

## At A Glance

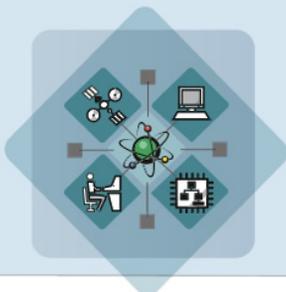
The AMPS (Automated Mission Planning & Scheduling) system is designed to be a key component of a ground system that can address the unique challenges of end-to-end constellation management.

## Features

- Multi-tier architecture
- CLIPS inference engine
- Java and XML-based GUI
- Data-driven
- Callable APIs
- CORBA

## Benefits

- Faster Dynamic Scheduling Algorithms
- Faster Expert System Algorithms
- More Sophisticated Constraint Checking
- Configurable on-the fly GUIs
- Multi-user System
- Resource Modeling
- Extensive API library
- Generic Data Storage



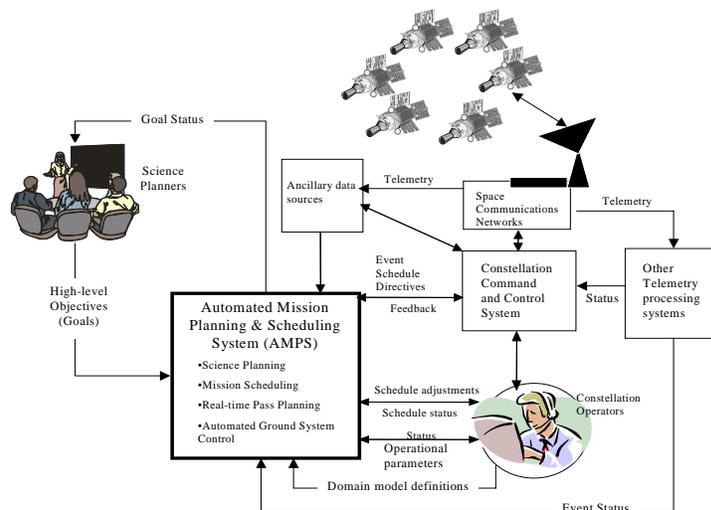
## The Next Generation Planning and Scheduling System for Constellations (AMPS)

### Product Overview

NASA's constellation environment poses a new level of complexity for spacecraft planning and scheduling. When multiple spacecraft are operated as a single constellation rather than individual elements, the complexity of such activities as dynamically allocating ground antennas to individual spacecraft increases by an order of magnitude. In addition, the degree of autonomy and orbital control of spacecraft will vary from constellation to constellation. Scheduling system performance must increase significantly, yet the cost of constellation management must remain on the order of that for a single spacecraft mission.

### The AMPS Solution

The AMPS system is designed to be a key component of a ground system that can address the unique challenges of end-to-end constellation management. AMPS acts as a ground-based "brain" that assists the user in specifying science and operational goals for the entire constellation, creates an optimized plan to achieve the high-level goals, and then schedules directives for ground system components and individual spacecraft based on current ground system and spacecraft states. AMPS integrates the latest software and scheduling technologies to provide a new level of automation and intelligence to both ground system element and spacecraft control.



NASA GSFC Mission Services Evolution Center, Code 581  
Greenbelt, Maryland 20771

<http://gmsec.gsfc.nasa.gov>

email: gmsec@nasa.gov

## AMPS Features

- AMPS is built on a **multi-tier architecture** using JAVA/JDBC, XML, C++, CORBA, CLIPS, and an RDBMS.
- Abstract planning, dynamic scheduling, and constraint checking algorithms implemented using the high-performance, cognitive-processing **CLIPS inference engine**.
- The **Java and XML-based GUI** is fully user-configurable, allow operators to set up or modify displays on the fly.
- Because the GUI is **data-driven**, displays are automatically “tailored” to reflect the environment of a particular constellation.
- A data interface of **callable APIs** allows external components and data to be integrated with the system.
- **CORBA** provides an open, hardware-independent infrastructure that supports transparent communication between AMPS elements.

## AMPS Design Incorporates new Technologies...

- **Abstract Planning** allows users to specify high-level goals instead of low-level directives, reducing the complexity and cost of science planning and mission scheduling.
- **Hierarchy of Domain Models** captures the data and relationships of deepening layers of domain objects from treating the constellation as a single entity to controlling individual spacecraft to commanding spacecraft science instruments.
- **Database Object Representation of Domain Models** streamlines setup, operations, and maintenance.
- **Model-based reasoning** separates the domain knowledge that defines object relationships from the domain data. This eliminates the rule population explosion so characteristic of systems that use expert system technology
- **Dynamic update** of domain-model data triggers the abstract planner to replan in response to FDF and other network feedback without user intervention.
- **Automated Pass Planning** determines the real-time commanding required in upcoming contacts, issues the commanding to the real-time system, and attempts error recovery commanding in the event of unsuccessful execution.

## And enhances existing ones...

- **Abstract Planning** allows users to specify high-level goals instead of low-level directives, reducing the complexity and cost of science planning and mission scheduling.
- **Hierarchy of Domain Models** captures the data and relationships of deepening layers of domain objects from treating the constellation as a single entity to controlling individual spacecraft to commanding spacecraft science instruments.
- **Database Object Representation of Domain Models** streamlines setup, operations, and maintenance.
- **Model-based reasoning** separates the domain knowledge that defines object relationships from the domain data. This eliminates the rule population explosion so characteristic of systems that use expert system technology.
- **Dynamic update** of domain-model data triggers the abstract planner to replan in response to FDF and other network feedback without user intervention.
- **Automated Pass Planning** determines the real-time commanding required in upcoming contacts, issues the commanding to the real-time system, and attempts error recovery commanding in the event of unsuccessful execution.

