Space Station Data Management System Assessment Methodology

William R. Jones
Aerospace Technologist, Spacecraft Analysis Branch, NASA Langley Research Center

David L. Bahrs
Member of the Executive Staff, Computer Sciences Corporation

SPACE STATION DATA MANAGEMENT SYSTEM ASSESSMENT METHODOLOGY

Mr. William R. Jones
Aerospace Technologist
Spacecraft Analysis Branch, M.S. 364
NASA Langley Research Center
Hampton, Virginia 23665

Mr. David L. Bahrs
Sr. Member of the Executive Staff
Computer Sciences Corporation
System Sciences Division
Silver Spring, Maryland 20910

ABSTRACT

A computer-aided modeling tool and methodology was developed and is currently being used to assess candidate designs for the Space Station Data Management System (DMS). The DMS will be a complex distributed computer system including processors, storage devices, local area networks, and software that will support all processing functions on board the Space Station. The methodology produces assessments of the performance, reliability, cost, and physical attributes of the candidate designs. This paper describes the architecture and design of the modeling tool and presents the modeling methodology.

INTRODUCTION

The conceptual design process for the Earth-orbiting Space Station requires the systematic assessment of architecture and technology options for all station subsystems before selecting the most promising candidates for development. A key element of the Space Station is the Data Management System (DMS), which will provide the command, control, data processing, and coordination functions for all subsystems within the station. The DMS architecture, hardware, and software alternatives eventually selected must be consistent with mission objectives and estimates of technology readiness and must satisfy DMS system requirements. To aid in the selections, digital computer models are used to represent candidate DMS designs and provide assessments of performance, reliability, and cost. Such models are applied during the design and development phases of the Space Station for the rapid evaluation of design and technology options and architectural configuration. In addition, such models improve station evolution by exposing future DMS technology needs and permitting cost and performance assessments of proposed DMS enhancements.

This paper describes a modeling methodology and an associated modeling tool, composed of a set of computer programs and databases developed by Computer Sciences Corporation in support of the NASA Langley Research Center to enable software models of alternative Space Station DMS design concepts to be rapidly built and evaluated.

The methodology should be of general interest to Space Station managers and engineers and of special interest to DMS designers. It is likely that new computer systems in other applications such as space platforms and ground support systems will have distributed architecture like the Space Station DMS. Designers of these systems should also find this methodology of interest. Requests for the methodology document or for the computer programs, databases, and user's guides should be directed to Mr. William R. Jones.

SPACE STATION DATA MANAGEMENT SYSTEM (DMS)

The DMS consists of the set of standard onboard processors, data storage units, local area networks, workstations, equipment interfaces, and software that collectively support the monitoring and control of all core and payload equipment and data functions on board the Space Station. Other Space Station subsystems will use the support services provided by the DMS. Figure 1 shows the reference configuration for the DMS that has been established by NASA as a departure point for further definition and preliminary design. The figure shows the broad range of functions performed by the Space Station DMS.

The Space Station DMS has two critical jobs: orchestrating the functioning of all onboard systems and interfacing with the station crew. Its architecture must be flexible, adaptable, and highly reliable because the system must resist obsolescence over a continuing life cycle. It must perform flawlessly with or without support from the crew and the ground. It must be able to recognize and report malfunctions and failures of all station-critical subsystems.

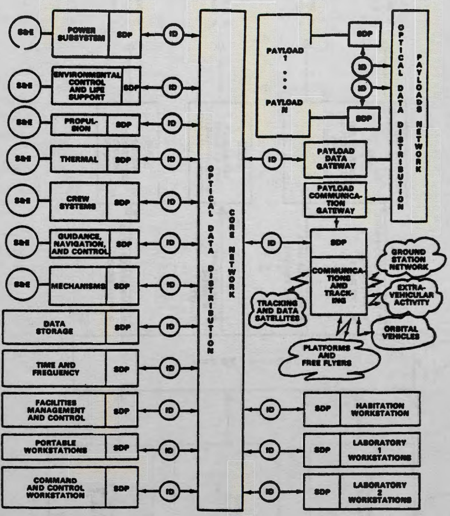
MODELING TOOL ARCHITECTURE

The overall architecture of the Space Station DMS assessment modeling tool is shown in Figure 2. The tool consists of an integrated set of databases and analysis algorithms that have been fashioned to support the construction of large, complex, distributed architecture models of DMS designs. The three databases shown at the left of the figure are populated with current DMS system requirements, various software design options, and hardware technology options for components of the DMS. These databases serve as libraries of requirements, design options, and technology options from which a modeling user can select items for inclusion in a specific candidate DMS model without having to reenter all the detailed parameters associated with each item. For example, a specific type of processor can be included in the technology options database, with its associated set of performance, reliability, cost, and weight parameters. A modeling user can include one or more instances of this processor type in a candidate DMS design model by merely referring to the processor type name when he/she defines the candidate model.

The requirement database consists of the functional and operational loading requirements levied on the DMS. Requirements are represented as end-to-end transactions. This database also provides a mechanism for function and data flow accountability. The design options database consists of sets of software designs for implementing candidate DMS architectures. The technology options database includes the information about specific hardware components that are candidates for inclusion in the DMS.

These databases can be interactively updated and extended as additional requirements, designs, and technologies mature. The databases provide input for performing system performance, reliability, and cost analyses for specific candidate DMS architectures. As an aid to the modeling user, these databases were populated with an initial version of a distributed architecture system used during the development of the tool. The databases can accept other architectures or variations as appropriate.

The user creates the candidate model, which is shown in the center of Figure 2, by selecting appropriate requirements, design options, and technology options from the databases



S&E — SENSORS AND EFFECTORS
SDP — SYSTEM DATA PROCESSOR
ID — INTERFACE DEVICE

Figure 1. Space Station DMS Reference Configuration

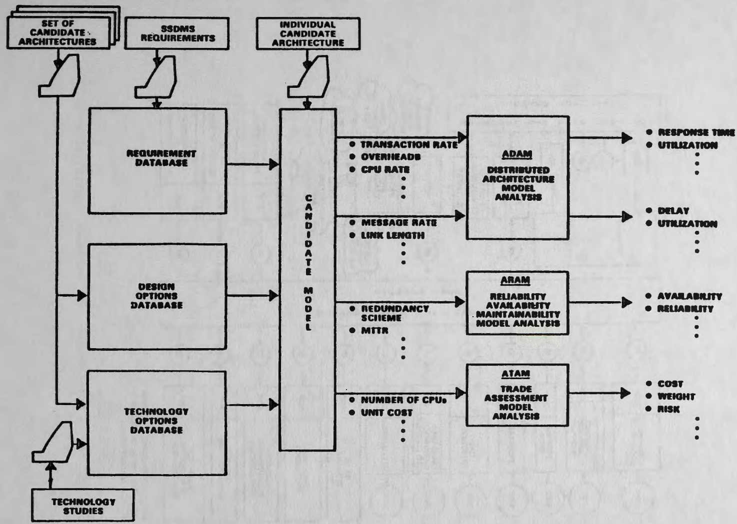


Figure 2. Space Station DMS Assessment Model

according to both the requirements and the characteristics of the candidate design being modeled. The candidate model will directly drive the analysis algorithms of the three model analysis programs provided (ADAM, ARAM, and ATAM). System performance characteristics, such as transaction response times and system component utilizations, are assessed by the Automated Distributed Architecture Model (ADAM) analysis program. The Automated Reliability, Availability, Maintainability Model (ARAM) analysis program uses the candidate architecture redundancy scheme, component mean time between failures (MTBF), component mean time to repair (MTTR), and repair person availability to predict system availability and failure rates. The Automated Trade Assessment Model (ATAM) analysis program contain the algorithms needed for design, hardware, and technology tradeoffs involving system cost, weight, volume, power, risk, and other parameters. The algorithms for these three model programs were developed to accomplish the objectives of the Space Station DMS assessment effort.

The database architecture has been designed to allow a DMS design to be modeled at layered levels of detail. This will allow the tool to be useful at initial DMS design stages when only coarse design details are available and to evolve with the design process to detailed DMS design stages when large parts of design detail are available.

MODELING TOOL DESIGN

The Space Station DMS Assessment Model has been designed to run on an IBM or IBM-compatible PC XT or AT personal computer having 640KB of main memory. It is written primarily in Microsoft FORTRAN and runs under the PC-DOS operating system. The Microrim RBASE 5000 database management system is used to manage many of the data files used by the tool and to generate many of the output reports.

The ADAM analysis program implements a set of analytic queuing algorithms that compute the utilization of hardware resources and response times for functional transactions. The model element types that it supports include processors, controllers, devices, network interface units, network routing linkages, transactions, transaction components, tasks, software modules, module paths, files, messages, and network protocols. Nearly 150 different parameter types are used to describe the characteristics of these model element types. Output reports include absolute loads and percentage utilizations for each hardware component at each priority level. Contributions to these loads by each transaction component, task, and module path are also reported. End-to-end response times are reported for each transaction, as well as the contributions to these totals by each transaction component and module path.

The ARAM analysis program uses an event simulation approach to predict hardware system availability. A simulation approach was adopted because the equations in an analytical approach rapidly become intractable as the configurations and repair disciplines increase in complexity. ARAM can generate both summary and detailed reports for the simulation. Summary reports present the computed availability for each hardware component and group, including the entire system. The number of times that each component or group cycles between available and failure states is also reported. Detailed reports trace the entire event simulation timeline indicating the time of each component failure and repair, intervals when components were queued awaiting a repair person, and intervals during which a component group or the entire system was down.

The ATAM analysis program performs simple algebraic computations on tradeoff parameter values to provide aggregate values for an entire system design. ATAM input includes a list of all hardware and software components in the candidate architecture; development, unit, and maintenance costs for each component; weight, volume, and power consumption parameters; and development risk estimates. Output reports include summations of tradeoff parameters, such as cost across all components, and weighted assessment incorporating several tradeoff parameters into an aggregate figure of merit.