Astronomical Data Analysis Software and Systems VIII
ASP Conference Series, Vol. 172, 1999
D. M. Mehringer, R. L. Plante, and D. A. Roberts, eds.

Small, Fast and Reusable: A Satellite Planning and Scheduling System

Steven Kleiner

Smithsonian Astrophysical Observatory, Cambridge MA 02176, Email: skleiner@cfa.harvard.edu

Abstract. The SWAS planning and scheduling system provides all functions from ephemeris processing to guide star selection to timeline generation. It embodies a dynamic scheduling algorithm which allows schedules to be rebuilt or slipped while conserving the scheduling effort already invested. A novel graphical interface provides interactive scheduling. The system uses a component design for efficient reuse. The software is available[1] for use by other missions.

1. Introduction

The Submillimeter Wave Astronomy Satellite is 600 lb NASA Small Explorer satellite launched into a 650 km orbit on 12/5/98. It is a radio telescope which observes the O_2, CI, H_2O, ^{13}CO and $H_2{}^{18}O$ spectral lines in the 500 GHz (0.6 mm) regime from molecular clouds in our Galaxy. These observations will help us to understand the chemistry, energetics and kinematics of star-forming regions.

The spacecraft was built by the NASA Goddard Space Flight Center. The mission is under the scientific direction of the Smithsonian Astrophysical Observatory. The PI is Gary Melnick. The NASA Astrophysical Data Processing program supports the distribution of this package for other missions. At the time of this writing, SWAS has completed its launch and early orbit testing and is taking beautiful data. The planning and scheduling system has worked faultlessly.

Noteworthy features of the SWAS planning and scheduling system include:

- full range of capabilities, speed and relative simplicity;
- efficient planning algorithms;
- dynamic scheduling algorithm;
- novel graphical user interface;
- a component design for convenient reuse;

[1] `http://www.quaternion.harvard.edu`

2. The SWAS Planning and Scheduling System

The planning and scheduling system provides all the functions required to plan, schedule and command SWAS. The system starts with a predictive ephemeris describing the orbit and produces a timeline which is then converted to a binary load and uplinked to the spacecraft. Between these two points the system must compute Keplerian orbital elements, planetary positions, Sun, Moon and Earth limb pointing constraints, target rise and set times, gyro calibration and waypoint targets, line-of-sight velocities for LO commanding, guide stars for the Ball CT601 star tracker, slew paths between targets, mapping offsets and attitude quaternions. The system must then generate commanding for the science instrument and the spacecraft data handling and attitude control subsystems.

The system is very fast. We can generate a week's worth of observation timelines, from ephemeris ingestion to guide star selection and command generation, in about three minutes on a SUN Ultra-10. The code is in C, uses Motif for the widgets, and consists of 18.5K lines of code.

3. Planning and Scheduling Algorithms

We have adopted a geometrical approach to planning calculations. We convert the Cartesian state vectors in the ephemeris file to Keplerian elements which describe the geometry of the orbit. We calculate such things as target rise and set times, shadow entry times, nominal roll positions, Sun avoidance satisfaction, pointing offsets, guide star selection and gyro calibration and waypoint target generation by clever use of coordinate transformations. We have entirely avoided the need to step through an orbit. We consistently use an R_{313} rotation corresponding to the classical Euler angles to do our transformations.

There is a paradox in scheduling. The better the schedule, the more fragile it generally is. A highly efficient schedule, for example, will rarely allow one to swap targets in and out or to slip the schedule for a week without degrading it. But the fact of real life is that schedules must be rebuilt or replanned all the time, and usually on very short notice. Too often a carefully crafted schedule must be thrown out in favor of a jury-rigged replan schedule. One goal of the SWAS planning and scheduling design was to build dynamic and conservative schedules – schedules which could be modified or slipped without degrading their efficiency while still retaining the scheduling effort already invested in them.

We have been able to realize this goal by redirecting our effort from building good schedules to building good order forms. An order form is a specification of the activities to be placed on a schedule, without reference to the time at which the activities are to be scheduled. The order form is a simple abstraction of the schedule, and a schedule is a concrete instance of the order form. This is an almost trivially simple differentiation, but it has powerful practical consequences. First, given a good order form, it is very simple to build a good schedule from that order form. Secondly, order forms are quite tolerant of changes. The schedules are not tolerant of change, but given that it is easy to generate a new schedule from an order form, we simply throw away the old broken schedules and build new good schedules when we change the order form. It works! The rule enforced in the SWAS planning and scheduling system is that the planner

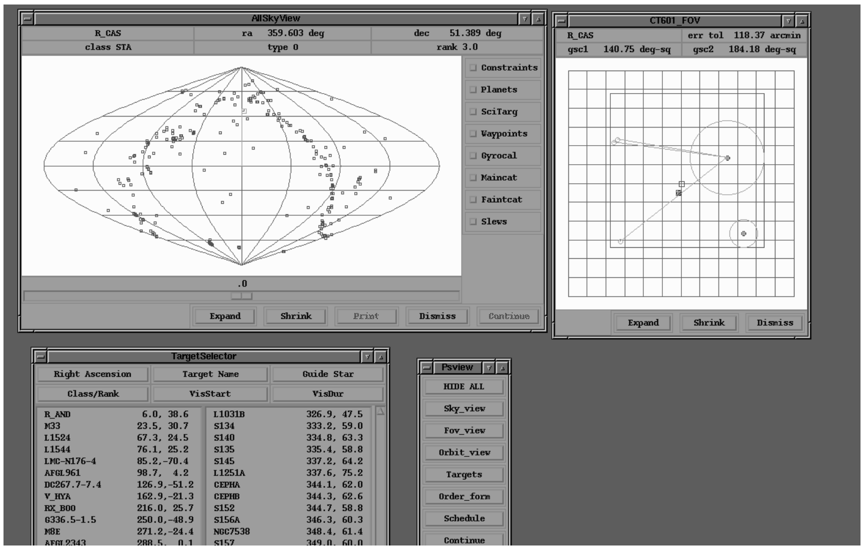

Figure 1. SWAS planning displays such as an all-sky map, the star tracker field of view with guide stars and a target browser which can select targets based on different criteria.

never interacts with the schedule directly. All changes are made to the order form and a new schedule is immediately run off from the order form.

4. Graphical User Interface

Some SWAS planning displays are shown in Fig. 1 The all-sky display shows how the SWAS target catalog is largely confined to the galactic plane. The field of view display to the right shows the sky as seen by the Ball CT601 star tracker and shows the guide stars for a given target. The target browser allows ones to examine targets based on their location in the sky, their visibility, the quality of their guide stars or other criteria.

Figure 2 shows the SWAS scheduling displays. The schedule to the left shows eight days of science observations. The waypoints which provide safe telescope pointings between science observations are not shown. The mixer to the right allows the planner to dynamically adjust the composition of the order form. The schedule display is then updated immediately to show the resulting new schedule.

Clicking on any gap in the schedule display highlights those activities on the order form which could be used to fill the gap. It is then simple to adjust the order form to fill the gap. Conversely, clicking on any scheduled activity highlights that activity on the order form, allowing one to eliminate the activity from the schedule. One can build or modify schedules in a very fluid and intuitive manner.

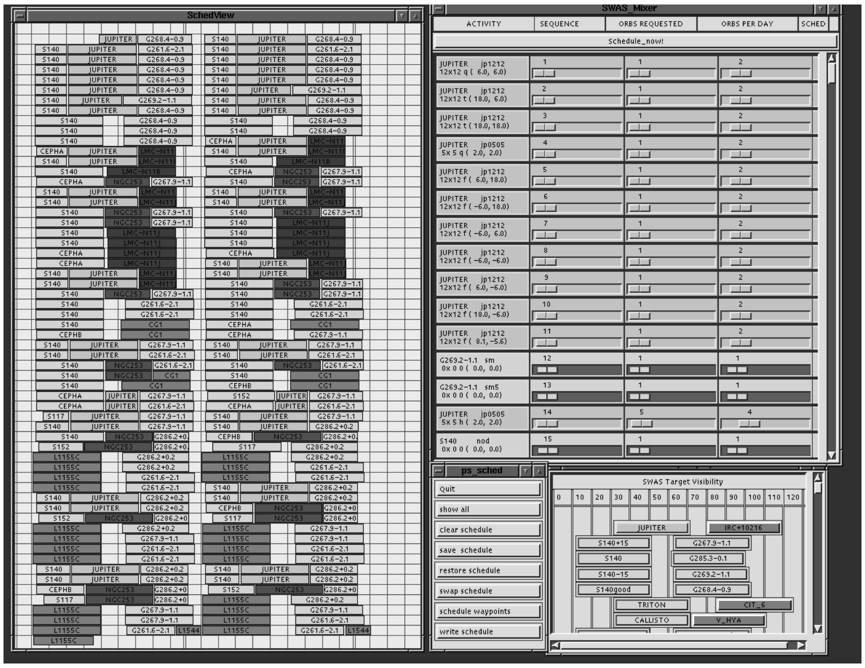

Figure 2. The SWAS scheduling displays showing an eight day schedule, the mixer which controls the composition of the schedule and a visibility display showing target rise and set times.

5. Design for Reuse

The SWAS planning and scheduling system is a set of components which can be assembled in different configurations to do different tasks. The design follows that of the CAMAC crates in nuclear physics which provide a chassis, a power supply and a bus. Different modular components can then be plugged into the crate to provide any needed functionality. We have one single, small, well-defined global PSTask data structure which acts as a bus and allows different components to communicate with each other. The data structure also carries strobe signals which synchronize the activities of the different components. We build a tools such as the SWAS scheduler or the SWAS timeline generator by selecting different components and plugging them into the core data structure. Components include a Target object which manages the SWAS target list and waypoint and gyrocal and planetary targets, an Orbit object which can either read an existing ephemeris or generate a model orbit, an ACS object which embodies the state of the spacecraft Attitude Control System and a Receiver object which manages the science instrument.

A web site, `http://www.quaternion.harvard.edu`, is under construction to describe the SWAS planning and scheduling code in greater detail and to make it available for use by other teams.