AD-A053 829 FEDERAL COMPUTER PERFORMANCE EVALUATION AND SIMULATIO--ETC F/G 5/1

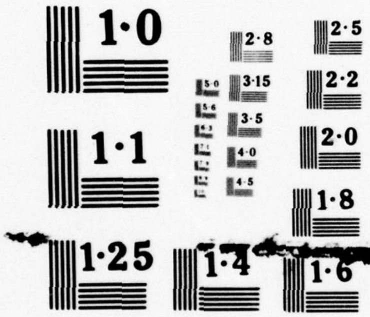
MV-027-033-ARMY EVALUATION OF DBMS MODELING APPROACHES.(U)

FEB 78

UNCLASSIFIED

NL

1 OF 2
ADA
053829


1·0
2·8
2·5
5·0
5·6
3·15
2·2
3·5
1·1
4·0
2·0
4·5
1·8
1·25
1·4
1·6

FEDSIM REPORT

12

SC

MV-027-033-ARMY

EVALUATION OF DBMS MODELING APPROACHES

FEBRUARY 1978

FEDERAL COMPUTER PERFORMANCE EVALUATION AND SIMULATION CENTER

Washington, D. C.

6

MV-Ø27-Ø33-ARMY

EVALUATION OF DBMS MODELING APPROACHES.

11 FEB[redacted]978

12 138 P.

PREPARED FOR:

U.S. ARMY COMPUTER SYSTEMS COMMAND
FT. BELVOIR, VA 22060

PROJECT PERSONNEL:

Directorate of Design
Assessment

APPROVED
David D. Igelman
DAVID D. IGELMAN
Colonel, USAF
Commander

FEDERAL COMPUTER PERFORMANCE
EVALUATION AND SIMULATION CENTER
WASHINGTON, DC 20330

hc

S/N 391711

PREFACE

This report is based on a detailed analysis of the Information Processing System Simulator (IPSS), DIMUI, and the Extendable Computer System Simulator (ECSS). The results address the specific problem of evaluating the capabilities of the above for the modeling and performance evaluation of data base management systems and their computer system environments. Questions related to the subject of this report or to the possibility of extending the stated conclusions should be addressed to its author at the Federal Computer Performance Evaluation and Simulation Center (FEDSIM).

ABSTRACT

This report presents the results of FEDSIM's evaluation of the Information Processing System Simulator (IPSS), the Extendable Computer System Simulator (ECSS), and DIMUI--A Performance Evaluation System for Data Base Management with respect to their data base management system/computer system modeling capabilities, modeling support facilities, and program product qualities.

TABLE OF CONTENTS

TABLES

I. INTRODUCTION

This report presents the results of FEDSIM's evaluation of the Information Processing System Simulator (IPSS),[1,2] the Extendable Computer System Simulator (ECSS),[3] and DIMUI--A Performance Evaluation System for Data Base Management.[4,5] The objective of the evaluation is to assess the Data Base System (DBS)[6] modeling capabilities, modeling support facilities, and program product qualities of each simulator.

[1] T. G. DeLutis, "A Methodology for the Performance Evaluation of Information Processing Systems," Final Report to the National Science Foundation, Office of Science Information Services, Grant No. GN 36622.

[2] T. G. DeLutis, "Information Processing System Simulator (IPSS): Syntax and Semantics," working document to NSF Grant No. GN 36622.

[3] D. W. Kosy, The ECSS II LANGUAGE for Simulating Computer Systems (The Rand Corporation, 1975).

[4] A. Reiter and B. Finkel, "Simulating a Virtual Data Machine," NTIS Document No. AD-A027 894, May 1976.

[5] A. Reiter, "DIMUI - IDMS User Manual, Version 1.2," Technical Report No. 101, Israel Institute of Technology, June 1977.

[6] "Data Base System" is used here to mean data base management system and its computer system environment.

II. METHODOLOGY

In order to establish a perspective from which to view and evaluate a DBS simulator, a structured set of evaluation criteria was developed. An evaluation technique was then developed to provide a means to specify the degree to which a simulator's facilities and qualities satisfy the criteria, and to facilitate a comparative evaluation of IPSS, DIMUI, and ECSS.

A. EVALUATION CRITERIA

The DBS simulator evaluation criteria are organized into three major categories: (1) modeling capabilities, (2) modeling support facilities, and (3) program product qualities.

1. Modeling Capabilities

DBS simulation imposes a need for modeling capabilities oriented specifically toward DBMS's and computer systems, in addition to those capabilities required for simulation in general. DBS simulator's modeling facilities can be evaluated by the extent that the simulator supports the modeling of the items listed in Table II-1.

2. Modeling Support Facilities

The overall utility of a DBS simulator is determined by its support of the entire simulation modeling process, as well as by the actual modeling facilities it provides. The support capabilities specified in Table II-2, which benefit all stages of the DBS simulation modeling process, are used to evaluate each DBS simulator's modeling support facilities.

3. Program Product Qualities

Aside from the functional and technical issues associated with DBS simulation modeling, a DBS simulator should also be evaluated as a program product. Table II-3 identifies the criteria used to evaluate the DBS simulator as program product.

B. EVALUATION TECHNIQUE

The foregoing categories of evaluation criteria establish the perspectives from which a DBS simulator is viewed and

° DATA BASE MANAGEMENT SYSTEM (DBMS)

- DBMS buffers
    - Description
    - Management algorithms
    - Management
    - Management/use operations
    - Buffer-data base access interaction
    - Utilization statistics collection
    - Queueing statistics collection
- Data base access control mechanisms
- Data base access scheduling algorithms
- Data base access management
- Request scheduling algorithms
- Data base management utility functions
- DBMS overhead
- Request processing statistics collection
- Request queueing statistics collection
- Data base access statistics collection
- Data base access locality statistics collection
- Data base access queueing statistics collection
- DBMS overhead statistics collection
- Data manipulation language (DML) operation statistics collection

MODELING CAPABILITIES: EVALUATION CRITERIA

TABLE II-1

° OPERATING SYSTEM

- Resource management strategies
- Resource management
- Separation from application program behavior
- Supervisory overhead
- Supervisory overhead statistics collection

° TELEPROCESSING RESOURCES

- Transmission control units
- Communications lines
- Terminal I/O devices
- Terminal control units
- TP device connectivity
- Message transmission operations
- Computation of message transmission time
- Message management strategies
- Message management
- Line control

MODELING CAPABILITIES: EVALUATION CRITERIA

TABLE II-1 (Cont'd)

° PROCESSOR RESOURCES

- Definition
- Execution rate
- Execution rate degradation due to primary memory contention
- Different execution rates for different instruction types
- Multiprocessor configurations
- Processor management strategies
- Processor management
- Priority-based interrupts
- Execution bursts
- Virtual memory reference patterns associated with execution bursts
- Multiprocessor management strategies
- Multiprocessor management
- Processes which implicitly require allocation of a processor
- Utilization statistics collection
- Queueing statistics collection
- Utilization-by-process statistics collection

MODELING CAPABILITIES: EVALUATION CRITERIA

TABLE II-1 (Cont'd)

- ° STORAGE RESOURCES: PRIMARY MEMORY
  - Definition
  - Memory management strategies
  - Memory management
  - Allocation/deallocation requests
  - Capacity utilization statistics collection
  - Capacity queueing statistics collection
- ° STORAGE RESOURCES: VIRTUAL MEMORY
  - Definition
  - Mapping to secondary storage
  - Mapping to primary memory
  - Page reference
  - Page transfer
  - Management strategies
  - Management
  - Page fault statistics collection
  - Utilization statistics collection
  - Queueing statistics collection

MODELING CAPABILITIES: EVALUATION CRITERIA

TABLE II-1 (Cont'd)

- ° STORAGE RESOURCES: SECONDARY STORAGE, DIRECT ACCESS
  - Definition
  - Capacity
  - Physical organization
  - Space overhead
  - Space management strategies
  - Space management
  - Space allocation/deallocation requests
  - Space utilization statistics collection
  - Space queueing statistics collection
- ° STORAGE RESOURCES: SECONDARY STORAGE, TAPE
  - Definition
  - Capacity
  - Space overhead

MODELING CAPABILITIES: EVALUATION CRITERIA

TABLE II-1 (Cont'd)

° I/O RESOURCES: DIRECT-ACCESS STORAGE DEVICES

- Definition
- Transfer rate
- Latency
- Cylinder access time
- Computation of cylinder access time
- Computation of data transfer time
- Read/write operations
- Seek operation
- Search operation
- Computation of search time
- Seek statistics collection
- Read/write statistics collection
- Search statistics collection
- Utilization statistics collection
- Queueing statistics collection

MODELING CAPABILITIES: EVALUATION CRITERIA

TABLE II-1 (Cont'd)

° I/O RESOURCES: TAPE DEVICES

- Definition
- Transfer rate
- Rewind rate
- Latency
- Computation of data transfer time
- Computation of rewind time
- Read/write operations
- Rewind operation
- Utilization statistics collection
- Queueing statistics collection
- Read/write statistics collection
- Rewind statistics collection

° I/O RESOURCES: UNIT RECORD DEVICES

- Definition
- Transfer rate
- Latency
- Computation of data transfer time
- Read/write operations
- Utilization statistics collection
- Queueing statistics collection

MODELING CAPABILITIES: EVALUATION CRITERIA

TABLE II-1 (Cont'd)

° I/O RESOURCES: CONTROL UNITS

- Definition
- Transfer rate
- Message capacity
- Utilization statistics collection
- Queueing statistics collection

° I/O RESOURCES: CHANNELS

- Definition
- Transfer rate
- Message capacity
- Utilization statistics collection
- Queueing statistics collection

° I/O RESOURCES: OTHER I/O DEVICES

- Definition
- Transfer rate
- Computation of data transfer time
- Read/write operations
- Utilization statistics collection
- Queueing statistics collection
- Read/write statistics collection

MODELING CAPABILITIES: EVALUATION CRITERIA

TABLE II-1 (Cont'd)

- ° I/O RESOURCES: DEVICE CONNECTIVITY
  - Definition
  - Dynamic path selection
  - Path management strategies
  - Path allocation/deallocation
  - Path management
  - Computation of effective path data transfer rate based on current states of all I/O devices on a path
  - Critical device statistics collection
  - Path utilization statistics collection
  - Path queueing statistics collection
- ° I/O RESOURCES: BUFFERS
  - Definition
  - Management strategies
  - Buffer management
  - Buffer allocation/deallocation requests
  - Buffer-I/O interaction
  - Utilization statistics collection
  - Queuing statistics collection

MODELING CAPABILITIES: EVALUATION CRITERIA

TABLE II-1 (Cont'd)

° DATA RESOURCES: FILES/FILE STRUCTURES

- Definition
- Record characteristics
- Space allocations
- Access methods
- Record-to-space allocation mappings
- Space allocation-to-physical storage media mappings
- Translation of file access into I/O activity
- File access requests
- Dynamic file creation/deletion
- Dynamic file space allocation
- File management strategies
- File management
- Access statistics collection
- Queueing statistics collection
- Utilization statistics collection

MODELING CAPABILITIES: EVALUATION CRITERIA

TABLE II-1 (Cont'd)

° DATA RESOURCES: DATA BASE

- Data structure (DS) representation
- Multiple DS's
- DS-to-DS mappings
- DS access/traversal operations
- Translation of access/traversal operations into processing time and physical I/O activity
- DS-to-secondary storage media mapping
- DS-to-virtual storage mapping
- DS-to-file (structure) mapping
- DML command characterization
- DML command invocation
- DS access paths
- Dynamic access path selection
- Access path statistics collection
- Data structure accessing statistics collection

° SOFTWARE RESOURCES

- Definition
- Management strategies
- Management
- Allocation/deallocation requests
- Utilization statistics collection
- Queueing statistics collection

MODELING CAPABILITIES: EVALUATION CRITERIA

TABLE II-1 (Cont'd)

° PROCESSES

- Definition/characterization
- Implicit processor requirement
- Time delays
- Initiation
- Termination
- Synchronization/communication/suspension
  - Wait for sub-process
  - Wait/post mechanism
  - Wait for I/O
  - Inter-process message passing
- Initialization, saving, and restoring of local process variables
- Process lists
- Process list entry discipline
- Process list operations
- Occurrence statistics collection
- Processor utilization statistics collection
- Processor queueing statistics collection
- Resource utilization statistics collection
- Resource queueing statistics collection
- I/O statistics collection

MODELING CAPABILITIES: EVALUATION CRITERIA

TABLE II-1 (Cont'd)

- WORKLOAD
  - Workload element representation
  - Workload element arrival pattern specification
  - Workload element processing statistics collection

- GENERAL-PURPOSE SIMULACRA
  - Events
    - Definition
    - Internal events
    - External events
    - Recurring events
    - Event scheduling operations
  - List data structures
    - Definition
    - Dynamic creation
    - List elements
      - Definition
      - Data aggregates
      - Dynamic creation/deletion
    - List processing operations
  - Random number generator
  - Probability distribution functions
  - Empirical distributions
  - Event and clock management
  - Recursive routines
  - Basic computational facilities

MODELING CAPABILITIES: EVALUATION CRITERIA

TABLE II-1 (Cont'd)

° STATISTICS COLLECTION CONTROL

- Suppression/resumption
- Selective suppression
- Selective omission
- Reset
- Cumulative collection
- Collection of statistics for specified time intervals

° STATISTICS REPORTING

- Automatic generation of reports in pre-defined formats
- Selective report generation
- Report generation scheduling
    - End-of-simulation
    - Periodically
    - Via external events

° SUPPLEMENTAL STATISTICS COLLECTION AND REPORTING

° MODEL BEHAVIOR TRACING

- Statement-execution level trace
- Change-of-state level trace
- Tracing control
    - Selective change-of-state tracing
    - Limited-range statement-execution tracing
    - Event-based scheduling of tracing activity

MODELING SUPPORT FACILITIES: EVALUATION CRITERIA

TABLE II-2

- ° POST SIMULATION PROCESSING
  - Selective saving of raw statistics data
  - Graphical display of statistics data
  - Generation of standard statistics reports from raw statistics data
- ° MODEL SYNTHESIS
  - Model source component library facilities
  - On-line interactive prompting/help facility for model source input
  - On-line model source editing
  - Macro processing
- ° MODEL VERIFICATION AND VALIDATION
  - Run-time diagnostic facilities
  - Model debugging facilities
    - Tracing
    - Formatted dumps
    - Behavior statistics
    - Routine traceback
    - Snapshot
  - Diagnostics at model source statement level as well as at base language level
  - On-line, interactive debugging facilities
  - Supplemental debugging facilities

MODELING SUPPORT FACILITIES: EVALUATION CRITERIA

TABLE II-2 (Cont'd)

- ° MODEL INITIALIZATION AND MODIFICATION
  - Pre-simulation input of model initialization data
    - System-defined capability for parameter value entry
    - Modeler-programmable parameter value entry and storage
  - Modification with minimal or no model source reprocessing
    - Assigning new values to model parameters
    - Hardware reconfiguration
    - Routine replacement
- ° MODEL PROCESSING
  - Model source syntax checking
  - Library facilities for the storage and retrieval of object/load modules
  - Diagnostics at each step of model processing
  - Conditional termination of model processing at any step due to error(s) in a previous step
  - Operating system-independent control language to initiate and control model processing
  - Inclusion of previously-processed model components

MODELING SUPPORT FACILITIES: EVALUATION CRITERIA

TABLE II-2 (Cont'd)

- ° MODEL EXECUTION
  - Run-time diagnostic facilities
  - Checkpoint/restart
  - Simulation termination control
    - Simulated-time based
    - Event-occurrence based
    - Error-occurrence based
    - Run-time based
  - Operating system independent language to initiate and control model execution
  - Run-time monitor providing on-line interaction with a model during execution
  - Multiple simulations during a single run
- ° MODEL OPTIMIZATION
  - Main storage space requirements
    - Program overlay
    - Deletion of unneeded data structures
    - Release of initialization and other routines not required after start of simulation
    - Omission of unneeded statistics collection
  - Executional efficiency
    - Optimizing compiler
    - Suppression/omission of collection of unneeded statistics

MODELING SUPPORT FACILITIES: EVALUATION CRITERIA

TABLE II-2 (Cont'd)

- ACQUISITION COSTS
  - Simulator
  - Other software products
- MAINTENANCE
  - Costs
  - Probability of continuing, long-term support
- RESOURCE REQUIREMENTS
  - Specific computer systems and/or operating systems
  - Main memory space
  - Language compiler(s), assemblers
  - Other software products
- PORTABILITY
- STABILITY/RELIABILITY

PROGRAM PRODUCT QUALITIES: EVALUATION CRITERIA

TABLE II-3

° DOCUMENTATION

  - User's manual
  - Language reference manual
  - Program logic manual
  - Texts, tutorials
  - Descriptions of:
    - Default algorithms
    - System-defined data structures
    - Random number generator algorithms
    - Probability distribution function algorithms
    - Timing routine
    - Event management
    - Process management
    - System-defined subprogram communication conventions

° TRAINING

  - Courses
  - Consultation

PROGRAM PRODUCT QUALITIES: EVALUATION CRITERIA

TABLE II-3 (Cont'd)

evaluated. Depending on the category, an evaluation technique ranging from qualitative assessment to quasi-quantitative measurement is used, as described below.

1. Modeling Facilities

Depending on the richness of a language's modeling constructs and features, a desired capability may be achievable with or without there being an explicit feature for it. Thus, it is not sufficient to simply state that a simulator "has" a particular capability or does not have it. To objectively evaluate a simulator's true modeling capabilities, a more quantitative measurement scale is needed. Such a measurement scale will also facilitate the comparison of languages with both similar and diverse features for satisfying the same set of requirements.

To achieve a desired modeling capability, a certain amount of modeling effort is required on the part of a modeler; and the amount of effort will vary depending on the features of the simulator being used. The degree to which a simulator supports a desired modeling capability can be "measured" in terms of the level of modeling effort required to achieve the capability using the simulator's modeling facilities. Thus, the quasi-quantitative measurement scale to be used for evaluating a DBS simulator's modeling capabilities will have level of modeling effort as a metric.

Specifically, the evaluation technique will, for each modeling capability, specify the following:

a. Whether or not the capability is explicitly provided (a measure of the richness of a simulator's modeling constructs).

b. The level of modeling effort required, encoded as one of the following:

- NONE: Capability is provided automatically/ implicitly
- DCL: Capability is achieved via a declarative
- STMT: Capability is achieved via one or very few in-line (high-level) statement(s)

- ROUTINE: Capability is achieved via a single (process) routine or a relatively small number of in-line statements

- MODULE: Capability is achieved via two or more routines and/or many in-line low-level statements

- MODULE+: Capability is achieved via two or more routines and/or many in-line low-level statements, plus new data structure(s)

- MODULE++: Capability is achieved via two or more routines and/or many in-line low-level statements, plus new data structure(s) and substantial modifications to system-defined data structures and/or system routines

c. Qualifying remarks regarding how the capability is achieved, limitations, etc.

2. Modeling Support Facilities

The modeling support facilities of a simulator are appraised using a qualitative assessment technique. For each support requirement, the following is specified:

a. An indication (YES or NO) of the availability of the modeling support facility.

b. If the support facility is provided, an appraisal of its comprehensiveness and/or limitations, if appropriate.

3. Program Product Qualities

A qualitative assessment is made of the qualities of each simulator as a program product per the specified evaluation criteria.

III. ASSUMPTIONS AND CONSTRAINTS

The results of the evaluation are based on (1) a review of all available documentation on IPSS, DIMUI, and ECSS; (2) FEDSIM's experiences with ECSS and IPSS; and (3) limited discussions with the developers of IPSS and DIMUI.

Additional insight into the capabilities and usability of each simulator for DBS modeling could be obtained by benchmark studies, involving the application of each simulator to a set of representative DBS modeling problems. Such studies are, however, beyond the scope of this project.

The developmental nature of IPSS and DIMUI is recognized and every effort has been made to represent as accurately as possible the capabilities of each. Capabilities which are inadequately documented or undocumented may not be accurately represented.

IV. FINDINGS AND RESULTS

A. IPSS

The results of the evaluation of IPSS with respect to its modeling capabilities, modeling support facilities, and program product qualities are presented below. Detailed evaluation results are presented in tabular format in Appendix A.

1. Modeling Capabilities

IPSS modeling constructs are embodied in a modeling language comprised of a large number of declarative and executable statements. Due to the large number of statement types and the often large number of parameters per statement, the IPSS language is not an easy language to learn. Learning the language is, however, facilitated by the natural way in which many of the statements correspond to components and operations within the DBS being modeled.

DBMS Modeling. IPSS provides facilities for modeling DBMS-related application programs, DBMS software, and DBMS buffer management. Built-in features facilitate characterizing data manipulation operations, modeling data structure traversal, and simulating data base access. All DBMS application programs, scheduling algorithms, utility functions, data manipulation operations, and other software strategies must be procedurally characterized. There is no implicit mechanism for modeling data base access management or DBMS request management. Statistics on data base accessing, DBMS request processing, and DBMS software execution are automatically collected and reported.

Operating System Modeling. IPSS does not represent the supervisory software of a computer system as a distinct model component. Resource management functions must be procedurally characterized. Supervisory overhead is not represented explicitly and is not reported as a separate component of resource utilization. IPSS does provide certain capabilities which will facilitate the characterization of resource management strategies, including the automatic association of a queue with each defined resource, queues and queue management, and process suspension and synchronization.

Teleprocessing Subsystem Modeling. IPSS does not provide explicit facilities for modeling teleprocessing (TP) environments. Certain IPSS definitional facilities can be adapted to describe TP hardware configurations and characterize TP message transmission; but extensive supplemental coding is required for modeling line control, message management, etc.

Processor Resource Modeling. Other than the capabilities to define a CENTRAL PROCESSOR or an INPUT OUTPUT PROCESSOR facility and to represent time delays within processes, IPSS provides no special support for simulating processors, interrupts, processor management, and processor usage.

Storage Resource Modeling. Primary memory is represented explicitly and may be characterized either as simply having a capacity or as consisting of some number of partitions, each of which has a capacity. Procedural operations for effecting memory allocation/deallocation are provided. No implicit memory management capability is provided. Memory utilization statistics are collected automatically in response to memory allocation/deallocation operations.

IPSS provides no explicit facilities for the representation of virtual memory, virtual memory management, or virtual memory usage.

Facilities are provided for the description of secondary storage devices, including direct-access and tape devices, and for the representation of secondary storage for space management purposes. No implicit secondary storage space management capability is provided.

I/O Resource Modeling. IPSS provides explicit capabilities for the description of secondary storage I/O devices, control units, channels, and other peripheral I/O devices, and the representation of I/O device configurations. I/O statistics are collected automatically for all I/O devices.

IPSS allows the description of buffer pools and provides built-in buffer allocation/deallocation operations. Buffer utilization statistics are collected automatically in response to those operations. There is no implicit buffer management capability.

Data Resource Modeling. IPSS provides extensive facilities for the description of files and file structures, the specification of file placement, and the representation of file accessing. File references are automatically translated into physical storage addresses. File (structure) access methods must be procedurally characterized. File accessing statistics are collected automatically.

Extensive facilities are also provided for describing data base structure and for characterizing data base access. Those facilities allow the description of arbitrary hierarchical and network data structures, data structure implementation techniques, and data structure access paths. IPSS automatically collects and reports statistics on data base accessing and access path utilization.

Software Resource Modeling. Software strategies are characterized procedurally, and are implicitly regarded as resources. Thus, they can be allocated and released. A process can be automatically delayed pending the availability of a software resource. Queueing and utilization statistics are automatically collected for all software resources.

Process Modeling. In IPSS, a process is called a "service," and there are two types. An "exogenous service" represents an external stimulus to the model (e.g., the arrival of a transaction), whereas an "endogenous service" represents some unit of internal processing (e.g., the execution of a software resource). There are built-in operations for process initiation, suspension, synchronization, communication, and termination. Process utilization and queueing statistics are collected automatically. Resource queueing and utilization statistics may be collected on a per process basis.

In addition to its special-purpose DBS-oriented modeling capabilities, IPSS provides a built-in multiple-seed random number generation capability, built-in probability distribution functions, and list processing capabilities. Basic programming capabilities are provided by IPSS's FORTRAN IV base language.

2. Modeling Support Facilities

Statistics Collection Control. IPSS provides for the automatic collection of many statistics. Control of statistics collection is limited to the resetting of statistics between consecutive simulations.

Statistics Reporting. IPSS produces statistics reports automatically at the end of a simulation. Aside from being able to suppress the automatic generation of reports, the modeler has no control over statistics reporting.

Supplemental Statistics Collection And Reporting. Additional statistics collection and reporting can be programmed by the modeler in FORTRAN IV.

Model Behavior Tracing. Limited tracing facilities are provided by IPSS. A routine-execution trace of either modeler-written and/or IPSS routines is supported; in addition, the contents of all active process and I/O transactions can be displayed.

Post Simulation Processing. The modeler has the capability to program post simulation analysis, which can occur after a simulation and/or between consecutive simulations. Analysis routines are programmed in FORTRAN IV.

Model Synthesis. IPSS provides model source component library facilities which allow the modeler to store model segments for future reference. Coupled with these facilities is a simple macro facility for effecting textual substitution during retrieval of a stored model segment.

Model Verification And Validation. IPSS's error-checking, tracing, and statistics collection and reporting facilities as well as the diagnostic facilities provided by the FORTRAN IV run-time system, assist in model verification/validation.

Model Initialization And Modification. Model parameter values may be entered at the beginning of or between simulations. Such parameters can be referenced from both declarative and procedural model components. Additional parameter value entry and storage can be programmed using FORTRAN IV.

IPSS allows a wide range of model modifications which can be made at the beginning of a simulation or between consecutive simulations.

Model Processing. The processing of an IPSS model involves IPSS language translation, FORTRAN IV compilation, and linkage editing. Model processing is controlled using an operating system-independent control language, thereby minimizing the use of operating system job control language and allowing for optimization of model processing by batching translation, compilation, and linkage edit operations.

Model Execution. IPSS provides simulation-time diagnostic facilities. As with model processing, model execution is initiated and controlled using IPSS language directives. IPSS provides a variety of simulation termination control capabilities. Multiple simulations may be performed during a single IPSS run.

Model Optimization. IPSS provides no specific facilities for optimizing a model's storage space requirements and/or executional efficiency.

3. Program Product Qualities

IPSS is a public domain product. It currently exists only as a prototype developmental system. As a consequence, only a very limited evaluation of its program product qualities can be made. A substantial amount of development work and documentation remains to be done. IPSS's stability and reliability can only be ascertained by subjecting it to extensive use. Its portability is restricted to IBM System 360/370 due to the need for a PL/I compiler and internal assembler routines.

B. ECSS

The results of the evaluation of ECSS with respect to its modeling capabilities, modeling support facilities, and program product qualities are summarized below. Detailed evaluation results are presented in Appendix B.

1. Modeling Capabilities

ECSS modeling constructs are embodied in a modeling language comprised of a relatively few number of declarative and executable statement types. Since ECSS is a superset of SIMSCRIPT II.5, the use of ECSS necessitates a knowledge of SIMSCRIPT II.5. The ECSS language itself is relatively easy to learn due to its English-like form and the natural way in which its modeling constructs correspond to the components and operations of the DBS being modeled. In addition, the set of ECSS statement types is easily extended through the use of ECSS macro facilities.

DBMS Modeling. ECSS provides no special support for simulating DBMS components and processing activities. ECSS modeling facilities, although partially satisfying DBMS-related modeling requirements, must be augmented with a substantial amount of lower-level coding and new data structures to achieve desired DBMS modeling capabilities.

Operating System Modeling. ECSS provides for the representation of a computer system's supervisory software as a distinct model component, definitionally independent of application software characterization. The concept of a "manager" mediating between requests for resources and the allocation of those resources is central to ECSS. Whenever a request occurs, or a resource is released, a manager routine is called to try to satisfy the request, or to reallocate the resource to a queued request, as necessary. ECSS supplies default managers for all definable resources. These managers embody typical resource allocation algorithms found in computer systems. Having such managers allows the characterization of supervisory software with minimal effort. The modeler is, however, not locked in to the default managers. To support the construction of new managers, the default managers are available for user modification and their operation is fully explained. Supervisor overhead can be represented explicitly and independently of other execution activity. ECSS automatically collects and reports statistics on supervisor overhead.

Teleprocessing (TP) Modeling. ECSS provides limited support for simulating TP resources and activities. Such support primarily pertains to the description of TP devices and device connectivity, and the representation of message transmission. A substantial amount of supplemental coding is required to achieve desired modeling capabilities related to line control, message management, and TP-related statistics collection.

Processor Resource Modeling. ECSS provides extensive facilities for simulating processors. Specific facilities are available for representing processors and multiprocessors, the memories in which they execute, the jobs that use them, and their management strategies. Processor management is an implicit aspect of simulated job execution. Processor utilization and queueing statistics are automatically collected and reported by ECSS.

Storage Resource Modeling. ECSS provides facilities for simulating arbitrary storage devices and storage management. A storage device may be represented as simply having a capacity which is incremented or decremented in response to allocations and deallocations. Alternatively, it may be represented as an address space which is allocated in distinct, contiguous pieces. In either case, default storage management algorithms are provided. Such storage device-related simulacra may be applied to the modeling of main memory, virtual memory, and secondary storage. ECSS automatically collects and reports storage utilization and queueing statistics.

I/O Resource Modeling. ECSS allows the definition of arbitrary I/O devices in terms of their simultaneous message capacity, nominal transfer rate, and maximum transfer rate for all concurrent messages. It also allows the arbitrary connection of any number of these devices to form an I/O data path. I/O management algorithms are provided, and can be easily modified or replaced. I/O management, including dynamic path selection, is implicitly performed for each simulated I/O operation. The representation of buffers, buffer usage, and buffer management, while not explicitly supported, may be readily accomplished using the concept of a storage device together with its associated storage management facilities and allocation/deallocation requests. ECSS automatically collects and reports statistics on I/O device utilization, queueing, and transmission rates, and I/O data path queueing and utilization.

Data Resource Modeling. ECSS does not support file (structure) description, file allocation to storage devices, or file accessing. It also does not support the representation of data structures, data structure implementation techniques, and data structure accessing. A substantial number of SIMSCRIPT statements and new data structures are required to achieve such capabilities. The ECSS macro facilities have been used to extend the ECSS language syntax with statements that realize many of the desired file-related modeling capabilities.

Software Resource Modeling. ECSS provides no explicit representation of software processes as resources.

Process Modeling. ECSS supports two types of processes: "external processes," which usually represent the dynamics of system loading or the behavior of equipment/users external to the system under study; and "jobs," which represent executable programs which implicitly require a simulated processor in order to execute. Extensive facilities are provided for process initiation, suspension, synchronization, termination, and interprocess communication. ECSS automatically collects and reports statistics on process execution; additionally, process types may be grouped into "families" for which ECSS automatically collects and reports statistics.

In addition to its special-purpose computer system-oriented modeling capabilities, ECSS also provides extensive general-purpose simulacra and basic programming capabilities via its SIMSCRIPT II.5 base language.

2. Modeling Support Facilities

Statistics Collection Control. ECSS allows the modeler to control the automatic collection and reporting of statistics. Control capabilities include the suppression/omission of selected statistics, reset of statistics, and collection of statistics for specified simulated-time intervals.

Statistics Reporting. In addition to producing statistics reports automatically at the end of a simulation, ECSS provides the modeler with run-time control of statistics reporting.

Supplemental Statistics Collection And Reporting. SIMSCRIPT II.5 provides special facilities for automatic statistics data collection and analysis based on modeler-specified directives. In addition, SIMSCRIPT II.5 provides special report generation features.

Model Behavior Tracing. ECSS provides extensive tracing facilities ranging from high-level statement execution tracing to detailed model change-of-state tracing. The modeler is provided with run-time control over trace options and trace start/stop times.

Post-Simulation Processing. ECSS provides the capability of selectively saving raw statistics data for post-simulation analysis. Behavior traces and standard statistics reports can be generated from the saved statistics data. Additional analysis can be programmed using SIMSCRIPT II.5.

Model Synthesis. A powerful macro facility is provided for modifying or extending the basic ECSS language syntax. Macros may be recursive, may have arguments, and may contain default values for arguments not specified at macro-call time.

Model Verification And Validation. The combination of ECSS's error-checking, tracing, statistics collection and reporting, and model snapshot facilities and SIMSCRIPT II.5's error-detection and debugging-oriented features provides the modeler with powerful capabilities for model verification and validation.

Model Initialization And Modification. Model parameter value entry and storage can be programmed in SIMSCRIPT II.5. Data elements in ECSS-defined data structures are referenceable by name, permitting easy access/modification. Substitution of one subprogram for another can be accomplished easily through the use of subprogram variables.

Model Processing. Model processing is a multi-step process, involving ECSS translation, SIMSCRIPT II.5 compilation, and linkage-editing/loading. Diagnostic facilities are provided at each step in the process. Model processing is initiated and controlled via operating system job control language statements.

Model Execution. ECSS provides run-time diagnostic facilities. Generally, an error is considered fatal, causing the simulation to stop. ECSS provides the modeler with the capability to terminate a simulation at a specified simulated-time. Simulation termination based on any other criteria must be modeler-programmed.

Model Optimization. ECSS/SIMSCRIPT II.5 provide the modeler with certain capabilities for optimizing a model's main storage space requirements and/or executional efficiency.

3. Program Product Qualities

ECSS is maintained by FEDSIM. Implementations of ECSS exist for the following computer systems: IBM System 360/370 series, Honeywell 600/6000 series, CDC 6000/7000 series, and UNIVAC 1100 series. Since the base language of ECSS is SIMSCRIPT II.5, a SIMSCRIPT II.5 compiler is required. ECSS is a stable program product whose reliability has been established by extensive use over a three-year period. ECSS courses and consultation are available from FEDSIM.

C. DIMUI

The results of the evaluation of DIMUI with respect to its modeling capabilities, modeling support facilities, and program product qualities are summarized below. Detailed evaluation results are presented in Appendix C.

1. Modeling Capabilities

As it currently exists, DIMUI is difficult to use due to the lack of a modeling language. DBS model synthesis involves the coding of FORTRAN IV subprograms and the input of model parameter values in fixed, pre-defined formats.

DBMS Modeling. DIMUI provides explicit facilities for simulating DBMS buffer usage and management, data structure traversal and accessing, data base access, data base access control and scheduling, and request scheduling. DIMUI provides a framework which the modeler tailors to a specific DBMS via the insertion of resource management strategies and characterizations of DML operations and other DBMS functions.

Operating System Modeling. DIMUI provides for the representation of certain computer system supervisory software functions as distinct model components, definitionally independent of application software characterization. Specifically, job scheduling, processor management, disk I/O scheduling, channel scheduling, buffer management, and disk space management strategies can be represented. DIMUI supplies default modules for each of the above supervisory functions. Resource management functions are performed implicitly in response to resource allocation/deallocation requests.

Teleprocessing (TP) Modeling. DIMUI provides no support for modeling TP resources and activities.

Processor Resource Modeling. Limited support is provided for simulating processors. Specific facilities are provided for representing processors, processor usage, interrupts, and processor management strategies. Processor management is an implicit aspect of simulated task execution. DIMUI provides a default uni-processor management strategy. Processor utilization statistics are automatically collected and reported.

Storage Resource Modeling. DIMUI allows the representation of disk and tape secondary storage devices. Facilities are provided for representing disk space usage and management, including a default disk space manager.

Main memory is representable only as a collection of buffers organized into one or more buffer pools. Facilities are provided for the representation of buffers, buffer usage, and buffer management.

I/O Resource Modeling. DIMUI supports the representation of certain I/O devices, specifically, tape and disk units and channels. A simple channel-disk/tape unit architecture is supported. I/O device management is an implicit aspect of I/O device modeling. Disk I/O requests are managed by a disk scheduler; each channel is managed by a channel scheduler. DIMUI supplies default disk and channel scheduling strategies.

Specific facilities are provided for the representation of buffers, buffer usage, and buffer management. Buffering is an implicit aspect of simulated data base access. A variety of alternative buffer management strategies are built into DIMUI.

DIMUI automatically collects and reports statistics on tape/disk device utilization, channel utilization, and I/O requests.

Data Resource Modeling. DIMUI supports file (structure) description, file allocation to storage devices, and simulated file accessing. In addition, it supports the representation of data structures, data structure implementation techniques, and data structure traversal/accessing. The data structure modeling capability is sufficiently general to allow the representation of arbitrary hierarchical and network data structures.

Software Resource Modeling. DIMUI provides no explicit representation of software processes as resources.

Process Modeling. DIMUI supports the representation of processes as "tasks," which represent executable programs consisting of a series of processing and data structure access/traversal activities.

In addition to its special-purpose DBMS-oriented modeling capabilities, DIMUI provides a built-in multiple-seed random number generation capability, built-in probability distribution functions, and modeler-defined distribution tables. Basic programming capabilities are provided via DIMUI's FORTRAN IV base language.

2. Modeling Support Facilities

Statistics Collection Control. DIMUI provides limited automatic statistics collection but does not provide the modeler with any control over the statistics collection process.

Statistics Reporting. Statistics reports are automatically produced at the end of a simulation.

Supplemental Statistics Collection And Reporting. The capabilities of FORTRAN IV are available for programming supplemental statistics collection and reporting.

Model Behavior Tracing. DIMUI's tracing facilities are limited to the automatic logging of all events and interrupts.

Post-Simulation Processing. Any post-simulation analysis would have to be programmed by the modeler in FORTRAN IV.

Model Synthesis. No special capabilities are provided for facilitating the model synthesis process.

Model Verification And Validation. DIMUI's error-checking, tracing, and statistics collection and reporting facilities provide limited assistance for model verification/validation.

Model Initialization And Modification. DIMUI provides extensive model parameterization, thereby facilitating model initialization/modification. Parameterization applies to selection of system-supported resource management strategies, selection of system-supported secondary storage device characteristics, selection of system-supported DBMS-related software strategies, workload description, data structure characteristics, and secondary storage hardware configuration. Additional model parameter value entry and storage can be programmed in FORTRAN IV.

Model Processing. Model processing is a multi-step process, involving FORTRAN IV compilation and linkage-editing. It is initiated and controlled via operating system job control language statements. No special capabilities related to model processing are provided.

Model Execution. DIMUI provides the modeler with the capability to terminate a simulation either when all events and interrupts have been processed or by an explicit command.

Model Optimization. There are no documented model optimization capabilities.

3. Program Product Qualities

DIMUI is a public domain product. It currently exists only as a developmental system. Consequently, only a very limited evaluation of its program product qualities can be made. Much additional documentation concerning DIMUI's modeling capabilities, internal operations, etc., is needed. DIMUI is portable, since it is implemented in standard FORTRAN IV.

V. CONCLUSIONS

The following conclusions have been reached regarding the basic DBS modeling capabilities of IPSS, ECSS, and DIMUI.

A. IPSS

Extensive support is provided for modeling data base structures, data base accessing, files, file accessing, and secondary storage device I/O and space management. No special support is provided for modeling TP subsystem components and activities. IPSS does not support the representation of computer system supervisory software as a distinct model component.

IPSS is most applicable to the modeling of a DBS from the perspective of its DBMS subsystem. With its focus on data base structure, data base accessing, DBMS workload, and secondary storage hardware and associated I/O activity, IPSS readily supports the examination of the influence on DBMS behavior of a variety of factors such as data structures, data structure implementations, data base access strategies, DBMS loading, secondary storage hardware configurations, and secondary storage resource management strategies.

The data base, file, and secondary storage related modeling capabilities of IPSS represent a significant contribution to computer system simulation modeling technology. FEDSIM recommends continuing interest in and support of the use and further development of those capabilities of IPSS.

B. ECSS

Extensive support is provided for modeling computer system supervisory software. The representation of a wide variety of computer system hardware is supported. Extensive support is provided for modeling processes and process interaction. Limited support for TP subsystem modeling is provided. No special support is provided for simulating DBMS components and activities.

ECSS is most applicable to the modeling of a DBS from a general computer system perspective, where the focus is on hardware, supervisory and application software, and processor and I/O activity. ECSS has been used successfully to model a wide variety of computer systems. While it does not currently provide special support for modeling certain DBMS and TP components and activities, its internal structure, powerful base language, and macro processor facilitate the extension of its basic computer system modeling facilities to modeling DBMS and TP subsystems.

In view of their different orientations, IPSS and ECSS have great potential for use as complementary tools for DBS modeling and performance evaluation. As an example, an IPSS model of a DBS's DBMS subsystem could be used to generate workload and performance data for use as inputs to an ECSS model of the DBS's computer system environment.

C. DIMUI

DIMUI supports the representation of data base structure, data base implementation, and data management related tasks. Representation of computer system supervisory software is limited to data management related tasks. Support for the representation of computer system hardware is very limited, and TP subsystem modeling is not supported.

The usability and generality of DIMUI's approach to modeling data base structure and implementation, data base accessing, and data management related tasks is difficult to assess due to the current lack of a modeling language and adequate documentation.

APPENDIX A

IPSS EVALUATION RESULTS

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
DATA BASE MANAGEMENT SYSTEM (DBMS)			
DBMS buffers			
Description	YES	DCL	BUFFER POOL statement
Management algorithms	NO	ROUTINE	No implicit buffer management capability
Management	NO		
Management/use operations	YES	STMT	GET/FREE BUFFER, FIND NEXT BUFFER statements
Buffer-data base access interaction	YES	STMT	
Utilization statistics collection	YES	NONE	Automatic statistics collection
Queueing statistics collection	YES	STMT	START/STOP QUEUE STATISTICS statements
Data base access control mechanisms	NO	MODULE+	No implicit access management capability
Data base access scheduling algorithms	NO		
Data base access management	NO		

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
DATA BASE MANAGEMENT SYSTEM (Cont'd)			
Request scheduling algorithms	NO	MODULE+	No implicit request management capability
Request management	NO		
Data base management utility functions	NO	ROUTINE	
DBMS overhead	NO	STMT	PROCESS/SERVICE TIME, I/O statements
Request processing statistics collection	YES	NONE	Automatic statistics collection
Request queueing statistics collection	NO	MODULE+	
Data base access statistics collection	YES	NONE	Automatic statistics collection
Data base access locality statistics collection	YES	NONE	Automatic statistics collection
Data base access queueing statistics collection	NO	MODULE+	
DBMS overhead statistics collection	NO	MODULE+	
DML operation statistics	YES	NONE	Automatic statistics collection

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
OPERATING SYSTEM			
Resource management strategies	NO	MODULE+	No implicit resource management capabilities
Resource management	NO		
Separation from application program behavior	NO		
Supervisory overhead	NO	STMT	PROCESS/SERVICE TIME, I/O statements
Supervisory overhead statistics collection	NO	MODULE+	
TELEPROCESSING RESOURCES			
Transmission control units	NO	DCL, MODULE	No explicit facilities are provided for modeling TP hardware. IPSS-defined I/O devices (DEVICE, ACCESS MECHANISM, CONTROL UNIT, CHANNEL, I/O PROCESSOR) could be adapted for describing TP hardware.
Communications lines	NO		
Terminal I/O devices	NO		
Terminal control units	NO		

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
TELEPROCESSING RESOURCES (Cont'd)			
TP device connectivity	NO	DCL, MODULE	No explicit facilities for describing TP device connectivity; PATH statement could be adapted for this purpose.
Message transmission operations	NO	MODULE+	No built-in capabilities explicitly for message management, simulated message transmission, etc.
Computation of message transmission time	NO		
Message management strategies	NO		
Message management	NO		
TP statistics collection	NO	MODULE+	
PROCESSOR RESOURCES			
Definition	YES	DCL	CENTRAL PROCESSOR, IO PROCESSOR statements
Execution rate	NO	MODULE++	The achievement of these processor-oriented modeling capabilities would require substantial low-level coding, additional data structures, and modifications to system-defined data structures and system routines.
Execution rate degradation due to primary memory contention			
Different execution rates for different instruction types	NO		
Multiprocessor configurations	NO		
Processor management strategies	NO		
Processor management	NO		

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
PROCESSOR RESOURCES (Cont'd)			
Priority-based interrupt mechanism	NO		
Virtual memory reference patterns associated with execution bursts	NO		
Multiprocessor management strategies	NO		
Multiprocessor management	NO		
Processes which implicitly require allocation of a processor	NO		
Dynamic degradation of execution rate	NO		
Execution bursts	YES	STMT	PROCESS statement
Utilization statistics collection	YES	STMT	START/STOP USAGE STATISTICS statements
Queueing statistics collection	YES	STMT	START/STOP QUEUE STATISTICS statements
Utilization-by-process statistics collection	NO	DCL, STMT	Utilization-by-process statistics can be collected using the TASK statistics collection mechanism.
STORAGE RESOURCES: PRIMARY MEMORY			
Definition	YES	DCL	MAIN STORAGE statement
Memory management strategies	NO	MODULE+	No implicit memory management capability is provided
Memory management	NO		

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
STORAGE RESOURCES: PRIMARY MEMORY (Cont'd)			
Allocation/Deallocation requests	YES	STMT	GET/FREE MAIN STORAGE, GET/FREE PARTITION statements
Capacity utilization statistics collection	YES	NONE	Utilization statistics are collected automatically in response to allocation/deallocation requests
Capacity queueing statistics collection	YES	STMT	Queueing statistics can be collected using START/STOP QUEUE STATISTICS statements
STORAGE RESOURCES: VIRTUAL MEMORY			
Definition	NO	MODULE+	No explicit facilities for the representation of virtual memory, virtual memory management, or virtual memory usage
Mapping to secondary storage	NO		
Mapping to primary memory	NO		
Page reference	NO		
Page transfer	NO		
Management strategies	NO		
Management	NO		
Page fault statistics collection	NO		
Utilization statistics collection	NO		

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
STORAGE RESOURCES: VIRTUAL MEMORY (Cont'd)			
Queueing statistics collection	NO	MODULE+	
STORAGE RESOURCES: SECONDARY STORAGE, DIRECT-ACCESS			
Definition	YES	DCL	DEVICE, VOLUME statements
Capacity	YES	DCL	
Physical organization	YES	DCL	CYLINDERS, TRACKS PER CYLINDER, TRACK CAPACITY, BLOCK SIZE, and SECTORS PER TRACK parameters of DEVICE statement
Space overhead	YES	DCL	SPACE OVERHEAD parameter of DEVICE statement
Space management strategies	YES	DCL, ROUTINE	SPACE ALLOCATION parameter of AREA statement
Space management	YES	NONE	Space management strategy is characterized by a modeler-written routine which is automatically invoked as needed. Any "map" of secondary storage reflecting allocated and free space must be built and maintained by the modeler-written space management routine.
Space allocation/deallocation requests	YES	STMT	CREATE/DESTROY DATA SET EXTENT statements
Space utilization statistics collection	NO	MODULE+	Secondary storage space utilization and queueing statistics collection must be programmed by the modeler.

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
STORAGE RESOURCES: SECONDARY STORAGE, TAPE			
Definition	YES	DCL	DEVICE, VOLUME statements
Capacity	NO	DCL	Determined by values of DENSITY parameter of DEVICE statement and TAPE LENGTH parameter of VOLUME statement
Space overhead	NO	DCL	Determined by values of DENSITY and IBG parameters of DEVICE statement
I/O RESOURCES: TAPE DEVICES			
Definition	YES	DCL	DEVICE, ACCESS MECHANISM statements
Transfer rate	NO	DCL	Determined by values of DENSITY and SPEED parameters of DEVICE statement
Rewind rate	YES	DCL	REWIND RATE parameter of DEVICE statement
Latency	YES	DCL	START STOP TIME parameter of DEVICE statement
Computation of data transfer time	YES	NONE	Computed automatically for each READ/WRITE PHYSICAL RECORD operation
Computation of rewind time	YES	NONE	Computed automatically for each REWIND TAPE operation
Read/write operations	YES	STMT	READ/WRITE PHYSICAL RECORD statements

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
I/O RESOURCES: TAPE DEVICES (Cont'd)			
Rewind operation	YES	STMT	REWIND TAPE statement
Utilization statistics collection	YES	NONE	START/STOP USAGE STATISTICS statements
Queueing statistics collection	YES	NONE/STMT	Queueing statistics are collected automatically if the WAIT FACILITY STATUS statement is used; otherwise, they may be collected using the START/STOP QUEUE STATISTICS statements.
Read/write statistics collection	YES	NONE	Automatic statistics collection
Rewind statistics collection	YES	NONE	
I/O RESOURCES: DIRECT-ACCESS STORAGE DEVICES			
Definition	YES	DCL	DEVICE, ACCESS MECHANISM statements
Transfer rate	YES	DCL	TRANSFER RATE parameter of DEVICE statement
Latency	YES	STMT	LATENCY TIME parameter of READ/WRITE PHYSICAL RECORD statements
Cylinder access time	YES	DCL	CYLINDER ACCESS parameter of DEVICE statement
Computation of cylinder access time	YES	NONE	Computed automatically for each SEEK operation

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
I/O RESOURCES: DIRECT-ACCESS STORAGE DEVICES (Cont'd)			
Computation of data transfer time	YES	NONE	Computed automatically for each READ/WRITE PHYSICAL RECORD operation
Read/write operations	YES	STMT	READ/WRITE PHYSICAL RECORD statements
Seek operation	YES	STMT	SEEK DASD CYLINDER statement
Search operation	YES	STMT	SEARCH statement
Computation of search time	YES	NONE	Computed automatically for each SEARCH operation
Seek statistics collection	YES	NONE	Automatic statistics collection
Read/write statistics collection	YES	NONE	
Search statistics collection	YES	NONE	
Utilization statistics collection	YES	STMT	START/STOP USAGE STATISTICS statements
Queueing statistics collection	YES	NONE/STMT	Queueing statistics are collected automatically if the WAIT FACILITY STATUS statement is used; otherwise, they may be collected using the START/STOP QUEUE STATISTICS statements.

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
I/O RESOURCES: UNIT RECORD DEVICES			
Definition	YES	DCL	DEVICE, ACCESS MECHANISM statements
Transfer rate	YES	DCL	TRANSFER RATE parameter of DEVICE statement
Latency	NO	DCL	Determined by values of CYCLE TIME, DATA TRANSFER OFFSET, and MODE parameters of DEVICE statement
Computation of data transfer time	YES	NONE	Computed automatically for each READ/WRITE PHYSICAL RECORD operation
Read/write operations	YES	STMT	READ/WRITE PHYSICAL RECORD statements
Utilization statistics collection	YES	STMT	START/STOP USAGE STATISTICS statements
Queueing statistics collection	YES	NONE/STMT	Queueing statistics are collected automatically if the WAIT FACILITY STATUS statement is used; otherwise, they may be collected using the START/STOP QUEUE STATISTICS statements
Read/write statistics collection	YES	NONE	Automatic statistics collection
I/O RESOURCES: CONTROL UNITS			
Definition	YES	DCL	CONTROL UNIT statement

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
I/O RESOURCES: CONTROL UNITS (Cont'd)			
Transfer rate	YES	DCL	MAX TRANSFER RATE parameter of CONTROL UNIT statement. No documented use for this value.
Message capacity	NO	MODULE	Input, storage, and use of this value as a factor affecting control unit availability and utilization must be programmed by the modeler.
Utilization statistics collection	YES	STMT	START/STOP USAGE STATISTICS statements
Queueing statistics collection	YES	NONE/STMT	Queueing statistics are collected automatically if the WAIT FACILITY STATUS statement is used; otherwise, they may be collected using the START/STOP QUEUE STATISTICS statements.
I/O RESOURCES: CHANNELS			
Definition	YES	DCL	DATA CHANNEL and/or IO PROCESSOR statements
Transfer rate	YES	DCL	MAX TRANSFER RATE parameter of DATA CHANNEL/IO PROCESSOR statements. No documented use for this value.
Message capacity	NO	MODULE	Input, storage, and use of this value as a factor affecting channel availability and utilization must be programmed by the modeler.
Utilization statistics collection	YES	STMT	START/STOP USAGE STATISTICS statements

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
I/O RESOURCES: CHANNELS (Cont'd)			
Queueing statistics collection	YES	NONE/STMT	Queueing statistics are collected automatically if the WAIT FACILITY STATUS statement is used; otherwise, they may be collected using the START/STOP QUEUE STATISTICS statements.
I/O RESOURCES: OTHER I/O DEVICES			
Definition	YES	DCL	DEVICE, ACCESS MECHANISM statements
Transfer rate	YES	DCL	TRANSFER RATE parameter of DEVICE statement
Message capacity	NO	MODULE+	
Computation of data transfer time	YES	NONE	Computed automatically for each READ/WRITE PHYSICAL RECORD operation
Read/write operations	YES	STMT	READ/WRITE PHYSICAL RECORD statements
Utilization statistics collection	YES	STMT	START/STOP USAGE STATISTICS statements
Queueing statistics collection	YES	STMT	Queueing statistics are collected automatically if the WAIT FACILITY STATUS statement is used; otherwise, they may be collected using the START/STOP QUEUE STATISTICS statements.
Read/write statistics collection	YES	NONE	Automatic statistics collection

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
I/O RESOURCES: I/O DEVICE CONNECTIVITY			
Path definition	YES	DCL	PATH statement
Dynamic path selection	YES	NONE	Built-in path selection mechanism bases path selection on facility status and facility select priority only; no consideration of path request priority or transmission capacities of path devices is taken.
Path management strategies	NO	ROUTINE	Path management must be modeler-programmed, and is facilitated by SELECT/WAIT ACCESS PATH statements and the built-in path request queueing and selection mechanism.
Path allocation/deallocation requests	YES	STMT	SELECT/WAIT ACCESS PATH statements
Computation of effective path data transfer rate based on current states of all I/O devices on a path	NO	MODULE	
Critical device statistics collection	NO	MODULE+	All path-related statistics collection must be modeler-programmed.
Path queueing statistics collection	NO		

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
I/O RESOURCES: BUFFERS			
Definition	YES	DCL	BUFFER POOL statement
Buffer management strategies	NO	ROUTINE	No implicit buffer management capability.
Buffer management	NO		
Buffer allocation/deallocation requests	YES	STMT	GET/FREE BUFFER, GET NEXT BUFFER statements
Buffer - I/O interaction	YES	STMT	
Utilization statistics collection	YES	NONE	Automatic statistics collection
Queueing statistics collection	YES	STMT	START/STOP QUEUE STATISTICS statements

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
DATA RESOURCES: FILES/FILE STRUCTURES			
Definition	YES	DCL	RECORD-TYPE statement
Record characteristics	YES	DCL	FORMAT parameter of RECORD-TYPE statement
Space allocations	YES	DCL	EXTENT statement
Access methods	NO	ROUTINE	
Record-to-space allocation mappings	YES	DCL	LOCATION parameter of RECORD-TYPE statement
Space allocation-to-physical storage media mappings	YES	DCL	AREA statement and AREA, AREA START parameters of EXTENT statement
Translation of file access into I/O activity	YES	STMT	GET RECORD ADDRESS and I/O statements
File access requests	YES	STMT	READ/WRITE PHYSICAL RECORD statements
Dynamic file creation/deletion	YES	STMT	CREATE/DESTROY DATA SET statements
Dynamic file space allocation	YES	STMT	CREATE/DESTROY DATA SET EXTENT, ALLOCATE/RELEASE DATA SET EXTENT statements
File management strategies	NO	ROUTINE	
File management	NO		

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
DATA RESOURCES: FILES/FILE STRUCTURES (Cont'd)			
Access statistics collection	YES	NONE	Automatic statistics collection
Queueing statistics collection	YES	STMT	WAIT/SET FACILITY STATUS or START STOP QUEUE STATISTICS statements
Utilization statistics collection	YES	STMT	START/STOP USAGE STATISTICS statements
DATA RESOURCES: DATA BASE			
Data structure (DS) representation	YES	DCL	The SCHEMA, RECORD TYPE, EXTENT, and SET statements provide a flexible mechanism for characterizing arbitrary hierarchical and network types of data structures.
Multiple DSs	YES	DCL	
DS-to-DS mappings	YES	DCL, ROUTINE	ORIGIN parameter of RECORD TYPE statement
DS access/traversal operations	YES	STMT	ACQUIRE/STORE/WAIT RECORD, COPY/CREATE/DESTROY/FIND/MODIFY ROUTE, ALTER/COPY SET, CREATE/IDENTIFY SET OCCURRENCE, FIND MEMBER, IDENTIFY OWNER statements

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
DATA RESOURCES: DATA BASE (Cont'd)			
Translation of access/traversal operations into processing time and physical I/O activity	YES	ROUTINE	
DS-to-secondary storage media mapping	YES	DCL, ROUTINE	
DS-to-virtual storage mapping	YES	DCL, ROUTINE	ORIGIN parameter of RECORD TYPE statement,
DS-to-file (structure) mapping	YES	DCL, ROUTINE	
DML command characterization	YES	ROUTINE	DML SERVICE routine
DML command invocation	YES	STMT	REQUEST DML SERVICE, WAIT DML SERVICE COMPLETE
DS access paths	YES	STMT	A DS access path is represented by a "route" which is established by a CREATE ROUTE statement.
Dynamic access path selection	YES	STMT	CREATE/FIND ROUTE statements
Access path statistics collection	YES	NONE	Automatic statistics collection
Data structure accessing statistics collection	YES	NONE	Automatic statistics collection

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
SOFTWARE RESOURCES			
Definition	YES	RTN	ENDO SERVICE routine
Management strategies	NO	MODULE	Software resource management must be modeler-programmed; it is facilitated by the implicit association of a queue with each software resource, process synchronization facilities, etc.
Management	NO		
Allocation/Deallocation requests	YES	STMT	WAIT/SET FACILITY STATUS statements
Utilization statistics collection	YES	NONE	Automatic statistics collection
Queueing statistics collection	YES	NONE	
PROCESSES			
Definition/characterization	YES	ROUTINE	ENDO/EXO SERVICE routines
Implicit processor requirement	NO	MODULE+	
Time delays	YES	STMT	PROCESS/SERVICE TIME statements
Initiation	YES	STMT/DCL	REQUEST SERVICE statement for an endogenous service; EVENT statement for an exogenous service
Termination	YES	STMT	TERMINATE SERVICE statement

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
PROCESSES (Cont'd)			
Synchronization/Communication/Suspension			
Wait for sub-process	YES	STMT	WAIT SERVICE COMPLETE statement
Wait/post mechanism	YES	STMT	WAIT/POST SIGNAL statements. The WAIT/POST SEMAPHORE statements allow the specification of a Boolean expression which must be true in order to terminate a wait condition.
Wait for I/O	YES	STMT	WAIT I/O COMPLETE statement
Inter-process message passing	NO	MODULE+	No capability is provided for inter-process communication via the passing of messages.
Initialization, saving, and restoring of local process variables	YES	NONE/STMT	Automatic save/restore facility; modeler may exercise specific control over saving/restoring of local process variables by using the GET/PUT SAVE AREA statements.
Process lists	YES	NONE/DCL	Process queues are implicitly associated with all resources; QUEUE statement can be used to define additional queues.
Process list entry discipline	YES	DCL	ENTRY DISCIPLINE parameter of QUEUE statement or PLACE IN QUEUE statement

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
PROCESSES (Cont'd)			
Process list operations	YES	STMT	PLACE/FIND IN QUEUE, MERGE QUEUES statements
Occurrence statistics collection	YES	NONE	Automatic statistics collection
Processor utilization statistics collection	YES	DCL	
Processor queueing statistics collection	YES	DCL	
Resource utilization statistics collection	YES	DCL	These statistics may be collected using TASK statistics collection mechanism.
Resource queueing statistics collection	YES	DCL	
I/O statistics collection			
WORKLOAD			
Workload element representation	YES	DCL, ROUTINE	EVENT statement and EXO SERVICE routine
Workload element arrival pattern specification	YES	ROUTINE	Inter-arrival time generation routine
Workload element processing statistics collection	YES	NONE	Automatic statistics collection

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
GENERAL-PURPOSE SIMULACRA			
Events			
Definition	NO	ROUTINE	ENDO/EXO SERVICE routines which execute in zero simulated time can be used to represent events.
Internal events	NO	ROUTINE	ENDO SERVICE routine that executes in zero simulated time.
External events	NO	MODULE	
Recurring events	NO	DCL, ROUTINE	The EVENT statement and its associated EXO SERVICE routine can be used to represent recurring events.
Event scheduling operations	NO	STMT	REQUEST SERVICE, PLACE IN QUEUE, EVENT statements
List data structures			
Definition	YES	DCL	QUEUE statement
Dynamic creation	NO	MODULE+	
List elements			An IPSS transaction (one or more 32-word blocks of a FORTRAN one-dimensional array) can be used as a list element.
Definition	NO	NONE	
Data aggregates	NO	STMT	IPSS built-in functions provide the capability of storing/retrieving different data types into/from an IPSS transaction.

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
GENERAL-PURPOSE SIMULACRA (Cont'd)			
Dynamic creation/deletion	YES	STMT	CREATE/DESTROY TRANSACTION
List processing operations	YES	STMT	FIND/PLACE IN QUEUE, MERGE QUEUES statements
Random number generator	YES	NONE	IPSS-supplied random number generator
Probability distribution functions	YES	STMT	IPSS supplies a variety of probability distribution functions.
Empirical distributions	NO	ROUTINE	
Event and clock management	YES	NONE	IPSS-supplied timing routine
Reentrant routines	YES	ROUTINE	ENDO/EXO SERVICE routines are reentrant
Basic computational facilities	YES		FORTRAN IV language programming facilities

SUPPORT FACILITY	AVAILABLE ?	REMARKS
STATISTICS COLLECTION CONTROL		
Suppression/resumption	NO	
Selective suppression	NO	
Selective omission	NO	
Reset	YES	
Cumulative collection	YES	
Interval collection	NO	
STATISTICS REPORTING		
Automatic report generation	YES	IPSS provides automatic generation of statistics reports in predefined formats.
Selective report generation	NO	
Report generation scheduling		
End-of-simulation	YES	
Periodically	NO	
Via external events	NO	

SUPPORT FACILITY	AVAILABLE ?	REMARKS
SUPPLEMENTAL STATISTICS COLLECTION AND REPORTING		Additional statistics collection and reporting can be programmed in FORTRAN. System-defined data structures are accessible using IPSS built-in functions and FORTRAN statements. IPSS system routines (FORTRAN) can be modified. Many system-defined data structures cannot be easily augmented due to their static tabular nature.
MODEL BEHAVIOR TRACING		
Statement-execution	YES	Routine-execution trace only. Additional tracing capabilities may be available using debugging facilities supported by the FORTRAN compiler.
Change-of-state	NO	
Tracing control		
Selective change-of-state tracing	NO	
Limited-range statement-execution tracing	NO	
Event-based scheduling of tracing activity	NO	
POST SIMULATION PROCESSING		Post-simulation processing operations can be programmed in FORTRAN. No capability exists for saving raw statistics data; such a capability would have to be modeler-programmed.
Selective saving of raw statistics data	NO	
Graphical display of statistics data	NO	
Generation of standard statistics reports from raw statistics data	NO	

SUPPORT FACILITY	AVAILABLE ?	REMARKS
MODEL SYNTHESIS		
Model source component library facilities	YES	
On-line, interactive prompting/help facility for model source input	NO	
On-line model source editing	NO	
Macro processing	YES	Macro capability provides for simple textual substitution, and is applicable only to model source stored in an 1PSS library.
MODEL VERIFICATION AND VALIDATION		
Run-time diagnostic facilities	YES	Error conditions are detected by both IPSS and the FORTRAN run-time system.
Model debugging facilities		
Tracing	YES	
Formatted dumps	NO	
Behavior statistics	YES	
Routine traceback	YES	FORTRAN run-time system may provide a diagnostic traceback.
Snapshot	NO	

SUPPORT FACILITY	AVAILABLE ?	REMARKS
MODEL VERIFICATION AND VALIDATION (Cont'd)		
Diagnostics at model source statement level as well as at base language level	NO	
On-line, interactive debugging facilities	NO	
Supplemental debugging facilities	YES	Supplemental debugging can be programmed in FORTRAN.
MODEL INITIALIZATION AND MODIFICATION		
Pre-simulation input of model initialization data		
System-defined capability for parameter value entry	YES	Parameter values may be entered into a global parameter array at the beginning of or between simulations. The global parameter array can be referenced from both declarative and procedural model components.
Modeler-programmable parameter value entry and storage	YES	Additional parameter value entry and storage can be programmed using FORTRAN.
Modification with minimal or no model source re-processing		IPSS allows a wide range of model modification; however, the modeler must in many cases anticipate at model definition time the types of changes to be made and the maximum number of certain model components. The latter restriction is due to the static tabular nature of many system-defined data structures.

SUPPORT FACILITY	AVAILABLE ?	REMARKS
MODEL INITIALIZATION AND MODIFICATION (Cont'd)		
Assigning new values to model parameters	YES	
Hardware re-configuration	YES	Hardware re-configuration is limited to changing a model's secondary storage I/O device configuration by enabling or disabling selected devices. No new devices can be added.
MODEL PROCESSING		
Model source syntax checking	YES	
Library facilities for the storage/retrieval of object/load modules	YES	
Diagnostics facilities	YES	
Conditional termination of model processing at any step due to error(s) in a previous step	NO	
Operating system-independent control language to initiate and control model processing	YES	
Inclusion of previously processed components	YES	

SUPPORT FACILITY	AVAILABLE ?	REMARKS
MODEL EXECUTION		
Run-time diagnostic facilities	YES	Error conditions are detected by both IPSS and the FORTRAN run-time system.
Checkpoint/restart	NO	
Simulation termination control		
Simulated-time based	YES	
Process/Event-occurrence based	YES	
Error-occurrence based	YES	
Run-time based	YES	
Operating system independent language to initiate and control model execution	YES	
Run-time monitor providing on-line interaction with a model during execution	NO	
Multiple simulations during a single run	YES	

SUPPORT FACILITY	AVAILABLE ?	REMARKS
MODEL OPTIMIZATION		
Main storage space requirements		
Program overlay	NO	Can be accomplished using linkage editor directives
Deletion of unneeded data structures	NO	
Release of initialization and other routines not required after start of simulation	NO	
Omission of unneeded statistics collection	NO	
Executional efficiency		
Optimizing compiler	YES	Optimization features of FORTRAN compiler
Suppression/omission of collection of unneeded statistics	NO	

PROGRAM PRODUCT QUALITY	REMARKS
ACQUISITION COSTS	
Simulator	None, since IPSS is a public domain product.
Other software products	Possibly the costs of PL/I and FORTRAN compilers
MAINTENANCE	
Costs	No maintenance support since IPSS currently exists only as a prototype developmental system.
Probability of continuing, long-term support	
RESOURCE REQUIREMENTS	
Specific computer systems and/or operating systems	IBM 360/370
Main memory space	Currently 300K bytes minimum
Language compiler(s), assemblers	PL/I compiler, FORTRAN IV compiler, IBM 360/370 assembler
Other software products	None
PORTABILITY	Portability of IPSS is restricted to IBM 360/370 systems due to the need for a PL/I compiler and internal 360/370 assembler routines.
STABILITY/RELIABILITY	IPSS is still under development; its stability and reliability cannot be assessed without subjecting it to extensive use.

PROGRAM PRODUCT QUALITY		REMARKS
DOCUMENTATION		
User's manual		No
Language reference manual		Yes
Program logic manual		No
Tutorials, usage examples		No
Descriptions of:		
Default algorithms		No
System-defined data structures		Yes, except formats of facility definitions are not shown.
Random number generator algorithms		No
Probability distribution function algorithms		No
Timing routine		Yes
Event management Process management	Transaction management	Yes
System-defined communication conventions		Yes

PROGRAM PRODUCT QUALITY	REMARKS
TRAINING	
Courses	No
Consultation	Yes

APPENDIX B

ECSS EVALUATION RESULTS

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
DATA BASE MANAGEMENT SYSTEM (DBMS)			ECSS provides no special support for simulating DBMS components and processing activities. ECSS modeling facilities, although partially satisfying DBMS-related modeling requirements, must be augmented with a substantial amount of lower-level coding and new data structures to achieve desired DBMS modeling capabilities.
DBMS buffers			ECSS provides no explicit representation of buffer management. However, the concept of an ECSS storage device together with its associated storage management facilities and allocation/deallocation requests is readily applicable to the representation of buffers, buffer usage, and buffer management.
Description	NO	DCL	HAS CAPACITY clause of SPECIFY statement
Management algorithms		NONE	The default storage management algorithm may require modification to reflect a particular buffer management strategy.
Management		NONE	Implicit storage management
Usage operations		STMT	GET, GIVE, FREE statements
Buffer-data base access interaction		ROUTINE	
Utilization statistics collection		NONE	Automatic statistics collection
Queueing statistics collection		NONE	

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVID, LIMITATIONS
DATA BASE MANAGEMENT SYSTEM (Cont'd)			
Data base access control mechanisms			
Data base access scheduling algorithms		MODULE+	
Data base access management			
Request scheduling algorithms		MODULE+	
Request management			
Data base management utility functions	NO	MODULE	
DBMS overhead		ROUTINE	
Request processing statistics collection			
Data base access statistics collection			
Data base access locality statistics collection		MODULE+	
DBMS overhead statistics collection			
DML operation statistics collection			

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
OPERATING SYSTEM			
Resource management strategies	YES	NONE	The built-in default resource managers may require modification/ replacement in order to model other resource management strategies.
Resource management	YES	NONE	Implicit resource management
Separation from application program behavior	YES	NONE	All supervisory functions are provided by the resource managers, and are therefore separate from the characterizations of application programs.
Supervisory overhead	YES	DCL/STMT	Both the THE...MANAGER INTERRUPTS...statement and the AS OVERHEAD clause of the EXECUTE statement allow the representation of supervisory overhead; overhead includes processor utilization only, no I/O.
Supervisory overhead statistics collection	YES	NONE	Automatic statistics collection

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
TELEPROCESSING (TP) RESOURCES			
Transmission control units			ECSS provides limited support for simulating TP resources and activities. ECSS's concept of a transmission device is applicable to the representation of TP hardware. TP device connectivity may be represented using the data path concept. ECSS's concept of a message, message-related operations, and inter-process message passing facilities assist in the representation of message transmission and message management. A substantial amount of supplemental coding is required to achieve desired modeling capabilities related to line control, message management, and TP-related statistics collection.
Communications lines			
Terminal I/O devices	NO	DCL, STMT, MODULE+	
Terminal control units			
TP device connectivity			
Line control strategies			
Message transmission operations			

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
TELEPROCESSING RESOURCES (Cont'd)			
Computation of message transmission time	NO	DCL, STMT, MODULE+	
Message management strategies			
Message management			
TP statistics collection			
PROCESSOR RESOURCES			
Definition	YES	DCL	EXECUTES clause of SPECIFY statement
Execution rate	YES	DCL	EXECUTES clause of SPECIFY statement
Execution rate degradation due to primary memory contention	YES	DCL/STMT	DEGRADES clause of SPECIFY statement; DEGRADE, UPGRADE statements
Different execution rates for different instruction types	YES	DCL	EXECUTES clause of SPECIFY statement
Multiprocessor configurations	YES	DCL	EXECUTES and STORES clauses of SPECIFY statements

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
PROCESSOR RESOURCES (Cont'd)			
Processor management strategies	YES	NONE	Default execution management algorithms may require modification/replacement to reflect other management strategies.
Processor management	YES	NONE	Implicit processor management
Priority-based interrupt mechanism	YES	NONE	The default processor management algorithms incorporate a priority-based interrupt mechanism.
Virtual memory reference patterns associated with execution bursts	YES	STMT	REFERENCING and MODIFYING clauses of EXECUTE statement
Multiprocessor management strategies	YES	NONE	Default multiprocessor execution management algorithm may require modification to reflect other management strategies.
Processes which implicitly require allocation of a processor	YES	ROUTINE	JOB routine
Dynamic degradation of execution rate	YES	NONE	Degrading/Upgrading of execution rate is done automatically by the default execution management algorithms.
Execution bursts	YES	STMT	EXECUTE statement

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
PROCESSOR PESOURCES (Cont'd)			
Utilization statistics collection	YES	NONE	EXECUTE statement
Queueing statistics collection	YES	NONE	Automatic statistics collection
Utilization-by-process statistics collection	YES	NONE	Automatic statistics collection
STORAGE RESOURCES: PRIMARY MEMORY			
Definition	YES	DCL	HAS CAPACITY clause of SPECIFY statement
Memory management strategies	YES	NONE	Default memory management algorithms may require modification/replacement to reflect other management strategies.
Memory management	YES	STMT	GET, FREE statements
Capacity utilization statistics collection	YES	NONE	Automatic statistics collection
Capacity queueing statistics collection	YES	NONE	

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
STORAGE RESOURCES: VIRTUAL MEMORY			
Definition	NO	MODULE+	ECSS does not provide a standard way to represent virtual memory. However, it does provide a way to associate a string of page references with simulated execution bursts and a way to use these references in modeling virtual memory usage and management.
Mapping to secondary storage	NO		
Mapping to primary memory	NO		
Page reference/modification	YES	STMT	REFERENCING and MODIFYING clauses of EXECUTE statement
Page transfer	NO	ROUTINE	
Management strategies	YES	ROUTINE	Skeleton virtual memory manager routine must be expanded to reflect desired management strategy.
Management	YES	NONE	Implicit virtual memory management
Page fault statistics collection	NO	MODULE+	
Utilization statistics collection	NO		
Queueing statistics collection	NO		

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
STORAGE RESOURCES: SECONDARY STORAGE, DIRECT-ACCESS			
Definition	YES	DCL	HAS CAPACITY clause of SPECIFY statement
Capacity	YES	DCL	HAS CAPACITY clause of SPECIFY statement
Physical organization	NO	MODULE+	
Space overhead	NO		
Space management strategies	YES	NONE	Default storage management algorithms may require modification to reflect other management strategies.
Space management	YES	NONE	Implicit space management
Space allocation/deallocation requests	YES	STMT	GET, FREE statements
Space utilization statistics collection	YES	NONE	Automatic statistics collection
Space queueing statistics collection	YES	NONE	

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
STORAGE RESOURCES: SECONDARY STORAGE, TAPE			
Definition	YES	DCL	HAS CAPACITY clause of SPECIFY statement
Capacity	YES	DCL	HAS CAPACITY clause of SPECIFY statement
Space overhead	NO	MODULE+	
I/O RESOURCES: TAPE DEVICES			
Definition	YES	DCL	TRANSMITS clause of SPECIFY statement
Transfer rate	YES	DCL	TRANSMITS clause of SPECIFY statement
Latency	YES	DCL	ABSORBS...PER MESSAGE clause of SPECIFY statement
Computation of data transfer time	YES	NONE	Implicit transmission management
Rewind rate	NO	MODULE+	
Computation of rewind time	NO		
Read/write operations	YES	STMT	RECEIVE/SEND statements
Rewind operation	NO	STMT	ABSORB statement

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
I/O RESOURCES: TAPE DEVICES (Cont'd)			
Utilization statistics collection	YES	NONE	Automatic statistics collection
Queueing statistics collection	YES	NONE	
Read/write statistics collection	YES	NONE	
Rewind statistics collection	NO	MODULE+	
I/O RESOURCES: DIRECT-ACCESS STORAGE DEVICES			
Definition	YES	DCL	TRANSMITS clause of SPECIFY statement
Transfer rate	YES	DCL	TRANSMITS clause of SPECIFY statement
Latency	YES	DCL	ABSORBS...PER MESSAGE clause of SPECIFY statement
Cylinder access time	YES	DCL	ABSORBS...PER SEEK clause of SPECIFY statement
Computation of cylinder access time	YES	NONE	Implicit transmission management
Computation of data transfer time	YES	NONE	Implicit transmission management
Read/write operations	YES	STMT	RECEIVE/SEND statements

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
I/O RESOURCES: DIRECT-ACCESS STORAGE DEVICES (Cont'd)			
Seek operation	YES	STMT	SEEK statement
Search operation	NO	MODULE+	
Computation of search time	NO		
Search statistics collection	NO		
Seek statistics collection	NO	MODULE+	
Read/write statistics collection	YES	NONE	Automatic statistics collection
Utilization statistics collection	YES	NONE	
Queueing statistics collection	YES	NONE	
I/O RESOURCES: UNIT RECORD DEVICES			
Definition	YES	DCL	TRANSMITS clause of SPECIFY statement
Transfer rate	YES	DCL	TRANSMITS clause of SPECIFY statement

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
I/O RESOURCES: UNIT RECORD DEVICES (Cont'd)			
Latency	YES	DCL	ABSORBS...PER MESSAGE clause of SPECIFY statement
Computation of data transfer time	YES	NONE	Implicit transmission management
Read/write operations	YES	STMT	RECEIVE/SEND statements
Utilization statistics collection	YES	NONE	Automatic statistics collection
Queueing statistics collection	YES	NONE	
Read/write statistics collection	YES	NONE	
I/O RESOURCES: CONTROL UNITS			
Definition	YES	DCL	Transmission device
Transfer rate	YES	DCL	TRANSMITS clause of SPECIFY statement
Message capacity	YES	DCL	TRANSFERS clause of SPECIFY statement
Utilization statistics collection	YES	NONE	Automatic statistics collection
Queueing statistics collection	YES	NONE	

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
I/O RESOURCES: CHANNELS			
Definition	YES	DCL	Transmission device
Transfer rate	YES	DCL	TRANSMITS clause of SPECIFY statement
Message capacity	YES	DCL	TRANSFERS clause of SPECIFY statement
Utilization statistics collection	YES	DCL	Automatic statistics collection
Queueing statistics collection	YES	DCL	
I/O RESOURCES: OTHER I/O DEVICES			
Definition	YES	DCL	Transmission device
Transfer rate	YES	DCL	TRANSMITS clause of SPECIFY statement
Message capacity	YES	DCL	TRANSFERS clause of SPECIFY statement
Computation of data transfer time	YES	NONE	Implicit transmission management
Read/write operations	YES	STMT	RECEIVE/SEND statements

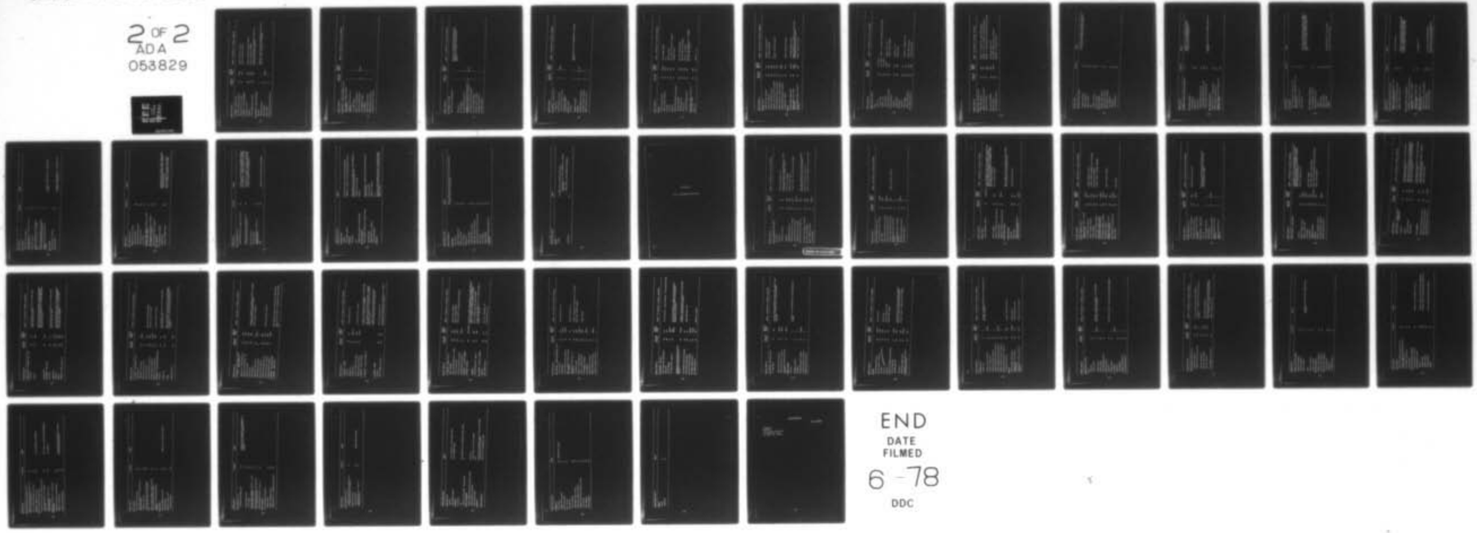
REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
Automatic statistics collection
PATH statement
Implicit data path management
A built-in default transmission management algorithm performs data path management; modification of default manager or associated routines may be required to reflect other management strategies.
Implicit data path management
Path allocation/deallocation is an implicit part of all data transfer and control operations.

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
I/O RESOURCES: I/O DEVICE CONNECTIVITY (Cont'd)			
Computation of effective path data transfer rate based on current states of all I/O devices on a path	YES	NONE	Performed as part of implicit transmission management
Critical device statistics collection	YES	NONE	Automatic statistics collection
Path utilization statistics collection	YES	NONE	
Path queueing statistics collection	YES	NONE	
I/O RESOURCES: BUFFERS			
Definition	NO	DCL	The concept of a storage device together with its associated storage management facilities and allocation/deallocation requests is readily applicable to the representation of buffers, buffer usage, and buffer management.
Buffer management strategies	NO	NONE	Default storage management algorithm
Buffer management	NO	NONE	Implicit storage management
Buffer allocation/deallocation requests	NO	STMT	GET, FREE statements
Buffer - I/O interaction	NO	ROUTINE	

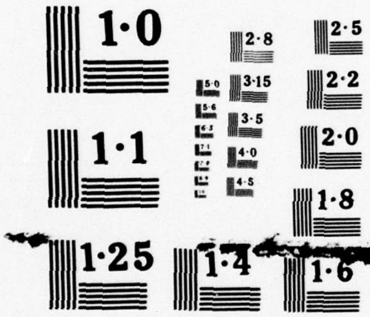
AD-A053 829　FEDERAL COMPUTER PERFORMANCE EVALUATION AND SIMULATIO--ETC　F/G 5/1
MV-027-033-ARMY EVALUATION OF DBMS MODELING APPROACHES.(U)
FEB 78

UNCLASSIFIED　NL

2 OF 2
ADA
053829



END
DATE
FILMED
6-78
DDC


1·0
2·8
2·5
5·0
3·15
2·2
5·6
3·5
6·3
2·0
1·1
4·0
4·5
1·8
1·25
1·4
1·6

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
I/O RESOURCES: BUFFERS (Cont'd)			
Utilization statistics collection	NO	NONE	Automatic statistics collection
Queueing statistics collection	NO	NONE	Automatic statistics collection
I/O RESOURCES: I/O DEVICE MANAGEMENT			
Management strategies	YES	NONE	Default transmission management algorithm
Management	YES	NONE	Implicit transmission management
Allocation/deallocation requests	YES	NONE	Allocation and deallocation are an implicit part of all data transfer and control operations.
DATA RESOURCES: FILES/FILE STRUCTURES			
Definition	NO	MODULE+	ECSS does not explicitly support the modeling of files or file structures, or the simulation of file accessing.
Record characteristics			
Space allocations			
Access methods			
Record-to-space allocation mappings			

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
DATA RESOURCES: FILES/FILE STRUCTURES (Cont'd)			
Space allocation-to-physical storage media mappings			
Translation of file access into I/O activity			
File access requests			
Dynamic file creation/deletion			
Dynamic file space allocation	NO	MODULE+	
File management strategies			
File management			
Access statistics collection			
Queueing statistics collection			
Utilization statistics collection			

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
DATA RESOURCES: DATA BASE			
Data structure (DS) representation	NO	MODULE+	ECSS does not provide modeling constructs for the representation of data structures, data structure implementation techniques, and data structure accessing.
Multiple DSs			
DS-to-DS mappings			
DS access/traversal operations			
Translation of access/traversal operations into processing time and physical I/O activity			
DS-to-secondary storage media mapping			
DS-to-virtual storage mapping			
DS-to-file (structure) mapping			
DML command characterization			
DML command invocation			

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
DATA RESOURCES: DATA BASE (Cont'd)			
DS access paths	NO	MODULE+	
Dynamic access path selection			
Access path statistics collection			
Data structure accessing statistics collection			
SOFTWARE RESOURCES			
Definition	NO	MODULE+	ECSS provides no explicit representation of a software resource.
Management strategies			
Management			
Allocation/deallocation requests			
Utilization statistics collection			
Queueing statistics collection			

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
PROCESSES			
Definition/characterization	YES	ROUTINE	JOB or EXTERNAL PROCESS
Implicit processor requirement	YES	NONE	JOB
Time delays	YES	STMT	WAIT FOR statement
Initiation	YES	STMT	START, SUBMIT, INITIATE, RUN statements
Termination	YES	STMT	TERMINATE statement
Synchronization/Communication/Suspension			
Wait for sub-process	YES	STMT	WAIT FOR...SUBJOBS statement
Wait/post mechanism	YES	STMT	WAIT FOR SIGNAL, SIGNAL statement
Wait for I/O	YES	STMT	WAIT FOR...I/O statement
Inter-process message passing	YES	STMT	PASS, WAIT FOR...MESSAGE, WAIT FOR...RESPONSES, WAIT FOR...INPUT statements
Initialization, saving, and restoring of local process variables	YES	NONE	
Process lists	YES	DCL	SIMSCRIPT II.5 set

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
PROCESSES (Cont'd)			
Process list entry discipline	YES	DCL	SIMSCRIPT II.5 set definition
Process list operations	YES	STMT	FILE, REMOVE statements
Occurrence statistics collection	YES	NONE	Automatic statistics collection
Processor utilization statistics collection	YES	NONE	
Processor queueing statistics collection	YES	NONE	
Resource utilization statistics collection	NO	MODULE+	
Resource queueing statistics collection	NO		
I/O statistics collection	YES	NONE	Automatic statistics collection
WORKLOAD			
Workload element representation	YES	ROUTINE	JOB, EXTERNAL PROCESS routines
Workload element arrival pattern specification	YES	STMT, ROUTINE	AUTOMATIC EVENT routine, SIMSCRIPT II.5 external events, JOB/EXTERNAL PROCESS scheduling statements
Workload element processing statistics collection	YES	NONE	Automatic statistics collection

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
GENERAL-PURPOSE SIMULACRA			
Events			
Definition	YES	DCL, ROUTINE	SIMSCRIPT II.5 declarative statements and event routine
Internal events	YES	DCL, ROUTINE	
External events	YES	DCL, ROUTINE	
Recurring events	YES	ROUTINE	AUTOMATIC EVENT routine
Event scheduling operations	YES	STMT	SCHEDULE, CANCEL statements
List data structures			
Definition	YES	DCL	SIMSCRIPT II.5 set
Dynamic creation	YES	STMT	CREATE statement
List elements			
Definition	YES	DCL	SIMSCRIPT II.5 temporary entities
Data aggregates	YES	DCL	
Dynamic creation/deletion	YES	STMT	CREATE, DESTROY statements
List processing operations	YES	STMT	FILE, REMOVE statements

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
GENERAL-PURPOSE SIMULACRA (Cont'd)			
Random number generator	YES	NONE	SIMSCRIPT II.5 multiple seed, random number generator
Probability distribution functions	YES	STMT	SIMSCRIPT II.5 built-in probability distribution functions
Empirical distributions	YES	DCL, STMT	SIMSCRIPT II.5 random step/linear variables; ECSS table functions
Event and clock management	YES	NONE	ECSS-supplied timing routine
Reentrant routines	YES	ROUTINE	JOB, EXTERNAL PROCESS routines are reentrant.
Basic computational facilities	YES		SIMSCRIPT II.5 programming facilities

SUPPORT FACILITY	AVAILABLE ?	REMARKS
STATISTICS COLLECTION CONTROL		ECSS provides the modeler with extensive capabilities for the control of automatic statistics collection.
Suppression/resumption	YES	
Selective suppression	YES	
Selective omission	YES	
Reset	YES	
Cumulative collection	YES	
Interval collection	YES	
STATISTICS REPORTING		
Automatic report generation	YES	
Selective report generation	NO	
Report generation scheduling		
End-of-simulation	YES	
Periodically	YES	
Via external events	YES	

SUPPORT FACILITY	AVAILABLE ?	REMARKS
SUPPLEMENTAL STATISTICS COLLECTION AND REPORTING	YES	SIMSCRIPT II.5 provides special facilities for statistics collection (ACCUMULATE and TALLY statements) and for report generation.
MODEL BEHAVIOR TRACING		
Statement-execution	YES	
Change-of-state	YES	
Tracing control		
Selective change-of-state tracing	YES	
Limited-range statement-execution tracing	YES	
Run-time control of tracing activity	YES	Both trace options and trace start/stop times can be specified.
POST SIMULATION PROCESSING		
Selective saving of raw statistics data	YES	
Graphical display of statistics data	NO	
Generation of standard statistics reports from raw statistics data	YES	

SUPPORT FACILITY	AVAILABLE?	REMARKS
MODEL SYNTHESIS		
Model source component library facilities	NO	
On-line, interactive prompting/help facility for model source input	NO	
On-line model source editing	NO	
Macros processing	YES	ECSS provides a powerful macro facility for modifying or extending the basic ECSS language syntax. Macros may be recursive, may have arguments, and may contain default values for arguments not set at macro-call time.
MODEL VERIFICATION AND VALIDATION		
Run-time diagnostic facilities	YES	
Model debugging facilities		
Tracing	YES	
Formatted dumps	NO	
Behavior statistics	YES	
Routine traceback	YES	Standard SIMSCRIPT II.5 routine traceback
Snapshot	YES	SHOW SYSTEM, SHOW STATUS statements

SUPPORT FACILITY	AVAILABLE?	REMARKS
MODEL VERIFICATION AND VALIDATION (Cont'd)		
Diagnostics at model source statement level as well as at base language level	YES	ECSS detected errors only
On-line, interactive debugging facilities	NO	
Supplemental debugging facilities	YES	SIMSCRIPT II.5 provides extensive debugging-oriented facilities, including free-format output (LIST statement), complex operation monitoring (BEFORE and AFTER statements), and monitored variables.
MODEL INITIALIZATION AND MODIFICATION		
Pre-simulation input of model initialization data		
System-defined capability for parameter value entry	NO	
Modeler-programmable parameter value entry and storage	YES	Using SIMSCRIPT II.5
Modification with minimal or no model source re-processing		
Assigning new values to model parameters	NO	
Hardware re-configuration	YES	Re-processing of SYSTEM DESCRIPTION only
(Process) routine replacement	YES	Routine replacement can be accomplished easily through the use of subprogram variables.

SUPPORT FACILITY	AVAILABLE ?	REMARKS
MODEL PROCESSING		
Model source syntax checking	NO	
Library facilities for the storage/retrieval of object/load modules	NO	
Diagnostics facilities	YES	
Conditional termination of model processing at any step due to error(s) in a previous step	NO	
Operating system-independent control language to initiate and control model processing	NO	
Inclusion of previously-processed model components	NO	This can be accomplished using a linkage editor or loader.
MODEL EXECUTION		
Run-time diagnostic facilities	YES	
Checkpoint/restart	NO	A checkpoint/restart capability is supported by the CDC version of SIMSCRIPT II.5.

SUPPORT FACILITY	AVAILABLE ?	REMARKS
MODEL EXECUTION (Cont'd)		
Simulation termination control		
Simulated-time based	YES	
Process/Event-occurrence based	NO	
Error-occurrence based	NO	
Run-time based	NO	
Operating system independent language to initiate and control model execution	NO	
Run-time monitor providing on-line interaction with a model during execution	NO	
Multiple simulaticns during a single run	YES	
MODEL OPTIMIZATION		
Main storage space requirements		
Program overlay	NO	Any desired overlay structure of ECSS routines and/or modeler-written routines must be built using the linkage editor/loader.
Deletion of unneeded routines	YES	Specific SIMSCRIPT set routines can be deleted when their associated operations are not used.

SUPPORT FACILITY	AVAILABLE ?	REMARKS
MODEL OPTIMIZATION (Cont'd)		
Suppression/deletion of unneeded data structures	YES	Allocation of data structures for unneeded statistics can be suppressed. Specific SIMSCRIPT set attributes may be deleted if not needed for desired set operations.
Release of initialization and other routines not required after start of simulation	YES	ECSS model initialization routines are released after initialization is complete. Modeler-written routines can also be released if and when appropriate.
Executional efficiency		
Optimizing compiler	NO	
Suppression/omission of collection of unneeded statistics	YES	
Other	YES	SIMSCRIPT attribute packing and word placement

PROGRAM PRODUCT QUALITY	REMARKS
ACQUISITION COSTS	
Simulator	$5000 one-time charge (includes first-year maintenance)
Other software products	SIMSCRIPT II.5 compiler must be purchased/leased from CACI, Inc.
MAINTENANCE	
Availability	FEDSIM maintains all four implementations of ECSS (SIMSCRIPT II.5 maintenance is available from CACI, Inc.).
Cost	$100/month (after first year)
Probability of continuing, long-term support	Excellent
RESOURCE REQUIREMENTS	
Specific computer systems and/or operating systems	IBM 360/370 series, Honeywell 600/6000 series, CDC 6000 series, UNIVAC 1100 series
Main memory space	220K bytes minimum
Language compiler(s), assemblers	SIMSCRIPT II.5 compiler
Other software products	None
PORTABILITY	Very portable. ECSS implementations exist for IBM 360/370 series, Honeywell 600/6000 series, CDC 6000 series, and UNIVAC 1100 series computer systems.

PROGRAM PRODUCT QUALITY	REMARKS
STABILITY/RELIABILITY	ECSS is a stable program product whose reliability has been established by extensive use over a two-year period.
DOCUMENTATION	
User's manual	Yes
Language reference manual	Yes
Program logic manual	No
Texts, tutorials	Yes
Description of:	
Default algorithms	Yes
System-defined data structures	Yes
Random number generator algorithms	No
Probability distribution function algorithms	No
Timing routine	Yes
Event management	Yes
Process management	Yes
System-defined subprogram communication conventions	Yes

PROGRAM PRODUCT QUALITY	REMARKS
TRAINING	
Courses	Yes, FEDSIM offers the following ECSS courses: 1. Introductory course, covering SIMSCRIPT II.5 and basic ECSS concepts. 2. Advanced course, covering advanced ECSS concepts.
Consultation	Yes

APPENDIX C

DIMUI EVALUATION RESULTS

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
DATA BASE MANAGEMENT SYSTEM (DBMS)			
DBMS buffers			
Description	YES	DCL	"Buffer pool description" card
Management algorithms	YES	DCL	"Buffer pool description" card identifies any of several system-supported buffer management strategies.
Management	YES	NONE	Implicit buffer management
Management/use operations	YES	STMT	ALLOCATE, RELEASE commands
Buffer-data base access interaction	YES	NONE	An implicit aspect of simulated data base accessing
Utilization statistics collection	NO	MODULE+	
Queueing statistics collection	NO		
Data base access control mechanisms	YES	STMT	LOCK, UNLOCK commands provide a mechanism for preventing/ permitting access to all or part of a data base.
Data base access scheduling algorithms	YES	NONE	Physical I/O operation scheduling only
Data base access management	YES	NONE	Implicit data base access management at physical I/O level
Request scheduling algorithms	NO	MODULE+	
Request management	NO		

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
DATA BASE MANAGEMENT SYSTEM (Cont'd)			
Data base management utility functions	YES	ROUTINE	
DBMS overhead	NO	STMT	
Request processing statistics collection	NO	MODULE+	
Request queueing statistics collection	NO		
Data base access statistics collection	YES	NONE	Automatic statistics collection
Data base access locality statistics collection	NO	MODULE+	
Data base access queueing statistics collection	NO		
DBMS overhead statistics collection	NO		
DML operation statistics	NO		

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
OPERATING SYSTEM			
Resource management strategies	YES	NONE/ROUTINE	Default management strategies are provided for job scheduling, processor management, disk access scheduling, channel scheduling, core buffer management, and disk space management; these may require modification/replacement to reflect desired management strategies.
Resource management	YES	NONE	Implicit resource management
Separation from application program behavior	YES		
Supervisory overhead	NO	MODULE+	
Supervisory overhead statistics collection	NO		
TELEPROCESSING RESOURCES			DIMUI provides no support for the modeling of TP resources, resource management, and processing activities.
PROCESSOR RESOURCES			
Definition	YES	DCL	"Processor description" card
Execution rate	YES	DCL	
Execution rate degradation due to primary memory contention	NO	MODULE+	

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
PROCESSOR RESOURCES (Cont'd)			
Different execution rates for different instruction types	NO	MODULE++	
Multiprocessor configurations	YES	DCL	"Processor description" cards
Processor management strategies	YES	NONE	A default processor management routine is provided.
Processor management	YES	NONE	Implicit processor management
Priority-based interrupt mechanism	YES	NONE	An implicit aspect of processor management
Virtual memory reference patterns associated with execution bursts	NO	MODULE++	
Multiprocessor management strategies	YES	MODULE++	
Multiprocessor management	YES	NONE	Implicit processor management
Processes which implicitly require allocation of a processor	YES	NONE	
Dynamic degradation of execution rate	NO	MODULE++	
Execution bursts	YES	STMT	CPUTIM command

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
PROCESSOR RESOURCES (Cont'd)			
Utilization statistics collection	YES	NONE	Automatic statistics collection
Queueing statistics collection	NO	MODULE+	
Utilization-by-process statistics collection	NO		
STORAGE RESOURCES: PRIMARY MEMORY			No explicit representation of primary memory; its only representation is as a collection of "buffer pools" (see I/O RESOURCES: BUFFERS).
Definition	NO	MODULE+	
Memory management strategies			
Memory management			
Allocation/deallocation requests			
Capacity utilization statistics collection			
Capacity queueing statistics collection			

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
STORAGE RESOURCES: VIRTUAL MEMORY			Virtual memory, while not represented explicitly, is basically represented by the totality of "core buffers." Virtual memory usage and management are representable using buffer usage and management operations.
Definition	NO	DCL	"Buffer pool description" card
Mapping to secondary storage	YES	ROUTINE	
Mapping to primary memory	NO	MODULE++	
Page reference	YES	STMT	FETCH, MODIFY commands
Page transfer	YES	STMT	FETCH, STORE, RELEASE, ALLOCATE commands
Management strategies	YES	NONE/ROUTINE	Several default buffer management strategies are provided.
Management	YES	NONE	Implicit buffer management
Page fault statistics collection	NO	MODULE+	
Utilization statistics collection	NO		
Queueing statistics collection	NO		

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
STORAGE RESOURCES: SECONDARY STORAGE, DIRECT-ACCESS			
Definition	YES	DCL	"Unit description" card identifies any of several system-supported DASD device types.
Capacity	YES	NONE	These characteristics are supplied for each system-supported device type. The characteristics of non-system-supported device types must first be assembled into an internal table before they can be referenced.
Physical organization	YES	NONE	
Space overhead	YES	NONE	
			require modification/replacement to characterize a particular space management strategy.
Space management	YES	NONE	Implicit secondary storage space management in response to allocation/deallocation requests.
Space allocation/deallocation requests	YES	STMT	ALBLOCK, FRBLOCK commands
Space utilization statistics collection	NO	MODULE+	
Space queueing statistics collection	NO		

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
STORAGE RESOURCES: SECONDARY STORAGE, TAPE			
Definition	YES	DCL	"Unit description" card identifies any of several system-supported tape device types.
Capacity	YES	NONE	These characteristics are supplied for each system-supported device type; the characteristics of non-system-supported device types must be assembled into an internal table before they can be referenced.
Space overhead	YES	NONE	
I/O RESOURCES: TAPE DEVICES			
Definition	YES	DCL	"Unit description" card identifies any of several system-supported tape device types.
Transfer rate	YES	NONE	These characteristics are supplied for each system-supported device type; the characteristics of non-system-supported device types must be assembled into an internal table before they can be referenced.
Rewind rate	YES	NONE	
Latency	YES	NONE	
Computation of data transfer time	YES	NONE	

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
I/O RESOURCES: TAPE DEVICES (Cont'd)			
Rewind operation	NO	MODULE+	
Computation of rewind time			
Rewind statistics collection			
Read/write operations	YES	STMT	READ, WRITE, WRITE-VERIFY commands
Utilization statistics collection	YES	NONE	Automatic statistics collection
Queueing statistics collection	NO	MODULE+	
Read/write statistics collection	YES	NONE	Automatic statistics collection
I/O RESOURCES: DIRECT-ACCESS STORAGE DEVICES			
Definition	YES	DCL	"Unit description" card identifies any of several system-supported device types.
Transfer rate	YES	NONE	These characteristics are supplied for each system-supported device type; those of non-system-supported device types must be assembled into an internal table before they can be referenced.
Latency			
Cylinder access time	YES	NONE	

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
I/O RESOURCES: DIRECT-ACCESS STORAGE DEVICES (Cont'd)			
Computation of cylinder access time	YES	NONE	
Computation of data transfer time	YES	NONE	
Read/write operations	YES	NONE	
Seek operation	YES	NONE	Low-level I/O operations are simulated as required in response to higher-level data base access requests.
Search operation	NO	MODULE+	Search operation not supported
Computation of search time			
Search statistics collection			
Seek statistics collection	YES	NONE	
Read/write statistics collection	YES	NONE	Automatic statistics collection
Utilization statistics collection	YES	NONE	
Queueing statistics collection	NO	MODULE+	
I/O RESOURCES: UNIT RECORD DEVICES			DIMUI provides no support for modeling unit record devices.
I/O RESOURCES: CONTROL UNITS			DIMUI provides no support for representing I/O device control units.

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
I/O RESOURCES: CHANNELS			
Definition	YES	DCL	"Channel description" card
Transfer rate	NO	MODULE++	
Message capacity	NO		
Utilization statistics collection	YES	NONE	Automatic statistics collection
Queueing statistics collection	NO	MODULE+	
I/O RESOURCES: OTHER I/O DEVICES			No support is provided for the modeling of I/O devices other than tape units, disk units, and channels.
I/O RESOURCES: I/O DEVICE CONNECTIVITY			Only a channel-disk/tape device architecture is supported; channel-device connectivity is specified on the "channel description" cards. A default channel scheduling algorithm is provided for controlling disk/tape device-channel connectivity.
I/O RESOURCES: BUFFERS			
Definition	YES	DCL	"Buffer pool description" card
Buffer management strategies	YES	DCL	The "buffer pool description" card identifies any of several system-supplied buffer management strategies.

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
I/O RESOURCES: BUFFERS (Cont'd)			
Buffer management	YES	NONE	Implicit buffer management
Buffer allocation/deallocation requests	YES	STMT	ALLOCATE, RELEASE commands
Buffer-I/O interaction	YES	NONE	Implicit aspect of simulated data base accessing
Utilization statistics collection	NO	MODULE+	
Queueing statistics collection	NO		
I/O RESOURCES: I/O DEVICE MANAGEMENT			
Management strategies	YES	NONE/ROUTINE	DIMUI provides default disk/tape device and channel scheduling strategies; these may require modification/replacement in order to characterize different management strategies.
Management	YES	NONE	Implicit disk/tape device and channel management
Allocation/deallocation requests	YES	NONE	I/O device allocation/deallocation is an implicit aspect of simulated I/O operations.
DATA RESOURCES: FILES/FILE STRUCTURES			
Definition	YES	DCL	"Area (file) description" card
Record characteristics	YES	DCL	"Extent description" card

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
DATA RESOURCES: FILES/FILE STRUCTURES (Cont'd)			
Space allocations	YES	DCL	"Extent description" card
Access methods	NO	ROUTINE	
Record-to-space allocation mappings	YES	ROUTINE	
Space allocation-to-physical storage media mappings	YES	DCL	"Extent description" card
Translation of file access into I/O activity	YES	NONE	Automatic translation of file access into I/O activity
File access requests	YES	STMT	FETCH, STORE, PREFETCH commands
Dynamic file creation/deletion	NO	MODULE++	
Dynamic file space allocation	YES	STMT	ALBLOCK statement
File management strategies	NO	MODULE+	
File management	NO		
Access statistics collection	NO	MODULE+	
Queueing statistics collection	NO		
Utilization statistics collection	NO		

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
DATA RESOURCES: DATA BASE			
Data structure (DS) representation	YES	DCL	"Area description," "record description," "set description" cards
Multiple DSs	YES	DCL	
DS-to-DS mappings	NO	MODULE++	
DS access/traversal operations	YES	STMT, ROUTINE	FETCH, STORE, PREFETCH are provided; additional operations oriented toward a particular DBMS must be procedurally characterized by the modeler.
Translation of access/traversal operations into processing time and physical I/O activity	YES	ROUTINE	Performed by the modeler-written routine which characterizes a particular access/traversal operation.
DS-to-secondary storage media mapping	YES	DCL	
DS-to-virtual storage mapping	YES	DCL	"Extent description" card
DS-to-file (structure) mapping	NO	MODULE++	
DML command characterization	YES	ROUTINE	
DML command invocation	YES	STMT	FORTRAN CALL statement

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
DATA RESOURCES: DATA BASE (Cont'd)			
DS access paths	NO	STMT	DS access paths are represented by stels (stack elements) which are explicit parameters of all DS accessed traversal operations.
Dynamic access path selection	NO	MODULE+	
Access path statistics collection	NO	MODULE+	
Data structure accessing statistics collection	NO	MODULE+	
SOFTWARE RESOURCES			
Definition			No support for the representation of a software module as a resource.
Management strategies			
Management			
Allocation/deallocation requests	NO	MODULE+	
Utilization statistics collection			
Queueing statistics collection			

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
PROCESSES			
Definition/characterization	YES	ROUTINE	Task description
Implicit processor requirement	YES	NONE	
Time delays	YES	STMT	CPUTIM command
Initiation	YES	DCL	"Workload description" card
Termination	YES	STMT	FORTRAN RETURN statement
Synchronization/Communication/Suspension			
Wait for sub-process	NO	MODULE+	
Wait/post mechanism	YES	STMT	"Block synchronization" facilities provide necessary mechanism for process suspension/synchronization.
Wait for I/O	YES	NONE	An implicit aspect of simulated data base accessing
Inter-process message passing	NO	MODULE+	
Initialization, saving, and restoring of local process variables	YES	NONE	

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
PROCESSES (Cont'd)			
Process lists	NO	MODULE+	No support for modeler-defined process lists (queues) and list processing operations.
Process list entry discipline	NO		
Process list operations	NO		
Occurrence statistics collection	NO	MODULE+	
Processor utilization statistics collection	NO		
Processor queueing statistics collection	NO		
Resource utilization satistics collection	NO		
Resource queueing statistics collection	NO		
I/O statistics collection	YES	NONE	Automatic statistics collection
WORKLOAD			
Workload element representation	YES	ROUTINE	Task description
Workload element arrival pattern specification	YES	DCL	"Workload description" card
Workload element processing statistics collection	YES	NONE	Automatic statistics collection

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
GENERAL-PURPOSE SIMULACRA			
Events			
Definition	NO		
Internal events	NO	MODULE++	DIMUI does not support the concept of a general-purpose event and its associated operations.
External events	NO		
Recurring events	NO		
Event scheduling operations	NO		
List data structures			
Definition	NO	MODULE+	DIMUI does not support a generalized list processing capability.
Dynamic creation	NO		
List elements			
Definition	NO		
Data aggregates	NO		
Dynamic creation/deletion	NO		
List processing operations	NO		

MODELING CAPABILITY	EXPLICITLY PROVIDED?	LEVEL OF MODELING EFFORT	REMARKS: HOW CAPABILITY IS ACHIEVED, LIMITATIONS
GENERAL-PURPOSE SIMULACRA (Cont'd)			
Random number generator	YES	NONE	Built-in multiple-seed random number generator
Probability distribution functions	YES	DCL	A limited set of probability distribution functions are supported.
Empirical distributions	YES	DCL	
Event and clock management	YES	NONE	DIMUI-supplied timing routine
Reentrant routines	YES	STMT	DIMUI supplies necessary data structures and routines to allow the modeler to write reentrant FORTRAN subroutines.
Basic computational facilities	YES		FORTRAN IV language facilities

SUPPORT FACILITY	AVAILABLE?	REMARKS
STATISTICS COLLECTION CONTROL		
Suppression/resumption	NO	
Selective suppression	NO	
Selective omission	NO	No statistics collection control capabilities are documented.
Reset	NO	
Cumulative collection	NO	
Interval collection	NO	
STATISTICS REPORTING		
Automatic report generation	YES	
Selective report generation	NO	
Report generation scheduling		
End-of-simulation	YES	
Periodically	NO	
Via external events	NO	

SUPPORT FACILITY	AVAILABLE?	REMARKS
MODEL SYNTHESIS		
Model source component library facilities	NO	
On-line, interactive prompting/help facility for model source input	NO	
On-line model source editing	NO	
Macro processing	NO	
MODEL VERIFICATION AND VALIDATION		
Run-time diagnostic facilities	YES	FORTRAN IV run-time diagnostic facilities
Model debugging facilities		
Tracing	YES	Selective logging of event and interrupt occurrences
Formatted dumps	YES	Diagnostic dumps can be produced using the "Dump" card.
Behavior statistics	YES	
Routine traceback	NO	May be provided by FORTRAN run-time system.
Snapshot	NO	

SUPPORT FACILITY	AVAILABLE?	REMARKS
MODEL VERIFICATION AND VALIDATION (Cont'd)		
Diagnostics at model source statement level as well as at base language level	NO	
On-line, interactive debugging facilities	NO	
Supplemental debugging facilities	YES	May be programmed using FORTRAN IV
MODEL INITIALIZATION AND MODIFICATION		
Pre-simulation input of model initialization data		
System-defined capability for parameter value entry	YES	DIMUI supports extensive model parameterization
Modeler-programmable parameter value entry and storage	YES	Using FORTRAN IV
Modification with minimal or no model source re-processing		
Assigning new values to model parameters	YES	The extensive model parameterization features facilitate model modification.
Hardware re-configuration	YES	
(Process) routine replacement	YES	

SUPPORT FACILITY	AVAILABLE?	REMARKS
MODEL PROCESSING		
Model source syntax checking	NO	
Library facilities for the storage/retrieval of object/load modules	NO	
Diagnostics facilities	YES	
Conditional termination of model processing at any step due to error(s) in a previous step	YES	
Operating system-independent control language to initiate and control model processing	NO	
Inclusion of previously-processed model components	NO	
MODEL EXECUTION		
Run-time diagnostic facilities	YES	FORTRAN run-time diagnostic facilities
Checkpoint/restart	NO	
Simulation terminal control		
Simulated-time based	YES	

SUPPORT FACILITY	AVAILABLE?	REMARKS
MODEL EXECUTION (Cont'd)		
Process/event-occurrence based	YES	Simulation can be terminated at the completion of a specified process or after completion of all processes.
Error-occurrence based	NO	
Run-time based	NO	
Operating system independent language to initiate and control model execution	NO	
Run-time monitor providing on-line interaction with a model during execution	NO	
Multiple simulations during a single run	NO	
MODEL OPTIMIZATION		
Main storage space requirements		
Program overlay	NO	
Deletion of unneeded routines	NO	
Suppression/deletion of unneeded data structures	NO	

SUPPORT FACILITY	AVAILABLE?	REMARKS
MODEL OPTIMIZATION (Cont'd)		
Release of initialization and other routines not required after start of simulation	NO	
Executional efficiency		
Optimizing compiler	YES	FORTRAN optimizing compiler
Suppression/omission of collection of unneeded statistics	NO	

PROGRAM PRODUCT QUALITY	REMARKS
ACQUISITION COSTS	
Simulator	None, since DIMUI is a public domain product.
Other software products	Cost of FORTRAN compiler
MAINTENANCE	
Availability	Not applicable due to developmental nature of DIMUI
Cost	
Probability of continuing, long-term support	
RESOURCE REQUIREMENTS	
Specific computer systems and/or operating systems	IBM 360/370
Language compiler(s), assemblers	FORTRAN compiler, IBM 360/370 assembler
Other software products	None
PORTABILITY	Use of PL/1 for preprocessor currently restricts DIMUI model processing to IBM 360/370 systems.
STABILITY/RELIABILITY	DIMUI is still a developmental system

PROGRAM PRODUCT QUALITY	REMARKS
DOCUMENTATION	
User's manual	Yes. DIMUI-IDMS User's Manual
Language reference manual	No
Program logic manual	No
Text, tutorials	No
Descriptions of:	
Default algorithms	Yes
System-defined data structures	Yes
Random number generator algorithms	No
Probability distribution function algorithms	Yes
Timing routine	Yes
Event management	Yes
Process management	Yes

PROGRAM PRODUCT QUALITY	REMARKS
TRAINING	
Courses	No
Consultation	No

DISTRIBUTION

NO. COPIES

Director
AIRMICS
Calculator Building
313 Georgia Tech
Atlanta, GA 30332

5