Paper ID (P-41)

Keywords: SIRTF, Uplink, JAVA, Object Oriented Design

# The Scientific Uplink System for SIRTF: Design to Implementation

Suzanne R. Dodd and Ingolf Heinrichsen, SIRTF Science Center / Infrared Processing and Analysis Center, California Institute of Technology, MS 220-6, 1200 E California Bvld, Pasadena, CA , 91125

## ABSTRACT

The Space Infrared Telescope Facility (SIRTF) is the last of NASA's Great Observatory missions, scheduled for launch in July 2002. SIRTF will perform an extended series of science observations at wavelengths ranging from 20 to 160 microns for five years or more. The California Institute of Technology is the home for the SIRTF Science Center (SSC). The SSC is responsible for supporting the scientific community with observation design and scheduling, and production and archiving of data products.

The SIRTF ground segment design is driven by the requirement to provide strong support to the entire astronomical community, while automating several functions in order to save operations costs. In the past year, the ground segment software has evolved from an early development system into an operational system, currently supporting both SIRTF Legacy Teams and Guarantee Time Observers.

This paper describes the Uplink Segment software developed at the SIRTF Science Center. The Uplink software architecture is split into three major parts:

The SIRTF Planning and Observation Tool (SPOT) is a front end Java application deployed at an astronomer's institution, providing the capabilities for the user to visualize, create, and edit observations.

The Astronomical Observation Request / Instrument Engineering Request Resource Estimator (AIRE) is the Java based application which calculates observation durations, data volumes, and other meta data used for scheduling and processing observation requests.

The Science Operations Application Server (SOAS) software is a Weblogic™ based middleware component which brokers all transactions between these client and server tools, and the Science Operations Database.

In addition to a software description, this paper addresses the remaining development work to be completed this winter, and gives a brief list of lesson's learned during the transition from system design to system implementation.

## INTRODUCTION

The Space Infrared Telescope Facility (SIRTF) consists of a 0.85-meter telescope and three cryogenically-cooled science instruments capable of performing imaging and spectroscopy in the 3 - 180 micron wavelength range. Incorporating the latest in large-format infrared detector arrays, SIRTF offers orders-of-magnitude improvements in capability over existing programs. SIRTF's mission lifetime requirement is 2.5 years, with an anticipated lifetime of 5 years. SIRTF represents an important scientific and technical bridge to NASA's new Origins program.

The SIRTF Science Center (SSC) is located at the Infrared Processing and Analysis Center (IPAC) on the campus of the California Institute of Technology in Pasadena. Its primary purpose is to ensure the scientific legacy of SIRTF. The SSC is charged with operating the SIRTF science mission, including the solicitation, review, and selection of all observing programs. It is the organization responsible for creating and securing the scientific legacy of SIRTF with a uniform and reliably reduced, calibrated, and readily accessible archive of all scientific and supporting engineering data.

The SIRTF Uplink system follows a three-tier architecture of distributed JAVA client applications, with a Weblogic™ middleware application server brokering the communications with the various server components and the SSC operational database systems. This architecture allows the deployment of sophisticated functionality on the users' home machines, while allowing load balanced operation of the SSC-based Science Operations Application Servers (SOAS). These servers perform computationally demanding tasks or operations that rely on SSC data, which could not be distributed to the users. This setup allows the users to save their work, both locally on their home machines during the planning stages and centrally via the SSC databases for accepted programs.

## SIRTF SCIENCE PLANNING AND OBSERVATION TOOL (SPOT)

SIRTF Science Planning and Observation Tool provides users with a tool to plan their observations. Today SPOT provides the following Functionality to the user: specify an Astronomical Observation Request (AOR) by filling Astronomical Observation Template (AOT); build a list of AORs; duplicate an AOR; input and output AOR ASCII files; calculate the exposure time and background estimates for an AOR; calculate the visibility windows for an AOR; edit observation and scheduling constraints for an individual AOR and a list of the AORs; build a list of targets to be accessible through each AOR so user can use the same target for a different instrument observation; modify and delete target; multiple coordinates input for target; visualization of images and observations; overlay of Star Catalog for visualization.

SPOT uses an Object Oriented design and Java for implementation. Data model, view and the controls are clearly separated. The tools are user friendly, flexible and robust. Validation is done at the field level and the content level. At each input stage, only valid data is allowed, minimizing mistakes the user can make. External files provide all text, help and field validation information. For Example, the minimum and maximum values for range checking are in an external file, making them easy to modify.

## ASTRONOMICAL OBSERVATION REQUEST / INSTRUMENT ENGINEERING REQUEST RESOURCE ESTIMATOR (AIRE)

The AIRE subsystem is an integrated part of the SIRTF Uplink System. It provides the following functionality:

A. Resource Estimation
AIRE provides accurate estimates for the total wall clock time, exposure time, downlink and uplink volume for all observing modes on SIRTF. The resource estimates given to the SPOT user are based on the actual high level expansion of his/her observation. The functionality is provided by the same Enterprise Java Bean (EJB) used to compute the high level expansions (see next item).

B. High Level Expansion
AIRE provides high level expansion into command and/or blocks for mission planning of all observing modes on SIRTF. This is implemented as a Java Enterprise Bean (EJB) deployed on an application server. The EJB will interface with the Science Operations Database (SODB) to

store expansions as either flat files or as objects. A separate scheduling software tool will obtain the expansions from the SODB. The content of the high level expansions is an ordered sequence relative in time of SIRTF commands or blocks, which form the scheduling units. In generating this sequence the expansion server will ensure, as far as possible at this stage, that no flight or mission rules are violated.

C. Background Estimation
AIRE provides infrared background estimates in the SIRTF wavelength range for a given position on the sky. This functionality is implemented via a "wrapper" Java class which uses Java Native Interface (JNI) to access pre-existing, mature background estimation subroutines written in native ("C") code.

D. Slew Time Prediction
SIRTF Flight Operations provides native ("C" language) pointing control subroutines used to estimate the time for a given spacecraft slew. This functionality is implemented via a "wrapper" Java class which uses Java Native Interface (JNI) to access the supplied subroutines. The Pointing Control System algorithms used to calculate the slew times are identical to the ones used by the onboard Flight Software. These algorithms are also used by the scheduling software and the sequence generation software.

E. Calibration and Instrument Engineering Request Support (IER Editor Tool)
The IER Editor tool is a versatile tool to generate and maintain sequences for the purpose of calibration and engineering activities (IERs) of the SIRTF instruments. It allows re-use of sequences or sequence parts. It makes the Command and Telemetry Dictionary available for use by the instrument specialists. This capability is implemented as a stand alone Java application with a Graphical User Interface. The application communicates with the SODB to import observation meta data (e.g. observer name, AOR ID number) and is capable of communicating the resulting high level expansion in the same way as the expansion server, i.e. via expansion files.

## SCIENCE OPERATIONS APPLICATION SERVER (SOAS)

The SOAS brokers communications with various server components and the SSC operational database system. It integrates the front end SPOT client with the back end services, such as the database, astronomical catalogs, and estimation and visualization software. It also provides load balancing for the Uplink Science Operations System.

The SOAS architecture is a n-tier, scaleable, transactional and service based design. It is based on object oriented analysis and design techniques. The system is implemented using the Java 2 Enterprise Edition architecture. The communication protocol between client and server is HTTP/HTTPS.

## FUTURE WORK

The remaining Uplink software development prior to SIRTF Launch is: completion of MIPS SED and IRS Spectral Mapping observing templates and visualization; support for Proposal Ingest via SPOT; calculation of on-source integration time; support for AOR viewing and modification; calculation of sensitivity estimates for an AOR.

## LESSONS LEARNED DURING THE DEVELOPMENT

The three-tiered architecture and use of Java have made the system robust, flexible and user friendly. Strong system engineering support interfacing between science users and programmers for requirements and design has been essential to completing functionality. Detailed attention was paid to the client-server interfaces which minimized system integration problems. Having a polished front end got users on board to try the software. Also, giving

demo versions of the software to the users was really the only way to get them to solidify and understand their requirements. Finally, the use of CVS and GNATs for software and change request configuration management has provided an orderly development framework.

## ACKNOWLEDGEMENT

This work was performed at the California Institute of Technology under contract to the National Aeronautics and Space Administration.

## REFERENCES

SIRTF on the WWW: http://sirtf.caltech.edu/

The Scientific Uplink and User Support System for SIRTF, HEINRICHSEN, INGOLF; CHAVEZ, JOE; HARTLEY, BOOTH; MEI, YI; POTTS, SU; ROBY, TREY; TUREK, GABRIELLA; VALJAVEC, EMMANUEL; WU, XIUQIN, Astronomical Data Analysis Software and Systems IX, ASP Conference Series, Vol. 216. Ed. N. Manset, C. Veillet, and D. Crabtree (San Francisco: ASP), 319.

Multi-tier Internet Architecture with Java, UML and OOA & D, CHAVEZ, JOE, Astronomical Data Analysis Software and Systems IX, ASP Conference Series, Vol 216. Ed. N. Manset, C. Veillet, and D. Crabtree (San Francisco: ASP), 75.

Infrared Array Camera (IRAC) for the Space Infrared Telescope Facility (SIRTF), FAZIO, GIOVANNI G.; HORA, JOSEPH L.; WILLNER, STEVE P.; STAUFFER, JOHN R.; ASHBY, M. L.; WANG, Z.; TOLLESTRUP, ERIC V.; PIPHER, JUDITH L.; FORREST, WILLIAM J.; MCCREIGHT, CRAIG R.; MOSELEY, S. H.; HOFFMANN, WILLIAM F.; EISENHARDT, PETER; WRIGHT, EDWARD L., Proc. SPIE Vol. 3354, p. 1024-1031, Infrared Astronomical Instrumentation, Albert M. Fowler; Ed.

Multiband Imaging Photometer for SIRTF, HEIM, G. B.; HENDERSON, M. L.; MACFEELY, KIM I.; MCMAHON, THOMAS J.; MICHIKA, DAVID; PEARSON, R. J.; RIEKE, GEORGE H.; SCHWENKER, JOHN P.; STRECKER, DONALD W.; THOMPSON, CRAIG L.; WARDEN, R. M.; WILSON, D. A.; YOUNG, ERICK T., Proc. SPIE Vol. 3356, p. 985-1000, Space Telescopes and Instruments V, Pierre Y. Bely; James B. Breckinridge; Eds.

Infrared Spectrograph for the Space Infrared Telescope Facility (SIRTF), ROELLIG, THOMAS L.; HOUCK, JAMES R.; VAN CLEVE, JEFFREY E.; RAKOWSKI, AL; STEWART, CHRISTOPHER K.; TAUDIEN, GLENN E.; HUISJEN, MARTY; BOLTON, MARY; SEIB, DAVID H., Proc. SPIE Vol. 3354, p. 1192-1203, Infrared Astronomical Instrumentation, Albert M. Fowler; Ed.