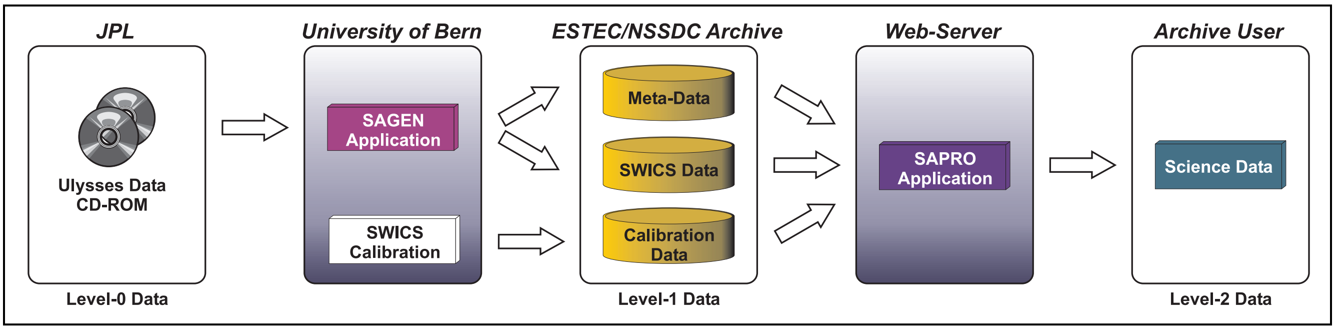
ESA/PRODEX Project - Ulysses/SWICS Data Archive

Project Overview

Started in Spring 1996, the ESA/PRODEX project for the archiving of the Ulysses/SWICS instrument data, involves the development of all software components needed for the setup and operation of the archive. Because the Ulysses mission is a joint ESA/NASA venture, the data archive will be located both at the European Space Research and Technology Center (ESTEC) in Noordwijk (NL) and at the National Space Science Data Center (NSSDC) near Washington (USA).
The following diagram gives an overview of the different steps required to establish the SWICS data archive and shows the different institutions involved:

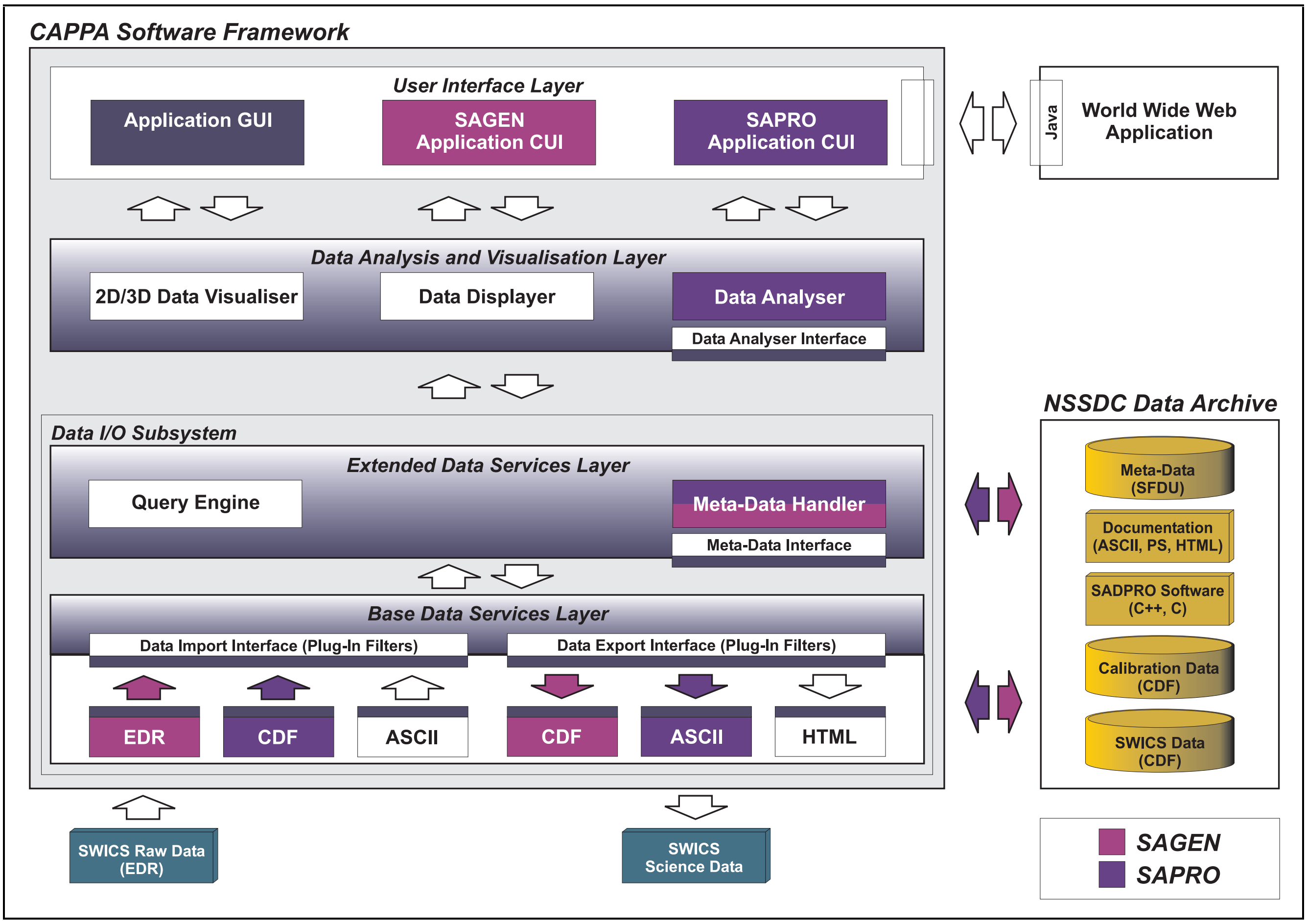


The Level-0 or raw data are delivered once per month on CD-ROM directly from the NASA Jet Propulsion Laboratory (JPL) to the Physics Institute of the University of Bern. These data, approximately 60 MB per month, contain not only SWICS mass spectrometer data, but also Ulysses telemetry and housekeeping data, which are of no relevance for the archive. The first step in preparing the archive data therefore is to extract the relevant science data through the SWICS Archive Generator (SAGEN) application and to store them in Common Data Format (CDF) files. These CDF files do not only contain the instrument data but also the corresponding metadata, which are an essential part for long-term data archiving. These data files and the accompanying SWICS instrument calibration file make up the Level-1 data and are archived at ESTED and NSSDC. Per month, approximately 15 MB of Ulysses/SWICS data are archived, which for an expected Ulysses mission lifetime beyond the year 2001 will sum up to more than 2 GB of archived data.
The advantage of archiving Level-1 instrument data is their calibration independence. Thereby whenever more accurate calibration data become available, there is no need to regenerate several thousand archived files, but only a single calibration data file needs to be updated.
Finally the SWICS Archive Processor (SAPRO) application is used to compute the actual science (Level-2) data from the archived CDF and calibration files. SAPRO has a simple Command-Line Interface, which allows the integration on a Web-Server at ESTEC/NSSDC, thereby making the Ulysses/SWICS archive on-line accessible.
The Institute of Technology Transfer (INTEC) is the prime contractor for this three man-year project and is responsible for designing and developing all software components necessary for the data processing and operation of the archive. The development is carried out in close collaboration with the Physics Institute of the University of Bern. The entire project is financed through the PRODEX program from the Federal Office for Education and Sciences (BBW) and from ESA.

System Architecture

The main goal in designing the system architecture consisted in developing the foundation of a universal software framework, which can not only be used for the Ulysses/SWICS project, but for a multitude of other scientific data processing applications. This has been achieved by specifying well defined software interfaces between all application specific modules. Thereby the individual software modules may easily be extended or replaced, which allows to implement a flexible system that can be adapted to new data formats and analysing requirements without the need to change the underlying software architecture. Through the use of object oriented design methods and the implementation with C++, software extendibility and the adaptability to new requirements are further improved.

The software framework, which is the basis to build the various scientific data processing applications from, is called *CAPPA (Customisable Architecture for Processing Pipeline Applications)*. The CAPPA framework uses a layer based architecture with standardised interfaces to all its external components. Each layer consists of several distinct modules which are dedicated to specific tasks. The following diagram shows a schematic overview of the system and its constituents:



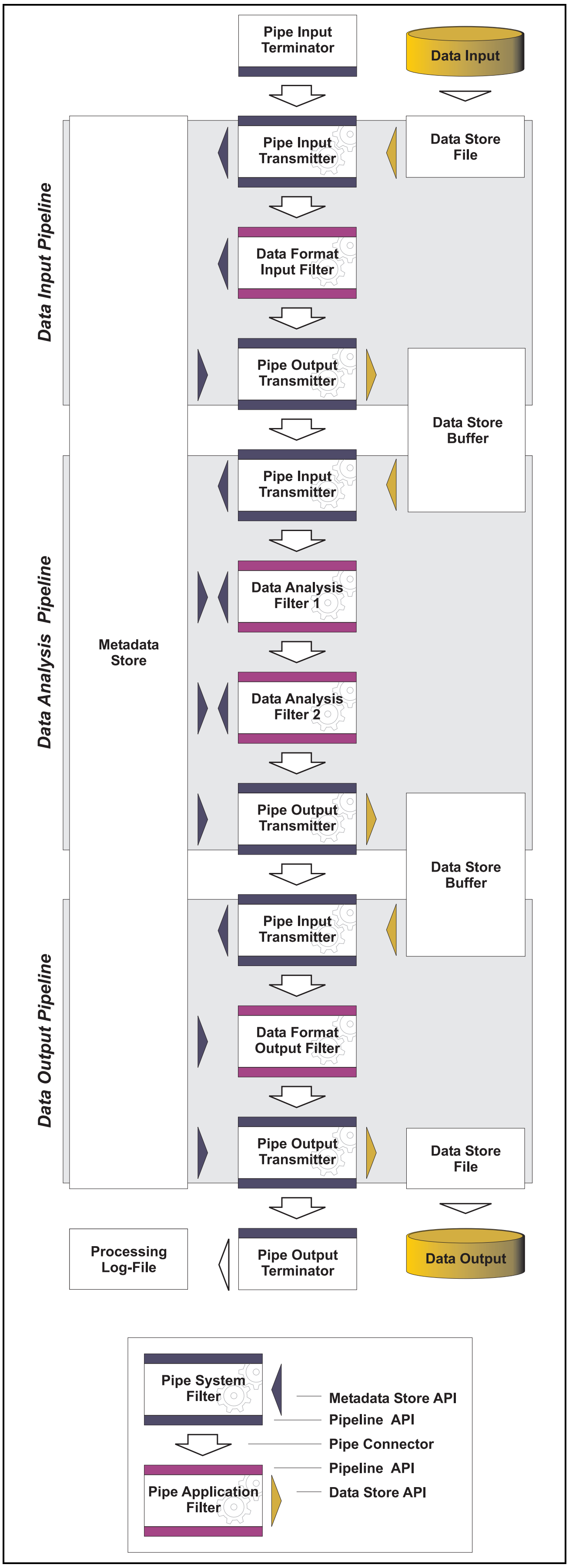
The diagram shows the CAPPA software framework composed of several modules and its external components consisting of the Web-Application and the different data files. Within the CAPPA framework the two simpler standalone Ulysses/SWICS applications, SAGEN and SAPRO, are depicted also. These applications are built from the same basic components as the CAPPA framework, but have less functionality and therefore do not need all the modules from the framework. The following paragraphs give a more detailed technical description of the framework and the Ulysses/SWICS applications.

The CAPPA Software Framework
The different modules of the CAPPA framework may functionally be arranged into four distinct layers. The most basic layer upon which all others are built, is the Base Data Services Layer. This layer allows the conversion between various data formats through different data import and export filters. These filters can be added to the system dynamically at runtime through a generic data import/export software interface. The following layer is the Extended Data Services Layer, which offers more sophisticated functions for data processing and searching. These two layers together make up the Data Input/Output Subsystem and have a close interrelation.
The Data Analysis and Visualisation Layer uses the services of the Data I/O Subsystem to analyse and pre-process the data for subsequent data display and visualisation. This layer also implements a generic software interface for the integration of plug-in filters for special purpose algorithms and analysing functions.
The User Interface Layer is responsible for providing an intuitive User Interface (UI) for the functionality contained in the underlying layers. Depending on the application usage, the UI may either be a sophisticated Graphical User Interface (GUI) or a more simple Command-Line User Interface (CUI) as it is the case for the SAGEN and SAPRO applications.

SWICS Archive Generator (SAGEN)
The SWICS Archive Generator will be the application used for the actual Ulysses/SWICS archive data production. The main capabilities needed to generate the archive data, are the extraction of the relevant records from the Experiment Data Record (EDR) file and the conversion to Common Data Format (CDF) formatted files. This functions will be part of the services being offered by the Base Data Services Layer. Additional functions are required for the generation of the corresponding metadata information, which need also to be stored in the data archive.
Because the SAGEN application uses only a simple Command-Line Interface, it can easily be ported to different platforms and can be used for the batch-processing of large data amounts.

SWICS Archive Processor (SAPRO)
The SAPRO application is the only software which will be part of the Ulysses/SWICS data archive. As the archive stores the SWICS instrument count rate data separately from the instrument calibration data, the SAPRO application is essential to produce the actual science data.
The functionality to accomplish this task requires the software to be able to read CDF files and the appropriate metadata information from the archive, as well as it needs to provide the necessary algorithms to extract and compute the physical solar-wind parameters. Because of portability reasons, the SAPRO software will reside in a single executable file and the User Interface will be command-line oriented. SAPRO may therefore be used as a standalone command-line application or may be invoked through a Web-Application, which will provide a user-friendly interface without restricting software portability.

Data Processing Pipeline Design
The application model used for system design and implementation is based on a view of consecutive data processing stages. The design pattern which best represents this model, is the pipes and filters architectural pattern. This design pattern divides the data processing task into several sequential processing steps. These steps are connected by a data flow through the system, whereas the output data of one step is the input to the following step. All processing steps are implemented by separate filter modules, which are connected through operating system pipes to implement the data flow between adjacent filters. Terminated by system filters for data input and output, these filters build the data processing pipeline. The data processing pipeline overview on the following page shows the different pipeline components and describes their interaction.
Starting with the pipeline data input, a specific system filter (Pipe Input Transmitter) is used to continuously read the input data from a file (or database) and writing it to the following Pipe Connector, which connects to the Data Format Input Filter. This filter parses the data stream,



converts the data into an internal representation and extracts the metadata information to be stored in the Metadata Store.
If the data need only to be converted into a different format (as it is the case for the SAGEN application), the Data Format Input Filter can directly be connected with the Data Format Output Filter. This filter reads the corresponding metadata information from the Metadata Store and converts the data read from the incoming pipe into the desired format. Then the filter writes the data to the outgoing pipe. This pipe is connected to another system filter, the Pipe Output Transmitter, which writes the converted data into an output file (or database). If additional data analysis is required, the processing pipeline can be extended through one or multiple Data Analysis Filters.
Because a filter can be implemented as a separate thread, each filter of the data processing pipeline can run logically in parallel. On a computer supporting symmetric multiprocessing, they run physically in parallel, which provides an excellent system scaleability, especially for large data volume or complex analysing tasks. A later framework version could even allow to use filters which run on separate machines within a computer network.
A further advantage of using the data processing pipeline application model, is the uniform software interface to integrate data format filters and data analysis filters. Through the use of plug-in filter modules, the processing pipeline can be configured and setup with specific filters in many different ways. This allows an easy adaptation to different analysis tasks by a simple reconfiguration and recombination of the appropriate filters. If the filters are linked as shared libraries, adding or replacing pipeline filters can be done dynamically at application runtime, without the need to recompile the whole application framework. This lowers application development time as well as memory usage, because only filters which are currently used by the framework are actually loaded into memory.
The CAPPA framework is distributed as Open Source Software. For further information please contact:

U. Künzler
Institute for Technology Transfer INTEC
Morgartenstr. 2c
CH-3014 Bern, Switzerland

E-Mail: urs.kuenzler@hta-be.bfh.ch
Phone: +41-(0)31-3355111