflows to link data sources and our workflow repository that allows queries to be re-run as the data in the data services changes.

In Section 2, we describe the e-Science challenge of heliophysics in more depth through a case study. In Section 3, we describe how we meet the challenges of cross-catalogue searches with non-trivial relationships between search regions and multiple data models. In Section 4, we describe the architecture we have built. In Section 5, we describe the scientific interface to the HELIO virtual observatory. In Section 6, we summarise the wider impact on eScience and how our methods will respond to technology developments (e.g., Cloud Computing).

[1] http://helio-vo.eu

2. Scientific case studies

2.1. Scientific setting

One of the key subjects studied by heliospheric physics is the release of large amounts of ionised particles, called plasma, that propagate through the heliosphere and interact with planetary environments. Particles are accelerated by large solar explosions called flares or by prominences that erupt and cause the ejection of "blobs" of plasma into interplanetary space. They are commonly known as coronal mass ejections (CMEs) [8]. Another phenomenon under study is called stream interaction regions (SIR) which forms when a fast solar wind stream overtakes a slower one [9]. SIR are representatives of a variety of perturbations in the solar wind ambient plasma. Both phenomena, CMEs and SIRs, originate by events produced at the Sun.

The analysis of these and similar phenomena involves the availability of multi-instrument and multi-wavelength data sampled at multiple locations. It also requires suitable propagation models so that it is possible to track the temporal and spatial evolution of phenomena with respect to triggering events (e.g., solar flares), interaction events during interplanetary propagation (e.g., particle beam acceleration and reflection at the shock front), and interaction processes with planetary magnetospheres and atmospheres (e.g., compression and energy transfer, particle injection). Hence, a scientist who wants to perform such an analysis has to:

1. Identify the ancillary heliospheric events that have concurred to determine the observational scenario focused on the primary heliospheric event of interest;
2. Identify the data sources;
3. Run propagation models to generate a time frame relevant to the occurrence of the various events;
4. Download the data sets;
5. Carry out the data integration;
6. Perform the physical modelling and interpretation.

Remote observations from ground- and space-based telescopes are usually needed because, besides detecting the event under study, they exhibit a contextual picture of the event. On the other hand, *in situ* multi-spacecraft observations at different locations play a fundamental role in characterising the propagating heliospheric event. Both kind of data are typically stored in dedicated archives, that can be accessed via web as standalone facilities or through portals which provide a common interface to different archives. There are virtual observatories that provide access to a variety of data sets, but they are usually sub-domain specific — i.e., they focus only on data relating to solar physics, space physics, magnetospheric physics, etc. Hence, the scientist had to manually follow the workflow outlined above, making the preparatory phase of the research in heliophysics a tedious and quite demanding activity. HELIO aims to simplify such tasks, helping also the discovery of new data-sets which otherwise the scientist would not be aware. A use case of the analysis of multiple events using HELIO is described below.

2.2. Case study: tracing a CME by auroral storms

CMEs are the largest-scale event in the heliosphere, they can carry up to 10^{13} kg of plasma at several hundreds of kilometres per second. They can produce auroral displays when they interact with planetary magnetospheres. However, the CMEs can cause other undesired effects to spacecraft and to the society at Earth (e.g., power-grid failures, disruptions in satellite communications and increased corrosion of pipelines to name a few) [10]. CMEs propagate through the heliosphere deforming the local magnetic field as they expand, thus making the exact prediction of their behaviour challenging. Figure 2 shows the complexity that entail the understanding and propagation of such events. The data used for this figure was generated by a simulation of a series of CMEs that erupted in October–November 2003, they were known as the *Halloween events*, and auroras were seen even in latitudes as low as 40° (e.g., in Barcelona, Spain).

The aim of the following use case is to study CMEs' properties that have been observed by *in situ* instruments first at Earth and then at Mars. To achieve such a study the scientist would like to analyse the properties measured, for each of the events, at the Sun at the time of the ejection, then the *in situ* properties when passed through Earth and Mars. Once all the data is gathered and integrated the scientist can proceed to study the properties of such event (e.g., mass, magnetic field, velocities,

[3] Image from HAFv2 model from EXPI [11, 12, 13].

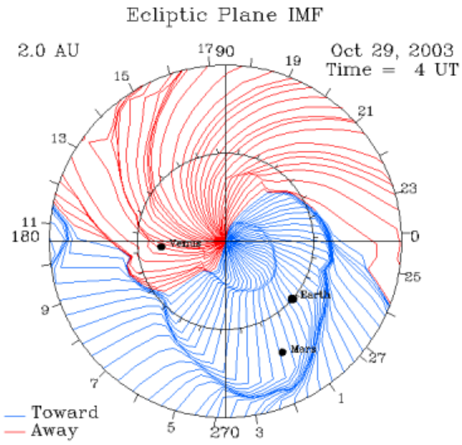


Figure 2: Instantaneous snapshot of a simulation of the magnetic field lines, showing distortions to the interplanetary magnetic field in the plane of the solar system caused by two CMEs during the *Halloween events*. Of particular note are the large distortions (from the close-to-ideal pattern in the top half of the figure) in field line directions due to the CME.[3]

etc.). A similar study has been already performed [14] and it is used here as a comparison.

This use case involves the usage of automated CME detection catalogues from coronagraphs (telescope attachment that creates an artificial solar eclipse by blocking out the solar disk allowing the observation of the outer corona orders of magnitude fainter than the disk). Also it includes observations of the solar disk from extreme ultraviolet or Hα telescopes to find out what solar features are correlated with the CME. Images or time-series in other wavelengths can also add value to understand the properties of such events (e.g., magnetograms, X-ray intensity increases). Radio observations can provide information of the density and the shock speed of the CME [15]. Few days after the CME erupted on the Sun and if it is heading towards Earth, the CME would be close to Earth, thus would be measured by *in situ* instruments orbiting around the Lagrangian L1 point (place between the Sun and Earth where their gravitational forces cancel each other). A couple of hours later, the effects of the CME on the Earth's magnetic field can be measured by ground-based magnetometers and be seen as aurora display. Finally, few days later it could be detected by *in situ* instruments orbiting Mars, assuming Mars and Earth are aligned in a suitable configuration.

As this example shows, data from many different sources had to be handled — i.e., identified, used in search requests based on dates and times according to the direction, speed and other properties of each CME. This process needs to be repeated for all the possible events which could hit Earth and Mars. It is a not-trivial process due to the intrinsic properties of the data, and the lack of full standardisation of the data descriptors because of the differences between the disciplines in which individual instruments were constructed.

HELIO provides to the scientist a common framework with an operational scenario for heliophysical data handling. Therefore, relieving the burden of data source identification and data integration as its web interface makes it possible to:

- place complex searches on multiple data repositories
- propagate diverse events to different points in the heliosphere
- use the new time-ranges to query for data at the relevant locations where detections of the event is expected to happen.

All this process is performed in a unified, user-transparent way, as reported in Section 5. This greatly facilitates the research, and creates a favourable operational environment for knowledge and data discovery. For example, the use case shown could be easily expanded to test whether any of the events found also hits Jupiter and/or Saturn, find whether there were *in situ* detections, and even if aurora displays were observed in either planet by astronomy telescopes, like Hubble Space Telescope [16].

3. The technical challenge of data in heliophysics

The technical challenges of data in heliophysics can be described as two major mapping problems. One is concerned with the mapping of events across space and time. This has to happen whenever an event is propagating from the Sun through the Heliosphere. The key problem is that different event types propagate on quite different paths and at very different speeds. Moreover, events can influence the propagation path and speed of other events. Scientists develop models for the propagation of events, but the complexity of the problem and the sparsity of instruments across the heliosphere makes this a challenging task. This problem will be discussed in detail in Subsection 3.1.

The second mapping problem is a semantic one. Heliophysicists use a variety of data formats, data dictionaries, and data models in their work. Depending on their special area of interest it is difficult to understand and use data products created by a group outside of that area. A lot of these data products do not contain enough metadata to enable scientists to interpret them without the assumed knowledge within the source community. Moreover, the phenomena being studied propagate from the Sun throughout the solar system at varying rates depending on their nature, which affects when and where the events associated with them should be searched for.

This is important because the HELIO infrastructure is not the only way for scientists to find and work with heliophysical data; different organisations (e.g., NASA, ESA) set up virtual observatories in which researchers can perform a subset of the functionality provided by HELIO. Which subset they cover depends on the concrete speciality these virtual observatories were designed to perform.

HELIO is designed to cross the boundaries between sub-communities within heliophysics. The project strives to provide services that can be used on their own but are also easy to integrate with one another and compose together to perform more complex search tasks. The data provided is encoded in the VOTable [17] format, an XML representation of tabular data that is widely used in the astronomy and astrophysics communities, which allows it to be rich in metadata, contain a full provenance trail, and be annotated using the community standards UCD [18] and UType. The UType attribute in particular provides a reference into a data model, which is key for driving semantic matching. The VOTable format can be interpreted by a large number of tools that the heliophysics community is already using to analyse their data. It was designed for the exchange of data and provides the necessary metadata fields to enable clear description of what the meaning of the data in the table is, even long after the query that generated the table was performed. In particular, it captures two notions of type associated with each table column:

1. The units that the data is expressed in, and the coordinate system that the data was measured within.
2. The higher-level meaning of the data; its intended interpretation.

We are also aiming to bridge the gap arising by different data standards in different communities within heliophysics by creating an ontology that maps terms from these data standards to each other.

3.1. Event propagation

The majority of the scientific requirements come down to the need to find correlated events, where the search space is non-trivial. In particular, there is a need to search for matching events that are occurring at locations other than the search anchor, the site of original interest. Moreover, because the majority of causes of events take time to propagate through the solar system, the time that those events at other locations could be expected to occur will be different to the anchor time.

Propagation paths and speeds are dependent on the kind of events and on events which occurred in timely proximity of the event of interest. So a CME event happening around the time of a flare changes the magnetic field lines on which the particles of the flare event are propagating and wherefore influencing the distance, direction, and the arrival time.

The final significant complication is that the propagation velocity is never known precisely, and is in fact in principle unknowable (without actual measurement) due to the natural variability of the Sun; though it is possible to apply magnetohydrodynamic modelling to the propagation problem (Figure 2 illustrates the output of such an exercise) this is not generally practical due to it being unrealistically difficult to determine the initial conditions with sufficient precision to be useful.

In practical terms this means that searching for correlated events is a matter of propagating the anchor event back to the Sun, iterating over the possible (and scientifically-interesting) types of event sources, propagating the event forwards to the site of an instrument that could have detected the event, seeing whether the instrument was actually observing at that time, and then looking to see if an event was actually captured within a reasonable time window (see Figure 3). Besides matching events from different catalogues which could be done with shifting time intervals depending on the location of the events compared, these steps could need some other requirements that depend on the event being studied. For example, some events are co-rotating with the Sun's rotation (≈ 27 days), therefore we could

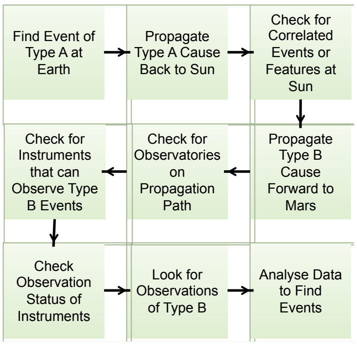

Find Event of Type A at Earth
Propagate Type A Cause Back to Sun
Check for Correlated Events or Features at Sun
Propagate Type B Cause Forward to Mars
Check for Observatories on Propagation Path
Check for Instruments that can Observe Type B Events
Check Observation Status of Instruments
Look for Observations of Type B
Analyse Data to Find Events

Figure 3: Steps involved in the integrated use of the propagation model with the information catalogues.

be interested in a previous or later rotation time in order to study either the changes observed by their origin in the Sun or the recurrence of the event on a certain place in the heliosphere. If done without supervision and with the actual information available, this introduces a lot of uncertainty, particularly as different types of events propagate at different rates, from the 45 000 km/s (15% of the speed of light) for the flux of protons at 10 MeV, to speeds of 400–600 km/s for the thermally-generated slow solar wind. Electromagnetic radiation is propagated radially at light speed as the gravitational or magnetic field effects can be negligible for most of the cases studied in heliophysics. The uncertainty is compensated for by expanding the search volume and lengthening the search time-slot; in effect, this is attaching an estimate of error to the spacetime location that the propagation model itself produces.

The Solar-Heliospheric Event Ballistic Algorithm (SHEBA) [19] has been developed within HELIO as a first approximation to solve the propagation problem. This algorithm, built on top of HELIO services, offers three different propagation modes (i.e., CMEs, SIRs and high energetic particles) through a simple 2D ballistic propagation model [20]. SHEBA allows to run the propagation in two directions, from the Sun outwards into the heliosphere, and from any object (planet or spacecraft) backwards to the Sun. Though SHEBA is a basic propagation model, it has shown satisfactory results for non-complex events (e.g., an individual CME propagating in a quiet environment) [19].

It should be noted that this propagation model is itself of considerable scientific interest and will be so in the future. In particular, complex simulations can provide properties of the plasma cells propagated (i.e., density, magnetic field, velocity) [21, 22] as well as time ranges of arrival. There is a clear demand from the heliophysical community to have the ability to replace one version of the propagation model with another.

3.2. Data models in heliophysics

The services in the HELIO system (see Section 4) are designed to work on their own, but in order to perform more complex tasks the user needs to execute these services in combination with each other, using the outputs of one service in defining the criteria for a call to another. Our primary instrument for such linking is the Taverna workflow system which is described in more detail in Section 4.6. We have created an overarching data model in which we define the content of the services, ensuring that the data resulting from one query will be semantically compatible with subsequent queries.

All HELIO services produce their output in VOTable format, which was designed for the exchange of data in tabular form by the IVOA (the standard body for virtual observatories). It is not specific to any content and does not contain any requirements of the description of the content. Even though the format can be read by a number of applications, only the author of such a file can make sure the content is meaningful to the recipient; this is enabled through the presence of two hooks in the VOTable format that are used to attach semantic content to the data. The first one is the “UCD”, which is a list of terms from a controlled vocabulary, though the current version of UCD, 1.23, does not contain appropriate terminology for heliophysics and the terms that are included do not provide the granularity required for a satisfactory mapping between tables. Since VOTable version 1.1, we can provide references into our own data models through the use of the second hook, the “UType”. That means we can create a data model describing the data exactly to the level of detail required to enable the use of the content between services.

An analysis of the existing data models in heliophysics showed that there are few well-defined data models, of which the most widely used data model is SPASE [24]. However, SPASE defines the structure of the data but does not deal with the meaning of the content, and the resulting UType references in a VOTable would be meaningless.

The HELIO data model was therefore constructed *de novo* with the idea to represent the semantics of the underlying data. Fields with the

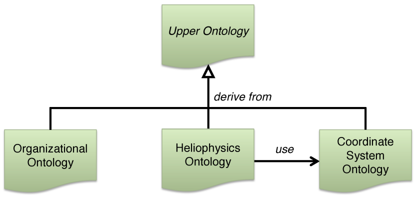


Figure 4: The structure of the ontologies, showing how they relate to each other with an upper ontology providing common concepts allowing the ontologies to be used together.

same content should have the same UType tag in the VOTable no matter which service has produced them. The resulting data model should not only be usable by the HELIO services, but should be well enough structured to be also easily usable by other community data providers. By creating a new data model, we run the risk that it will not be used outside of this project, but our hope is that by tackling the semantics of the data in ways that other data models did not do, we can reach adopters outside our group.

3.3. A heliophysical ontology

Different parts of the heliophysics community use different data models, data dictionaries, and file types to describe, store and exchange their data. These different data products were developed completely independently of each other, and often use different keywords to describe what the same thing is physically.

We addressed this through creating an ontology describing the whole heliophysics discipline. In the first stage of this process, we created an "Upper Ontology" that contains the basic concepts used in heliophysics, so creating a semantic skeleton for the science. It is logically structured (see Figure 4) in a way which makes maintaining the parts easier and consists of:

Organisational Ontology This contains the structure and properties of data, infrastructure and people.

Coordinate System Ontology This contains classifications of coordinate systems, and parameters relating to coordinate systems.

Heliophysics Ontology This contains the domain concepts of this community. It uses the Coordinate Systems Ontology.

Upper Ontology This includes all concepts of the previous three ontologies, and adds properties which bind concepts of the different subontologies together.

In the second stage, we used the "Upper Ontology" to map terms from the SPASE data model, the PDS data dictionary [26], the EGSO data model [27], and the HELIO data model onto that structure. We created individual annotation types for each of the data products and, where appropriate, hierarchies of annotation types. This enables us to use different levels of detail in the integration. Annotations have the advantage that they can be created for both classes and individuals, but a problem is that common ontological tools cannot reason over them, which needs to be considered when queries are constructed. The resulting ontology (see Figure 5) can be queried for terms in these different data products that represent the same concept or a related higher or lower level concept. Of course, the ontology can only provide these terms for areas where these data products actually cover the same ground, such as in the terms used for spacecraft names. Beyond the mapping of terms, the ontology can also inform about the concepts used in heliophysics and their relation to one another.

The ontology is integrated into a Semantic Mapping Service, which is a web service providing a SOAP interface to its functions (a part of the metadata category described in Section 4.2). This service allows the ontology to be integrated into other the parts of the HELIO system, or to provide semantic mappings to workflows.

4. The architecture of the HELIO infrastructure

4.1. Overall architecture

The HELIO infrastructure is based on the concepts of a Service Oriented Architecture [28]. SOAs feature a set of loosely coupled components, and so have two main advantages for HELIO:

1. The components can be deployed redundantly at different locations, increasing the overall stability of the system.
2. The components can be developed independently at different locations by different teams, so supporting the distributed nature of the project consortium.

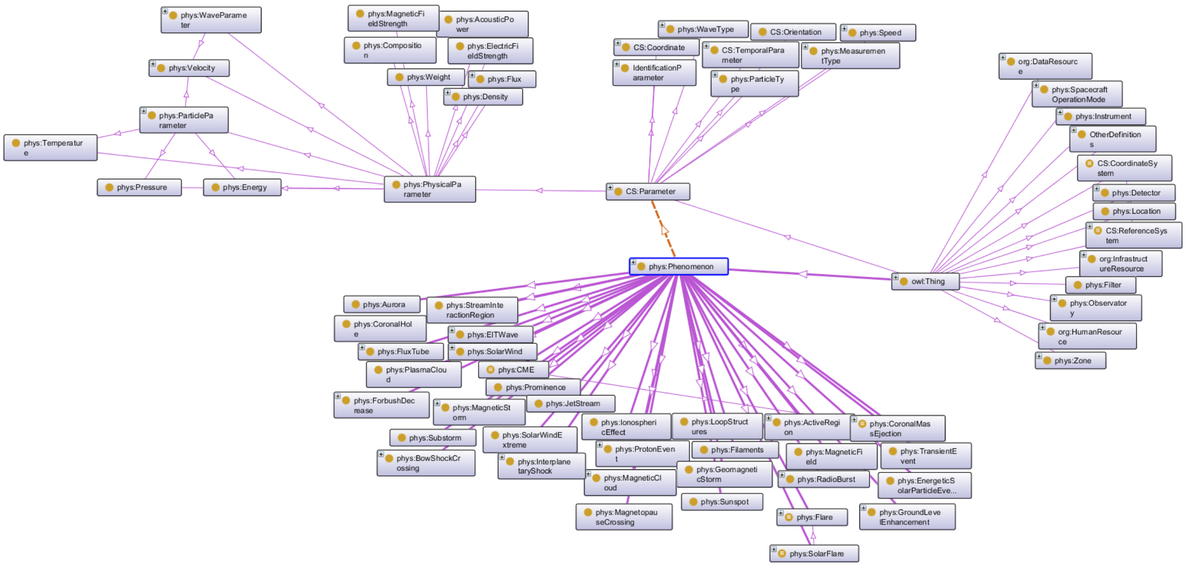


Figure 5: The classes in the HELIO ontology [25], together with selected types of inter-class relation (principally "is-a").

Another key aspect of the use of a SOA is that the primary data resources (notably images and spectra) are large and generated at a high rate[4] so keeping the catalogues describing that data close to the depository institutions minimises the number of large transfers that need to occur. This naturally leads to distributed catalogues due to the substantial number of organisations participating in heliospheric-related research.

The other principles used in the design of the HELIO architecture [30] were that services should, to as great an extent as possible, permit multiple access methods (minimally including both integrated and standalone modes), that no particular client or workflow system should be specially favoured, that the system should respect the security policies of the service providers to as great an extent as possible (a real issue when some processing services require significant resources to operate), and that as much non-scientific information as possible should be hidden from the scientific users (i.e. that technical details of logins, data location, etc. should be kept shrouded unless specifically requested).

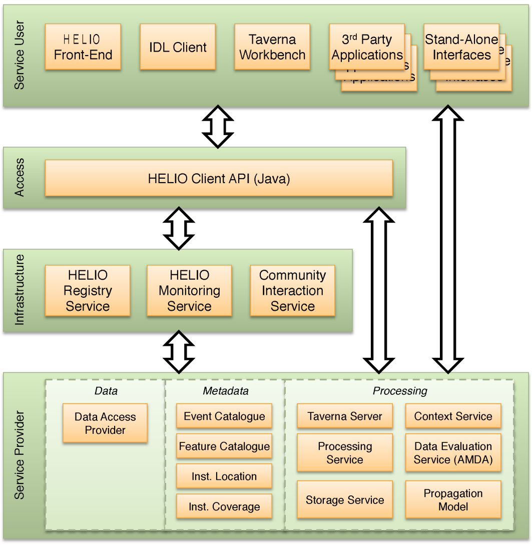


Figure 6: Structural view of the main components involved in the HELIO infrastructure. The arrows denote communication flows.

Figure 6 shows the conceptual architecture of the components involved in the HELIO infrastructure. The diagram is divided into four main areas.

Service Provider Components that implement services providing access to data, metadata, on-demand processing and storage capabilities.

Infrastructure Components that are required for management, maintenance and security han-

[4] A *single* space-based observatory such as the SOHO satellite can produce 0.5Gb/day continuously [29]. Ground-based observatories can have much higher data rates. There are over 50 observatories, with over 200 instruments producing many different types of data, and with collection happening over many decades; the oldest complete datasets start in the nineteenth century.

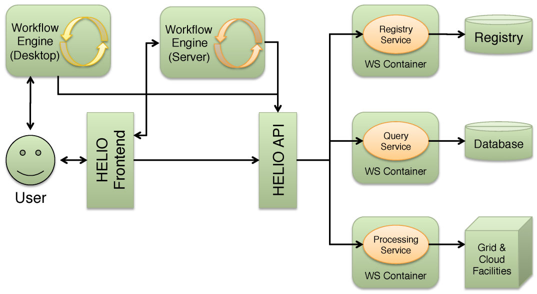


Figure 7: Information flows in complex HELIO usage, showing how a workflow engine may be used within the front-end, or instead of it.

dling of the HELIO infrastructure. Consumed principally by the access layer.

Access Responsible for connecting and integrating the underlying services and for facilitating access to the infrastructure for different "service user" components. This layer handles security, failover, service resolution, etc.

Service User Components that provide the interface between human beings and the underlying infrastructure.

Although in simple usage, information proceeds along the major flow directions identified in Figure 6, this is not the only way in which things can work. For example, Figure 7 shows more complex interactions that can exist when a workflow server is in use. As can be seen, users can connect to a centralised Graphical User Interface (the HELIO Front-End, see Section 5) that uses an instance of the Taverna Server (see Section 4.6), as well as using a local instance of a Taverna Workbench [31] to define workflows. They can also access the HELIO API directly through Java [32] or IDL (Interactive Data Language [33]) code. Finally, some of the services also offer standalone graphical user interfaces that offer advanced functionality not available in the HELIO Front-End.

Heliophysical data processing can potentially involve substantial amounts of data processing and computation, so it is important principle of the architecture that the resources allocated to any particular piece of processing or storage are limited and will be eventually reclaimed if not in use. This is done according to the principles set out in [34] with the level of resources assigned to a user according to their identity[5] and with the resources being de-allocated after a time-out. The length of time-out depends on the nature of the resource; the results held in the database that backs up the front-end website typically have a longer life-span than the processes spawned by the HELIO Processing Service or the workflow runs created by Taverna Server.

4.2. Service provider components

The service provider components are divided into three categories: data, metadata and processing.

The *data* category contains the Data Provider Access Service (DPAS), which provides uniform access to a multitude of archives with data about heliospheric observations. The DPAS implements connectors to various types of archives, such as FTP- and HTTP-archives, web services, relational databases and virtual observatories outside the heliophysics field.

The *metadata* category contains services for accessing secondary catalogues and other types of metadata. These catalogues include collections of events observed on the Sun and in the heliosphere, features of the Sun as they evolve over time (e.g., filaments, active regions, coronal holes), and descriptions of what instruments were observing and where they were located at the time.

Access to the metadata catalogues is given primarily through the HELIO Query Interface (HQI), a common interface implemented by all HELIO metadata services that supports both REST and SOAP styles of use. Service users may use either interface style to get the same results. Conceptually, it supports a parametric query style to query tabular data; parametric queries are best suited to express and implement cases where the data model is sufficiently well defined. The results of the queries are formatted as VOTables.

The *processing* category holds services for on-demand processing of data or metadata. Depending on the scientific question asked to HELIO some information cannot be prepared in advance but has to be computed based on given parameters. HELIO provides several types of processing components:

[5] As a matter of policy we permit limited use by anonymous users, only accessing small fractions of the total datasets available and running only pre-determined processing operations that are known to consume resources to a level acceptable to the hosting organisation.

Taverna Server A workflow engine suited to combine multiple services into a more complex workflow (see Section 4.6).

HELIO Processing Service The HPS provides access to a high performance computing infrastructure, used for resource intensive data processing such as image analysis.

HELIO Storage Service The HSS acts as utility service for the HPS and other services to store large result sets.

Context services These give access to predefined plotting services, e.g., to create a timeline plot of solar activity for a given date range.

Data Evaluation Service This acts as an interface to the Automated Multi Dataset Analysis (AMDA) infrastructure [35], which provides a collection of tools to access and analyse heliophysical data.

Propagation Model This simulates the propagation of the effects of a solar event through space and time, allowing the scientist to relate observations made at different locations and time to the same event.

4.3. Infrastructure components

The infrastructure category provides helper services for the management of the HELIO infrastructure; these components are usually transparent to the end user and do not provide any information of scientific value.

The *HELIO Registry Service* (HRS) is a directory service to enable service discovery. Additionally, it provides information on how to use the services. The *HELIO Monitoring Service* (HMS) monitors the system by frequently polling the status of the service. In combination with the HRS it provides the failover and load balancing capabilities of the infrastructure.

The *Community Interaction Service* (CIS) implements the basement for authentication in HELIO. Moreover, it manages user profiles in a central place.

4.4. Access component

Depending on their needs client applications may choose to directly access individual HELIO services or they may use the Java-based HELIO API. The HELIO Java API facilitates access to the system by shielding users from the underlying infrastructure. It offers:

1. Transparent discovery of services,
2. Load balancing and failover (through the use of the HRS and HMS),
3. Automatic handling of security and user profile management,
4. Client stubs to access different service providers in a uniform way, and
5. Utilities to combine services to solve more complex tasks.

4.5. Service user components

We support user access to HELIO services through multiple methods. The principal ones are:

HELIO Front-End The HFE provides an integrated browser-based interface to the HELIO services; it allows users to perform common searches and data retrieval actions in a user friendly way (discussed in more depth in Section 5). The HFE is meant for novice users that need only little or even no knowledge about the services and the data model provided by HELIO.

Taverna Workbench This allows users to define custom workflows for their own specific scientific use cases using a visual composition and configuration environment. It also supports the sharing of these workflows through social media [36]. Thus, Taverna users can start from an existing workflow and adapt it to their needs. This allows even inexperienced users to use HELIO for science.

HELIO IDL Client This enables access to HELIO through the IDL scripting environment. With IDL, users can interactively communicate with the HELIO system, and in this way combine the HELIO capabilities with advanced data analysis tasks. The HELIO IDL client is meant for expert users that have deep knowledge of the HELIO data model.

Stand-alone Interfaces Most HELIO services have their own stand-alone interfaces, allowing them to be directly used over the web without integration with any other system. The standalone interfaces are mainly intended for development and test purposes and to better understand certain aspects of the system.

We also support third-party application access, so that other virtual observatories, data warehouses, and graphical client applications can integrate with HELIO.

4.6. Workflows in HELIO

HELIO uses Taverna [31, 37] as its exemplar workflow system, as it provides a relatively simple mechanism for orchestrating multiple services into a single unit of processing. In addition to supporting the use of the Taverna Workbench, HELIO has an installation of Taverna Server which can execute workflows that have been created through the Workbench and stored in the myExperiment workflow repository [36, 38]. These stored workflows[6] are annotated with metadata that enables them to be automatically exposed to users through the HELIO Front-End, enabling workflow use and reuse without the users having to install a complex piece of software like the Workbench.

The workflows are principally comprised of processing elements that access HELIO's services (especially the query interfaces) via SOAP method invocations, interleaved with extra processing elements to extract and combine results. One example of this (see Figure 8) is a composite query which takes a time period identified by the invoking user, during which they want to search for correlated features (coronal holes, etc.) and events (e.g., X-Ray flares) originating from the same part of the Sun (i.e., within a certain distance across the sun's disk). In the workflow diagram different colours represent different types of operators. Green ones are SOAP services, pink operators are nested workflows (standalone workflows which are included to provide a specific functionality), brown operators are scripts to do operations within the workflow (such as string operations to transform a output of a service to the required input patterns for a next service), blue operators are string constants, and purple operators perform the composition and decomposition of SOAP messages. In general, a SOAP service element could be accessing any web service, but within this paper they are used exclusively to access metadata catalogues that follow the HQI specification of Section 4.2.

Architecturally, Taverna Server is a web service that is hosted within a Java web container. The server provides job and file management where the jobs are specialised to executing workflows created by the Taverna Workbench. Workflows are executed in a different local user account for each distinct user through the use of an impersonation module, allowing for the application of per-user security policies and accounting. Since it is necessary to use impersonation to achieve this execution model, the server installation is not shared with any other services. Security tokens are mapped transparently from the service call context to a parameter in the execution environment so that the (limited) number of workflows that need to access secured services can do so automatically. Note however that the large majority of heliophysics workflows in use (currently including all those in myExperiment) only use public services.

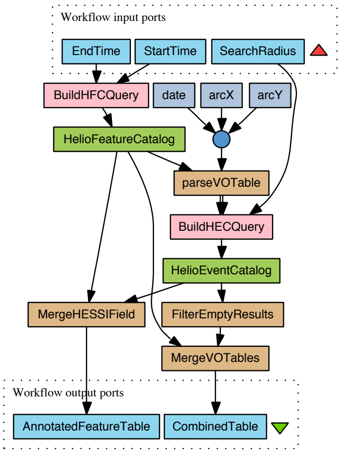


Figure 8: Taverna workflow for querying the HELIO Feature Catalogue and HELIO Event Catalogue and merging the results. [39]

4.6.1. CME science case as a workflow

The workflow in Figure 9 creates a VOTable based on events in the CACTus SOHO CME catalogue [41, 42] which is filtered to events which could have been seen at Earth and at Mars. Only events which happen at a time when Earth and Mars are not further apart than 80 degrees have a possibility of effecting both planets. The HELIO Instrument Location Service contains information about the position of spacecraft and planets which we use to query for suitable time periods. A database query to this service reduces the initial search period. The time periods need some adjustment for the time a CME would need to arrive at the planets. Therefore we perform a backwards propagation of the start

[6] There are 46 shared workflows at the time of writing.

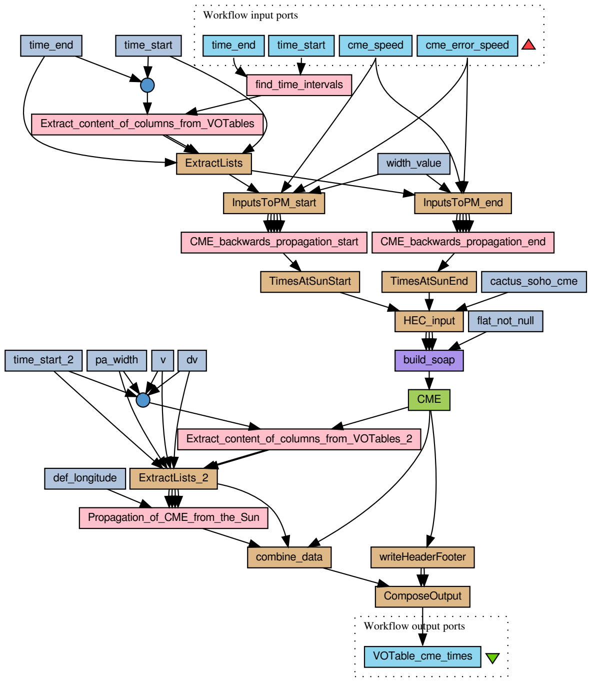


Figure 9: Taverna workflow to select CME events which could have been observed at both Earth and Mars. [40]

and end times using the HELIO processing service running the CME backwards propagation model. The times generated we use to query the CACTus SOHO CME catalogue of the HELIO Event Catalogue service for Halo events. Halo CME events are events which propagate forwards to Earth. The resulting list of events gets forward propagated and only events where the propagation model predicts a path including both Earth and Mars are used to generate the final VOTable. The VOTable includes all fields from the event catalogue and the predicted times at Earth and Mars. With this information the scientist can look for signatures of CMEs in the data of the appropriate instrument and time. This is a next step for which the Data Evaluation Service could be employed.

The construction process of a workflow can be split down into larger steps which can be developed separately and can be included in the complex task of a workflow. In this way, asynchronous service calls (e.g., calling the processing service) can appear as a single step in workflow diagram (represented as pink boxes in Figure 9) making the process easier to understand and workflow parts easier to reuse. Automatic re-tries of failed service calls make the process more robust. Looping over datasets removes a level of complexity as well, as the operators only need to handle as simple types as required for the operation, and enables easy parallelisation of operations.

5. The user interface

5.1. The design challenge

The major challenge of the user interface has been to support the concurrent selection and combination of data from many instruments so as to support heliophysics research such as described in Section 2. Though many user interfaces exist already to perform single steps of the study, it is generally recognised that none of them provide an easy way of composing different analyses to answer larger research questions. Former projects, such EGSO [43], did provide an embryonic integrated user interface, but the previously existing systems did not fully support the way scientists want to work. As such, a major rethink was required in order to increase the acceptance of the HELIO user interface approach.

This HELIO approach is to present the underlying capabilities as interface modules within a web portal with a common look and feel. The interface modules can be chained together as the user requires. The chaining is based on the common underlying data model while access to the underlying services is shielded from the user. The principal focus of the user interface is on the catalogue entries, images and spectra that constitute the heliophysical observations. The approach also provides means for customisation of the interface to specifically support sub-communities of heliophysics. Individual modules can be adapted to use updated algorithms and additional data sources, or they can be restricted to provide only the core subset of the functionality of a module.

The user interface was made available to our testers early in the development process, and frequently updated according to user feedback. Their experience and feedback was essential for ensuring that we support the methods of working of such groups. Moreover, this helps to promote the adoption of the interface by the wider heliophysics community. This is essential to ensuring that HELIO produces a sustainable suite of practical working tools for new scientific discovery.

Two of the major challenges for the design are:

- The user groups within the heliophysics community and their needs (and expectations) are highly heterogeneous. Therefore, the user interface has to provide multiple different types of search task (e.g., searching by time and searching by location) and both user-guided and system-guided interaction styles.

- There are many heterogeneous data sets from different repositories, as well as complex search results, and their relationships have to be represented in a comprehensible way. Following Shneiderman's "Visual Information seeking Mantra" [44], we use representation techniques supporting overview, zoom/filter and details-on-demand visualisation that differ from the conventional user interfaces previously used in the heliospheric domain.

Therefore, the user interface provides techniques for the search tasks (where the focus is on how to present an input search range, how to manage the result, and how to pick a subset of the results) as well as tools to navigate through the data sets (with a particular focus on interactive visualisation of the data sets through fetching appropriate previews).

5.2. Implementation

The user interface is realised as the *HELIO Front-End* (HFE), a Rich Internet Application [45] written in Javascript [46] that provides access to the underlying HELIO API (see Section 4.4) and which mimics much of the capabilities that would be expected in a desktop application through the use of AJAX techniques [47]. This is in line with the recent trend of several major IT companies to abandon support for third party browser plugins such as Adobe Flash [48] or Microsoft Silverlight [49] and to shift towards modern HTML5 [50] technologies.

The HFE is centred around the data — which may either originate from catalogues within the system or from an uploaded VOTable — and the tasks performed on it. This distinguishes the HFE from both normal web applications, which are more workflow-oriented, and traditional scientific systems, which are function-oriented. In a function-oriented approach, input data is feed into a function, processed and new data is generated. Function-oriented applications are more suited for data processing, while HELIO focuses on data retrieval and exploration. Data processing is performed outside of HELIO after interesting data has been found.

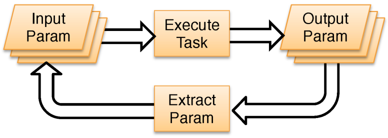


Figure 10: Flowchart diagram of a HFE task. Input parameters are fed into a task which creates output parameters. An extraction step allows to derive input parameters for a new task from the output.

In a data-centric system, the data processing is done at an abstract level from the user's perspective. For a given data product, the user is presented with a set of tasks that can be applied to this data. These tasks are presented in a domain specific, but natural language like: "Get observations for a given time range" or "See what instruments covered this period" rather than as potentially cryptic function names. This data-centric approach supports the novice users to perform common tasks without having deep knowledge of the detailed science, while not preventing more advanced users from working with the data.

Any task accessible through the HFE follows a generic user interaction pattern:

1. Select a task to be executed. This is driven by identifying the scientific question to be addressed.
2. Configure the input parameters required for the selected task. The input parameters may be entered manually or they may be reused from a previously executed task. Generally, tasks have sensible defaults for most input parameters.
3. Execute the task on the HELIO infrastructure.
4. Visualise the output of the task in order to enable users to analyse it and to extract new input parameters from it. The extracted parameters can then be used for succeeding tasks.
5. Continue with step 1

This pattern allows the HFE to be designed and implemented in a modular way. It is reflected in the flowchart diagram in Figure 10.

5.2.1. Workflows in the HELIO Front-End

HELIO tasks cover many areas such as to retrieve data or metadata, join data tables, upload a VOTable and access plots or quicklook images. Of particular interest is the group of tasks to run processing services like Taverna workflows. Typically a task is executed iteratively, i.e., the task is first run with default input parameters, which are then progressively adjusted to improve the result.

The input parameters of a task may be entered manually, may come from a HELIO service such as the event or feature catalogue or they may be read from an external data source through an uploaded VOTable. The input parameters are divided into two categories: cross-search parameters and task-specific parameters.

Cross-search parameters are the core parameters of the heliophysics domain. Typically, they are used to connect the output of one task with the input of another. Examples of these parameters are time ranges, locations, instruments and event lists.

Because there are only a few cross-search parameters, we were able to provide a custom tailored input dialog for each of them. Some of these dialogs even provide several input modes. For instance, a time range can be entered by choosing a start and end time from a calendar widget or by marking the range in a time line plot showing the activity of the Sun.

Task-specific parameters configure the behaviour of a particular task but do not generally make any sense when shared with other tasks. In HELIO, task-specific parameters are further restricted to be of simple type, such as numeric, boolean, string or an array of a simple type. This allows us to automatically generate a generic dialog for capturing these values and like this significantly reduce development costs. Figure 11 shows a sample input dialog for parameters specific to a propagation model task.

Once the input parameters for a workflow task are specified the HELIO Front-End sends these parameters to the Taverna Server for execution of the workflow. The HFE polls the Taverna Server until a result is ready for download.

If the result contains tabular data encoded in the VOTable format, the HFE will visualise it such that a scientist can filter out data relevant to a scientific question. This work needs to be done manually by selecting scientifically meaningful rows from the tables. Only cross-search parameters can be extracted, which greatly reduces the complexity of the HFE while still addressing the majority of the use cases.

If the result is comprised of raw data such as FITS [51] files, PNGs or custom formats the HFE offers tools for viewing and downloading the data. The actual processing of such data is beyond the scope of HELIO and will be done in dedicated data analysis software packages.

At a higher level, the HFE interface is focused around representing results from previous queries as graphical data items that can be dragged and dropped into different parts of the user interface. This provides a direct method for users to compose queries together in an *ad hoc* mechanism to explore their scientific questions.

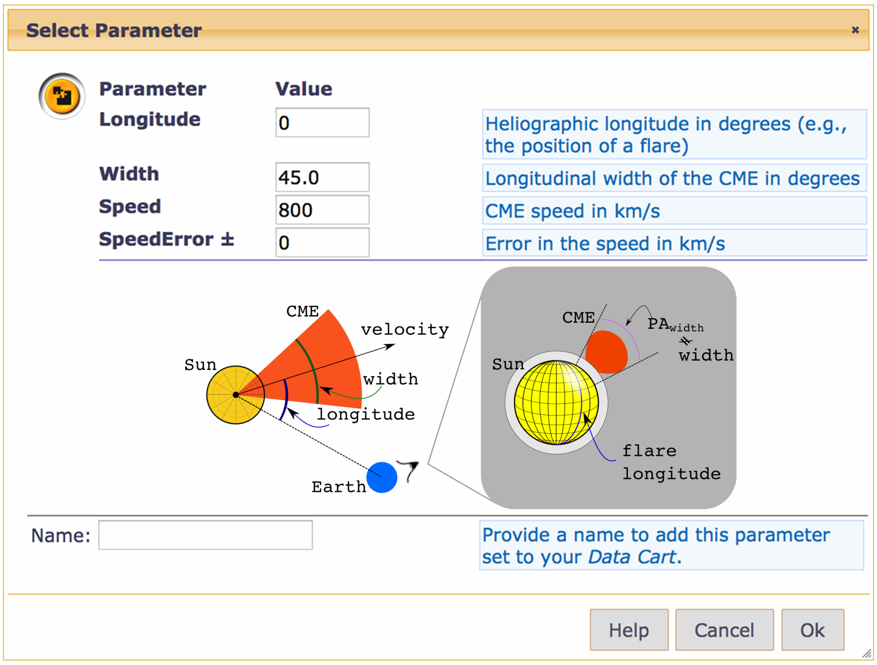
Integration of new workflow tasks in the HFE is currently a manual process. The main reason for this is that most registries used by the different service providers do not provide native support for UI-specific configuration parameters; for example, the annotations supported by the T2FLOW format used by Taverna is slightly too limited to support the type metadata that would be required for automated UI construction[7].

6. Conclusions and future work

The HELIO infrastructure is largely complete and key use cases are being deployed. The community consultation is proceeding via a series of workshops in which the requirements of the heliophysicists are being mapped onto the services and the services are linked in dynamic workflows that execute across a back-end infrastructure that transparently uses Grid and Cloud resources. The workflows represent a key resource for the community, just as they do in other disciplines and are shared via the myExperiment repository [36]. We have been careful to reuse previous work by the virtual observatory and eScience communities, and we believe that our success in doing this is a mark of the progress of eScience to becoming a more mature field of research. As a result, we are able for the first time to address the whole nature of heliophysics.

The current status of the system is that all services are working and are available via the HELIO Web site. Half the services are currently replicated on geographically disparate servers for failover and resilience. The workflow repository on myExperiment currently has 74 items, mostly workflows (62) but also the HELIO ontology and explanatory presentations. On this site it can be seen that workflows have been downloaded between 90 times from

[7] SCUFL2 *will* be able to support sufficient richness of annotation, but was not available in time for us to use as a development target.



Select Parameter

Parameter	Value	
Longitude	0	Heliographic longitude in degrees (e.g., the position of a flare)
Width	45.0	Longitudinal width of the CME in degrees
Speed	800	CME speed in km/s
SpeedError ±	0	Error in the speed in km/s

Name:

Provide a name to add this parameter set to your *Data Cart*.

Figure 11: Generated dialog for input parameters specific to the propagation model. Note particularly the explanatory text for each of the parameters, and the context illustration below.

the most accessed down to workflows that have less than 5 downloads. The project has held four Coordinated Data Analysis Workshops during the software has been tested on realistic science queries and explorations. Performance testing of individual services indicates that the system can handle of the order of 100 concurrent queries with an acceptable query return time of less than ten seconds. Thus the status of the system is that has been tested by a modest number of users but those that have tested it have performed worthwhile scientific tasks and posters and papers based on use of the system have been presented at workshops and conferences on geophysics, space weather and solar system science. Part of future work will be to continue to engage the heliophysics communities and to record statistics and usage via the web site and the workflow repository.

Studying the heliophysics discipline in a systematic manner will bring new challenges, and methods will be developed that can be applied in other data-centric sciences. Future work will integrate the data gathered from observations with models of the energetic processes of interplanetary space, allowing for example the models to be continuously calibrated with data in a similar manner to data ingestion in weather forecasting.

Acknowledgment

The Hakamada-Akasofu-Fry Solar Wind Model version 2 (HAFv2) real-time forecast products (see Figure 2) were provided by Exploration Physics International, Inc. (EXPI). This work was funded by the European Union as part of the Seventh Framework Programme (FP7/2007–2013) under grant agreement No. 238969.

[1] J. Kappenman, L. Zanetti, W. Radasky, Geomagnetic storms can threaten electric power grid, Earth in Space 9 (1997) 9–11.

[2] R. Langley, GPS, the Ionosphere, and the Solar Maximum, GPS World 11 (2000) 44–49.

[3] W. Fricke, Definition of the celestial reference coordinate system in fundamental catalogues, in: IAU Colloq. 26: On Reference Coordinate Systems for Earth Dynamics, volume 1, pp. 201–222.

[4] A. Szalay, J. Gray, The world-wide telescope, Science 293 (2001) 2037–2038.

[5] R. Bentley, A. Csillaghy, J. Aboudarham, C. Jacquey, M. Hapgood, K. Bocchialini, M. Messerotti, J. Brooke, P. Gallagher, P. Fox, et al., HELIO: The Heliophysics Integrated Observatory, Advances in Space Research (2010).

[6] I. Foster, H. Kishimoto, A. Savva, D. Berry, A. Djaoui, A. Grimshaw, B. Horn, F. Maciel, F. Siebenlist, R. Subramaniam, et al., The open grid services architecture, Version 1 (2005) 1–61.

[7] S. Krishnan, P. Wagstrom, G. Von Laszewski, GSFL: a workflow framework for grid services, Preprint

ANL/MCS-P980-0802, Argonne National Laboratory 9700 (2002).

[8] P. F. Chen, Coronal mass ejections: Models and their observational basis, Living Reviews in Solar Physics 8 (2011).

[9] L. Jian, C. Russell, J. Luhmann, R. Skoug, Properties of stream interactions at one AU during 1995–2004, Solar Physics 239 (2006) 337–392.

[10] T. Pulkkinen, Space weather: Terrestrial perspective, Living Reviews in Solar Physics 4 (2007).

[11] K. Hakamada, S. Akasofu, Simulation of three-dimensional solar wind disturbances and resulting geomagnetic storms, Space Science Reviews 31 (1982) 3–70.

[12] C. Fry, W. Sun, C. Deehr, M. Dryer, Z. Smith, S. Akasofu, M. Tokumaru, M. Kojima, Improvements to the HAF solar wind model for space weather predictions, Journal of Geophysical research 106 (2001) 20985–21.

[13] C. Fry, M. Dryer, Z. Smith, W. Sun, C. Deehr, S. Akasofu, Forecasting solar wind structures and shock arrival times using an ensemble of models, Journal of Geophysical Research 108 (2003) 1070.

[14] T. V. Falkenberg, A. Taktakishvili, A. Pulkkinen, S. Vennerstrom, D. Odstrcil, D. Brain, G. Delory, D. Mitchell, Evaluating predictions of ICME arrival at Earth and Mars, Space Weather 9 (2011) S00E12.

[15] M. Reiner, M. Kaiser, J. Bougeret, Coronal and interplanetary propagation of CME/shocks from radio, in situ and white-light observations, The Astrophysical Journal 663 (2007) 1369.

[16] R. Prangé, L. Pallier, K. Hansen, R. Howard, A. Vourlidas, R. Courtin, C. Parkinson, An interplanetary shock traced by planetary auroral storms from the Sun to Saturn, Nature 432 (2004) 78–81.

[17] F. Ochsenbein, R. Williams, C. Davenhall, D. Durand, P. Fernique, R. Hanisch, D. Giaretta, T. McGlynn, A. Szalay, A. Wicenec, VOTable: Tabular data for the Virtual Observatory, in: P. Quinn, K. Górski (Eds.), Toward an International Virtual Observatory, volume 30 of *ESO Astrophysics Symposia*, Springer Berlin / Heidelberg, 2004, pp. 118–123. 10.1007/10857598_18.

[18] A. Martınez, S. Derriere, N. Gray, R. Mann, J. McDowell, T. Mc Glynn, F. Ochsenbein, P. Osuna, G. Rixon, R. Williams, The UCD1+ controlled vocabulary, IVOA Semantics WG Recommendation (2005).

[19] D. Pérez-Suárez, S. A. Maloney, P. A. Higgins, D. S. Bloomfield, P. T. Gallagher, G. Piearatoni, X. Bonnin, B. Cecconi, V. Alberti, K. Bocchialini, M. Dierckxsens, A. Opitz, A. Le Blanc, J. Aboudarham, R. B. Bentley, J. Brooke, B. Coghlan, A. Csillaghy, C. Jacquey, B. Lavraud, M. Messerotti, Studying Sun-planet connections using the Heliophysics Integrated Observatory (HELIO), Solar Physics 280 (2012) 603–621.

[20] E. Parker, Dynamics of the interplanetary gas and magnetic fields, The Astrophysical Journal 128 (1958) 664.

[21] H. Xie, Cone model for halo CMEs: application to space weather forecasting, Journal of Geophysical Research 109 (2004) 1–13.

[22] A. Aran, B. Sanahuja, D. Lario, SOLPENCO: a solar particle engineering code, Advances in Space Research 37 (2006) 1240–1246.

[23] M. Ohishi, International Virtual Observatory Alliance, Proceedings of the International Astronomical Union 2 (2006) 528–529.

[24] J. Dickinson, et al., SPASE collaboration, Nucl. Inst. Meth. A 440 (2000) 95.

[25] A. Le Blanc, HELIO ontology, in myExperiment repository, `http://www.myexperiment.org/files/703.html`, 2012.

[26] S. K. McMahon, Overview of the Planetary Data System, Planetary and Space Science 44 (1996) 3–12. Planetary data system.

[27] K. Reardon, R. Bentley, M. Messerotti, S. Giordano, A Solar Data Model for use in Virtual Observatories, in: Bulletin of the American Astronomical Society, volume 36, p. 796.

[28] R. Perrey, M. Lycett, Service-oriented architecture, in: Applications and the Internet Workshops, 2003. Proceedings. 2003 Symposium on, IEEE Computer Society, pp. 116–119.

[29] V. Domingo, B. Fleck, A. Poland, The SOHO mission: an overview, Solar Physics 162 (1995) 1–37.

[30] G. Pierantoni, B. Coghlan, E. Kenny, The Architecture of HELIO, in: Proc. Cracow Grid Workshop (CGW'10), pp. 1–8.

[31] D. Hull, K. Wolstencroft, R. Stevens, C. Goble, M. Pocock, P. Li, T. Oinn, Taverna: a tool for building and running workflows of services, Nucleic acids research 34 (2006) W729.

[32] J. Gosling, The Java language specification, Prentice Hall, 2000.

[33] B. Stern, Interactive Data Language, in: Proceedings of SPACE 2000: The Seventh International Conference and Exposition on Engineering, Construction, Operations and Business in Space, American Society of Civil Engineers, 1801 Alexander Bell Drive, Reston, VA, 20191-4400, USA,, p. 1011.

[34] I. Foster, C. Kesselman, S. Tuecke, The Anatomy of the Grid: Enabling Scalable Virtual Organizations, International Journal of High Performance Computing Applications 15 (2001) 200–222.

[35] C. Jacquey, V. Génot, E. Budnik, R. Hitier, M. Bouchemit, M. Gangloff, A. Fedorov, B. Cecconi, N. André, B. Lavraud, et al., AMDA, Automated Multi-Dataset Analysis: A web-based service provided by the CDPP, The Cluster Active Archive (2010) 239–247.

[36] C. Goble, D. De Roure, myExperiment: social networking for workflow-using e-scientists, in: Proceedings of the 2nd workshop on Workflows in support of large-scale science, ACM, pp. 1–2.

[37] W. Tan, R. Madduri, A. Nenadic, S. Soiland-Reyes, D. Sulakhe, I. Foster, C. Goble, CaGrid Workflow Toolkit: A taverna based workflow tool for cancer grid, BMC bioinformatics 11 (2010) 542.

[38] W. Tan, J. Zhang, I. Foster, Network analysis of scientific workflows: A gateway to reuse, Computer 43 (2010) 54–61.

[39] A. Le Blanc, D. Fellows, Associate hessi flares with active regions, in myExperiment repository, `http://www.myexperiment.org/workflows/2181.html`, 2011.

[40] A. Le Blanc, Filter CME events to events which could have been observed at Earth and Mars, in myExperiment repository, `http://www.myexperiment.org/workflows/2775.html`, 2012.

[41] E. Robbrecht, D. Berghmans, Automated recognition of coronal mass ejections (CMEs) in near-real-time data, Astronomy and Astrophysics 425 (2004) 1097–1106.

[42] E. Robbrecht, D. Berghmans, R. A. M. V. der Linden, Automated LASCO CME catalog for solar cycle 23:

Are CMEs scale invariant?, The Astrophysical Journal 691 (2009) 1222.

[43] R. Bentley, A. Csillaghy, I. Scholl, The European grid of solar observations, in: Proceedings of SPIE, volume 5493, Citeseer, pp. 170–177.

[44] B. Shneiderman, The Eyes Have It: a task by data type taxonomy for information visualizations, Visual Languages, IEEE Symposium on 0 (1996) 336.

[45] M. Driver, R. Valdes, G. Phifer, Rich Internet Applications are the next evolution of the Web, Gartner Research (2005).

[46] D. Flanagan, JavaScript: the definitive guide, O'Reilly, 1998.

[47] J. Garrett, Ajax: A new approach to web applications, blog posting, available online at `http://www.adaptivepath.com/ideas/ajax-new-approach-web-applications`, 2005. Downloaded June 2011.

[48] J. Allaire, Macromedia Flash MX—A next-generation rich client, Macromedia White Paper (2002) 1–2.

[49] C. Wenz, Essential Silverlight 2, O'Reilly Media, Incorporated, 2008.

[50] I. Hickson, D. Hyatt, HTML 5 Working Draft, W3C Working Draft, `http://www.w3.org/TR/html5/`, World Wide Web Consortium (W3C) (2012).

[51] R. Hanisch, A. Farris, E. Greisen, W. Pence, B. Schlesinger, P. Teuben, R. Thompson, A. Warnock III, Definition of the Flexible Image Transport System (FITS), Astronomy and Astrophysics 376 (2001) 359–380.