AD NUMBER

AD811682

NEW LIMITATION CHANGE

TO

Approved for public release, distribution unlimited

FROM

Distribution: Further dissemination only as directed by Director of Computers Electronic Systems Div., AFSC, Hanscom Field, MA, MAR 1967, or higher DoD authority.

AUTHORITY

ESD USAE ltr, 24 Jul 1968

ESD-TR-67-2

A METHODOLOGY FOR COMPARISON OF GENERALIZED DATA MANAGEMENT SYSTEMS:

PEGS (PARAMETRIC EVALUATION OF GENERALIZED SYSTEMS)

Anthony J. Dowkont
William A. Morris
T. Dwight Buettell

March 1967

DIRECTORATE OF COMPUTERS
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
L. G. Hanscom Field, Bedford, Massachusetts 01730

(Prepared under Contract No. AF 19(628)-5936 by Informatics Inc., 5430 Van Nuys Boulevard, Sherman Oaks, California.)

(18) (19)

ESD-TR-67-2

(6)

A METHODOLOGY FOR COMPARISON OF GENERALIZED DATA MANAGEMENT SYSTEMS:

PEGS (PARAMETRIC EVALUATION OF GENERALIZED SYSTEMS).

(9) Final rept. Mar-Nov 66,

(10) Anthony J. Dowkont,
William A. Morris
T. Dwight Buettell

(11) Mar[redacted]67 (12) 286p.

(15) AF-19(628)-5936

(16) AF-2801

DIRECTORATE OF COMPUTERS
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
L. G. Hanscom Field, Bedford, Massachusetts 01730

(17) 280114

(Prepared under Contract No. AF 19(628)-5936 by Informatics Inc., 5430 Van Nuys Boulevard, Sherman Oaks, California.)

mt

1473 ef

(178 030)

FOREWORD

This Technical Documentary Report was prepared by Informatics Inc., Sherman Oaks, California, under USAF Contract AF 19(628)-5936; Project No. 2801, Task No. 2801.14, "Data Base System Technology". The work was performed under the direct supervision of T. Dwight Buettell, Informatics' Project Manager, and under the general direction of John A. Postley, Vice President, Advanced Information Systems Division. The work was accomplished between March 1966 and November 1966 and was administered under the direction of the Electronic Systems Division, Deputy for Command Systems, Computer and Display Division, L. G. Hanscom Field, Massachusetts, with Lt T. M. Sparr, ESVPT, serving as task monitor.

This report has been reviewed and is approved.

CHARLES A. LAUSTRUP, Colonel, USAF
Chief, Computer and Display Division

ABSTRACT

The objective of this research study contract was to develop a practical technique for evaluating generalized data management systems. This report describes the technique that was developed for the quantitative evaluation of the relative effectiveness of large on-line generalized data management systems.

Parametric Evaluation of Generalized Systems (PEGS) is a procedure based on analysis of user-oriented system parameters. The utility of a system is measured in terms of its usefullness in a particular application environment. The overall effectiveness of the system is evaluated, rather than any individual hardware or software component.

A large number of system parameters is described. Each parameter is a value attribute of a data management system, with respect to its capability, its ease of use, or its performance.

Techniques are specified for measuring the utility of a system to the user in terms of each parameter. These measurements of individual parameter utility are expressed as ratings based on a standard scale. Each rating is weighted by a measure of its relative importance in a particular application. Finally, a single numeric figure-of-merit is computed for each generalized data management system evaluated.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS/ACRONYMS

GDMS: Generalized Data Management System

PEGS: Parametric Evaluation of Generalized Systems

FCM: File Creation and Maintenance

Section I

INTRODUCTION

This report is organized in the following manner. Section I provides the background for the study; describes a GDMS (Generalized Data Management System) and related topics; and discusses measures of effectiveness.

Section II is an overview of the PEGS evaluation procedure, and describes the steps, such as weighting and rating, that an evaluator follows.

Section III describes parameter organization and measurement and lays the groundwork for the following section, which describes the parameters. Various views on parameter organization are developed; examples of parameter measurement are illustrated; and the Parameter Worksheet is described.

Section IV contains the parameter descriptions and constitutes the major portion of the report. The section is divided into six subsections: Data Definition and Data Organization; File Creation and Maintenance; Retrieval; Processing; Output; and Environmental Considerations. The first part of each major subsection contains descriptive text, and the other part consists of the corresponding Parameter Worksheets.

Section V covers other approaches of interest which were developed during the course of the study.

The last section contains a summary of the advantages of PEGS; a brief discussion of the problems of quantitative evaluation; and comments on future work.

1.1 PURPOSE AND OBJECTIVES

A need exists within the Air Force for evaluating large on-line data management systems. This need arises from the necessity for a military commander to choose, among the increasing number of generalized systems that are available, the system that will be most useful for his data management applications. In recognition of this need, Electronic Systems Division awarded a contract to Informatics Inc. for the development of a methodology for evaluating the total effectiveness and power of a generalized data management system. Under this contract, Informatics has developed PEGS (Parametric Evaluation of Generalized Systems).

The objective of the study was to develop a parametric analytical approach which incorporates the following pragmatic advantages:

- It is practical to use
- It is capable of evaluating complex systems
- It provides meaningful results.

These objectives have been achieved; in addition, PEGS provides a systematic approach for analyzing GDMS's and other related systems. Computations are simple and a computer program is not required as is usually the case with simulations. The resultant score for a GDMS is meaningful in terms of the standard rating scale that was developed. The weighting and rating methods are flexible and are suitable for a wide range of applications. The parameter list is broad and open-ended. These and additional advantages of PEGS are discussed throughout the report and summarized in Section 6.1.

PEGS can be described by outlining the procedure for its use. The first step in using PEGS is to formulate the objectives and requirements of both the evaluation and the application environment. The application requirements are formulated in terms of parameters developed in the study; a comprehensive list of parameters was developed for this purpose. The selected parameters are assigned weights according to their relative importance in the application. Next, the GDMS's being evaluated are analyzed and their capabilities measured and rated in terms of their effectiveness in fulfilling requirements. Then, parameter scores are computed and aggregated into an overall system score for each GDMS evaluated. Last, the weights, ratings, and scores are reviewed, adjusted if necessary, and a final score computed.

1.2 GDMS EVALUATION PROBLEM

The state of the art of data processing technology is advancing at a rapid rate. Computers are becoming faster and more powerful while the cost per instruction executed has been steadily declining. Improvements in software continue to be made, yet the time and cost to implement a system, using conventional programming techniques, has not been reduced dramatically. This is true despite the development of a proliferation of programming languages, operating systems, utility programs, etc.

Major users of data processing equipment, such as the Air Force, are becoming more and more concerned with the lack of better programming techniques for a certain class of applications. These applications are characterized by large complex data bases and/or by unknown query requirements. In the past, the solution of these applications has required a substantial amount of initial programming and subsequent program modification. This results in high cost, long elapsed implementation times, limitations on operational capability, and inflexibility to changes in requirements.

The development of generalized programming techniques in the past few years is showing great promise towards overcoming the problems just mentioned. Generalized data management systems have been and still are being developed to facilitate the solution of a variety of applications, including those with large data bases and complex query requirements. These GDMS's vary in power, design, acquisition cost, operating cost, ease of use, etc., and there is a growing need for a technique to evaluate GDMS's.

The determination of the power and effectiveness of a GDMS is not a trivial task regardless of whether the measure of power is quantitative or qualitative. A GDMS consists of many capabilities and features; typically, only a subset of these capabilities is employed in an application, and their relative importance varies among applications. Hence, the power of a system simply is not obvious.

1.3 GENERALIZED DATA MANAGEMENT SYSTEMS

1.3.1 Description

A Generalized Data Management System (GDMS) has the capability of handling a wide range of file management applications; the system is generalized in that it has to be adapted for use in each application. The main objective of the generalized capability is to reduce the total time and cost required for problem solution. A GDMS can be designed with many different objectives; the purpose of the study is to evaluate the effectiveness of the system and not the validity of the design objectives.

Although the capabilities of a GDMS can be accomplished with conventional programming techniques, GDMS's have proven useful for one or more of the following reasons:

- Reduced costs
- Ease of use
- Faster implementation time
- Direct user access to data base
- Indirect improvement in system design of an application

These benefits are achieved through the adaptation of generalized capabilities for specific applications. It is unlikely, however, that any existing or proposed GDMS is sufficiently generalized to handle any problem; if it were, it probably would amount to a conventional programming language or system.

Since a generalized system is not designed for a single application, the desired functions and file definitions for a given problem must be specified and furnished to the system. The GDMS either interprets the specifications at execute time or compiles an object program incorporating the specifications. The task of conventional computer programming for a file management application is replaced (or drastically reduced) by the usually easier job of defining problem specifications in a GDMS language. A significant saving in both time and cost of implementation is thus achieved.

It is difficult, if not impossible, to define precisely what a GDMS is and to decide whether a specific set of computer programs is or is not a GDMS. This is especially true when a computer program possesses a limited degree of generalized capability. In addition, most GDMS's have some special purpose as well as generalized capabilities, and these capabilities (both special and general purpose) vary considerably among systems. This is a result of differences in design objectives, in design approach, in resources available for development, and in the hardware used.

A GDMS as defined here includes all of the necessary hardware and software to operate the system. The major components of a GDMS are:

1) A set of generalized file management programs
2) The required operating system or its equivalent and other software (if any) to operate and support the file management programs
3) The specific configuration of computing hardware used to execute the foregoing programs

It is necessary to include hardware in the analysis since a GDMS may be operable on more than one computer or on more than one configuration of a computer. The operating software and computing hardware will not be evaluated as such; their effect will be evident in overall GDMS performance characteristics and capabilities.

The generalized file management capability consists of the following major functions:

- Data Definition and Data Organization
- File Creation and Maintenance
- Retrieval
- Processing
- Output

These functions are consistent with the organization of parameters developed in the study (Section 3.1).

There is an implicit definition of a GDMS in the parameter list in that most (if not all) of the functions and components of a GDMS are described. A precise definition of a GDMS is of little consequence insofar as the use of PEGS is concerned. The technique covers a broad spectrum of capabilities which should cover most GDMS's and many other systems as well. The open-ended nature of the parameter list provides for adding parameters to accommodate any type of capability or requirement.

Some GDMS's possess on-line capabilities. The definition of an on-line system has been the subject of many papers and much discussion; a simple definition will suffice here. An on-line system provides the capability for a person to communicate directly with the system and to receive a rapid response from the system. For example, the capability to enter a query into a system using a teletype terminal and to receive a response in a matter of seconds is considered on-line. The response may be the answer to the query or an indication that the query has been received and is being processed. The subject of on-line capabilities is discussed further in Section 4.3.3.

Some GDMS's have document retrieval capabilities, such as keyword processing, that are not covered in the previous definition and the parameter list. The specialized capabilities required for document retrieval and text processing were considered to be outside the scope of the study.

1.3.2 Users of GDMS

The user of a GDMS is defined as the person who utilizes or otherwise has a need of the outputs of a GDMS. The user has one or more file management applications which collectively constitute a problem-mix. This problem-mix, in turn, is being processed by a GDMS. Since we are studying on-line systems, at least some of the file management applications are assumed to have an on-line requirement of some kind.

The user is thought of as being middle or upper management and is probably not concerned with all of the detailed capabilities of a GDMS. His main interest is the fulfillment of the requirements of his problem-mix in a broad sense, that is, "getting the job done." Although user satisfaction with a GDMS is important, it is unlikely that a user is qualified to evaluate GDMS's.

Some users may personally interact directly with a GDMS by means of an on-line terminal device. Other users, however, may be serviced by assistants or an operating organization, and may have no direct contact with a GDMS other than receiving output reports or output information. There may be more than one user for a given application or problem-mix, and each of several users could have his own set of requirements distinct from any other user.

A further consideration in evaluating two GDMS's is that a user (or group of users) may have:

1) A problem-mix that is actually being processed by a GDMS

2) A problem-mix that is being supported by conventional programming techniques

3) A proposed problem-mix that is being considered for implementation on a GDMS

1.3.3 Operating and Support Personnel

A data processing organization operates a GDMS for a user. The maintenance of the file management programs and other required software as well as the operation of the computing facility are all performed by data processing personnel. In addition, assistance in problem analysis and GDMS implementation is provided. The extent of assistance in the preparation of queries and other tasks depends upon the role of the individual user in a GDMS.

Except for possible on-line activity by the user, operating and support personnel provide the interface between a GDMS and a user. Many of the characteristics of a GDMS, therefore, are of more than casual interest to those personnel.

1.3.4 Evaluator of GDMS

The person charged with the responsibility for evaluating a GDMS can have many different orientations or viewpoints: he may be a user, a user's superior, a GDMS operator, a GDMS designer, an outside consultant, etc. His background is bound to have some effect on an evaluation, and the inclusion or exclusion of such bias will be discussed later. Of primary importance here is the need for the evaluator, regardless of his organizational interests, to be an experienced systems analyst with considerable knowledge of computer concepts and programming. If the evaluator does not possess this technical background, he should have qualified persons assisting him in any analysis of a GDMS.

All future references to an evaluator assume that he is the person who will by applying the techniques developed in this study, and that he is (or has access to persons who are) knowledgeable in systems analysis and computer sciences. Who the evaluator should be and the desirability that he be familiar with GDMS techniques will be treated later on.

The evaluator should have extensive experience in business systems, computer programming, computer applications, file management systems, and analysis and evaluation techniques. Since it is unlikely that many individuals have sufficiently extensive experience in all of these areas, it is recommended that an evaluation team consisting of two or more members be used for an evaluation. For a two member team, one person should be primarily a systems analyst and the other primarily a computer programmer; both members should have at least some familiarity with file management systems.

1.3.5 Problem-mix

A problem-mix consists of one or more applications, each of which involves file management. The requirements of these applications collectively constitute the requirements of the problem-mix. Each evaluation of a GDMS must be made with a specific problem-mix in mind, with the requirements of the problem-mix stated in terms of the GDMS list of parameters.

It is possible that several different requirements for a single parameter can exist when a problem-mix consists of more than one application. Both the minimum and desired requirements can vary, and the weighting of a parameter could be different within each application. Further, the importance of each application could be different. If such conditions exist, it may be necessary to perform an evaluation for each application as a problem-mix, and to weigh these results according to the importance of each application.

1.3.6 Data Organization

Data organization involves physical and logical organization of data in a file and is discussed in detail in Section 4.1. The definitions that follow relate to the logical organization of data, and they will be used throughout the report. There are many different terms and concepts in use, and there is no known list of standard definitions that is univerally accepted. The terms to be defined describe the hierarchic levels of data organization, that is, the logical relationship of data groupings in an aggregation of data. Some examples of hierarchic terminology are:

- Data set, record
- File, record, segment, field
- File, record, group of elements, element of data
- File, record (master), record (detail), data field
- File, object, group, property
- File, entry, subfile, data fields
- File, record, subrecord, field
- File, property, sub-property
- File, subfile, data set
- Record, subrecord
- Subrecord, segment, group
- Field, data element, element, item

The hierarchy used in this report is:

- Data base
    - File
        - Record
            - Segment
                - Field

Data Base

A data base is the aggregate of all of the data available in a GDMS, including users data files, working files, computer programs, etc. The highest level of data organization within a data base is a file.

File

A file is an organized collection of one or more logically related records.

Record

A record consists of one or more logically related fields. A key, consisting of one or more fields in the record, identifies a particular record in a file. In addition, a record may have groups of one or more logically related subordinate segments.

Segments

A segment is a subordinate logical unit within a record. A segment consists of one or more logically related fields. A key consisting of one or more fields in the segment identifies a particular segment in a group. A segment, in turn, may have groups of one or more logically related subordinate segments.

Field

A field contains one item of information which describes one property of the subject of the record.

1.4 MEASURES OF EFFECTIVENESS

The statement of work specifies that the study should develop a technique for a quantitative evaluation of the total effectiveness of a generalized data management system based on an anlaysis of relevant system parameters. A quantitative parametric evaluation approach has been developed, but it became apparent during the course of the study that a great deal of care is necessary both in the development and in the use of such an approach.

The effectiveness of a system can be measured by analyzing its parameters and estimating the:

1) Degree of excellence in fulfilling a requirement
2) Degree of excellence measured against a theoretical "ideal" system
3) Degree of excellence measured against another system being evaluated.

The first approach listed above is used in PEGS. Degree of excellence is measured by analyzing a system and evaluating the effectiveness of its capabilities in fulfilling a requirement. The requirement can reflect almost any desired characteristic, such as capability (e.g., requirement for a certain record size), performance (e.g., requirement for a specific response time or better), etc.

Effectiveness also can be analyzed in terms of the value of the output of system rather than the value of its capabilities. A GDMS has value if its outputs:

1) Are produced more cheaply than with another system
2) Are produced faster than with another system (more timely)

3) Contain information which is used to achieve cost savings in the operating organization

4) Are of value to a user (who is willing and able to pay for them)

5) Are produced with less effort (not directly measurable in terms of time and cost) than with another system.

To some extent, items in the above list can be requirements that can be evaluated using system parameters. If such items are requirements, there may be overlap or double measuring of costs; this will be discussed later.

1.4.1 Costs

The subject of cost is always of more than casual interest in an evaluation; in this technique the cost of using a GDMS should be compared with its power and effectiveness. The primary purpose of this study is to develop a technique for quantitatively measuring power and effectiveness of a GDMS. This qualitative measure is not stated in dollar units and does not explicitly include the cost of using a GDMS. It is intended, however, that the total measure of effectiveness developed for a GDMS be compared with the total costs of using that GDMS.

The design of a GDMS involves two considerations of cost: How much to spend on the development of specific GDMS capabilities, and how much will a capability cost to use? Both of these considerations involve cost trade-offs when more than one method of designing a capability exists, and differential cost analysis techniques are applicable. The evaluation of design and development, costs, of course, is beyond the scope of this study.

Some elements of cost are considered in the ratings for several parameters. It is not the objective of the technique, however, to include all costs in the overall rating for a GDMS. The cost of using a GDMS is a major consideration which warrants separate treatment. It is felt that the development and comparison of total cost and overall effectiveness as two separate figures usually provides a better basis for evaluation than a single combined measure of effectiveness which includes cost.

The ratings for some parameters are based on processing times and man-hours. Processing times are included in order to evaluate performance in terms of requirements, and man-hours to perform tasks such as data definition are used to evaluate ease of use of a capability. Both computer time and man-hours would be extended into dollar costs in a cost analysis.

1.4.2 Total Costs

The comparison of total costs with an index of effectiveness is complicated by the problem that a single total cost figure may be difficult to formulate. The cost of using a GDMS may consist of an initial one-time cost, a recurring cost each time an application is run, and a program maintenance costs. The initial cost includes the cost of acquiring the GDMS program, training cost, and implementation costs. These initial costs should be allocated among all applications; this is usually a problem since all of the eventual applications of a GDMS are not known at the outset.

Recurring costs for an application consist of equipment, supplies, and services (computer, peripherals, keypunch, communications, etc.) costs, operator costs, programmer/analyst (problem analysis, data definition, task specification preparation, etc.) cost, and other manpower (data preparation) costs. These costs vary within an application depending on whether a file is being generated, maintained, or queried and on the complexity of each task being performed. Accounting for equipment costs may not be precise, may vary depending on utilization or load factors (and whether purchased or leased), and may involve policies on allocation of overhead charges.

Program maintenance costs include work on the basic GDMS program and on other required software (e.g., operating systems, utility routines, etc.).

The major components of cost discussed above include:

1) Initial
   - Acquisition
   - Implementation
   - Training
   - Facilities
2) Operating or Recurring
   - Machine Time
   - Communications
   - Operator
   - Programmer/Analyst
     — Problem Analysis
     — Programming
   - Data Preparation
   - Facilities Upkeep
   - Supplies
3) Maintenance
   - Basic GDMS Program
   - Operating System and Utility Software
   - Hardware

The foregoing is intended to illustrate that a determination of costs is not a trivial matter; nonetheless, every effort should be made to develop total cost data for each GDMS evaluated. The effectiveness rating that is developed for a GDMS should not be used as the sole basis for evaluation.

The total cost of using a GDMS is of primary interest. Although detailed costs of capabilities or functions are made primarily for the purpose of arriving at a total cost figure, such detail can be useful in comparing systems.

The cost of and the effectiveness of a system can be arrived at by using different parameters. The list of parameters in this study is not intended to serve as a framework for cost estimation. The parameters describe capabilities, performance, and other considerations.

1.4.3 Cost Effectiveness

Ideally, a cost effectiveness study would provide the dollar value and cost of using a GDMS or of conventional programming to fulfill a problem requirement. The value derived from using a GDMS would be used as a measure of effectiveness and would be evaluated against the cost of use. The evaluation of two or more GDMS's would entail a comparison of values and costs for all systems. If the system that offers the most value is also the lowest in cost, then the choice is clear-cut. However, if a higher cost system also provides greater value, the choice of the most effective system is not obvious and requires judgement as well as consideration of other factors.

The value or effectiveness of a GDMS can be estimated in total or by component and summed to a total. It is extremely difficult to develop a technique for this approach using monetary units for value. Furthermore, the "total" approach is not practical even with quantitative non-monetary units. The analysis of the effectiveness of the detailed parameters of a system does provide a reasonable basis for a rating of overall effectiveness.

In a cost-effectiveness study the cost of doing something is compared against the effectiveness (or value) of doing it; the effectiveness can be measured in dollars or some other unit. It is difficult to measure the dollar value of the effectiveness of each parameter in a GDMS; for this reason, effectiveness is measured in terms of how well a capability meets a requirement.

Some parameters describe optional methods or capabilities which result in faster or more efficient operations. The use of these capabilities contributes to lower operating costs or to convenience of use. Also, a range of hardware options may be available which affect the cost and performance of a GDMS. This study does not attempt to analyze the cost-effectiveness of such optional capabilities in order to determine the optimum use of a GDMS. The evaluation process and the examination of costs, however, should provide some insights as to the value or usefulness of the options available.

The value or effectiveness of a system is not necessarily the sum of the values of its components. The value of a system stems from the fulfillment of a requirement which consists of a set of detailed specifications. Some of these specifications may be mandatory; others may be variable or optional. The capability to meet all mandatory specifications collectively can be evaluated; the capability to fulfill any one of many mandatory specifications is of little interest unless all mandatory requirements are fulfilled. Similarly, the value of performing optional specifications is based on the premise that mandatory requirements are met.

Section II

EVALUATION PROCEDURE

The major steps performed during an evaluation are to: determine objectives and requirements; assign weights; analyze and measure GDMS capabilities; rate parameters; and compute and review scores. The detailed steps that an evaluator must perform during an evaluation depend on the objectives of the evaluation, the GDMS's being evaluated, and the complexity of the requirements. Some steps will be performed once, whereas others will be repeated one or more times. The detailed steps are:

1) Determine objectives and requirements
   - Determine evaluation objectives
   - Select applications
   - State application objectives
   - Formulate application requirements
   - Translate application requirements to parameter requirements
   - Development bench mark problems
   - Select parameters

2) Assign weights

3) Analyze and measure GDMS capabilities
   - Analyze and select GDMS's
   - Measure GDMS capabilities
   - Check mandatory requirements

4) Rate parameters

5) Compute and review scores
   - Compute and accumulate scores
   - Review and adjust entries
   - Compute final scores for the problem-mix
   - Overall evaluation of final scores

Each of the above steps is discussed in this section; the measurement and rating steps are treated in depth in Sections III and IV.

2.1 DETERMINE OBJECTIVES AND REQUIREMENTS

2.1.1 Determine Evaluation Objectives

The first step in any evaluation is to determine the objectives of the evaluation. This is necessary since the specific evaluation tasks to be performed and their execution sequence depend on the nature of the objectives. In writing about objectives, Hitch and McKean have stated: "Choice of objectives is fundamental: if it is wrongly made, the whole analysis is addressed to the wrong question."*

The evaluation technique can be applied in a number of different ways and for a variety of objectives. The major type of use is to evaluate two or more GDMS's for a given set of requirements. The GDMS's can be:

- Two or more existing systems
- Two or more proposed systems
- Any combination of existing and proposed systems

Earlier it was felt that only two GDMS's could be evaluated at one time, and that an evaluation of more than two systems would

---

*Hitch, Charles J., and McKean, Roland N., The Economics of Defense in the Nuclear Age, R-346. The RAND Corporation, March 1960, p. 118.

require an evaluation of each combinatorial pair of systems. This would be the case if the result of a technique is a single figure of relative merit which indicates which of two systems is superior. The technique developed in this study provides an absolute measure of effectiveness for each system evaluated; the overall score is a weighted rating ranging from 0 to 10. The scores for any number of systems can be readily compared to determine relative effectiveness. However, the comparison of such scores should be done only when the evaluation scores have been developed on a consistent basis. This may require that the same evaluator (or team of evaluators) perform all evaluations when overall scores are to be compared.

Another use of the technique is to determine which application, if any, in a problem-mix is suitable for implementation with a given GDMS, as well as to determine the effectiveness of the GDMS for each application. This is essentially the reverse situation from the primary use described earlier. In the major use, a problem-mix is given and two or more GDMS's are evaluated; in this case, a GDMS is given and evaluated for one or more applications in a problem-mix. Other variations should be obvious to the reader.

Still another possible use of the technique is to employ it during the design of a new GDMS. This use was not anticipated, but is a by-product of the study. The parameter list is used as a check list and a guide. In this case, a set of requirements is formulated and used to determine the characteristics of each parameter in the new GDMS. Or, an existing system can be evaluated and used as a vehicle for a new design.

The technique can also be used to determine what modifications should be made of a GDMS to bring it up to a desired level of effectiveness.

2.1.2 Select Applications

The application (or applications constituting the problem-mix) may have been selected in advance or the evaluator may have to select applications for the GDMS's to be evaluated. In the first instance, the problem is one of determining which GDMS does the best job. In the other situation, GDMS's also are evaluated, but the applications (if any) which are suitable for GDMS implementation must be determined first.

The applications selected should have at least some requirements that are best fulfilled by a GDMS. The evaluator should recognize that some applications should not be implemented with a GDMS, and that another type of programming system or approach is more appropriate (e.g., RPG, COBOL, etc.)

2.1.3 State Application Objectives

The objectives of each application selected should be determined in order to formulate requirements, weight parameters, and provide a basis for the final overall evaluation. Examples of application objectives are:

- Lowest cost
- Fastest response time
- A specific response time (or better)
- Ease of use for a specific level of user
- Faster implementation time

Some application objectives, such as fastest response time, can be treated as requirements and reflected in the parameter list. Other objectives, such as lowest cost, however, are evaluated by a separate analysis and compared with the score for power and effectiveness. Since the distinction between application objectives and requirements is not clear-cut and not of special significance, the important consideration is that all aspects of an application are included in an evaluation.

2.1.4 Formulate Application Requirements

Once the application or problem-mix has been selected and the application objectives determined, a set of requirements can be developed. The parameter list should be used as a basis for formulating requirements. Every parameter should be analyzed to determine if a requirement exists, and if one does exist, it should be noted on the Parameter Worksheet illustrated in Figure 8 and described in Section 3.2.2.

The formulation of requirements is a mandatory and important step in the evaluation procedure. This is true whether the applications are existing, proposed, future, or generally unknown, since the basis of the technique is a comparison of requirements and capabilities. It is recognized, however, that the detail available on application requirements will vary considerably, and that the evaluator will have to estimate or assume requirements for many parameters. Even in the worst case, where a GDMS is to be used for largely unknown applications in the future, a set of requirements should be assumed based on past experience and whatever information is available. Usually, there will be some basis for making estimates of future requirements. A total lack of knowledge of future requirements is unlikely since GDMS evaluations may be presumed to involve file management or file management related applications.

The determination and analysis of application requirements is not a simple step in the evaluation procedure. Although the formulation of requirements or specifications for applications implemented with conventional programming methods is reasonably straightforward, this is not so for a GDMS. A GDMS has pre-programmed generalized capabilities that must be adapted for each application; this can result in a modification of requirements either to accommodate the GDMS or to take advantage of "free" capabilities that already exist. A good deal of iterative analysis will probably be required for each GDMS in an evaluation.

2.1.5 Translate Application Requirements to Parameter Requirements

A full statement of application requirements may contain specifications which are not directly expressable as parameters or which do not appear on the parameter list of all. In such cases, the evaluator must interpret the requirements in terms of the parameter list and should add parameters as required. If the evaluator feels that, for some compelling reason, he cannot or should not translate an application requirement into a parameter requirement, he should make sure that such requirements are considered in the overall evaluation of final scores (Section 2.5).

2.1.5.1 Minimum and Desired Requirements. Many parameters involve a range of capability, such as maximum record size; capability in excess of requirements, however, can account for capabilities above a minimum need. A requirement for a capability can be a single value, or these can be a minimum value and a desired value.

A GDMS must meet the minimum value to be acceptable, and it is given a higher rating on the rating scale for capability up to the desired level (maximum rating of 10). Whether or not capability above the desired level is reflected in the rating is a judgement the evaluator must make for each parameter.

The minimum and desired requirement levels can be the same, of course, and a GDMS either does or does not have the required capability;

only ratings of 0 or 10 are allowable in this case. A requirement, however, often will not be numeric, but will express a directional preference (e.g., faster is better).

2.1.5.2 Mandatory Requirements. The definition of a mandatory requirement is something which a GDMS must satisfy in order to be used at all. The requirement can be a matter of degree (such as query response time) or a yes/no function (such as the requirement for a specific terminal device). A mandatory requirement may or may not be accounted for in the parameter list. In any event, mandatory requirements must be fulfilled for a GDMS to be acceptable.

The question of whether parameters which are mandatory requirements should be rated is difficult to resolve. On one hand, it can be argued that a GDMS should be rated only on those features which are not mandatory, since systems that qualify for rating must possess the required capabilities. On the other hand, several arguments can be made for rating all required capabilities, whether mandatory or not. First, a measure of effectiveness should be based on all capabilities of a system that are useful, and not just on those that are extra for a problem-mix. Second, a mandatory requirement may involve a parameter whose quality among systems varies.

2.1.6 Develop Bench Mark Problems

Bench mark problems should be developed when it is not practical to implement, for evaluation purposes, the actual problem-mix used in the evaluation. This is the case when the problem-mix is:

- Already implemented on a computer, but it is too costly or too time consuming to run on a GDMS for evaluation purposes.

- Not implemented on a computer and requires extensive system work prior to implementation.
- Known on a general level but lacking in detail (e.g., it is anticipated that certain files will be queried, but the exact nature of the queries is unknown).

2.1.7 Select Parameters

Two closely related tasks in the evaluation are the selection of parameters to be rated from the list, and the addition of "other" parameters to the list. Refinement of the parameter list and the parameter selection technique will occur as experience is gained by performing evaluations for actual application.

The evaluator should not feel obligated to rate each parameter; only those parameters that are reflected in requirements derived from a problem-mix and its environment need be rated. Requirements can be actual or assumed, and can be rather general. For example, a certain capability can be deemed useful even though an exact requirement for it cannot be pinpointed. This is done in order to avoid rating capabilities in a system simply because they exist, and to avoid rating a parameter according to a fixed notion of what is useful.

Although the list of parameters does include, except for the exclusions discussed elsewhere, the major areas that should be evaluated, it does not cover every detailed consideration that could be important in an evaluation. For this reason, the evaluator may find it necessary to add parameters in order to account for all problem-mix requirements. The list is "open-ended" and new parameters can be defined, rated, and weighted in the same pattern used for listed parameters.

The selection of parameters to be weighted and rated is a by-product of the requirements determination steps. The identification of parameters to be weighted and rated is accomplished by elimination. Parameters without any requirements have no bearing on the evaluation are assigned weights of zero, and excluded from further analysis; the remaining parameters are carried through the evaluation procedure.

2.2 WEIGHTING

2.2.1 General

The weighting and aggregating technique in this section describes the mechanics of weighting, but it does not address the problem of methodology for determining weights for specific parameters. The actual selection of a weight is an arbitrary judgement of relative importance made by an evaluator (a person or group of persons).

The use of experts, consultation with users, etc., in order to determine a weight may be of help in arriving at a "fair" weight, but the fact remains that weighting is a purely subjective matter. The main point of the weighting step is to allow the introduction of the evaluator's opinion (or user's) as to which parameters are more important than others for a problem-mix. For this reason, there is no preassignment of weights in this study; weighting must be done for each evaluation performed. Where figures are listed, they are examples shown for illustrative purposes only.

The initial allocation of weights among parameters should be done prior to the analysis and rating of a GDMS in order to promote objectivity. It is felt that the evaluator is less apt to be biased in his initial choice of weights if he does not have knowledge of a system's capabilities. The weights should indicate relative importance of parameters in terms of application requirements, and they should not be used to express personal preferences or to favor one GDMS over another. Although the weighting process is subjective, the evaluator should make every effort to be objective by excluding personal biases and preferences whenever possible. For example, an evaluator may personally feel that ease of use of a GDMS language is unimportant and that power of the language (regardless of complexity) is important. Such an evaluator should not reflect this opinion when he is weighting parameters in a situation where ease of use is known to be an important requirement.

Another reason for weighting prior to rating is to determine which parameters have zero weights. Such parameters, of course, do not require further analysis and rating, and their identification early in an evaluation helps to eliminate unnecessary analysis.

2.2.2 Definition of a Weight

The degree of importance which is attached to a parameter is referred to as a weight. A basic assumption of the evaluation method is that weights are expressed numerically and will be used to adjust the parameter rating according to relative importance judgements. The nature of and the ranges of this numerical measure are explored in this section.

The weights to be applied are subjective estimates of the importance of the parameter, subparameter, or group of parameters under consideration. There are other interpretations, definitions, and usages of the term "weight" than to signify relative importance, however. For example, for certain mathematical and statistical problems it is appropriate to use weights as a measure of the confidence or reliability of an estimate or of a sample variable. If weights were used as a measure of confidence, the evaluator would be instructed to assign a relatively low weight to a parameter for which he has little confidence in the basis of judgement or for parameters for which the judgement is strictly a matter of (perhaps contentious or divided) opinion. However, weighting confidence levels as a procedural step in the evaluation method is not recommended.

The technique as developed in the current study is not sufficiently precise to permit the addition of yet another subjective measurement. The simplifying assumption is made that if confidence level is a sufficiently compelling consideration, it will be treated subjectively by the evaluator as a part of the weighting process and not as a separate procedural step in the evaluation.

2.2.3 Weighting Method

For purposes of illustration, an example hierarchy of parameters was used together with ratings which it is assumed were derived by using the standard scale for individual parameters and subparameters.

A method using percentage weights was developed for use in the evaluation technique. This method may be undertaken either in a bottom-up or top-down order. The basic method may be stated in a general way as a single step which is:

> Determine the relative importance of sibling parameters and apportion weight on a percentage basis. (The total weight for each set of sibling parameters is unity.)

It is seen that when the ratings and weights are extended and totaled the resulting score for each hierarchical level retains the value significance of the standard scale, e.g., a score of 9.3 is "good", one of 2.7 is "poor", etc.

Each parameter which contains or is composed of subparameters is not rated. The score for such a parameter is obtained by computing a summation of the weighted scores of the subparameters. In other words, the whole is considered to be the sum of the parts. This implies that no parameter will have only one subparameter. (Such a subparameter would be a redundant restatement of the parent parameter). A parameter may have two or more subparameters or none at all.

The process is illustrated in columns 4 – 7 of Table 1. The relative importance of Parameters II.C.2.a, II.C.2.b, and II.C.2.c are determined to be 0.20, 0.50, and 0.30 respectively, as shown in column 4. Succeedingly higher hierarchical levels are determined in columns 5, 6, and 7. For example, in column 6 the relative importance of Parameter

I. A versus parameter I. B is shown to be 0. 70 - 0. 30. It is noted that the selecting of relative weights for Parameters I, II, and II (column 7) involves determining the relative importance of parameter groups, not individual parameters. In the functional organization of parameters which has been developed in this study, examples of such parameter groups are "Data Definition", "File Creation and Maintenance", "Retrieval", etc.

It is noted again, that either a bottom-up or top-down order may be used; in fact, any order (inside-out) will be workable since any set of sibling parameters may be considered independently.

The percentage weights shown in columns 4 – 7 have no absolute significance, e.g., the weights given Parameter III. A. 4 (Line 35) and III. D. 1 (Line 40), both given weights of 0. 20 in their respective groups are not equally important. Therefore, in order to obtain relative weights for all parameters, a system weight may be computed for each parameter (column 8).

This is done by simply multiplying the weight at the lowest hierarchical level times the weights at each succeeding higher level. For example, Subparameter I. A. 1 (Line 3) is shown to constitute 21% of the system capability since it constitutes 60% of a parameter which is weighted at 70% of a parameter group which is considered to comprise 50% of the system capability (0. 60 x 0. 70 x 0. 50 = 0. 21).

The method of computing a system score by the evaluator analyst is shown in Columns 11 – 22. The scores for the lowest hierarchical level (in this case II. C. 2. a, II. C. 2. b, and II. C. 2. c) are evaluated in Columns 11 – 13 in order to obtain a rating for use at the next level. In the example shown, a score of 7. 4 is computed (column 13). This score, although weighted, retains a significance on the standard "goodness" scale. To paraphrase this significance, the evaluator analyst might conclude "when both the quality

Table 1. Weighting Method and Computation of System Score (Sheet 1 of 2)

Line (1)	Parameter Hierarchy (2)	Rating (3)	WEIGHTS II.C.2.a, etc. (4)	I.A.1, etc. (5)	I.A, I.B etc. (6)	I, II, III (7)	System Weights (8)	Computation Using System Weights (9)
1	I					.50	.50	
2	A				.70		.35	
3	1	7		.60			.21	1.47
4	2	9		.40			.14	1.26
5				1.00				
6	B				.30		.15	
7	1	6		.50			.075	.45
8	2	4		.10			.015	.06
9	3	2		.10			.015	.03
10	4	9		.30			.045	.405
11				1.00				
12					1.00			
13	II					.30	.30	
14	A	9			.20		.06	.54
15	B				.30		.09	
16	1	9		.90			.081	.729
17	2	5		.10			.009	.045
18				1.00				
19	C				.50		.15	
20	1	6		.20			.03	.18
21	2			.40			.06	
22	a	4	.20				.012	.048
23	b	9	.50				.03	.27
24	c	7	.30				.018	.126
25			1.00					
26	3	8		.30			.045	.36
27	4	7		.10			.015	.105
28				1.00				
29					1.00			
30	III					.20	.20	
31	A				.50		.10	
32	1	10		.60			.06	.6
33	2	0		.10			.01	–
34	3	10		.10			.01	.1
35	4	10		.20			.02	.2
36				1.00				
37	B	9			.30		.06	.54
38	C	4			.10		.02	.08
39	D				.10		.02	
40	1	9		.20			.004	.036
41	2	5		.80			.016	.08
42				1.00				
43					1.00			
44						1.00		7.714

Table 1. Weighting Method and Computation of System Score (Sheet 2 of 2)

Line (10)	Method of Computation Used by GDMS Evaluator											
	Eval. of II.C.2			Eval. of I.A, etc.			Eval. of I, II, III			System Score		
	Rating (11)	Weight (12)	Score (13)	Rating (14)	Weight (15)	Score (16)	Rating (17)	Weight (18)	Score (19)	Rating (20)	Weight (21)	Score (22)
1										7.35	.50	3.675
2							7.8	.70	5.46			
3				7	.60	4.2						
4				9	.40	3.6						
5				Total I.A		7.8						
6							6.3	.30	1.89			
7				6	.50	3.0						
8				4	.10	.4						
9				2	.10	.2						
10				9	.30	2.7						
11				Total I.B		6.3						
12							Total I		7.35			
13										8.01	.30	2.403
14							9.0	.20	1.80			
15							8.6	.30	2.58			
16				9	.90	8.1						
17				5	.10	.5						
18				Total II.B		8.6						
19							7.26	.50	3.63			
20				6	.20	1.2						
21				7.4	.40	2.96						
22	4	.20	.8									
23	9	.50	4.5									
24	7	.30	2.1									
25	Total II.C.2		7.4									
26				8	.30	2.40						
27				7	.10	.70						
28				Total II.C		7.26						
29							Total II		8.01			
30										8.18	.20	1.636
31							9.0	.50	4.50			
32				10	.60	6.0						
33				0	.10							
34				10	.10	1.0						
35				10	.20	2.0						
36				Total III.A		9.0						
37							9.0	.30	2.70			
38							4.0	.10	.40			
39							5.8	.10	.58			
40				9	.20	1.8						
41				5	.80	4.0						
42				Total III.D		5.8						
43							Total III		8.18			
44										System Total		7.714

and the relative importance of these factors (the subparameters of II. C. 2) are taken into account, a score of 7.4 is obtained." According to the narrow definitions of rating and score used here, a score (computed from a rating and a weight) is used as a "rating" in order to compute a score for the next higher hierarchical level.

The 7.4 score is then carried forward in the rating column of the next hierarchical level (II.C). A weighted score for II. C is computed in columns 14 – 16, Line 20 – 27 (7.26). This score becomes the rating to be used to compute the score for Parameter Group II. The score for II. A, III. B, and III. C appear for the first time in the third set of columns (17–19) because they have no subheadings on a lower hierarchical level to be evaluated. The total for II is 8.01 and this value is weighted together with the scores for I and III to obtain the system score, 7.714. It is noted that this value is the same as that derived by using the system weights in column 9. The question arises as to why the direct computation is not sufficient. Apart from the rather trivial advantage of checking arithmetic accuracy by independent computation by two methods, a more important advantage is seen. By aggregating the score from the bottom up at each step, a score of merit is obtained. This score may be inspected for reasonableness and may be adjusted by changing the ratings or weights if they are felt to be in error.

2.3 ANALYZE AND MEASURE GDMS CAPABILITIES

The analysis and measurement of specific GDMS capabilities is a major task, and is discussed in detail in Sections III and IV. A few general comments, however, are made here.

2.3.1 Analyze and Select GDMS's

As was discussed earlier, the GDMS's to be evaluated are either assigned to or selected by the evaluator, depending on the objectives of the evaluation. If the GDMS's are to be selected, this step serves as a preliminary screening of systems. Candidate GDMS's are analyzed and those that are judged suitable are selected for further evaluation. Sources of information for this analysis and subsequent rating include:

- Manuals and other documentation
- Interviews with developers of GDMS's
- Interviewers with users and operators of GDMS's
- Personal evaluation and operational experience

PEGS is useful for screening GDMS's in that it can be used as a check list for analysis.

2.3.2 Measure GDMS Capabilities

Once the selection of GDMS's has been completed, each GDMS is further analyzed in order to determine or measure the capabilities of each GDMS for the selected parameters. Observations are made; input data, data definition, and task specifications are prepared; and bench mark problems are executed (or estimated). Access to persons with detailed knowledge about the specific GDMS's being evaluated is essential.

The measurement methods to use include:

- Benchmark study
- Observation
- Consensus
- Estimate
- Supplementary analysis

2.3.3 Check Mandatory Requirements

The evaluation technique was not explicitly designed with a provision for checking mandatory requirements. It was assumed that the determination of minimum acceptability is performed prior to the evaluation process; in other words, GDMS's to be evaluated are acceptable, and the prupose of the evaluation is to measure the relative power and effectiveness of the system. The technique is a tool for selecting the best system (according to evaluation objectives and weighting), and it is not a method for screening suitable vs. unsuitable systems. This does not preclude the use of the technique, however, as an aid in checking for minimum requirements.

2.4 RATING

The problem of obtaining quantitative measurements for parameters was not solved easily. The measurement problem turned out to be two distinct steps; the first step is to determine (measure, observe, etc.) the capability of a system for a parameter. The second step is to measure the effectiveness of the system by comparing the capability with the requirement, and arriving at a rating.

The definitions of normalizing, rating, conversion (or translation) of measurements to ratings, etc., involve semantic problems that are difficult to resolve. At one time during the study, normalization was thought of as a necessary and separate step in the evaluation procedure. The measurements for the various capabilities consists of many different units and size ranges, and these measurements had to be "normalized" to a standard range or ratio (such as a percent) in order to be able to aggregate an overall index of effectiveness. As the procedure now stands, this is the rating and not the normalization step. The normalization of measurements is built into the rating procedure in the sense that all ratings of measurements are based on a standard rating scale.

The conversion from a capability measurement to a rating ideally should be based on a function which relates measurements to ratings. The function can be simple, complex, linear, discrete, etc., and it should be formulated prior to analysis of a GDMS, if possible. The functional relationships which are considered typical are graphically illustrated in Figure 1 . Other relationships also are shown in tabular form for various parameters in Section 3. 2. The vertical axis represents the rating scale, and horizontal axis the capability measurement. The functions shown are not absolute, but should be thought of in reference to a specific requirement.

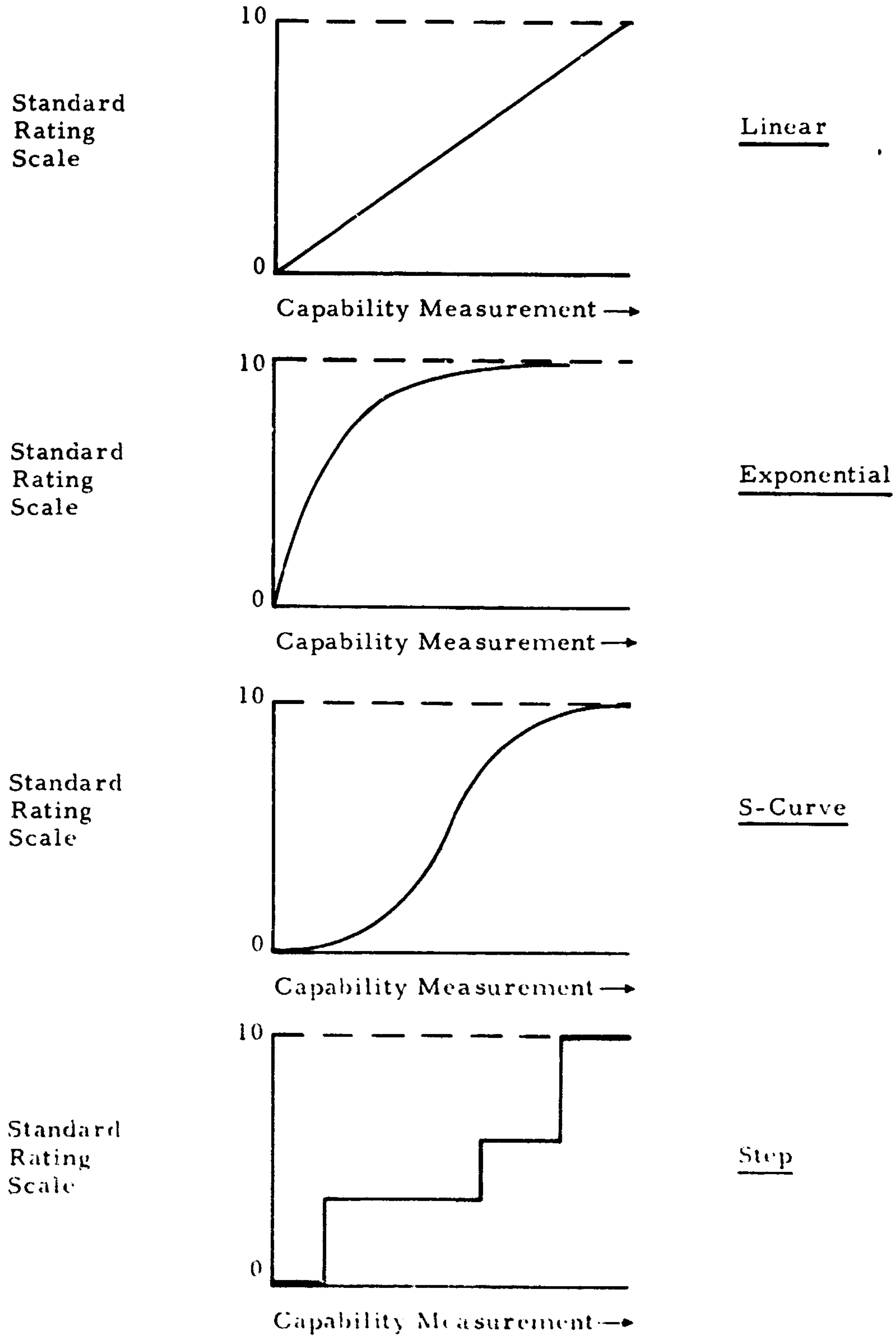

Figure 1. Examples of Functional Relationships

In many cases, however, it is realized that the evaluator does not have sufficient information to determine such functions, and he exercises judgment in directly estimating a rating for a capability. This is so since most parameters are not numerical in nature. They are either of the yes/no variety or of the qualitative judgment type, and they pose serious quantification problems. The parameters that can be directly stated in numeric terms also present estimating problems, especially for systems that are proposed or under development.

For parameters that are not amenable to quantification, it is undesirable to try to force a quantified measurement prior to rating. Instead, such parameters are directly quantified by arriving at a rating based on qualitative considerations.

2.4.1 Effective Range of Scales

A numerical scale for evaluation may take several forms and use various effective ranges such a 0 to 1, used to measure probability, -1 to +1, used for measurements of coefficients of correlation, or percentage. The scales may be continuous such as a percentage measurement or contain discrete values such as scoring 1 or 0 for a yes/no answer or scoring 4, 3, 2, 1 for Excellent, Good, Fair or Poor.

One of the most common usages of scales for scoring is the 0 to 100 scale which may be thought of as a percentage score. The prevalence of this grading system is almost certainly attributable to its almost universal use in educational institutions. The scale, as used in school systems, is curiously insensitive, however, since it is effective over a relatively small portion of the entire scale, i.e., approximately 60-100. A grade of 60 usually is a failing grade, a grade of 80 medium, (although generally accorded a euphemistic "good"), and 100 the theoretical limit of excellence. Thus, although the nominal scale is 0-100, the effective scale is 60-100. Development of a scale for grading, which utilized the 0-100 scale as an effective range, would result in the scale such as:

0 - 20	Failing
20-40	Poor
40-60	Average
60-80	Good
80-100	Excellent

This scale could readily be adopted in traditional letter grading (A-F) used in education. The 0-100 scale is also more amenable to the concept of comparison ratios. For example, a grade of 80 could be interpreted as approximately twice as good as a grade of 40. Currently, the advantage of a 90 score compared to a score of 70 can hardly be considered as adequately expressed as a ratio (e. g., 9:7).

Although the 0-100 scale (effective range) would appear to offer more opportunity for sensitive measurement, it would be difficult to persuade the educational community that making such a change would result in significant advantages. The question of the effective range is important. The traditional grading methods have probably inculcated a bias of sorts which would tend to introduce a dampening effect by the evaluation analyst.

Would a composite evaluation of 60 for a system have a connotation of "poor" to the decision maker or "better than average" (since it is higher than 50)? Would two closely matched "average" systems receive grades of 77 and 81? Or, should the same systems get evaluation indexes of 46 and 55; indicating slightly worse than, and slightly better than average, respectively?

The scale range could be considered as merely a subjective and arbitrary consideration were it not for a basic assumption which is explicit in the overall approach of this study. This assumption is that we will agglomerate quality factors (as expressed on a common scale of excellence), quantity factors (e.g., a capability may exist for one system and not another; thus one system may have more features to be evaluated additively), and importance factors (weights). It is imperative, therefore, that the standard scale be uniformly significant over the entire range since all ratings, whether they are in the high or low region of the scale, are weighted and aggregated.

2.4.2 Standard Scale

In order to provide a uniform basis for rating parameters, a scale consisting of the values 0 through 10 has been adopted for use. This scale is called the "standard scale" and is used for rating all parameters. Several sets of descriptive adjectives are shown as examples, and an appropriate set will be selected for each of the various types of ratings to be made. The rating can be based on capability, performance, or ease of use; in all cases, however, the rating is a measure of excellence of a system in relation to an application requirement.

Every effort should be made to make full use of the entire 0 to 10 range. This provides for a more sensitive and unbiased rating technique and does not result in clustering of values at 0 or the high (6 to 10) region of the scale.

In order to make full use of the scale, the following viewpoint may be of use. A rating of 0 is unacceptable, and a rating from 1 to 10 is acceptable with the range 1 to 10 measuring the degree of goodness or excellence. Hence, a rating of 1 indicates that a capability is acceptable and of poor quality and a rating of 10 indicates that a capability is acceptable and of excellent quality.

The sets of descriptors on the following page (Table 2) illustrates several different approaches for rating a parameter. Sets 1, 2, and 3 are all scales of excellence; set number 1 provides guidance only as to the high and low ratings and the evaluator would formulate his own descriptors or intermediate ratings. Set number 3 suggests a full range of adjectives. Set number 4 would be used for yes/no parameters without gradation of capability. Set number 5 pertains to ease of use; set number 6 refers to capability. The last set combines capability and requirements.

2.4.3 Yes/No Parameters

There are two kinds of parameters that are yes/no functions. In one type, there are only two distinct conditions represented by "yes" or "no", such as whether a system does or does not exactly meet a requirement. In the other type, the "yes" condition can take on a range of values, and it should be rated accordingly. An example of the latter type is a requirement for a general type of output device, which can be measured as "yes" or "no" for a system, but which also can be rated as to adequacy, quality, etc., if "yes".

The rating of parameters that are distinct yes/no functions poses an unusual problem. The rating guidelines (Section 2.4.7) state that "no" is rated as 0 and "yes" as 10. Yet, for parameters that can be rated anywhere on the 0 to 10 scale, a rating of 10 is excellent, (or the highest possible). The rating of 10 for "yes" implies that the capability is excellent, when in fact "yes" only means that the capability exists. This tends to promote an upward bias in the overall ratings; the upward bias may be insignificant because:

Table 2. Examples of Rating Descriptors

Std. Scale	Set No. 1	Set No. 2	Set No. 3	Set No. 4	Set No. 5	Set No. 6	Set No. 7	Std. Scale
10	Excellent	Excellent	Superior	Capability exists	Very easy to use	Very high capability	Capability meets all requirements	10
9			Excellent					9
8		Good	Good		Easy to use	High capability	Capability meets most require-ments	8
7			Fair					7
6		Satisfactory	Above average		Average ease of use	Medium capability	Capability meets some require-ments	6
5			Average					5
4		Adequate	Below average		Hard to use	Low capability	Capability meets few requirements	4
3			Inferior					3
2		Poor	Poor		Very hard to use	Very low capability		2
1	Poor		Marginal				Capability barely meets minimum requirements	1
0	No rating	Unacceptable	Unacceptable	Capability does not exist	Not usable	No capability	No capability for these require-ments	0

- There are few yes/no parameters rated 0 or 10
- Of weighting
- The ratings of 0 and 10 average out close to the average of other ratings.

There are several ways to eliminate this potential bias, such as using ratings other than 0 and 10 for yes/no parameters. For example, a rating of 7 for "yes" and 3 for "no" could be used if upward bias is a significant problem. Most yes/no parameters, however, do have gradations of capability and should be rated appropriately; for the few yes/no parameters that are of the presence/absence variety, the rule of 0 and 10 should be used.

The rating of yes/no parameters can be made more sensitive by considering the capabilities afforded relative to one or more of the following criteria of judgment rather than by considering the presence (or absence) of the capability as a yes/no function.

- Availability
- Quality
- Reliability
- Compatibility
- Ease of use
- Flexibility

2.4.4 Rating with Incomplete Information

The evaluator probably will be faced with the problem of trying to rate a parameter with incomplete information about a GDMS. This is bound to occur with a GDMS that is proposed or under development, and can happen with an operational system if documentation is not complete or information is not readily available. In such situations, the evaluator should make a rating for a GDMS only when he feels he has reasonable confidence in his rating; otherwise the parameter should not be rated for that GDMS. What reasonable confidence means is hard to define, but the evaluator should trv to think of a "threshold of significance". A rating should be made only when the evaluator feels that the significance of his rating is above the minimum threshold.

Since the degree of confidence in a rating will vary among ratings, the possibility of reflecting confidence in the scores was explored. The selection of ratings and weights based on confidence or an adjustment of a rating or score with a confidence factor were considered. The handling of two different degrees of confidence for two GDMS's for the same parameter further complicates the issue. As a result, confidence limits are not specified.

During the course of an evaluation, there may be insufficient information to rate a parameter for a GDMS that has already been rated for another GDMS. The first consideration is whether a mandatory requirement exists for this parameter; if one does exist, then it is necessary to obtain more information to complete the evaluation. If the requirement is not mandatory, the two major choices are to exclude the parameter from the evaluation or to rate one GDMS with the information on hand and rate the other GDMS as 0 for this parameter. Neither alternative seems to be obviously fair; in one case a rating which can be made for one system is not used, and in the other case, a system is penalized with a 0 rating for lack of information.

2.4.5 Composite Ratings

A composite rating may have to be developed for some parameters. This situation arises when a parameter is defined as two or more capabilities or considerations that are not rated individually. A judgment is made for the parameter by collectively analyzing the capabilities as a group.

A composite rating also is made when there are two or more different requirements for a parameter. This can result from a single complex application or a problem-mix with several applications. Both the rating and the weighting of such parameters is more difficult than for those with simple requirements.

Theoretically, each evaluation should be performed using a set of requirements that consists of one requirement (or at most two: minimum and desired) for each parameter rated, with an overall score developed from the weighted ratings. Two or more sets of requirements should be evaluated separately, and the overall scores for each set weighted in order to arrive at an aggregate score for all requirements. As a practical matter, however, such a procedure can become lengthy and tedious, and the composite approach may yield sufficiently accurate results.

2.4.6 Rating vs. Weighting

Throughout the parameter list, the evaluator is reminded not to intermix rating with weighting. The rating for a parameter is a measure of degree of excellence in fulfilling requirements. The weight for a parameter is an expression of the importance of that parameter relative to other parameters. This is analogous to the academic grading where class grades are weighted by semester hours for an average grade. The overall grade is not a simple average of class grades and the weighting reflects the relative importance of each class independent of the grade for that class.

2.4.7 Rating Guidelines

The following guidelines apply except where special instructions are listed in the evaluation instructions for a parameter:

1. No parameter can be rated below 0, that is, with a negative rating. The lowest rating is 0.
2. No parameter can be rated above 10 in an effort to give credit for excess capabilities. The highest or best rating is 10.
3. For parameters that describe capability which either exist or do not exist and are yes/no functions, the "yes" condition is rated as 10 and the "no" condition as 0. The yes/no parameters that possess a gradation of values should be rated using the entire 0 to 10 scale.
4. If a parameter is specified as a mandatory requirement, a rating of 0 would disqualify the system. Therefore, a rating of 1 or higher for such parameters is necessary for a system to be acceptable.
5. Whenever feasible, the evaluator should formulate a function which relates the measure obtained for a parameter with the standard rating scale.
6. All parameters, especially those rated on non-numeric considerations, require careful comparisons among systems being evaluated in order to insure that the relative ratings are as accurate as judgment allows. For example, a rating of 8 for System A and 6 for System B for a parameter should indicate that System A is superior to System B for that parameter.
7. For parameters rated on a comparison of benchmark results, the best result is given the highest (not necessarily the maximum, however) rating, and the other results are rated lower relative to the highest rating.
8. A rating must be a whole number or integer; fractions or decimals should not be used.
9. The ratings for each parameter must be determined by the evaluator for every evaluation. Ratings in examples in other sections of this report are shown for illustrative purposes only.

2.5 COMPUTE AND REVIEW SCORES

2.5.1 Compute and Accumulate Scores

The computation and accumulation of scores by extending ratings and weights is a clerical process described in Section 2.2 on the description of weighting. Since there are many factors, such as number of parameters rated, number of GDMS's evaluated, personal preferences of the evaluator, etc., the design of a summary worksheet can take on many formats. Therefore, the layout of the summary worksheet is left up to the evaluator. Any columnar format based on Table 1 (used to illustrate weighting and aggregating) should suffice.

2.5.2 Review and Adjust Entries

The selection of ratings and weights is an iterative process. Although the initial choice of ratings and weights may be satisfactory, a comparison of scores may lead to adjustments of initial entries. The purpose of this is not to bias the results towards some preconceived answer. Iteration allows the evaluator to reconsider all of his entries after a detailed analysis of two or more systems for the following purposes.

1) To adjust for overlap in the parameter list
2) To adjust for overlap caused by the functional organization of a GDMS
3) To take an overall view of the entries
4) To adjust for parameters that were added as a result of the analysis

2.5.3 Compute Final Scores for a Problem-Mix

The final score for a problem-mix is computed by weighting the scores for each application and accumulating using the same procedure for computing parameter scores (Section 2.2). This permits the more important applications to be assigned higher weights and thus have a greater influence in the final score for a problem-mix. This step would be skipped, of course, when a single application constitutes a problem-mix.

2.5.4 Overall Evaluation of Final Scores

In some cases the prior step will be the last one; in other cases the final scores computed with the evaluation technique will be compared with other factors such as total cost. This is done when the evaluator has elected to treat certain parameters outside of the evaluation technique. For example, the technique may have been used to determine which systems, if any, are acceptable, and the final selection of a system is based on total costs. The technique should be used as a tool, and not necessarily the final answer, in an evaluation or selection procedure. Hitch and McKean have written the following about economic analysis and it is applicable here:

> "Where mathematical models and computations are useful, they are in no sense alternatives or rivals to good judgment; they supplement and complement it. Judgment is always of critical importance in designing the analysis, choosing the alternatives to be compared, and selecting the criterion. Except where there is a completely satisfactory one-dimensional measurable objective (a rare circumstance), judgment must supplement the quantitative analysis before a choice can be recommended."*

*Hitch, Charles J., and McKean, Roland N., The Economics of Defense in the Nuclear Age, R-346, The RAND Corporation, March 1960, p. 120.

Section III

PARAMETER ORGANIZATION AND MEASUREMENT

The organization of parameters supplies an implied definition of GDMS systems. It is seen as an evolving structure which should be selectively utilized and further refined at the time of each evaluation. The organization, as conceived for this report, is therefore to be regarded as a framework to which factors of importance (parameters), which pertain to particular applications requirements not anticipated here, may be added, and from which parameters of doubtful importance or which are redundant or inappropriate to the particular evaluation may be removed.

3.1 PARAMETER ORGANIZATION

A few comments concerning the basis for organization are in order here. One of the difficulties of a parameter organization which purports to measure so complex an entity is a GDMS is that there are many methods of categorization and many viewpoints and criteria which may be used. Some of the approaches are:

- System analysis approach
- Language analysis
- Performance measurement
- Procedural analysis
- User-oriented approach
- Functional analysis
- Operations analysis

It is virtually impossible to define a list of parameters that accommodates pertinent judgemental criteria, reflects both GDMS capabilities and system task requirements, does not have overlapping characteristics, and is reasonably straightforward to interpret. The reasons for this are several. The designs of GDMS's vary greatly; requirements continue to grow more complex. A complex item such as a GDMS is not amenable to simple, straightforward evaluation. A system such as a GDMS must be regarded as an entity whose totality transcends the sum of its parts. The field of generalized systems is new and is undergoing rapid technological development. Capabilities that are lacking or are absent in present systems undoubtedly will be commonplace in future systems.

However, an even more basic difficulty is that of semantics. The language used for naming and describing parameters may be framed in structural, procedural, functional, and operational terms. No single one of these categorization viewpoints was found to be adequate for expressing all the variables and determinants of system worth. Even more basic to the problem of parameter definition are the assumptions to be made concerning the criteria to be used for determining system value. A number of subtle questions should be resolved regarding criteria for evaluation, for example:

- Is quality to be regarded as a virtue even when it does not result in utility?
- Is "more" better?
- Is convenience a valid criterion if it does not result in decreased costs (e.g., fewer man hours)?
- Is "value to the user" the all important basis for evaluation or can some "points" be given for advancing the state-of-the-art independent of user values?
- Is "intrinsic excellence" a ratable value, apart from whatever worth can be shown to result from this quality?

These and other philosophical points relating to theories of value are largely problem dependent. Therefore, as was discussed in Section 2.1, one of the first steps in the evaluation procedure is to determine the objectives and criteria of the evaluation. Since these criteria may vary, the parameter organization should anticipate and accommodate as many such criteria as possible. Therefore, various viewpoints of evaluation are represented, even at the risk of parameter overlap.

The basic organization of parameters selected for this report is based on the functional division of tasks for which a GDMS is primarily intended. This parameter organization is shown in Figure 2. For some systems these divisions are not sharply delineated, or may vary in important respects. However, the selected parameter groups should serve as a convenient basis for further modification even in these cases.

The functional categorization of GDMS operation and performance was selected because it is the most nearly oriented towards user values of the classification schemes investigated. These values are those which we are most interested in evaluating.

The functional groupings are those shown in the first five headings of Figure 2:

I. Data Definition and Data Organization

II. File Creation and Maintenance

III. Retrieval

IV. Processing

V. Output

In addition to the parameters considered in these groups, there are a number of environmental and support considerations which, Functional Parameters

Parameter group	I Data Definition and Data Organization	II File Creation and Maintenance	III Retrieval	IV Processing	V Output		VI Environmental Considerations
Capability Parameters	I.A Field Definition	II.A File Creation	III.A Selection	IV.A Computation	V.A Formats	Hardware	VI.A Computer System Characteristics
Capability Parameters	I.B Record/ Segment Definition	II.B FCM Operators	III.B Data Extraction	IV.B Summarization	V.B Formats-User Specified	Software	VI.B Operating System
Capability Parameters	I.C File Definition	II.C File Maintenance	III.C On-Line Capabilities	IV.C Sorting	V.C Editing	Software	VI.C Available Programming Languages
Capability Parameters	I.D Input Media	II.D Input to FCM	III.D Input	IV.D Data Conversion	V.D Page Numbering and Control	Software	VI.E Interfaces with Other Systems
Capability Parameters	I.E - Storage & Modification of Data Definitions	II.E - Storage & Modification of FCM Task Specifications	III.E Storage of Queries	IV.E Statistical	V.E Output Media	Software	VI.E System Support
Capability Parameters	I.F - Capability to Read Files from Other Systems	II.F Conditional Maintenance	III.F File Security	IV.F Own Code	V.F-Capability to Prepare Input for another System		VI.F Installation Planning
Ease of Use Parameters	I.G Ease of Use	II.G Ease of Use	III.G Ease of Use	IV.G Ease of Use	V.G Ease of Use		VI.G Personnel
Performance Parameters	I.H Performance	II.H Performance	III.H Performance	IV.H Performance	V.H Performance		VI.H General System Characteristics

Figure 2. Parameter Organization

though important, are not easily identifiable under the functional headings. We have grouped these under the heading:

VI. Environmental Considerations

The matrix characteristic of this organization is noted in the foregoing discussion. Grouping the elements of the functional headings, horizontally, these parameters may further be categorized as:

- "Capability" Parameters
- "Ease of Use" Parameters
- "Performance" Parameters

Investigation of the important parameters under each of the parameter groupings reveals that many of the same factors are important for each functional group. For example, ease of use and performance criteria are important considerations for each of the functional groupings. One of the plausible alternate parameter organizations is thus easily apparent if a horizontal rather than vertical division of the parameter matrix shown in Figure 2 is assumed. (For such an organization subparameters for Performance might be: Performance for Data Description; Performance for File Creation and Maintenance, etc.)

The matrix characteristics of this organization are noted here in order to suggest that in certain of the PEGS evaluation steps (notably, weighting) it may be appropriate to perform cross checks of certain groups of parameters. For example, the systems weights for all "Ease of Use" parameters (a horizontal grouping in the matrix) may be totaled and reviewed and a reasonableness check made.

3.1.1 Capability Parameters

The capability parameters are shown in the first six subsections of each of the first five parameter groupings. These parameters describe functional characteristics of the GDMS. Many of the parameters are capabilities which have been referred to as yes/no, or binary functions. The basic criteria for judgement of parameters of this type are:

- Does the capability exist for the candidate GDMS?
- Does the user want or need this capability?
- Does the capability permit the user additional flexibility or conversely does it impose an additional burden upon the user?

Other capability parameters may be measured quantitatively. For these parameters, a normalization step is required to translate the raw score to a standard scale rating.

3.1.2 Ease of Use Parameters

A basic tenet of the evaluation technique is that the selection of parameters reflect a user orientation. Accordingly one of the primary criteria of judgement is ease of use. This parameter is included under each of the GDMS functional divisions in order that the evaluator be encouraged to consider the system capability in these terms explicitly. It is recognized that a degree of overlap with capability and performance is unavoidable in this appraoch since ease of use may stem from various capabilities afforded and superior performance may result (particularly in terms of minimizing man hours) from the satisfaction of the ease of use criterion. Resolution of this overlap is accomplished by the following evaluator processes:

- A careful examination of the listed parameters with a continuing attention to the point of view implied by the parameter type (i.e., from the point of view of capability, ease of use, or performance).
- Sensitive adjustments of the closely associated parameters of these major types in the weighting process.

The ratings for capability parameters may in some cases vary inversely with the ratings of ease of use parameters. It may be difficult for a system to have a broad range of capabilities and at the same time emphasize the "ease of use" criteria. The ease of use capabilities have been classified in three categories.

- Language Considerations
- Ease of Learing
- Skill Level Required

The degree of sophistication of a dialog or conversational method may be a function of both advanced equipment and programming methods. It is therefore possible to evaluate both the means and the ends. Evaluation of the latter is the goal; evaluation of the former may provide insight to an evaluation of the latter, however.

Apart from power and capability measured by capability parameters, language features may also be measured on a scale of convenience and/or ease of use. A residue of language considerations remain for discussion, even if most capability factors are categorized elsewhere in the parameter list. A number of these items which the evaluator may wish to consider are listed to follow:

- Capability for user-defined additions to the language
- String set substitution capability
- Capability to refer to conditional statements by label (name)
- Free-form language characteristics
- Ability of the system to detect and correct minor breaks in the users syntax.
- Capability to add own code.

It is the large differences in system philosophy reflected in the language design which must be evaluated as a part of the "Language Considerations" parameter. However, language factors which can be directly associated with system capabilities, identifiable elsewhere in the parameter list, should be analyzed from that standpoint and effectively removed from consideration in this parameter.

3.1.2.1 Language Consideration. In considering the specification for a particular GDMS, there is a tendency to evaluate the entire system in terms of its language features. The language features are usually documented more thoroughly than the other aspects of the system design (particularly the environmental and operational ones). It is thus possible to develop a list of parameters which appears relatively thorough, but composed almost entirely of language considerations.

The approach implicit in the parameter organization is that the GDMS language is not analyzed as one integrated topic but rather that the effect of the language is reflected in the constituent capabilities which may be identified. Thus, the repertoire of the language is not considered as a major parameter topic, but elements of this parameter appear in several parts of the parameter list as appropriate (e.g., III.A.1, Repertoire of Comparators).

However, apart from the individual operators and other features of the language as reflected by individual capabilities, there are basic language methods to be evaluated. These methods may vary substantially in different systems.

The language may be essentially free form in terms of the conventions for composing acceptable expressions. Or the programming method may involve a highly structured coding sheet wherein columnar arrangement of input specifications is highly significant. The language may be interpreted by a language processor (compiled) for subsequent execution; it may be executed interpretively and/or on-line to provide immediate response characteristics. There may be more than one programming method permitted. For example, there may be a rather conventional set of programming languages for use by system programmers for input of data files or other system tasks, and a user-oriented on-line query language for use by those users who wish to interrogate the data files.

3.1.2.2 Ease of Learning. The ease of use of a language is often thought of in terms of how easy it is to learn. This aspect of this parameter is not as important as the economy of effort after the learning phase is complete. The distinction is easily seen if one considers that a highly sophisticated, possibly complex, language may provide the easiest method of problem definition for computer solution and yet may be difficult to learn. A rudimentary language may be the simplest to learn while awkward and cumbersome to use effectively.

An indication of the relationship of these parameters is shown below:

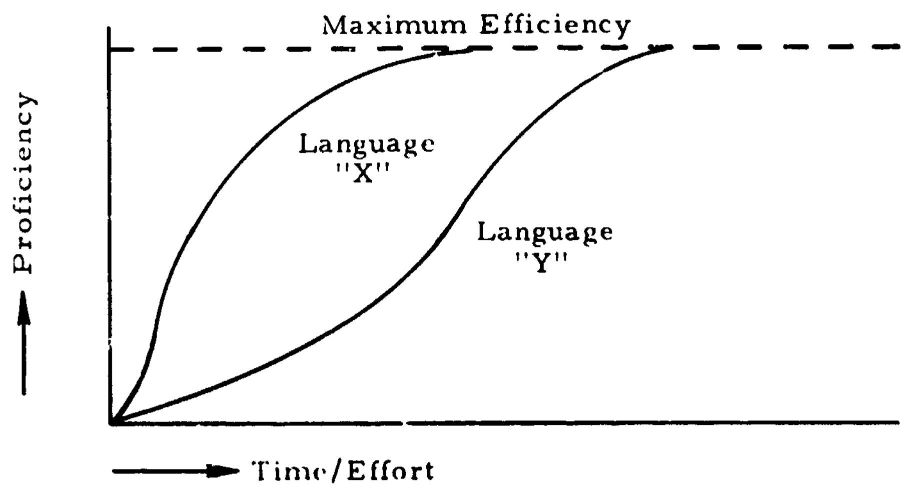

Figure 3. Ease of Learning: Programmer "A"

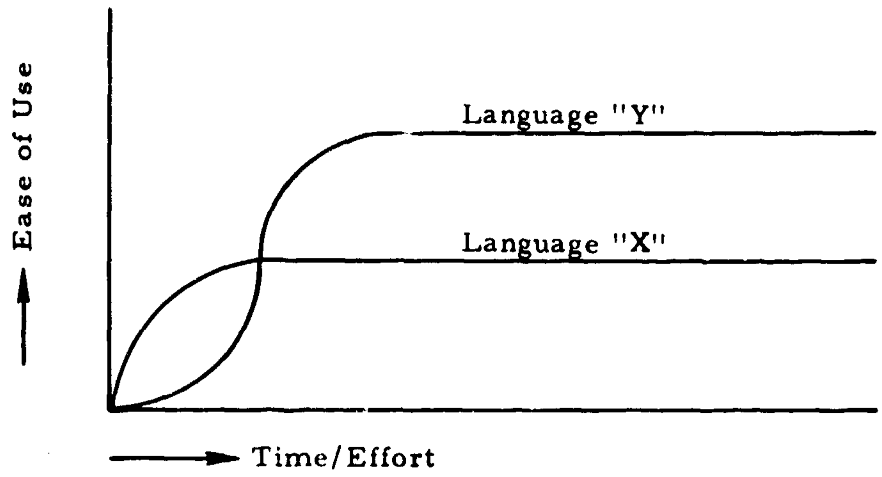

Figure 4. Ease of Use: Programmer "A"

It is thus seen that Ease of Learning is a parameter which is significant only during the learning phase, while Ease of Use is a parameter of more enduring significance.

For some systems the method of describing the problem for solution does not involve programming or coding in the traditional sense. A highly structured set of forms may provide, by arrangement of row and column, a matrix for insertion of the job description parameters. Other methods of prescribing the problem details are coding by questionaire and by a dialogue using keyboard and/or display equipment in a time shared environment.

Ease of learning may be measured by the amount of time and effort required for the user (programmer, coder, analyst, operator, etc.) to attain a required or desired degree of proficiency. Other considerations in the evaluation of this parameter are the amount of time available for learning and the amount of training effort required by other individuals to train the neophyte.

3.1.2.3 Skill Level Required. An important variable in determining ease of use is the skill level, training, and experience required of the practitioner. This item may be narrowly defined by the application requirement or a range of skill/experience levels that may be defined for the personnel who are expected to use the language(s). It is noted that the requisite skill levels may be different for the various functional divisions (e.g., data definition, retrieval, etc.).

Skill level is closely associated with the other ease of use parameters, in particular ease of learning. This relationship is indicated in Figure 5 for programmers of varying experience levels.

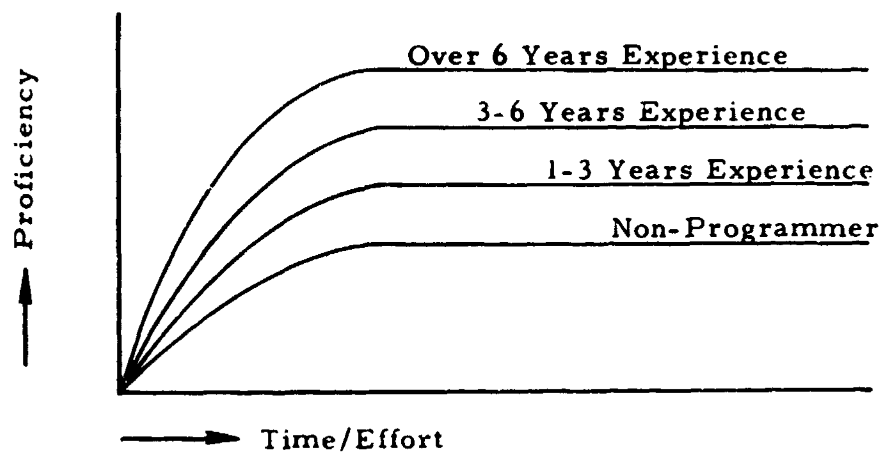

Figure 5. Curve of Learning and Proficiency Level for Selected Experience Levels: Language "X"

Typically, skill level required is inversely related to ease of use: the higher the skill level requirement, the lower the ease of use factor. Although skill level should be rated in this manner, care should be taken not to oversimplify this relationship. For example, if data definition is simple enough for clerical personnel, yet there is not intention that anyone other than a systems analyst will undertake this task, there is little benefit to be derived from such simplicity unless the task of the systems analyst is thereby made easier.

3.1.3 Performance Parameters

The performance parameters are intended to measure problem execution efficiency. These parameters are subdivided into three categories of measurement.

- Man hours
- Elapsed time for problem execution
- Machine time

These parameters have the following elements in common:

- They are measured in units of time.
- The preferred method of measurement involves recording the actual times for sample jobs or job mixes.

Thus, this particular categorization is determined partly by the practical consideration, i.e., the classification of parameters which are measured in the same general way into the same grouping.

The performance parameters introduce the element of cost into the evaluation. Although not primarily a cost or cost/effectiveness technique, the method outlined in this report includes measurements of cost such as man hours, machine time, etc. The justification for including cost elements is that capability as a function of cost is a better measure of system "value" than capability alone.

Measurement of this parameter should utilize performance data which are already available, may be computed by the use of standard estimates, or may be obtained by executing sample jobs. The latter method would ordinarily derive the most accurate results since it would measure an actual situation. However, in some cases standard estimates yield better comparative information. This may be the case for evaluations for which actual performance statistics may be obtained for only one system. Although an estimate would ordinarily be expected to be less accurate in all cases than measurement of actual performance, the crucial element of using the same ground rules, conventions, and assumptions is lost in such a case.

An illustration of this is seen in the EDP Reports in which are published comparative statistics for performance of a standard file maintenance problem. These figures are computed on the basis of carefully defined problems using standard estimates. In some cases, actual routines to do the identical task have been prepared which have been measured to have slightly different performance results. However, even with the knowledge of the "better" information, the standard estimates are still retained and represent a better comparative index of hardware capability than would comparison of actual routines which might involve other variables (e.g., programming methods, operating system differences, etc.).

There are several methods of ascertaining problem execution efficiency, listed below in the order of decreasing reliability:

- Measurement of the subparameters on the basis of the actual preparation of a job mix which is typical of the expected problems.
- Measure of a single job which is felt to be representative.
- Computation of a performance index on the basis of standard estimates of the subprocesses required for problem execution.

- An estimate of these subparameters based on knowledge of the known characteristics of the system.

If benchmark studies are undertaken to measure the performance parameter, a problem mix should be chosen which is as representative of the applications requirements as possible. The jobs measured should be performed, if possible, by persons who normally would do the job or by persons with equivalent training and experience.

A system under development may be unavailable for the execution of an application or benchmark problem. Under these circumstances, processing times must be estimated using whatever information is available. In the absence of any estimating guidelines, gross estimates of relative performance of two or more systems can be based on the characteristics of the computing hardware used in the systems. In some cases, part of the resulting efficiency of a job run may be attributable to the effectiveness of the operating system. Since the operating system is rated separately, its effect should be removed from this parameter, if possible.

3.2 PARAMETER MEASUREMENT

A general description of the evaluation and rating method as developed in this study is contained in Section 2.4. Additional specific information relating to parameter measurement and parameter work sheets follows:

3.2.1 Conversion to the Standard Scale

As an indication of the subjective process involved in selection of the appropriate ratings, it is appropriate to examine in detail the thought processes which will go into the determination of these ratings. For purposes of illustration, we examine three representative parameters, and, in effect, admit the reader to the flow on consciousness which might typify this complex evaluator judgement.

For this purpose, we will consider the parameter, Ease of Learning, which is a subheading in several of the functional divisions of parameters described in Section IV, and two on-line characteristics parameters for which normalization of raw scores is required.

One of the assumptions of our analysis methodology is that the evaluator will consider all known information regarding applications or problem-mix and system requirements. We cannot anticipate what these background considerations may be for the individual case. However, we can postulate several sets of conditions which we may then use for illustrative purposes.

3.2.1.1 Ease of Learning Rating Illustration

Situations

"A" Formulation of problem solutions (programming or coding) is expected to be accomplished by all levels (with respect to experience) of programming personnel; however, most programming will be done by junior personnel. The system is oriented toward batch processing with little if any on-line programming activity. A large number of personnel will be required to use this language.

"B" Applications will be "programmed" by programmers and non-programmers, with emphasis on problem presentation to the computer primarily via local, and possibly remote, terminals. The emphasis of this system i on man/machine integration for dynamic problem solution. A relatively small number of people will use this language.

"C" The particular applications requirements are not known, but it is expected that the system will be used for a wide assortment of applications. The language will probably be used by professional personnel, many of whom will not be computer personnel. It is also expected that many people (possibly thousands) will be exposed to, and eventually use this language at this installation for short periods.

Scale Gradations

The following levels of excellence are defined for purposes of fitting values on the standard scale.

I. The language includes a self teaching program module in the computer which may be used to teach the basic rules of the language. No other training requirement.

II.	The language may be understood and used by a professional programmer by study of a manual. Non-programmers can generally attain understanding after a four hour course.
III.	The language requires either a formal training program (one week) or a lengthy period of study and on-the-job-training. Documentation is complete.
IV.	The language has many features which must be learned. A training course of two weeks duration is suggested. Documentation fair.
V.	The language is difficult to learn and many of its features may be exercised appropriately only by professional programmers with considerable experience in the use of this language.

Having defined several levels of classification for the parameter "ease of learning", the evaluator must then determine a way of translating these gradations in terms of a standard score. His judgment will be affected by what he knows about the requirements, (e.g., situation A, B, or C or other may pertain). He will select from a number of appropriately shaped curves on the common scale. For example, for situation A, B, C any of the following rating scales might be deemed appropriate by the evaluator.

	"A"				"B"				"C"			
	a	b	c	d	a	b	c	d	a	b	c	d
I	10	9	—	—	—	—	10	9	10	10	10	10
II	9	9	10	9	10	9	9	10	9	7	8	8
III	8	7	8	7	8	7	8	7	7	6	5	6
IV	6	5	5	4	7	6	7	6	4	5	2	4
V	2	3	—	1	6	3	6	5	3	—	1	2
			↑					↑			↑	

Let us assume that the judgement of the evaluator indicates that the appropriate curve forms for the situations A, B, C are those numbered c, d, and c, respectively. A table of grades is obtained according to the requirements assumptions shown below:

	"A"	"B"	"C"
	c	d	c
I	—	9	10
II	10	10	8
III	8	7	5
IV	5	6	2
V	—	5	1

Only one of the sets of situations (A, B, C) applies to our example. Assuming the conditions described in C above, the grade for this parameter for two systems might result as follows:

	Observations		Rating	
	Syst. X	Syst. Y	Syst. X	Syst. Y
I	√		10	
II				
III				
IV		√		2
V				

3.2.1.2 On-line Characteristics Rating Illustration. It is important that significant discrete levels of capability be identified if such exist. A mandatory system requirement may constitute such a level; on the other end of the scale, a level of capability may exist beyond which no useful value may be assigned. For example, it is possible that if the desired level is provided, e.g., the required number of on-line user terminals, that additional capability (in this case, additional terminals) may represent an unneeded excess capability. Such excess capacity may have significance, however, in terms of system expansion capabilities. The minimum mandatory requirement level may have little if any rating significance since it is generally assumed that this evaluation technique will be used primarily for comparing systems which have already qualified in respect to such minimum standards. However, this level may be used as a base for the rating scale selected. In addition to the boundary upper and lower limit levels, several significant intermediate levels may be used.

The significant levels of capability (if discrete rather than continuous) must be defined in the application/requirements information gathering phase of the evaluation effort. The range of the numerical scales used are very dependent on this data and cannot be anticipated here. An indication of the normalization process is illustrated in Figures 6 and 7. These relate to several numerical measures of on-line capability. It should be emphasized that these figures are for illustration and should not be used as a basis for a particular evaluation. The actual standard scale ratings (left blank in Figures 6 and 7) would be filled in, of course, by the evaluator during an evaluation.

Levels of Capability	No. of On-Line Terminals	No. of Input Channels	No. of On-Line Users	Max. No. of Simultaneous On-Line Users	Standard Scale
					10
Expanded Capability (Needed for System Expansion)	100	128	200	60	
Desired Capability	60	64	80	40	
Capability Defined by System Requirements	40	32	60	20	
Mandatory Minimum Requirement	20	16	40	10	
					0

Figure 6. Illustration of Standard Scale Ratings: On-Line Traffic Volume

Response Time	Standard Scale
	10
Immediate Response	
0.5 Second Response	
2 Second Response	
5 Second Response	
Deferred Response* (Indication that query is being processed for later output)	
	0

*A deferred response might be entitled to a higher rating if it is anticipated procedurally as a part of system design.

Figure 7. Illustration of Standard Scale Ratings: Response Time for Typical Request, On-Line

3.2.2 Parameter Worksheet Description

The Parameter Worksheet, Figure 8, is intended to provide a flexible framework for the collection of observations obtained for both applications requirements information and GDMS specifications information. It is essentially free form since the parameter characteristics vary so widely that a single form designed to record all particulars is not feasible.

The upper portion of the form contains the parameter identification information and identification of the specific study. The body of the form contains the brief descriptive information of the parameter, subparameter, or attribute being measured. Columns are provided for recording observations, ratings, weights, and scores. The bottom of the form is used for recording of notes, comments, or other pertinent information. The elements of the parameter worksheet are described in more detail to follow and keyed to the circled letters in Figure 8.

A. Parameter Group

This refers to the division of parameters for the parameter to be analyzed. These parameter groupings were listed in Section 3.1. Example: II. File Creation and Maintenance.

B. Parameter

This identifies the major parameter to be analyzed. For some parameters, several worksheets will be required to contain the required information. Example:

II. C File Maintenance

C. Date and Evaluator

When the worksheet is used for a specific evaluation study, it should be identified. The date on which observations were made should be recorded. The name of the evaluation or analyst should be entered. These items are useful for filing, indexing and future reference.

D. GDMS

Each worksheet will record observations of one Generalized Data Management System. It is essential that the system be identified on every worksheet. Observations of a competing GDMS will be similarly identified and collected in a separate set. It may also be desirable to note here the application that is being studied. However, no specific space is provided for this because the entire study of competing systems for a particular application will normally be compiled into a report.

E. Data Source

This is an optional entry that may be used by the evaluator to record his authority or source of information. It may list a specification manual or other publication. It may name an expert who has provided data, estimates or opinions. It may contain words that denote direct observation, such as "TIME STUDY", "SAMPLE" or "DIRECT MEASUREMENT".

F. Parameter Description

This space is used to identify the parameter to be evaluated. Under each parameter, subdivisions may be listed which may be any of the following:

- Subparameters
- Attributes of the parameter
- A list of gradations of capability
- Other description information

G. Applications Requirements: Required and Desired

The first two columns are intended to be used by the analyst to record applications requirements information of a numerical nature. Two designated levels of capability may be defined; required and desired characteristics. In some cases, only one level may be appropriate to define. In some instances the two columns may be used for other purposes; for example:

- To contain two values which indicate an acceptable range.
- To contain checkmarks or X to indicate the presence or absence of a requirement or desired capability.
- For a Yes/No.
- To contain other information (e.g., dates, time intervals, etc.

PARAMETER WORKSHEET

Parameter Group: (A)

Parameter: (B)

Date: (C) Evaluator: (C)

GDMS: (D) Data Source: (E)

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
(F)	(G)	(G)	(H)	(I)	(J)	(K)

NOTES: (L)

Figure 8, Parameter Worksheet

H. System Observations: Observation

The third column is used to record observed data for the system being evaluated. The data entered here may be any of the types described for "G" above.

I. System Observations: Rating

The rating for the parameter or subparameter is entered in the fourth column. It is determined by a subjective process as described in Sections 2.4 and 3.2.1.

J. Computation of Score: Weight

The weight or importance value for the parameter or subparameter is entered in this column. It is determined by the techniques described in Section 2.2.

K. Computation of Score: Score

The score is generally computed by multiplying Rating x Weight. The rationale and the procedure for this computation is described in Sections 2.2 and 2.5.

L. Notes

A space is provided for the evaluator's notes. At the bottom of each page, he may enter additional observations, details, non-quantitive observations, qualifications, cross-references or other data.

Section IV

PARAMETER DESCRIPTIONS

The parameters developed in the study are listed and described in this section. The parameters are listed on parameter worksheets immediately following the sub-section describing each parameter group. For convenience, the term "parameter" is used as a synonym for "sub-parameter" throughout the text; this should cause no difficulty for the reader, since the distinction between the two terms is made clear when necessary. The hierarchy used in the text and on the worksheets is:

I. Parameter Group
    A. Parameter
        1. Sub-parameter
            a. Sub-parameter
                1) Sub-parameter

For example, Sub-parameter II. D. 5. a is: "Modification of item size", and it appears in the parameter hierarchy as follows:

II. File Creation and Maintenance
II. D. Input to FCM
II. D. 5. Input edit
II. D. 5. a. Modification of item size

4.1 DATA DEFINITION AND DATA ORGANIZATION

Data definition is one of the major functions of any generalized data management system. The following group of parameters measure the relative effectiveness of competing systems in performing this function.

The objective of data definition is to describe to the system the particular data on which it must operate. The structure and format of the data must be described so that the system can recognize and interpret the data.

The primary classes of data that must be defined are:

- Master files (called data sets, databanks, or simply files)
- Input files, for file creation and maintenance.

For any file of data, the structural components must be defined. Fields of data must be defined sufficiently to permit the system to find, delimit, decode, and interpret an item of data. Records and record segments must be defined sufficiently to permit blocking and deblocking and the organization of logically related segments or header-trailer records. Files are described to permit input/output, chaining, sequencing (or sequence checking) and other operations involving file structure.

Data definition may be required to provide information on the procedural or control language set that is being used for a particular application. In addition, data definition will often provide information on the standard treatment of each field in printed output such as table lookup, output edit, output conversion, and standard report column headers.

Data definition must be performed for each file that is processed. In most generalized systems, each file is defined once and the same data definition is used for all applications involving that file. Typically, the data definition is stored as part of the system.

Some systems required that data definition must be entered for each application or each run. This is generally considered undesirable because of the repetitive effort required and the possibility of introducing error.

There are two aspects of this parameter group:

1) That which relates to the procedure of producing the data organization which a user may require,

2) That which relates to the data organizations which are permitted for a system.

Although the emphasis of the discussion to follow is from the first viewpoint, it is important to recognize that to a large extent the data organization characteristics of the system are also being measured. The objective of the system data organization is to provide a framework in order that the user may build any reasonable logical structure according to his individual needs.

Data definition procedures may be optional or required. The mandatory nature of definition requirements may introduce a negative aspect to certain features. For example, is it burdensome to accomplish certain operations? Do some systems require definition of data characteristics which other systems would provide automatically?

It is necessary, therefore, to recognize the negative elements of certain capabilities. Since negative ratings are not permitted by the scoring method devised in this study, such characteristics may only be reflected by appropriately low ratings.

4.1.1 Field Definition

This parameter measures the capability of the system for describing the form and content of data fields on which it operates. The purpose of the field description is to permit the system to identify each field for subsequent retrieval, processing, or output.

The capabilities listed in Worksheet I.A are for the most part either binary (yes/no) functions or are to be estimated by the evaluator as a value judgement as expressed by selection of a rating on the standard scale. An exception is Parameter I.A.3, Number of fields which may be defined for each record, for which a raw score may be obtained. This score may have little absolute significance, however, and the evaluator may wish to consider this simply in terms of whether the number is sufficient or not.

Parameter I.A.5, (Re-)definition of Fields, although primarily intended to describe redefinitions, includes sub-items which apply to initial field definition also.

Parameters I.A.6, I.A.7, and I.A.8 relate to parameters discussed elsewhere (output editing, columnar headers for reports, and security control). The capability to be measured here is whether these functions may be specified by Data Definition Procedures.

4.1.2 Record/Segment Definition

This parameter measures the capability of the system for describing the structure of logical and physical records in the file. Typically, a logical record consists of a related set of data fields that all pertain to the same entity. For example, the entity of interest in a personnel file is a person. The logical record therefore consists of the fields of data that describe a person or are related to a person.

A segment is usually defined as a logical subdivision of a variable length logical record. Each segment consists of one or more fields of data. Segments are also called subrecords, repeating groups, etc. (see Section 1.3.6). A fixed length record is usually considered to consist of a single segment. The segments of a record need not be physically contiguous. Chained records exhibit this property. Segments may be repeated within a record. For instance, a personnel record may contain segments describing a man's education. Each segment may consist of fields for degree, major, year, and grade-point average. A record for one man may contain a variable number of these segments, one for each school attended. One record may contain several kinds of segments. Determination of the existence of some of the capabilities listed in Worksheet I. B may be difficult in some cases because some systems define records and/or segments as a part of field definition or file definition. The capabilities listed in Worksheets I. A, I. B, and I. C relate to data organization features at the various hierarchical levels and the items listed may be used as a check list from which the appropriate subparameters may be selected.

The evaluator must be careful not to bias his ratings in favor of the system which has the most complex set of capabilities, or simply on the basis of the total number of features available; unless such features result in recognizable advantage from the user viewpoint. The capability being measured is the ability of the system to recognize the logical record structure, not the complexity of the definition entries.

4.1.3 File Definition

File definition provides information to a generalized data management system to permit the compilation of input/output routines that provide physical access to the file. Capability of a generalized data management system to define files is measured by considering the specific capabilities for defining the varieties of files that may be encountered.

File Identification, I. C. 1, and Device Identification I. C. 2, relate to the manner in which the system differentiates between logical files and physical files. The criteria for judgement will be based on whether the user has the identification capabilities he needs, and whether he has the degree of freedom (options) which he may find useful. For different application requirements different rating orders may be appropriate. For example, in some cases Parameter I. C.2. a is preferable to I. C. 2. c; in other situations, the reverse may be true. Files will originate from two sources:

1) File creation by the system,
2) A file creation and maintenance activity outside the system.

Those capabilities which relate to files from the latter source are discussed in Section 4.1.6.

Parameters I. C. 2, I. C. 4, I. C. 5, and I. C. 6 are definition capabilities for corresponding functions that exist elsewhere in the parameter organization. For example, File Security Capabilities are outlined in Worksheet III. F. It is only the ability to specify them as a part of the data definition which is to be considered here.

Parameter I. C. 6 relates to methods which are designed to minimize access time for file maintenance and retrieval functions. Therefore, excellence in this area may be directly reflected in the performance capabilities of these functions (Section 4.1.8). The evaluator must be aware therefore of the potential overlap of these two areas.

The presence of a hardware associative memory (I. C. 6. e. 2) in the system may require a special analysis. Typically the associative memory would be used to contain the search criteria used for conditional retrieval or conditional update operations. This utilization would dramatically reduce search times for some applications.

Parameters I. C. 7 and I. C. 8 are special definition capabilities which are permitted for some systems. It is important for the evaluator to recognize the open endedness of these lists and to list other pertinent capabilities which may apply in certain systems, or which may be needed for the application.

4.1.4 Input Media

This evaluation parameter, considered here initially in relation to Data Definition functions of the GDMS is also to be evaluated in the File Creation and Maintenance and Data Retrieval parameter groups. This parameter measures the flexibility of the system in accepting input from a variety of sources. Sub-parameters may be conveniently identified which correspond to each of the input methods permitted. Rating of this parameter may be undertaken individually on the basis of a scale of capability which seems appropriate. An example of such a scale for which gradations of capability may be assigned is shown to follow:

1) Special features of the system permit unusually easy, fast, or inexpensive input via this medium/technique.
2) System accepts input from this medium/technique efficiently.
3) The normal system accepts input from this medium, but it is cumbersome.
4) The normal system does not accept input from this medium, but specific provision is made in the system to permit this capability to be provided.
5) The system cannot accept input from this medium.

Other aspects of the input media parameter which are ratable are:

- Data Transfer Rates
- Number of Devices On-line
- Simultaneity Features

The first two items are easily quantifiable. However, to some extent these considerations will overlap the measurements obtained for the performance parameters (I. H) and care should be taken to avoid duplicate accounting if these criteria are used.

Examples of rating considerations pertaining to specific media are:

1) Punched Cards: A system that accepts a number of different input card formats in one pass would ordinarily be rated higher than a system that requires a separate pass for each card format.

2) Paper Tape: A system that accepts a variety of punched paper tape codes would be rated higher than one that accepts only one code; a system that accepts 5, 6, or 8 channel tape would be rated higher than one that accepts only one width.

3) Magnetic Tape: Items for consideration include:

- Ability to read or skip labels
- Size of input area or buffer
- Ability to identify more than one input record type
- Ability to read variable-length input records

Although many more input devices and input media are possible for input to the GDMS (e.g., keyboard entry, light pen, or optical reader), it is not as likely that these less usual input methods would be used for Data Definition than for the other function areas; in particular, Data Retrieval (Section 4. 3. 4).

For some analyses, consideration of input media for data definition will not be appropriate since the system procedurally expects to execute this function as a part of FCM or Data Retrieval. If this is the case, this parameter should be disregarded.

The general topic of "input" is considered in more detail in Sections 4. 2 and 4. 3 in connection with File Creation and Maintenance and

Retrieval functions. It is a more important consideration for repetitive functions such as those, than it is for data definition. However, if the evaluator considers the more detailed structure of parameters (as shown on Worksheet II. D) as appropriate for data definition also, it may be substituted for Worksheet I. D.

4. 1. 5 Storage and Modification of Data Definitions

This parameter measures the power and flexibility of the system in making data definitions available for use at the time a specific run is compiled. Gradations of this capability are listed to follow:

- Data must be defined anew for each run.
- Data definitions may be stored and are available for reuse and are callable by file name.
- Modification of a data definition may be accomplished by means of a data definition specification. (Presumably less effort is required for this than for a complete data (re)definition task specification.)
- Modification of a data definition at the time of use for other GDMS functions (file maintenance, retrieval)

4. 1. 6 Capability to Read Files from Other Systems

One of the important capabilities of a generalized file management system is to retrieve information from files created by other systems. Parameter I. F measures the capability of a system to define data structures to be consistent with those created by other systems, so that existing files can be interpreted and processed.

For some applications the requirement to be compatible with another system may not exist. However, some of the features listed in Worksheet I. F are measures of system flexibility and are additive to those listed for parameters I. A, I. B, and I.C. They may be rated, therefore, even though no compatibility requirement exists.

4.1.7 Ease of Use

This parameter measures the simplicity of technique for data definition. Ease of Use is evaluated primarily by consideration of the language features which contribute to the convenience or efficiency of the data definition task. Other factors which are included in measurement of this parameter are the requisite skill level of those who prepare the data definition and the ease which the technique for data definition is learned.

The weighting of the parameter, Ease of Use, is likely to be much lower relative to other parameters in the data definition group than for other groups (e.g., Data Retrieval). The reason for this lower assessment of importance is that the data definition is done much less frequently than the other GDMS tasks. The "capability" parameters are therefore much more important in this group than are either the "ease of use" parameters or the "Performance parameters."

4.1.7.1 Language Considerations/Ease of Use. A number of language considerations may be identified which contribute to the convenience and efficiency of the data definition task. However, to a large extent, many of these are already reflected in the "capabilities" parameters described in foregoing sections. It is also noted in a previous section that one of the measures of the language excellence will be reflected in a measurement of the performance characteristics. For weighting purposes, it is therefore important to recognize that language characteristics have already been considered to some extent in these other areas.

Somewhat fewer language characteristics relating to Ease of Use may be identified for the data definition than for other functional areas; in some cases a single overall rating estimate may be preferable to a consideration of the listed items.

4.1.7.2 Skill Level Required. The skill level requirement for the data definition task may tend to be somewhat higher than for other functional areas. An important fact to consider here is that the data definition task has far reaching effects on the efficiency of other operations (e.g., data retrieval); however, the task is undertaken much less often.

4.1.7.3 Ease of Learning. This parameter, described in detail in Section 3.1.2.2, can be measured in terms of cost if experience has shown what typical training requirements have been. Items for consideration are listed in Worksheet I.G.3.

4.1.8 Performance

The performance aspects of data definition are evaluated primarily as a function of time. These are categorized on Form I.H as follows:

- Total Man Hours
- Response Time for Completion of Data Definition Task
- Machine Time Required for Data Definition

The first item is amenable to analysis based on selection of a sample task mix and measurement of the human effort involved. However, the latter items in some cases may be difficult to ascertain the data definition is not a segregated system function and is integrated with the File Creation and Maintenance and Data Retrieval functions. If this is the case, the evaluator may prefer to disregard these measurements in this group of parameters and consider them as a part of the performance characteristics to which the data definition function is procedurally associated (i.e., parameters II.H and III.H).

The measurement technique for "performance" parameters is described in Section 3.1.3. The selection of a task mix for analysis which are typical of the anticipated applications must be made with knowledge of the particular methods and procedural techniques of each of the candidate systems. An example of a task mix is shown to follow:

1) A simple task involving definition of:
   - Six or eight fields
   - One record type (fixed length, unblocked)
   - A simple file structure
2) A complex task that requires the use of as many data definition capabilities as possible.
3) Another complex task that requires a different combination of these system capabilities.
4) Evaluate the coding required to describe to the system a sample file structure which would include normal complexities - multilevels, multiple segments, variable fields, variable length records.

PARAMETER WORKSHEET

Parameter Group: I. Data Definition and Data Organization

Parameter: I. A. Field Definition

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
I. A Field Definition						
1. Field Identification						
a. Fields are identifiable by name						
b. Synonyms are permitted						
2. Field Coding which may be specified:						
a. Number representation						
1) Fixed point						
2) Floating point						
b. BCD						
c. EBCDIC						
d. ASCII						
e. Other (list)						
3. Number of Fields that may be defined for each record						
4. Data conversion may be defined for:						
a. Input encoding						
b. Output decoding						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: I. Data Definition and Data Organization

Parameter: I. A. Field Definition (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
5. (Re)definition of Fields:						
a. Fields may be (re)defined as subdivisions of other fields						
b. Fields may be (re)defined that overlap other fields						
c. Fields may be (re)defined as an arithmetic or logical function of other fields						
6. Capability to define output editing of fields						
7. Capability to define column headers for printed output						
8. Security control of fields may be defined for:						
a. Writing						
b. Reading						
9. Techniques for defining data field location in the data record on card, disc, tape, etc.						
a. Field location compiled during execution						
b. Field location compiled before execution						

NOTES:

PARAMETER WORKSHEET

Parameter Group: I. Data Definition and Data Organization

Parameter: I. B. Record/Segment Definition

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
I. B Record/Segment Definition						
1. Record/Segment Identification						
a. Records may be referenced by name						
b. Segments may be referenced by name						
c. Synonyms are permitted						
d. Implicit definition capability (capability to define a field as a function of other fields)						
e. More than one record definition per file is permitted						
f. Types of identification permitted (list)						
2. Record Length						
a. Maximum length of physical record						
b. Maximum length of logical record						
c. Maximum length of segment						
d. Number of segments per record permitted						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: I. Data Definition and Data Organization

Parameter: I. B. Record/Segment Definition (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
3. Logical Organization of Record/Segments						
a. Variable length records may be defined						
b. Number of types of segments which may be defined						
c. Capability to define several different kinds of segments at each organization level of the record						
d. Capability to define several organizational levels (hierarchical levels) in a record, with one kind of segment at each level						
e. Capability to combine c and d above; to define several organizational levels in a record with more than one kind of segment permitted at each level						
4. Relationships with other records or segments in the file may be defined Can links or chains be defined in data definition?						

NOTES:

PARAMETER WORKSHEET

Parameter Group: I. Data Definition and Data Organization

Parameter: I.C. File Definition

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
I.C File Definition						
1. File Identification						
a. Files are identifiable by name						
b. Synonyms are permitted						
2. Device Identification – data definition may specify that:						
a. The file is always located on a standard device						
b. A variety of devices may be used for input of the file						
c. Device identification occurs at execute time						
1) For file maintenance						
2) For retrieval						
3. Sequence Control Specification of records in a file						
a. Sequence Control is specified by Data Definition						
b. Sequence Control is specified but may be overridden by file creation and file maintenance functions						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: I. Data Definition and Data Organization

Parameter: I.C. File Definition (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
4. Sequence Checking						
a. Sequence Checking may be specified						
b. Sequence Checking may be specified but may be subject to override						
5. File security restrictions may be defined for:						
a. Writing						
b. Reading						
6. Indexes to files or other access aids may be defined						
a. Index to a sequential file						
b. Multi-level indexes, e.g., index to the index, to improve speed of locating a record						
c. Multiple indexes, e.g., access to each record through more than one field						
d. An algorithm is provided for computing a record address for direct access						
e. Associative memory techniques						
1) Software						
2) Hardware						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: I. Data Definition and Data Organization

Parameter: I.C. File Definition (Cont'd.)

Date: | **Evaluator:**

GDMS: | **Data Source:**

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
7. Relationships between files may be defined:						
a. For merging files						
b. To determine precedence						
c. For disposition of						
1) Matches						
2) Mismatches						
d. For correspondence of records						
1) One-to-one correspondence						
2) One-to-many correspondence						
8. Special File Structures						
a. Split records						
b. List structures						
c. Links, chains, others						
d. Inverted file						
e. Segmented files						

NOTES:

PARAMETER WORKSHEET

Parameter Group: I. Data Definition and Data Organization

Parameter: I. D. Input Media

Date: **Evaluator:**

GDMS: **Data Source:**

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
I. D Input Media						
1. Magnetic Tape						
2. Magnetic Disc File						
3. Magnetic Disc Pack						
4. Magnetic Cards						
5. Punched Cards						
6. Punched Paper Tape						
7. Typewriter						
8. Teletype						
9. Remote terminal						
10. Display console/keyboard						
11. Light pen						
12. Other (list)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: I. Data Definition and Data Organization

Parameter: I.E. Storage and Modification of Data Definitions

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
I.E Storage and Modification of Data Definitions						
1. Capability for storing data definition						
a. Data must be defined for each run						
b. Data definitions are available for reuse, callable by file name						
2. Capability for modification of data definition						
a. Modification of data definition requires a complete redefinition run						
b. Modification may be accomplished without complete redefinition run						
c. Modification may be accomplished as a part of the task specifications for:						
1) File maintenance functions						
2) Data retrieval						

NOTES:

PARAMETER WORKSHEET

Parameter Group: I. Data Definition and Data Organization

Parameter: I.F. Capability to Read Files from Other Systems

Date: **Evaluator:**

GDMS: **Data Source:**

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
I.F Capability to read files from other systems						
1. Compatibility of storage media						
a. Magnetic tape						
b. Magnetic disc file						
c. Magnetic disc pack						
d. Magnetic cards						
e. Punched cards						
f. Punched paper tape						
2. File labels and control data						
a. Ability to ignore (pass over) labels						
b. Ability to check label ID						
c. Ability to make additional label checks (i.g., creation date, reel no., etc.)						
d. Ability to pass EOF following label, if one exists						
e. Ability to pass multiple label records						
f. Ability to check EOF trailer records						
g. Ability to differentiate between EOF trailer records and end of reel trailer records						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: I. Data Definition and Data Organization

Parameter: I.F. Capability to Read Files from Other Systems (Cont'd.)

Date: **Evaluator:**

GDMS: **Data Source:**

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
3. Ability to define the physical organization of a file for file structures created by another system						
a. Partitioned files						
b. Random file organization						
c. Chained files, using a variety of definitions for the chaining address field						
4. Ability to define more than one record structure per file						
5. Ability to define a variety of logical structures for segments within a record						
a. Identify type of segment by a distinctive symbol or an identifier code						
b. Identify end of variable-length segment by a terminator symbol						
c. Locate each segment by use of:						
1) Segment length definition						
2) Repeated segment count in each record						
6. Ability to accept an existing data definition						
a. From the data file						
b. From a file other than the data file						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: I. Data Definition and Data Organization

Parameter: I.F. Capability to Read Files from Other Systems (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
7. Physical record organization:						
a. Grouped fixed-length records may be defined						
b. Grouped variable-length records may be defined						

NOTES:

PARAMETER WORKSHEET

Parameter Group: I. Data Definition and Data Organization

Parameter: I.G. Ease of Use

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
I.G Ease of Use						
1. Language Consideration/Ease of Use						
a. Capability for user-defined additions to the language						
b. Free form language characteristics						
c. Routines for performing data definition may be stored in the GDMS for later use						
d. Other (list)						
2. Skill Level Required						
a. System specialist						
b. Programming specialist						
c. Other professional (specify)						
d. Clerical						
e. Other (specify)						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: I. Data Definition and Data Organization

Parameter: I.G. Ease of Use (Cont'd.)

Date:

Evaluator:

GDMS:

Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
3. Ease of Learning						
a. Training required						
b. Number of practitioners						
c. Tutorial capabilities of system						

NOTES:

PARAMETER WORKSHEET

Parameter Group: I. Data Definition and Data Organization

Parameter: I.H. Performance

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
I.H Performance						
1. Total Man Hours						
a. Preparation of data definition						
b. Keypunch						
c. Operation and support						
2. Response Time — (Response time for typical Data Definition task)						
3. Machine Time for Sample Problems — (List selected problems and record timing results. Attach subsidiary analysis sheets)						

NOTES:

4.2 FILE CREATION AND MAINTENANCE

For a number of systems, file creation is effected by the same means as file maintenance and for most systems the correspondence of sub-parameters in the two groupings is great. Therefore, File Creation and File Maintenance (FCM) functions are considered under one heading in the parameter organization.

File Creation capabilities which are distinct from or in addition to File Maintenance capabilities are considered in Section 4.2.1. Other capabilities, which apply to both areas or to File Maintenance only, are discussed in the sections following. Performance and Ease of Use parameters (Sections 4.2.7 and 4.2.8) are intended to measure both functional areas.

4.2.1 File Creation

In addition to the potential overlap of File Creation and File Maintenance, it is also possible that certain confusion may exist as to the role of data definition in the process of file creation. The two are sometimes effected in the same operation, and if this is the case, the evaluator must exercise judgement as to the best method of accounting for the capabilities. For certain systems, the need for File Creation (as a parameter) may be obviated. This would be the case, however, only if the function were adequately accounted for by appropriate factors in the Data Definition and File Maintenance parameters.

One of the considerations of the File Creation parameter is the source of the new file. In general, the source data may be:

1) Read from an input device,
2) Constructed from existing files based on user-defined or pre-determined standard selection criteria, and
3) Entered from a user terminal.

For files obtained from existing files, merging, sorting, rearranging, or other processing may be required. In some cases, a choice between considering this topic as part of processing (Section 4.4) and/or file creation may be required.

For new files, the user may or may not be required to provide a new file description. This aspect of file creation may be viewed from two standpoints: 1) that he may need the capability to do so, and 2) that he may wish to avoid the burden of doing so.

To provide the capability for selection of data in the file creation process, conditional selection capabilities may be required. Conditional selection is discussed in detail in Section 4.3.1 in connection with the general topic of retrieval to which it closely relates. This capability is, therefore, not considered in detail here, except to consider the general question, "Are the conditional selection features of the system available for file creation? If so, to what degree of flexibility and power?"

File creation capabilities are not listed in detail in this section since for the most part these are the same as for file maintenance, considered in sections to follow. However, a few factors, which pertain to file creation in particular, are:

- Capability for conditional selection
- Edit capability for file creation
- Reliability of initial file preparation
- Validity checking for file creation

The reliability of the initial file preparation is particularly important. If additional man hours and/or machine time is required to make corrections to erroneous data, performance statistics may have to be modified. A benchmark analysis will not give a reliable measure of this sub-parameter.

4.2.2 FCM Operators

This parameter is a measure of the capability of the system for file update and other file maintenance activities. It is evaluated by considering the individual operators available for file maintenance. Most file maintenance operations are essentially yes/no functions (i.e., the operation is available or not). Therefore, the overall merit of this parameter is determined primarily by the weighting accorded to each available operation rather than by a value judgement according to the standard scale. However, there may be several gradations of capability for certain operations (e.g., table look-up conversions, etc.) for which ratings should be made.

A list of typical file maintenance operations are given below:

- Add a record
- Delete a record
- Replace a record
- Change the value in a field
- Arithmetic operations on a field-
  - Algebraic sum of original data and input value
  - Algebraic difference of original data and input value
  - Multiply original data by input value
  - Divide original data by input value
- Table look-up conversion

The list is not complete and should be expanded to include the particular capabilities of the competing systems. Consideration of the operators permitted is essentially a language analysis. The evaluation in this section should be from the standpoint of capability, not ease of use (Section 4.2.7).

Many more capabilities could be listed which are specific to one GDMS or another. The evaluator should list as many such capabilities as he is able to determine. It is noted that many additional update capabilities are found elsewhere in the section under other descriptive headings. For example: the capability to store maintenance task specifications and call for their reuse , obviating the need for reentry (Parameter II. E), and input data validation (Parameter II. D. 4).

4. 2. 3 File Maintenance

This parameter measures the file maintenance or update capabilities in user terms (for Conditional Maintenance Capabilities, see Section 4. 2. 6). These capabilities are in some cases made possible by the specific operations (operators) permitted as listed in Form II. B. Analysis both from the standpoint of available operators (language) and from the standpoint of functional capabilities provided, listed below, is useful in arriving at a more sensitive measure of file maintenance capability. However, care should be taken that duplicate consideration not be given to substantially identical capabilities. The effects of possible overlap may be mitigated to some extent in the weighting process.

This parameter is measured by weighting the individual update capabilities listed. This parameter may be scored, therefore, entirely as a function of the weighting process if all parameters are regarded as yes/no functions. However, the nature of some of these capabilities are such that graduations of capability may be recognizable. If this is the case, more sensitive measures (than 0 and 10, only) may be used for rating.

The items on this list should be regarded as optionally important depending on the nature of the applications requirements. The list should also be considered open-ended, and other specific file update capabilities should be included as their importance is recognized.

- Capability to merge two or more files
- Capability to reformat the records in a file
- Capability to establish a working file for further processing
- Capability to perform arithmetic operations in the update operation
- Capability to override data specifications in file dictionaries
- Creation of new data fields computed from arithmetic or logical relationships of one or more other data fields
- Capability to override data validation specified in file dictionaries
- Capability for use of literals for insertion or other update functions
- Capability to automatically maintain indexing controls
- Capability to perform file maintenance operations using data from more than one data file
- Capability to add a segment from a variable-length entry
- Capability to resequence segments within a variable length item
- Capability to resequence entries in a data set when a specified sort key changes
- Capability to batch input data (i.e., collect and hold data until enough is received to warrant a file update)
- Capability to query a file while it is being updated
- Special updates by user specification

The last item on the list above could be expanded to enumerate the special capabilities which may be specified by the user.

The output of FCM functions is typically an updated master file. However, the user may be permitted additional output options. A number of such output options are implied in the list of update capabilitis, above. However, additionally, the user may be permitted capabilities to:

- Select media for production of FCM reports
- Produce the new master file with a different format from that of the old one
- Produce a report listing all FCM transactions
- Produce a report showing all updated (changed) records
- On-line printed output of all items affected by transactions

The evaluator may rate such output capabilities as a part of this parameter, or alternately, as a part of output, Section 4.5.

4.2.4 Input to FCM

The input considerations for FCM functions are considered under three headings: Input Media, Input Sources, and Input Validation.

4.2.4.1 Input Media. The subject of input media was treated in Section 4.1.4. The general remarks given there apply for FCM functions. The primary distinction regarding FCM is that it is more likely to be performed on-line than is Data Definition; however, less likely than for Retrieval. A number of methods for input of file maintenance transaction are listed below:

- Card input
- Punched paper tape input
- Card images on tape
- Fixed-length tape records
- Blocked fixed-length tape records
- Variable-length tape records
- Input transactions on cards and one (or more) tape(s) simultaneously
- On-line data entry from console
- Console input processed singly or batched in transaction file

The on-line input capability is measured by Parameter II. D. 1. g On-line Terminal Devices.

Only those media that are liable to be needed by the application should be evaluated. The elimination of other media is effected by assigning weight of 0.

4.2.4.2 Input Sources. This parameter measures the flexibility of input sources from a procedural rather than a media standpoint. Several possibilities are listed on Form II. D which indicate the user choices for input of both FCM data and task specifications.

Since one of the primary objectives of a GDMS is reduction of the programmer effort, a primary consideration here is that tasks may be specified in a manner which minimizes the effort required for FCM task specification. It is noted, however, that the capabilities which achieve this (e.g., parameter-driven FCM procedures for task specification) will be reflected in the performance parameters, II. H.

4.2.4.3 Input Validation. This parameter measures the capability to validate task specifications and data input, and will reflect the time and effort saved by the avoidance of erroneously prepared tasks and/or data. Ordinarily, erroneous data should be prevented from using up machine time.

Validation of Input Data

Validation of input data protects files from being updated with erroneous data that can be detected with edit checks; processing of files with faulty data is thus reduced. Examples of edit checks are:

1) A check of data type; i.e., numeric, alphabetic, etc.
2) A range check to ensure that input data falls at or between specified limits for a data field.
3) A table check of all possible values that can be entered into a field.

Validation of Task Specification

This can include checks for current use of language, for references to fields and files, and for sequence, completeness and continuity. Required checks normally include checking for:

- Required inputs are present
- Specifications are in correct sequence
- Control codes contain permissible values
- Field, record, and file identifications are legal
- Specifications for a task are compatible

Invalid task specifications should be flagged for correction. The type of error and the specific input should be noted on a printed report, a console display, or a typed output.

4.2.5 Storage and Modification of FCM Task Specifications

The capability to store task specifications and call for them subsequently is an important capability. Considerations for evaluation of this parameter are:

1) Capability to store task specification and refer to it in an assigned name for future use
2) Capability to modify stored task specifications prior to reuse
3) Capability to store a skeleton task specification and supply variable parameters prior to use

Gradations of these capabilities are listed in Parameter Worksheet II. E.

4.2.6 Conditional Maintenance

In addition to the foregoing capabilities, a powerful capability is added if conditional update functions are permitted. This capability permits the specified update operation to occur if certain conditions are met.

For example, the conditional update feature would provide the capabilities to:

- Revise a field in all entries that satisfy certain logical criteria
- Blank a field in all entries that satisfy certain logical criteria
- Sum an external value and the contents of a field for all entries that satisfy certain logical criteria
- Eliminate all segments that satisfy certain logical criteria
- Add a segment to all items that satisfy certain logical criteria

Definition of these conditions is typically stated in terms of logical statements involving comparison expressions and boolean connectors. This type of logical operation is more often provided in connection with the retrieval function of a GDMS. A detailed treatment of selection capabilities for retrieval contained in Section 4.3.1. This discussion is also pertinent here if the same or a subset of the conditional logic capabilities are used for file maintenance. The general question is asked here, then, "Are the selection and extraction capabilities used for retrieval also available to provide a conditional update capability for file maintenance?" If so, the same organization of sub-parameters (described in 4.3.1) may be used with similar rating and weighting techniques. Alternately, the ratings obtained for the selection and extraction obtained

(Parameter III. A) could be used as a basis, and modified depending on the effectiveness of the selection logic for File Creation and Maintenance.

4.2.7 Ease of Use

This parameter measures the simplicity of techniques for File Creation and Maintenance. Ease of use is evaluated primarily by consideration of the language characteristics which contribute to the convenience or efficiency of task preparation and execution. Other factors which are included in measurement of this parameter are the requisite skill level of those who prepare the FCM task specification and the ease with which the technique for data definition is learned. As noted in a foregoing section, the management of this parameter is used primarily as an index of ease of use rather than for the transitory advantage of quickly obtaining useful work from the new user.

Discussion of ease of use parameters is found in Section 3.1.2. Specific language features, which may be considerations for the evaluation of this parameter, are listed in Form II. G.

4.2.8 Performance

The performance of the system in executing file maintenance functions is measured by this parameter. If the use of a benchmark analysis is possible, the resulting statistics will yield raw scores which may be normalized. Three aspects of performance efficiency are measured:

- Total man hours for preparation and running of file maintenance tasks.
- Response time for completion of file maintenance task
- Total machine time for execution of file maintenance task

4.2.8.1 Man Hours – File Maintenance. This parameter measures the human effort for preparation and running of a File Maintenance task. The units of measurement are hours/minutes for preparation of typical file maintenance tasks. Elements in the measurement of this parameter may include:

- Number of man-hours required to enter transaction data on line, directly through a terminal facility
- Number of hours required to record data on transmittal sheets
- Number of man-hours required to monitor and operate computer during update operations

The sample tasks should be defined according to the information obtained in the determination of applications requirement. The tasks should be undertaken by personnel who possess the skill level which is expected for performance of these tasks in the operational environment, if possible. In some cases it may be necessary, however, for the evaluator to perform the sample tasks in order to hold constant the variable of proficiency of the user.

4.2.8.2 Elapsed Time for Completion of Job Run File Maintenance. This parameter measures the response characteristics for file maintenance tasks. The measurement is taken of the interval from the time a job is submitted until it is completed.

This parameter is highly dependent on the application requirements and the desired or required response time of the user. In some cases the response time will be a function of the priority of the task in relation to other tasks. Response may be a discretionary matter dependent primarily on installation policy and standard operating procedure. In such cases, actual response should not be rated as a capability except as it may pose a limitation on the user. Graduations of response may be characteristic for which a range of values rather than a single value may be more appropriate.

The response characteristics may assume a different aspect for a real time or on-line operation with input from remote or local terminals. For on-line file maintenance operations, there is possibility of overlap with the Parameter II. H. 3, Machine Time for Sample Problem (File Maintenance). One of the considerations for this parameter is the nature of the waiting times at each of the work stations (time in queue). This accumulation occurs for both on-line and batched maintenance operations. Typical elapsed time elements for each type of operation which are likely to occur are listed to follow:

Batched File Maintenance Operation	On-Line File Maintenance Operation
Time in queue – keypunch	Wait for access to terminal
Keypunch	
Time in queue – verify	Terminal operation time
Verify	
Time in queue for stacking input	Time in input queue
Time for execution of file maintenance operation	File maintenance compute time

This parameter will be measured in appropriate units of time; days/hours/minutes/seconds. The requirements should be studied to determine the range of values which are appropriate to measure response time. This range of values will provide a basis for conversion from time units to the standard scale.

4.2.8.3 Machine Time – File Maintenance. This parameter is a measure of the performance for the computer system. It is measured most accurately by a benchmark analysis of a typical file maintenance problem. An alternate approach is the use of standard estimates such as those derived for major commercial computers by the Standard EDP Reports service. These estimates assume a standard task of processing a detail file against a master file. Various assumptions are made in respect to activity rate and computer equipment configurations in order to facilitate comparison for alternative computer systems for various applications requirements.

This parameter is one of several performance parameters and, if possible, should reflect only file maintenance functions. If the distinctions between maintenance and retrieval functions (and possibly report generation as well) are difficult to define, an overall system performance parameter, which attempts to measure hardware performance for the intended applications, may be a preferable alternative approach.

PARAMETER WORKSHEET

Parameter Group: II. File Creation and Maintenance

Parameter: II. A File Creation

Date: **Evaluator:**

GDMS: **Data Source:**

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
II. A File Creation *						
1. Capability to accept source data from:						
a. An input device						
b. Existing files (creating a file from existing files on disc or tape)						
c. User terminals						
2. Specific File Creation Capabilities						
a. Capability for conditional selection						
b. Edit capability for file creation						
c. Reliability of initial file preparation						
d. Capability to override data specifications in file dictionaries						
e. Capability to resequence or rearrange data from existing files for creation of new files						
f. Validity Checking for File Creation						
*"Ease of Use" and "Performance" considerations for File Creation is rated as a part of Parameters II. G and II. H, respectively.						

NOTES:

PARAMETER WORKSHEET

Parameter Group: II. File Creation and Maintenance

Parameter: II. B FCM Operators

Date: **Evaluator:**

GDMS: **Data Source:**

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
II. B FCM Operators						
1. Add a record						
2. Delete a record						
3. Replace a record (segment)						
4. Change the value in a field						
a. Change the value in several fields						
b. Blank (erase) the value in a field						
5. Arithmetic operations on a field:						
a. Algebraic sum of original data and input value						
b. Algebraic difference of original data and input value						
c. Multiply original data by input value						
d. Divide original data by input value						
6. Table look up conversion						
7. Other (list)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: II. File Creation and Maintenance

Parameter: II. C File Maintenance

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
II. C File Maintenance						
1. Capability to merge two files (or more)						
2. Capability to reformat the records in a file						
3. Capability to establish a working file for further processing						
4. Capability to perform arithmetic operations in the update operation						
5. Capability to override data specifications in file dictionaries						
6. Creation of new data fields computed from arithmetic or logical relationships of one or more other data fields						
7. Capability to override data validation specified in file dictionaries						
8. Capability for use of literals for insertion or other update functions						
9. Capability to automatically maintain indexing controls						
10. Capability to perform file maintenance operations on more than one data file						
11. Capability to eliminate a segment from a variable-length entry						
12. Capability to add a segment to a variable-length entry						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: II. File Creation and Maintenance

Parameter: II. C File Maintenance (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
13. Capability to resequence segments within a variable-length item						
14. Capability to resequence entries in a data set when the sort key changes						
15. Capability to batch input data (i.e., collect and hold data until enough is received to warrant a file update)						
16. Capability to query a file while it is being updated						
17. Special updates by user specification						
18. Capability for user selection of output file media						
19. Capability to produce the new master file with a different format from that of the old one (reformat)						
20. Capability to produce a report listing all FCM transactions						
21. Capability to produce a report showing all updated (changed) records						
22. Capability for on-line printing output of all items affected by transactions						

NOTES:

PARAMETER WORKSHEET

Parameter Group: II. File Creation and Maintenance

Parameter: II. D Input to FCM

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
II. D Input to FCM						
1. Input Media						
a. Magnetic Tape						
b. Disc File						
c. Disc Pack						
d. Magnetic Cards						
e. Punched Cards						
f. Paper Tape						
g. On-line Terminal Devices						
1) Typewriter						
2) Teletype						
3) Remote Terminal						
4) Display Console/Keyboard						
5) Light Pen						
h. Other (list)						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: II. File Creation and Maintenance

Parameter: II. D Input to FCM (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
2. Alternate sources of FCM data input						
a. From a specified source file						
b. Directly from an input terminal						
c. Capability to accept input from multiple input streams						
d. As part of the task specifications (the use of literals)						
3. Sources of input of FCM task specifications						
a. From conventional media (CR, tape, etc.)						
b. From input terminal						
c. From system storage in the form of retained procedures						
1) Parameters for each run must be input						
2) Parameters may be input at user option						
3) Parameters neither required nor permitted						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: II [illegible] ation and Maintenance

Parameter: II. D. Input to [illegible]CM (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION		APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
4. Input Validation						
a. Data						
1) Minimum value						
2) Maximum value						
3) Between limits						
4) Leading zeros supplied						
5) Input Sequence checked						
6) Specific characters accepted						
7) Specific characters rejected						
8) Fields compared for consistency						
9) Identification checked for:						
a) Fields						
b) Records						
c) Files						
b. Task Specification (data definition; maintenance, retrieval, processing, and output procedures)						
1) Sequence checking of logical order of specification steps						
2) Control codes checked for legality						
3) Task specifications checked for validity						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: II. File Creation and Maintenance

Parameter: II.D Input to FCM (Cont'd.)

Date: **Evaluator:**

GDMS: **Data Source:**

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
4. Input Validation (Cont'd.)						
c. Input Validation Specified (when)						
1) As a part of Data Definition						
2) As a File Maintenance Function						
5. Input Edit						
a. Modification of item size						
b. Addition of information to fields						
c. Deleting items						
d. Selection sort						
e. Specified (when)						
1) As a part of Data Definition						
2) As a File Maintenance Function						

NOTES:

PARAMETER WORKSHEET

Parameter Group: II File Creation and Maintenance

Parameter: II. E Storage and Modification of FCM Task Specifications

Date: **Evaluator:**

GDMS: **Data Source:**

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
II. E Storage and Modification of FCM Task Specifications						
1. Capability for storing FCM Task Specifications						
a. FCM task must be defined for each run						
b. Task Specifications are available for reuse, callable by file name						
2. Capability for modification of task specifications						
a. Modification of specification requires a complete rerun						
b. Modification may be accomplished without complete respecification						
3. Capability to store a skeleton task specification and supply variable parameters prior to use						

NOTES:

PARAMETER WORKSHEET

Parameter Group: II File Creation and Maintenance

Parameter: II. F Conditional Maintenance

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
II. F Conditional Maintenance						
1. Revise a field in all entries that satisfy certain logical criteria						
2. Blank a field in all entries that satisfy certain logical criteria						
3. Sum an external value and the contents of a field for all entries that satisfy certain logical criteria						
4. Eliminate all records (segments) that satisfy certain logical criteria						
5. Add a segment to all items that satisfy certain logical criteria						
6. Other (list)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: II File Creation and Maintenance

Parameter: II.G Ease of Use

Date: **Evaluator:**

GDMS: **Data Source:**

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
II.G Ease of Use						
1. Language Considerations/Ease of Use						
a. Capability for User defined additions to the language						
b. Free form language characteristics						
c. Routines for performing a particular job may be captured for repetitive use						
d. Other (list)						
2. Skill Level Required						
a. Systems Specialist						
b. Programming Specialist						
c. Other professional (specify)						
d. Clerical						
e. Other (specify)						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: II File Creation and Maintenance

Parameter: II.G Ease of Use (Cont''d.)

Date: **Evaluator:**

GDMS: **Data Source:**

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
3. Ease of Learning						
a. Training Required						
b. Number of Practitioners						
c. Tutorial Capabilities of System						

NOTES:

PARAMETER WORKSHEET

Parameter Group: II. File Creation and Maintenance

Parameter: II. H. Performance

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
II. H Performance						
1. Total Man Hours						
a. Preparation of FCM Task Specification						
1) Batched input						
2) On-line input						
b. Keypunch						
c. Operation and Support Activities						
2. Response Time — Response time for typical FCM task)						
3. Machine Time for Sample Problems (List selected problems and record timing results. Attach subsidiary analysis sheets)						

NOTES:

4.3 RETRIEVAL

This section describes the measures of power and effectiveness of GDMS in the performance of retrieval functions. From the user's point of view, retrieval capabilities are those which enable him to use the data in his files. The user may wish to select, manipulate, combine, replace, and output data. It is the selection process which is considered in this section.

The term query is used to describe the process of presenting a user request to the system. Although this term connotes an on-line request, suggesting an immediate or nearly immediate response, a broader interpretation is used here which includes the possibility of batching queries for subsequent processing and output.

4.3.1 Selection

The key to effective retrieval is the logical selectivity capability of the system.

This parameter measures the ability of the system to select items for retrieval. Data may be retrieved on the basis of its location in the file, or it may be retrieved on the basis of logical statements which define the conditions for retrieval. Retrieval may be conditional upon a set of comparison criteria which define the conditions which are needed for the desired retrieval to occur. Comparisons may be between fields, or between a field and a value introduced externally as a part of the retrieval specification input. The data used for comparison purposes are not necessarily the same as the data to be retrieved.

A selection based on a comparison of the contents of fields or on a comparison of the contents of a field and a value typically takes the general form:

IF (comparand) (comparator) (comparand)

Such an expression implies that if the specified condition exists, then certain data are selected for retrieval. For some systems, logical expressions may be combined with connectors, typically AND or OR. Negation of a condition set may be specified by NOT or AND NOT.

If the criteria for selection are met, the items to be retrieved must then be specified for output or other processing. The process of obtaining the selected data is sometimes referred to as extraction.

The general form of a logical equation shown above is typical of systems which utilize a free form coding (i. e., little or no columnar conventions are required) of logical statements to describe the values to be retrieved. There are several other methods to provide similar selection capabilities. For example, connectors may be implied, the comparator may be determined by insertion of a unique character in a particular column of a coding sheet, comparands may be indicated by number rather than name, etc. The methods for specifying the selection process may also vary considerably depending on whether it is an on-line request. On-line capabilities may include a dialogue method which will require that the evaluator must determine whether equivalent capability exists in a useable form. It is important, however, that these considerations which are those of task specification format be relegated to their appropriate parameter and not considered here, since this parameter is intended to measure the capabilities for selection only.

For some systems, the conditional and logical powers of retrieval are made available for other GDMS functions (e.g., for File Update functions). If so, the capability is measured for the other functional area, as appropriate (See Section 2.1).

The organization of the "Selection" parameter is according to the logical expression format discussed in the foregoing paragraph. The subparameters, to be discussed in the following sections are:

- Repertoire of Comparators
- Connectors (Boolean)
- Types of Comparands
- Data Selection

To these are added one further consideration of the "Selection" parameter which treats the combination aspects of the elements listed above.

- Complexity of logical relationships for selection.

4.3.1.1 Repertoire of Comparators. The parameter may be evaluated on a quantitative basis by considering the number and types of conditional relationships which may be specified. Using this method, each comparator is treated as a yes/no function and the scoring of the parameter is accomplished by the assignment of weights according to the applications/requirements needs. An alternative method is for the evaluator to make a judgement of the collective power of the repertoire of comparators permitted and rate the parameter on the standard scale.

Comparators which are commonly permitted are:

- Equal
- Not Equal

- Greater Than
- Less Than
- Greater than or equal
- Less than or equal

Others which are less usual but may be of value are:

- Between limits
- More than but not high-order value
- Less than but not blank
- Matched to a specified character pattern
- Matched to a specified masked pattern
- Keyed to a change in value encountered when moving from one record (or field) to the next

Other operators which may be used which are cumulative in nature rather than comparative are:

- Maximum
- Minimum
- Total

And finally, the combination of conditions is possible in some systems. It is noted, however, that this capability is logically identical to combining conditional criteria in compound statements. This capability is, therefore, better described by the subparameter to follow.

4.3.1.2 Boolean Connectors. The typical boolean connectors used for compound logical statements are AND or OR. For some systems the NOT operation is permitted (sometimes called AND NOT). This capability is obviously of value if there are many cases anticipated for which exclusion of data which has certain properties or characteristics is desired. It is once again emphasized here that the method of specifying the AND, OR, or NOT is not a part of this parameter. These operations may be specified by any symbol, may be implicitly designated by columnar position in a coding sheet, may be designated by a particular keyboard input convention from a terminal device or specified explicitly by a compiler language statement. The purpose of this parameter is to determine whether the capability exists, and not to evaluate the method used to specify the capability.

It is also noted that this parameter does not include consideration of the number of connectors that may be used in a statement, or in general the complexity of the logical expressions permitted. These characteristics are considered in Section 4. 3. 1. 4 (Parameter III. A. 4).

It is noted that the comparators concerning character patterns or masking operations may relate more closely to the topic of comparands. Depending on the method of task specification and the language emphasis (whether the capability is best described as a noun or a verb), this capability could be considered a part of either subparameter (but not both).

4. 3. 1. 3 Comparands. This parameter measures the number and type of values or data which may be used as criteria for selection. These items for comparison may be a part of the data in the files, may be data introduced as a part of the query input, or may be specified in the query input as literals. As an ideal, it should be possible for the user to specify as comparands all or part of any file/record/segment/field, any data introduced as a part of the query input process, or any value introduced as a part of query input. However, typically, the kinds of data or values which may be specified as conditional criteria for selection are determined by the needs of the anticipated application mix and only a limited number are permitted. The measure of this parameter should therefore be oriented closely to a consideration of what the applications/requirements data indicate would be useful. This parameter measures the degree of flexibility which the user has in specifying

test values which may be used for conditional retrieval. It is noted, however, that this parameter must be an indication of the number and type of comparands which may be specified only and should not measure the entire logical expression (i.e., the comparators or boolean operators discussed in the foregoing sections and the degree of complexity of the logical statements permitted as a whole, which is considered in Section 4.3.1.4).

Various possible combinations of comparands (fields/values/segments/characters) are indicated in the list to follow. This subparameter measures the ability to retrieve an item conditional on the comparison of the contents of a field with:

- An external value
- Another field in the same record
- The same field in another record in the same file
- The same field in a record in another file
- Another field in a record in another file
- The results of another comparison or calculation.

Further capability may derive from the ability to specify partial fields, multiple fields, overlapped fields, or selective field segments (or bits) as specified by a mask definition.

Other conditions for retrieval may involve the accumulation of a total beyond a specified threshold value, or the detection of a change condition in a field (Section 4.3.1.1), or the determining of the maximum (minimum) value of a field in order to retrieve associated fields or records.

Another capability which may exist is that arithmetic operations may be performed on selected fields; the results of which may in turn be used as a comparand. Here again, the evaluator has the choice of regarding these in terms of either the operator verbs or the operand nouns.

4.3.1.4 Complexity of Logical Statements for Conditional Retrieval. This parameter measures the power and sensitivity of the conditional selection logic in terms of the provisions for combinative statements and/or expressions. One of the primary capabilities to consider in this parameter is whether nesting of expressions is permitted. The term "complexity" is not to be regarded as a system virtue, per se, except as it contributes to the power and sensitivity for conditional retrieval. Indeed, depending on the objectives and criteria of judgement indicated by requirements information, complexity may be considered to be a detrimental factor.

This parameter typically will be rated subjectively based on the evaluator's judgement and assessment of the requirement for logical selection. Quantitative considerations for evaluation might include:

- Number of conditional expressions which may be combined.
- Number of nesting levels permitted.

However, these considerations should be evaluated only on a basis of actual utility. For example, the provision to combine 20 expressions may be of slight advantage, if any, in comparison with the capability to handle only 10. Five nesting levels are probably substantially as good as ten would be. Furthermore, it is unlikely that these numerical measures would be as significant as the overall subjective evaluation of this factor.

4.3.2 Data Extraction

The process of obtaining the data after it has been identified is sometimes referred to as "Data Extraction". The data selected for extraction may or may not be the same as the data which are used for search criteria (i.e., the comparands). For example, it may be possible to retrieve all items in Field A which are between limits x and y; or to retrieve

Fields B, C, and D for all records for which the contents of Field A are between limits x and y. In the latter case, the capability for specifying Field A as a comparand is a part of parameter III. A. 3, the capability for specifying Fields B, C, and D as those fields to be used for retrieval is measured by III. B.

4.3.2.1 Specific Capabilities. It is important to measure the accuracy and sensitivity of the system in retrieving the desired information; however, much of this sensitivity has already been described as part of the logical equations discussed in the foregoing sections. Measurement of this sub-parameter must relate only to the variety of and flexibility for retrieval of items after the conditional aspect of the task specification for retrieval has been measured. Thus, the ability to retrieve from various levels of hierarchical levels of data; from fields, segments, records, or files should be considered independent of the conditional statements or expressions.

Although a subjective judgement may be called for here, it may be based on a number of rather straightforward considerations which relate closely to the expected data structures involved, and the applications/ requirements information. A number of such considerations are listed to follow:

Does the capability exist to:

1) Retrieve from data sets of the type created by this system.
2) Retrieve from data sets of a type not created by this system.
3) Retrieve from any one of many different data sets by selecting appropriate data definitions from file identification only.
4) Retrieve simultaneously from two or more files (multifile query).
5) Retrieve data from one file based on selection criteria found in another file.

6) Retrieve data in any order from selected items. (Extracted fields may be different from selection fields.)

7) Retrieve any desired characters or partial fields from selected items.

8) Retrieve all repetitions of repeated fields, or only specified repetitions.

9) Perform many separate retrieval jobs in one pass.

The question arises as to what is to be done with the data retrieved and whether any subsequent process is to be included in this subparameter. The capability to establish a working file for further processing should be included; however, whether the retrieval data is to output by cards or tape, or is to be kept for report generation are matters to measure with other parameters.

4.3.2.2 Relevance. The technical details relating to retrieval method discussed in the foregoing sections do not directly consider the relevance of the response. The system may be accurate in its response, yet not get the desired information. It can be argued that the question of relevance stated generally as "Did we get what we wanted?" transcends any single parameter discussion and may be evaluated only as a composite of many subitems. In the identification of the many technical details of a system, it is easy to become concerned only with details of method, and in questions of "form" rather than "content". In respect to the selection parameter, it seems appropriate therefore to permit the evaluator the discretion of a relevance judgement which is content oriented.

4.3.3 On-Line Capabilities

In general, on-line capability permits the user to communicate directly with the system and to receive a rapid response to the query he has introduced. The response may be an answer to the query, i.e., the data selected for retrieval, or it may be an indication that the query has been received and is being processed for later output. The purpose of the query may be to condense and summarize decision making data, to answer pertinent questions, or to select appropriate data from a control data repository.

The capability must be evaluated in terms of the requirements. For example, some questions directed to a data base require a rapid response to be useful. This is typical of the command/control environment. In some cases an immediate response will permit the user to improve the query.

Typically, on-line systems deal with communications to and from terminals at either remote or local locations. These communications usually have well defined format characteristics which, to a large extent, depend upon the communications equipment. Knowledge of the interface is mandatory for an accurate and useful definition of the communications input/output. In on-line systems the most important interface is that between user and the system. The characteristics of the input/output should be only indirectly a function of the equipment. They should be primarily a function of the way the user can most conveniently handle the data. Important aspects besides the transfer rates and formats are the degree of buffering and the kind of "attention" mechanism which will be used, i.e., whether it will be available on interrupt, alert, or continuously. One important consideration is whether the user has a convenient choice between direct access on-line or of having his query processed on a scheduled basis.

Every on-line system has a group of people who use it. These users are of different types: those who sit at consoles and query the data base and those who service and man the equipment (e.g., computer operators). For each of these groups certain acceptable procedures must be planned and defined. The procedures must allow the data processing system to operate without interruption. In addition, many systems operate in one or more modes depending upon the demands of the application. These modes and procedures are an integral part of the design and often govern to a large extent both the nature of the language and the resulting effectiveness of its use.

Factors which should be considered in assessing other on-line capabilities of a system are:

1) Capability for communicating with multiple on-line users.
2) Capability for communicating with users at distant locations.
3) Mean time to initiate processing in response to a user request.
4) Maximum rate of simultaneous on-line query traffic.
5) Processing time for each query. (This parameter is strongly dependent on the type of query and the amount of processing required. The range of processing times for a wide variety of queries should be determined.)
6) Computer guidance for the inexperienced or inept user.
7) The response to an unacceptable query, in terms of detecting the user's error and responding with instructions for correcting the error or otherwise proceeding.

In the evaluation process it may be convenient for the evaluator to consider this parameter categorized in terms of three subparameters pertaining to different aspects of on-line capabilities.

- On-line traffic volume
- Specific capabilities
- Methods (interrupt, priority, simultaneity, and data access)

It is also appropriate to measure on-line capabilities in terms of specific query language features and in terms of response time. In accordance with the parameter organization assumptions, however, these items are considered to be ease of use and performance parameters, and are discussed under those headings. (See Sections 4.3.7 and 4.3.8).

4.3.3.1 On-Line Traffic Volume. This characteristic of system capability is particularly amenable to quantification; however, normalization to a value scale of utility may involve a subtle evaluation of differential cost, consideration of system limitations, and a careful attention to the possibility of parameter overlap. Several numerical measures may be pertinent such as:

- Maximum number of on-line users
- Maximum number of simultaneous on-line users
- Average number of simultaneous on-line users
- Number of input channels
- Number of consoles/terminals
- Number of queries

Not all of the above characteristics would ordinarily be used for a single analysis. The type of numerical measure selected may vary depending on the application/requirements information. The important measure may be the one which is most liable to be a system limitation. Interrelating numerical relationships may determine which are the most significant measures. For example, it may be that the number of simultaneous on-line users is limited by the number of terminals, the number of input channels, or by processing limitations in the computer system. Other capacity or volume considerations may be obtained by tabulating data transfer rates or the volume of on-line query traffic. The number of queries may not be a dependable statistic, however, since it may be possible in a given system configuration to process a large number of simple queries or a small number of complex ones.

Certain of these relationships may be illustrated by example. JOSS, the in-house on-line system employed by RAND Corporation, provides a problem-solving capability for scientific and engineering personnel. The number of potential outlets is 200; however, these are simply wall plugs which permit convenient access by employees. A total of 30 consoles may be utilized (plugged in) at any one time. Whether all thirty on-line users have instant, merely adequate, or insufficient response depends on the total processing demands that are currently being made. If these demands should become too great, or if it becomes necessary to increase the on-line capacity to 50, then greater computing capability would be needed. This illustrates the tradeoffs which may be involved in considering:

1) Number of users
2) Processing capacity (e.g., a measurement of peak conditions)
3) Response time

The evaluator may have difficulty in avoiding an overlap in scoring these related parameters. The number of users may directly effect response time (to be discussed in the next section) which in turn is related to the processing capacity of the computer system.

The identification of cause and effect is not the prime concern of the evaluator; however, it is important that duplicate accounting not occur. The evaluator should, in such cases, try to eliminate rating of the parameter which provides the less meaningful measure of system capability and retain only the more descriptive and sensitive measure.

Normalization of traffic volume factors was illustrated conceptually in Figure 6.

4.3.3.2 Specific Capabilities. A number of specific capabilities may be identified which may be important criteria for evaluation of on-line capability. However, this list should be considered incomplete — to be modified and expanded according to the particular characteristics and requirements of each evaluation.

- Capability for console programming.
- Capability for user to specify the type of output media or specific peripheral unit.
- System response to change in load.
- Confirmation of on-line input message.
- Ability to erase (e.g., backspace) erroneous data input.
- Capability to accommodate a wide range of terminal devices.
- Guidance capabilities for the inexperienced or inept user
  - error responses
  - self-teaching program
- Input message check to assure content consistency.

4.3.3.3 Methods. The on-line capability may be measured by the number of users served, by consideration of specific capabilities and by response time and query language characteristics, and it can be argued that these considerations summarize the on-line capability entirely. However, the variations in the methods which make possible on-line capability vary widely and may involve many important considerations of operational efficiency. Several methodological considerations which should be considered are:

- Interrupt methods
- Priority logic and queueing algorithms
- Simultaneity of Operations
- Data access methods

An automatic interrupt system is a powerful system capability which permits events external to the computer system to be registered in the computer program in a timely manner. This will in turn permit the computer system to respond to new situations. Typically, interrupt programming methods will include provision for a return to the place in the program where interrupted and will include save and restore logic.

Response characteristics will generally be enhanced if provision for levels of priority and data accessibility is made. This may involve:

- Recognition of classes of users.
- Classification of data according to frequency of use in the memory hierarchy.

The latter consideration might dictate that high priority data would be stored on drum or disk and lower priority data would be stored on tape.

The priority logic will depend on the selected queueing algorithms. These methods are sufficiently diverse to suggest that subsidiary analysis might be appropriate. A number of possible considerations are:

- Are priorities determined by the user or imposed by the system via a scheduling algorithm?
- Are tasks serviced on a round-robin basis?
- Are priorities a function of:
  - the type of user?
  - the task size?
  - the terminal identification?
  - the system status?
- Are there differential queries for classes of users?
- Is the system cyclical? If so, is the cycle time a function of the number of users?
- Do priorities change dynamically?
- Do on-line tasks compete with background jobs?

Data access methods is a general topic which transcends the "on-line capabilities" topic. It is measured to a large extent by performance parameters. It is also treated as a part of Parameter VI. A. However, if data must be quickly accessible in order to provide on-line response, this variable should be considered explicitly as it applies to on-line capability. A method of data indexing, random access storage, or use of a hardware associative memory, may permit an on-line response which otherwise would not be possible.

Another criterion for evaluation is the degree of simultaneity of operations which the program permits, e.g.:

- Communication simultaneity.
- Simultaneity of CPU and peripheral equipment.
- Simultaneity among computer programs (multiprogramming)
- Simultaneous interrogation of the same file by different users.

Although it may be difficult to associate a reliable rating with each of these considerations, it may be somewhat easier to judge the relative effectiveness of two candidate systems in these respects. For example, it may be ascertainable that interrupt capabilities for System A are somewhat superior to those of System B and therefore, should be reflected by a higher rating for System A. It is important, however, that if these capabilities are adequately reflected by other parameters (e.g. , response time) that it not be rated again here.

The recommended method for evaluation is to select those on-line features which constitute identifiable system capabilities and rate them according to a subjective analysis of the value they represent in terms of on-line system performance.

4.3.4 Input

Input considerations are detailed in Section 4.2.4, which apply to File Creation and Maintenance. The structure of subparameters there outlined are largely adequate for measurement of the input considerations for retrieval functions. However, greater emphasis should be placed on the on-line aspects of the query characteristics in this section. The various input subparameters for retrieval are listed on Parameter Worksheet III. D.

4.3.5 Storage of Queries

The storage of a query task specification represents an important capability which may be measured in reduction of the time and effort which would otherwise be required to input frequently used query. Following are several forms of this capability:

1) Capability to store a query in the system and to call the query by an assigned name for future use.
2) Capability to recall a stored query and modify it prior to reuse.
3) Capability to provide variable parameters to a stored query at execution time. In this type of query, some of the specifications, such as field names in conditional comparisons, are not pre-stored and must be supplied when the query is performed.

The capability to store queries is important for several reasons. The primary reason for desiring this capability is to permit fast, easy, and foolproof direct access to the system by non-technical users. This is one of the main objectives of generalized data management systems.

The storage of repeated queries may be of little or no benefit, however, if the reduction of effort is insignificant. A simple query may be just as easy to enter in full each time it is used as it is to retrieve it from the system for reuse. The storage of lengthy queries, however, is a distinct advantage, reflected in faster and more efficient operation.

4.3.6 File Security

The protection of user programs and data files is an important consideration. This is particularly vital if there is a large number of users. If there is shared access to common files of data, the problem of data protection is especially crucial. This means of protection may be generally divided between hardware and software, i.e., hardware protect features and software supervisory control. Although the former is usually considered somewhat the more dependable, the main concern of the evaluator is the degree of protection afforded, not the method by which it is achieved.

The rating of this parameter may be on a basis of degree of capability, (degree of file or program security) or it may be on a basis of particular capabilities for which a system value (utility) may be attached. Examples of these capabilities are:

1) Protection features for user program storage.
2) Protection of file data against
   - Unauthorized access
   - Accidental update
3) Provisions for supervisory override of security specification.
4) Designation of authorized user categories.
   - By classes of users
   - By individual user
5) Capability to protect specified fields/records within file.

4.3.7 Ease of Use

The aspects of this GDMS criteria were described in earlier sections (3.1.2, 4.1.7, and 4.2.7). The measurement of this parameter for retrieval functions should consider the same sub-items:

- Language Considerations
- Skill Level Required
- Ease of Learning

Greater emphasis should be placed on the on-line characteristics of the query language for retrieval than for the other functional areas.

4.3.7.1 Language Characteristics. It was suggested in foregoing sections that language attributes should be identified with appropriate capabilities and for the most part accounted for elsewhere in the parameter list. However, special language features and basic language philosophy differences may be measured under the "language" parameters.

The language considerations which are to be considered here are those which pertain exclusively to on-line functions and operations. Some of the pertinent language considerations are listed to follow.

- Simplicity of query language for typical queries.
- Dialog or conversational capability.
- Capability for user-defined additions to the language.
- String set substitutions capability.
- Free form language characteristics.
- Capability to refer to procedures by name.

The capabilities for on-line specification of retrieval functions are related to the flexibility of the terminals, consoles, and display equipment, which are partially accounted for elsewhere in the parameter organization.

4.3.7.2 Skill Level and Ease of Learning. Measurement of these variables are described in Sections 4.1.7 and 4.2.7. It is noted however, that the ratings of these parameters may vary among the functional GDMS parameter groups and, indeed, that is the reason for considering them separately.

4.3.8 Performance

The performance of the system in executing data retrieval tasks, as is the case with other performance parameters, is measured by:

1) Total man hours for preparation and running of data retrieval task.
2) Response time for completion of data retrieval task.
3) Machine time for execution of data retreival task.

The measurement of this parameter will vary considerably depending on whether an on-line environment for the retrieval task is assumed. Both environments (on-line and off-line retrieval) must be considered if the applications requirements information indicate that both methods will be used.

4.3.8.1 Total Man Hours. This parameter measures human effort for preparation and running of a data retrieval task. For on-line queries the measurement will include the time required to formulate the query, (possibly prior to the use of the on-line terminal), the time required to enter the data request and the delay time, if any, in receiving the response.

4.3.8.2 Response Time. There are two aspects of response time to be considered depending on whether the retrieval request is an on-line request or not. For a time-sharing environment for which immediate or near-immediate response is desirable, the range of acceptable performance is much different than for tasks which involve selection of considerable data for output in report form.

Elapsed Time for Completion of Job Run-Data Retrieval

This parameter is a measure of the response characteristics for Data Retrieval tasks which are presented to the computer in some manner other than the on-line query as discussed below. Typically, this implies a batched processing operation for which printed reports are expected as output. The methods for measurement of this parameter were discussed in Section 4. 2. 8. 2.

Response Time — On-Line

The response characteristics of an on-line retrieval request may be in units of milliseconds, seconds, minutes, or hours. One definition of the on-line capability includes the proviso that each on-line user may consider that the system is at his disposal. In this respect, it is of interest that experience has shown that human perception of time intervals is such that a response is generally considered (subjectively) to be immediate if it is in the order of one-half second or less. Time intervals in excess of that amount are perceived as time delays.

Time delays may be a function of the number of users who are using the system at the time, and also of the processing power of the computer system (unless this capability is sufficient to ensure that under no circumstances is it a limiting system factor).

Therefore, in assessing this parameter, it will be necessary to derive a set of operating assumptions which may be regarded as typical with respect to anticipate on-line traffic and assumptions regarding computation capability. With these assumptions as a basis, standards of performance may be set which can then be interpreted (normalized) in terms of a common scale value.

One method of measurement is to derive the mean-time from the completion of an inquiry until completion of the response. An enumeration of types of user requests, organized into discrete groupings and weighted according to expected frequency may be useful to determine this measurement more accurately. Another variable to be evaluated in the average delay time as a function of the numbers of queries in queue (for example, a measurement of the average delay time until the acceptance).

Measurement of the response time parameter may also be effected by an analysis of the component intervals which in the aggregate comprise total response time. For example:

- Processing time
- Time-in-queue
- Setup
- Input thinking and querying time
- Output

Output further may be analyzed based on the nature or characteristics of the output media, such as:

- Printing speed of the high-speed printer
- Typewriter speed of on-line remote inquiry stations
- Computing speed of the generative routines for displays
- Display unit ability to restore and hold a display

Output performance is also measured by Parameter V. H, however, and the overlap should be resolved by either removal of this variable from consideration here, or by a weighting adjustment which takes the overlap problem into account.

The response time elements may be evaluated in terms of sample measurements of individual performance, or as a function of statistical computations of response characteristics (e.g., mean-time to initiate processing in response to user request). The suggested method of quantification of this parameter is to develop standards of performance which may be plotted on the standard scale and normalized according to the evaluator's judgement of response time requirements. An illustration of the normalization process is shown in Figure 7.

4.3.8.3 Machine Time. Measurement of this subparameter is discussed in Section 4.2.8.3. Further discussion of performance measures and Figure of Merit analysis is contained in Section 4.4.8.1.

PARAMETER WORKSHEET

Parameter Group: III. Retrieval

Parameter: III. A. Selection

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
III. A Selection						
1. Repertoire of Comparators						
a. Equal						
b. Not equal						
c. Greater than						
d. Less than						
e. Greater than or equal (not less than)						
f. Less than or equal (not greater than)						
g. Between limits						
h. More than but not blank						
i. Less than but not blank						
j. Matched to a specified character pattern						
k. Matched to a specified masked pattern						
l. Keyed to a change in value encountered when moving from one record (or field) to the next						
m. Maximum						
n. Minimum						
o. Total						
p. Other (list)						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: III. Retrieval

Parameter: III. A. Selection (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
2. Boolean Connectors						
a. AND						
b. OR						
c. NOT						
3. Comparands						
a. Capability to retrieve an item conditional on the comparison of the contents of a field with:						
1) An external value						
2) Another field in the same record						
3) The same field in another record in the same file						
4) The same field in a record in another file						
5) Another field in a record in another file						
6) The results of another comparison or computation						
b. Capability to specify as a comparand						
1) Partial fields						
2) Multiple fields						
3) Overlapped fields						
4) Partial field as specified by a mask						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: III. Retrieval

Parameter: III. A. Selection (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
4. Complexity of Logical Statements for Conditional Retrieval						
a. Number of Conditional Expressions which may be combined						
b. Number of nesting levels permitted						
c. Overall evaluation of complexity						

NOTES:

PARAMETER WORKSHEET

Parameter Group: III. Retrieval

Parameter: III. B. Data Extraction

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
III. B Data Extraction						
1. Specific Capabilities – The capability to:						
a. Retrieve items from all hierarchical levels of data base						
b. Retrieve items used as search criteria						
c. Retrieve items different from search criteria items						
d. Retrieve simultaneously from two or more data sets that contain complementary data						
e. Retrieve data in any order from selected items. (Extracted fields may be different from selection fields.)						
f. Retrieve any desired characters or partial fields from selected items						
g. Retrieve all repetitions of repeated fields						
h. Retrieve specified repetitions of repeated fields						
i. Retrieve from any one of many different data sets by selecting appropriate data definition from file ID						
j. Retrieve from data sets of a type not created by this system						
k. Capability to establish a working file for further processing						
l. Perform many separate retrieval jobs in one pass						
2. Relevance of Data Extracted						

NOTES:

PARAMETER WORKSHEET

Parameter Group: III. Retrieval

Parameter: III. C. On-Line Capabilities

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
III. C On-Line Capabilities						
1. On-line traffic volume						
a. Maximum number of on-line users						
b. Maximum number of simultaneous on-line users						
c. Average number of simultaneous on-line users						
d. Number of input channels						
e. Number of consoles/terminals						
f. Number of queries						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: III. Retrieval

Parameter: III. C. On-Line Capabilities (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
2. Specific Capabilities						
a. Capability for console programming						
b. Capability for user to specify the type of output media or specific peripheral unit						
c. System response to change in load						
d. Confirmation of on-line input message						
e. Ability to erase (e.g., backspace) erroneous data input						
f. Capability to accommodate a wide range of terminal devices						
g. Guidance capabilities for the inexperienced or inept user						
1) Error responses						
2) Self-teaching program						
h. Input message check to assure content consistency						
i. Other (list)						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: III. Retrieval

Parameter: III. C. On-Line Capabilities (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
3. Methods						
a. Interrupt Methods (e.g., from console or remote user terminals)						
b. Priority Logic						
1) How determined:						
• By users						
• Scheduling algorithm						
• Round robin						
2) Priorities are a function of:						
• Type of user						
• Task size						
• Terminal ID						
• System status						
• Other						
3) Priorities change dynamically						
4) On-Line tasks compete with background jobs						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: III. Retrieval

Parameter: III. C. On-Line Capabilities (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
3. Methods (Cont'd.)						
c. Data Access Methods						
1) Sequential access						
2) Indexed sequential access						
3) Random access						
4) Associative memory						
d. Simultaneity						
1) Communication simultaneity						
2) Simultaneity of CPU and peripheral equipment						
3) Simultaneity among computer programs (multi-programming)						
4) Simultaneous interrogation of the same file by different users						

NOTES:

PARAMETER WORKSHEET

Parameter Group: III. Retrieval

Parameter: III. D. Input

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
III. D Input						
1. Input Media						
a. Magnetic tape						
b. Disc file						
c. Disc pack						
d. Magnetic cards						
e. Punched cards						
f. Paper tape						
g. On-line terminal devices						
1) Typewritten						
2) Teletype						
3) Remote terminal						
4) Display console/keyboard						
5) Light pen						
h. Other (list)						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: III. Retrieval

Parameter: III. D. Input (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
2. Input Sources						
a. From conventional media, CR, tape, etc.)						
b. From input terminal						
c. From system storage in the form of retained procedures						
1) Parameters for each run must be input						
2) Parameters may be input at user option						
3) Parameters neither required nor permitted						
3. Input Validation						
a. Task specification						
1) Sequence checked						
2) Control codes checked for legality						
3) Task specification checked for compatibility						
b. Input Validation Specified (when)						
1) As a part of date definition						
2) Before being used for compilation of retrieval program						

NOTES:

PARAMETER WORKSHEET

Parameter Group: III. Retrieval

Parameter: III. E. Storage of Queries

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DLS.	OBS.	RATING	WEIGHT	SCORE
III. E Storage of Queries						
1. Capability to store queries						
a. On-line entry						
b. Scheduled or batched						
2. Capability to recall a stored query and modify it prior to reuse						
3. Capability to recall a stored query and supply parameters for specific request						
a. Scheduled or batched jobs						
b. On-line insertion of parameters						

NOTES:

PARAMETER WORKSHEET

Parameter Group: III. Retrieval

Parameter: III. F. File Security

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
III. F File Security						
1. Protection features for user program storage						
2. Protection of file data against						
a. Unauthorized access						
b. Accidental update						
3. Provisions for supervisory override of security specification						
4. Designation of authorized user categories						
a. By classes of users						
b. By individual user						
c. By source of input (e. g., terminal I. D.)						
5. Capability to protect specified fields or records within a file						

NOTES:

PARAMETER WORKSHEET

Parameter Group: III. Retrieval

Parameter: III.G. Ease of Use

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
III.G Ease of Use						
1. Language Characteristics						
a. Off-line characteristics						
1) Capability for user-defined additions to the language						
2) String set substitution capability						
3) Capability to refer to conditional statements by label (name)						
4) Ability of the system to detect and correct minor breaks in the user's syntax						
5) Free form language characteristics						
6) Capability to add own code						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: III. Retrieval

Parameter: III. G. Ease of Use (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
1. Language Characteristics (Cont'.)						
b. On-line query characteristics						
1) Simplicity of query language for typical queries						
2) Dialog or conversational capability						
3) Capability for user-defined additions to the language						
4) String set substitutions capability						
5) Free form language characteristics						
6) Capability to refer to procedures by name						
2. Skill Level Required						
a. Systems specialist						
b. Programming specialist						
c. Other professional (specify)						
d. Clerical						
e. Other (specify)						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: III. Retrieval

Parameter: III. G. Ease of Use (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
3. Ease of Learning						
a. Training required						
b. Number of practitioners						
c. Tutorial capabilities of system						

NOTES:

PARAMETER WORKSHEET

Parameter Group: III. Retrieval

Parameter: III. H. Performance

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
III. H Performance						
1. Total Man-Hours						
a. Preparation of retrieval task specification						
1) Batched input						
2) On-Line input						
b. Keypunch						
c. Operation and support activities						
2. Response time for typical request						
a. Scheduled or batch job						
1) Time in queue						
2) Set up time						
3) Processing time						
4) Output						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: III. Retrieval

Parameter: III. H. Performance (Cont'd.)

Date:

Evaluator:

GDMS:

Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
2. Response time for typical request (Cont'd.)						
b. On-line request						
1) Input and transmission						
2) Time in queue						
3) Processing time						
4) Output speed characteristic						
• Transmission speed						
• Printing speed, etc.						
3. Machine time for sample problem (List selected problems and record timing results. Attach subsidiary analysis sheets.)						

NOTES:

4.4 PROCESSING

This group of parameters measures the computation, summarization, sorting, statistical, conversion, and other processing capabilities of the system. An optional parameter form is provided to measure document processing and retrieval capabilities in case this type of system capability is required. Many of the capabilities described herein have application to other of the parameter group functions. However, this section of parameters is included because these capabilities may not adequately or completely be dealt with under the other functional headings.

The evaluation of processing capabilities requires an analysis of the capabilities of the system to perform an operation:

- By GDMS language
- By conventional "programming"
- By the use of a single operator (rather than a procedure or routine)

For example, it may not be possible to compute an average value for a field in a GDMS, except by adding own code. If an average can be computed, the capability to specify the computation with a single operator is more valuable than the capability to calculate an average by specifying the detailed steps of summing field values, counting the number of field values, and dividing the sum by the count.

A complete processing facility, such as might be typical for a scientific computing installation, would ordinarily be beyond the scope of the evaluator's consideration, even if such capabilities were easily available because of the colocation of such a system with the GDMS. However, the capability to call for specialized processing facilities by a convenient linkage with another system constitutes a valuable capability if such processing is needed for the GDMS purposes. It therefore resolves to a question of the application requirements for such processing and the selected criteria for the particular evaluation.

4.4.1 Computation

This parameter measures the computation capabilities of the system. It will relate closely to the capability of the CPU and in some cases an approximation of the capability may be determined from the instruction repertoire. However, to some extent this parameter is considered to be a language capability measurement. Therefore, a computer capability would not qualify unless it could be specified by the user.

This parameter therefore is organized into three categories:

1) Operators
2) Operand Specification
3) Control

The sub-parameters to consider are listed below and again on forms IV.A.1, IV.A.2, and IV.A.3. It should be emphasized that these lists of items are not complete and should be modified and extended depending on the computation needs indicated by applications requirements information.

These capabilities usually take the form of statements or expressions containing operators (actions to be performed) and operands (data fields to be acted upon), and control operations to determine sequence of operation and disposition of results.

Operators

- Addition
- Subtraction
- Multiplication
- Division
- Exponentiation
- Trigonometric functions
- Square root
- Boolean operators

Operand Specification

The capability to specify as operands:

- Input transaction fields in file creation and maintenance
- Fields in master files (data sets or data banks)
- Retrieved data fields (including operations among several fields, such as cross column arithmetic)
- Output detail fields (e.g., fields in detail print line)
- Constants (either literal or named constants)
- System-supplied constants
- Results of prior computation
- Results of summation (summary fields)

Control

- The capability to iterate or repeat the execution of a processing statement.
- The capability to specify the execution sequence of processing statements.
- The capability to execute a processing statement based on the results of prior processing.
- The capability to use the results of processing statements to establish new data fields.

An alternative method of evaluating this parameter would be by means of a subsidiary analysis of the instruction repertoires of the competing systems. The results of this comparison could then be used as an index to arrive at a rating for this parameter as well as for other related parameters (e.g., Language Considerations, IV.G.1).

4.4.2 Summarization

The capability to summarize detail data is evaluated here. To rate, compare requirements with capabilities for:

- The number of fields that can be totaled at any level.
- The number of levels of totals and subtotals of a ~~field~~ that can be accumulated.
- Controlling a particular level of total based on a change of value in a corresponding control field or according to specified control conditions.

Summarization is closely associated with report preparation. The flexibility of the summarization in respect to specifying subtotal levels will largely effect whether the information presented is pertinent and relevant to the users needs.

4.4.3 Sorting

Sorting is one of the most significant capabilities provided as part of GDMS processing. It enables data to be arranged in more useful forms and to be organized according to specific user needs. Functional requirements of users frequently require that printed reports be in different sequences from the master file. Examples of sort functions are:

- The ordering of a transaction file for processing with a master file.
- The ordering of retrieved data according to a sort key to prepare a stratified report.
- The ordering of data obtained from several files to create a new file.

The above tasks are representative of file maintenance, data retrieval, and file creation sorting tasks, respectively. The sorting capability is thus implied in the capabilities evaluated in the other parameter groups. The capability is evaluated explicitly by parameter IV. C.

Sorting may be measured by considering the number and extent of sorting capabilities afforded, the variety of items which may be sorted, (e.g., the number of fields that may be used as sort keys in a file), and by evaluation of limitations and constraints which effect the overall sorting performance and efficiency.

Sorting capabilities are classified in Form IV.C under the subheadings:

1) Sources of Data to be Sorted
2) Sorting Characteristics
3) Limitations of Sort Operation
4) Specification of Sort Operation

Sorting is sometimes used as a primary measurement of computer performance. The application is amenable to analysis on the basis of well known sorting formulas. However, it is noted that sorting performance is not evaluated in this parameter specifically; however, it is considered in Section 4.4.8.

Some systems require that input data be pre-sorted in master file sequence; other systems accept input data in any order and perform an internal sort. The capability to accept unsorted input can be useful, but it should be evaluated carefully. Internal sorting probably increases processing time and cost, and it may be cheaper to pre-sort input by other means. Furthermore, input data may already be in proper sequence as an output of another operation, and automatic sorting of input can be a needless operation.

Sequence checking of input data that is supposed to be in a specified order will detect out-of-sequence input and minimize loss of computer time. This capability, however, should be evaluated only when a GDMS requires pre-sorting.

The option to utilize special coding supplied by the user, i.e., own coding, is sometimes provided in connection with the sorting operation. Own coding is considered in a later section as a language feature. It may be evaluated also here since it may supply an augmented sort capability if it is available at sort time. Uses for own code at sort time include special editing functions, rearranging or translation of sort keys, and other minor processing.

4.4.4 Data Conversion

This parameter covers two types of conversion: the form of number representation and encoding/decoding.

4.4.4.1 Number Conversion. The capability to specify conversion from one form of number representation to another form is evaluated in this subparameter. Conversion from binary to decimal and from decimal to binary is one capability to consider. Another capability to evaluate is conversion from fixed-point to floating-point and from floating-point to fixed-point.

Both of these capabilities can be useful by providing greater capability to accept different forms of input data and by providing a wider range of representation options for output data. These and other conversion capabilities which may exist should be rated in terms of their usefulness in meeting requirements.

4.4.4.2 Encoding/Decoding. This capability provides for the conversion of a field on input into the value that will be stored in the file, and for the conversion of a field into the value to be printed or displayed for output. The conversion or encoding and decoding process can be accomplished by table look-up or by subroutine. The tables or subroutines used for encoding/decoding may be specified in data definition or at execution time (dynamic table look-up).

Some of the advantages of using encoding/decoding capability are:

- Saving of file space by using short codes for long fields and using fixed length codes for variable length fields.
- Using dynamic tables for various calendars such as Fiscal Year, calendar year, etc., at run time for changing entire tables during processing for simulation or predictive analysis; and for accommodating existing files that contain codes.

4.4.5 Statistical

The capability to generate statistical values is measured with this sub-parameter. The following capabilities are examples of operators available in some systems:

1) The capability to find the maximum or minimum value of a field.
2) The capability to calculate the average value of a field.
   - Arithmetic mean
   - Mode
   - Median
3) The capability to calculate a running average for a field (i.e., the average value for N consecutive records).
4) The capability to calculate the standard deviation of a field.
5) The capability to count the number of entries of a field.
6) The capability to count the number of unique values of a field.
7) The capability to calculate percent of total for a field.
8) The capability to calculate coefficient of correlation and regression equation coefficients for a pair of variables.
9) The capability to assign a rank number for a specified field.
10) Linear Programming Capabilities.

4.4.6 Own Code

As its name indicates, own coding is written by the individual user or by an analyst to provide additional capability. Typically, this capability may be called for only at specified times (e.g., in conjunction with sort and collate programs) of the GDMS operation.

The capability to add own code may enhance the power of a GDMS. It can provide the means for modifying existing capabilities in a GDMS or for adding functions that do not exist in a GDMS. The capability to add own code should not be confused with the capability to modify the GDMS program or to perform other systems programming tasks. The own code capability provides for the addition of sub-routines and customizing the task specification. Although a powerful capability, the extent to which it is found useful or necessary may indicate a lack of GDMS design features that anticipate user needs.

The following points should be considered:

1) Can own code be added during or immediately after the desired function? Are linkage points or "own code exits" provided in the system? Does the system provide for compiling and loading the own code routines?
2) Language:
   - Languages available
   - Power
   - Ease of use
3) Ease of adding, deleting, or modifying own code
4) Size restrictions, modularity requirements, linkages

5) How added:

- Compiled at execute time
  - As part of GDMS
  - As subroutine in library
- Precompiled
  - As part of GDMS
  - As subroutine in Library
  - As independent subroutine

6) Effect on GDMS efficiency

7) Number of linkage points

4.4.7 Ease of Use

This parameter measures the simplicity of technique for defining processing tasks. Ease of use is evaluated primarily by consideration of the language features which contribute to the convenience or efficiency of the data definition task. Other factors which are included in measurement of this parameter are the requisite skill level of the practitioners and the ease which the techniques for problem definition are learned.

As noted in previous sections, the language considerations contributing to ease of use are determined by exception. Many language factors have already been noted in connection with specific capabilities. Those language features which are more descriptively classified as capability parameters are analyzed as such; others which are clearly intended primarily to reduce effort are considered as a part of Parameter IV.G.

The language features may show a close correspondence with the instruction repertoire; however, for more sophisticated GDMS languages it is likely that conventional programming is replaced by data-oriented procedural statements. In some cases, there may be several languages to evaluate ranging from the user-oriented dialog languages to conventional assembly and compiler languages for use by system programmers. The languages may be variously described as:

- Declarative (e.g., for definition)
- Command languages (goal-oriented intended to evoke action)
- User-oriented languages
- Procedural languages
- Problem-oriented languages

In the context of the GDMS functions, there may also be languages or language elements called:

- Data description languages
- Retrieval languages
- Query languages
- File update languages
- Report generation languages

In connection with this variety of language types and problem description flexibility, it may be appropriate to evaluate:

- Ability to intersperse declarative statements with retrieval requests and computations.
- Capability for user defined additions to the query languages. Ability to:
  - Make new operands from existing ones
  - Specify new procedures, callable by name

4.4.8 Performance

The evaluation of performance for the processing functions will be measured somewhat differently than for the previous sections. Man hours and the elapsed time for completion of a task are not measured here. It is assumed that this aspect of performance is already measured in performance parameters of the previous sections. The processing capabilities are typically integrated (at least procedurally) in one or more of the other functional areas (i.e., File Creation and Maintenance or Data Retrieval).

The performance factor to be measured here approximates the traditional and well developed techniques for measurement of computational power. It is, therefore, largely a hardware measurement and is primarily concerned with the capability of the Central Processing Unit (CPU) of the system.

Any of several well known computer performance measurement techniques may be used. Among these are:

1) The computation of a Figure of Merit
2) The bench mark technique

4.4.8.1 Figure of Merit. The Figure of Merit is commonly derived from a combination of various factors using a Figure of Merit equation based on assumptions which relate the relative importance of these factors. Examples of these factors are:

1) Access time
2) High speed memory capacity (usually a logarithmic function is used)
3) Word length (in bits)
4) Add time (representative of simple computations)
5) Multiply (representative of complex computations)

Figures of merit may be used to good effect to evaluate the relative power of competing central computers. However, the results usually may not be taken literally since a number of judgmental questions of importance which are not amenable to quantification in a figure of merit computation are likely to arise. In any case, although the results may be computed precisely, formulation of the equation is inherently a subjective determination of the importance of individual computer characteristics. For this reason, no particular figure of merit equation is recommended here. Further, the processing capability may not be entirely a function of CPU performance. In many cases the limiting factor is found elsewhere in the system (e.g., peripheral equipment in an I/O bound application, channel capacity, number of terminals, etc.). The resulting figure of merit if used for this parameter measurement would require a normalization step to convert the figure of merit to the standard scale.

4.4.8.2 Bench Mark Technique. This approach to measuring performance is problem-oriented. This technique is recommended as the appropriate method for evaluation assuming that realistic information is obtainable concerning the problem mix to be anticipated. It can be quite accurate but may be costly if undertaken in great detail. The competing systems are evaluated on the basis of their ability to perform selected problems or selected parts of problems. The problems may be real or simulated to approximate the real probl

4.4.8.3 Evaluation of Performance — Processing Functions. The inclusion of computational capabilities in a GDMS is, to some extent, outside the basic requirement. However, as computing systems have developed it is increasingly evident that the various capabilities of one type of system are often seen as useful and important adjuncts to a system of another type. An example of this is the Report Program Generator. Its original purpose was, as its title implies, to generate reports (or report programs); however, current specifications of this type of program call for the sufficient computing and processing capabilities to qualify the RPG as a complete programming method which, for some computing systems, may be the only programming method.

It should first be noted that performance is assumed to have been measured to a large extent by Parameters I. H, II. H, and III. H. The framing of bench mark problems may have already incorporated many of the processing capabilities (e.g., Sorting) considered in Section 4.4.3. However, after making allowance for the possibility of such overlap it should provide a useful measurement to evaluate the performance of those processing capabilities described in this section.

PARAMETER WORKSHEET

Parameter Group: IV. Processing

Parameter: IV. A Computation

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
IV. A Computation						
1. Operators						
a. Addition						
b. Subtraction						
c. Multiplication						
d. Division						
e. Exponentiation						
f. Trigonometric Functions						
g. Square root						
h. Boolean operators						
i. Other (list)						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: IV. Processing

Parameter: IV.A Computation (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
2. Operand Specification – The capability to specify as operands:						
a. Input transaction fields in file creation and maintenance						
b. Fields in master files (data sets or data banks)						
c. Retrieved data fields						
d. Constants						
e. System-supplied constants						
f. Results of prior computation						
g. Results of summation (summary fields)						
h. Other (list)						
3. Control						
a. Capability to iterate or repeat the execution of a processing statement						
b. Capability to specify the execution sequence of processing statements						
c. Capability to execute a processing statement based on the results of conditional comparisons of fields or of results of prior processing						
d. Capability to use the results of processing statements to establish new data fields						

NOTES:

PARAMETER WORKSHEET

Parameter Group: IV. Processing

Parameter: IV. B Summarization

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
IV. B Summarization						
1. Number of fields that can be totaled at any level						
2. Number of levels of totals and subtotals that can be accumulated						
3. Summarization of subtotal levels may be made conditioned on a change in value of a specified control field						

NOTES:

PARAMETER WORKSHEET

Parameter Group: IV. Processing

Parameter: IV. C Sorting

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
IV. C Sorting						
1. Source of Data to be sorted						
a. External Source (specify which)						
b. File Data						
2. Sorting Characteristics						
a. Number of sort keys						
b. Size of sort keys						
c. Order of Sort						
1) Ascending						
2) Descending						
3) Other specified sequence						
d. Operating Characteristics						
1) Automatic multi-pass merge						
2) Multi-reel one pass merge						
3) Internal Sort						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: IV. Processing

Parameter: IV.C Sorting (Cont'd.)

Date: **Evaluator:**

GDMS: **Data Source:**

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
2. Sorting Characteristics (Cont'd.)						
e. Sorting Methods						
1) N-way merge						
2) Cascade						
3) Poly phase						
4) Other						
3. Limitations of Sort Operation						
a. Size of file						
b. Number of tape units available						
c. Maximum record size						
d. Memory limitations						
e. Presort required (of external data)						
4. Specifications of Sort Operation						
a. Parameter driven sort routine						
b. Own code permitted						
c. Specification of source of sort data by user is permitted						
d. Other						

NOTES:

PARAMETER WORKSHEET

Parameter Group. IV. Processing

Parameter: IV. D Data Conversion

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
IV. D Data Conversion						
1. Number Conversion						
a. Binary to BCD						
b. BCD to Binary						
c. Fixed Point to Floating Point						
d. Floating Point to Fixed Point						
e. Other (list)						
2. Encoding/Decoding						
a. Table Look-up Method						
b. Computed Encoding						

NOTES:

PARAMETER WORKSHEET

Parameter Group: IV. Processing

Parameter: IV.E Statistical

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
IV.E Statistical						
1. The capability to find the maximum or minimum value of a field						
2. The capability to calculate the average value of a field						
a. Arithmetic mean						
b. Mode						
c. Median						
3. The capability to calculate a running average for a field						
4. The capability to calculate the standard deviation of a field						
5. The capability to count the number of entries of a field						
6. The capability to count the number of unique values of a field						
7. The capability to calculate percent of total for a field						
8. The capability to calculate coefficient of correlation and regression equation coefficients for a pair of variables						
9. The capability to assign a rank number for a specified field						
10. Linear Programming Capabilities						

NOTES:

PARAMETER WORKSHEET

Parameter Group: IV. Processing

Parameter: IV. F Own Code

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
IV. F Own Code						
1. Linkage points						
2. Own code languages available (list)						
3. Capabilities for modification of own code						
4. Compiled						
a. At execute time						
b. Precompiled						

NOTES:

PARAMETER WORKSHEET

Parameter Group: IV. Processing

Parameter: IV.G Ease of Use

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
IV.G Ease of Use						
1. Language Considerations						
a. Capability for user defined additions to the language						
b. Free form language characteristics						
2. Skill Level Required						
a. System specialist						
b. Programming specialist						
c. Other professional (specify)						
d. Clerical						
e. Other (specify)						
3. Ease of Learning						
a. Training required						
b. Number of practitioners						
c. Tutorial capabilities of system						

NOTES:

PARAMETER WORKSHEET

Parameter Group: IV. Processing

Parameter: IV. H Performance

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
IV. H Performance						
1. Figure of Merit Analysis						
a. Access Time						
b. High Speed Memory Capacity						
c. Word Length						
d. Add Time						
e. Multiply Time						
f. Other factors (list)						
2. Performance Measures						
a. Sample Computation						
b. Sorting Problem						
c. Statistical Problem						
d. BCD to Binary						

NOTES:

4.5 OUTPUT

The output capabilities and characteristics of a GDMS are considered to constitute a primary functional division of parameters. The treatment of this set of parameters differs from that accorded input considerations. From the user standpoint, input may be regarded as a means to an end and it has therefore been considered as a part of other functions (e.g., data definition – input media, data retrieval input, etc.). Output, however, is given greater emphasis since it constitutes the end product of the GDMS to the user.

The parameters considered in this section are divided into eight groups:

1) Formats
2) Formats – User Specified
3) Editing
4) Page Numbering and Control
5) Output Media
6) Capability to Prepare Input for Another System
7) Ease of Use
8) Performance

It is evident from this categorization that these parameters deal with form rather than content. The question of output content which may involve the degree of relevance of requested information, the completeness or accuracy of the information or, on the other hand, the presence of unneeded, confusing, or redundant information is not treated specifically here. The generai question which the user may wish to pose, "Did I get what I want?", as it relates to content, must be measured by consideration of the detailed logical selection capabilities contained elsewhere in the parameter organization (in particular, Parameter III. A). The output characteristics measured in this section, then, relate largely to matters of format and the mechanics of output preparation.

4.5.1 Formats

The preparation of output reports involves specifying the location of data on a page or display. The two major types of formats are columnar and graphical.

Columnar reports consist of lists of numeric or alphanumeric characters defined in a task specification; the output data can be file data or the results of processing. The capability to specify an exact arrangement of data depends on the range of options available for formatting. Output specifications can appear in data definition or in task specifications, or in both.

4.5.1.1 Report Format Capabilities – Automatic. In some systems an output can be requested with a minimum of specifications; in such cases, format features that are not detailed are automatically provided according to fixed rules. The capability to request output with a minimum of specifications should be evaluated in terms of the acceptability of the format produced. Examples of automatic features are:

- The capability to automatically calculate column width taking into account edit symbols (such as dollar signs), extra digits in totals, and column headings.
- The capability to adjust format to the width of form or to the output device, and to instruct the printer operator to mount proper paper.
- The capability to automatically select a field name or title from data definition for column headings.

- The capability to automatically select a report title from the task statement (e.g., task name).
- The capability to automatically apply edit and decoding specifications from data definitions.
- The capability to order columns across the page according to predetermined rules or implied order of the task specifications.

4.5.1.2 Headings. The capability to print descriptive information in addition to requested data contributes to the readability or a report. The capability may not be useful for the printing of simple lists such as a roster of names, but it can be most helpful in interpreting a multi-column numeric report.

Page and Report Headings

1) The capability to print the following should be evaluated:
   - Report titles
     - Multi-line titles.
     - Is title centered?
   - Page title
     - Multi-line titles.
     - Centered?
     - Repeated on every page?
   - Multi-line titles permit a description of the report, distribution list, or other information to be attached to a report to facilitate its handling and use.

Column Headings

The capability to supply and print column headings from the following sources should be considered:

- Name or number of the field being printed
- Title specified in data definition

- Title specified in output specifications
- Automatically supplied sequence number of the column

Trailers

The capability to print trailer information at the bottom of the page is sometimes required. Trailer information can consist of security classification, etc.

Other

Examples of other capabilities which may exist are:

1) The capability to print descriptive labels for totals and subtotals.
2) The capability to print a consecutive line number (ascending or descending sequence) for each line or group of lines in a report.

4.5.1.3 Graphical Output. Graphical output is prepared on devices with line drawing capability such as plotters and CRT's. Examples of the types of graphic which may be available are:

- Cartesian graphs
- Pie charts (other charts in circular coordinates)
- Bar charts (vertical and horizontal bars, Gantt charts)
- Time-series graphs
- PERT charts
- Maps and other pictorial

Graphical output capabilities that apply to one or more of the above types of output can be evaluated independently of the abilities of a system to produce the various types. Some of these component capabilities are:

- Media flexibility (e.g., 35 mm film, dry copy, full scale photo copy, CRT display, projection capability).
- Straight line capability.
- Curved line capability beyond concatenation of straight lines.
- Symbol generation (e.g., alphanumeric special symbols, pictorial building blocks).
- Scaling (from an input parameter, or from limits of data).
- Non-linear scales (e.g., logs, log-log, calendar dates).
- Background generation (grid pattern, map, etc.).

4.5.1.4 Audio Output

The preparation and sequencing of audio output can be thought of as a format function. If it is available in competing systems, its capability may be measured by vocabulary.

4.5.2 Formats – User Specified

Here, as noted elsewhere in the parameter organization, the evaluation may be dealing with two criteria which are at cross-purposes. These may be stated generally as:

1) The user may wish to have an automatic capability and thus be relieved of the necessity of specifying or describing to the system, or
2) The user may wish the capability to describe in a sensitive manner to the system exactly his needs or preferences.

Perhaps the optimum solution is to provide him with the option to choose and also a backup automatic capability if he prefers not to exercise this option.

The following list of capabilities which may be user specified should be rated somewhat higher if, in the absence of user specification, a standard assumption is made automatically:

- The capability to vary horizontal (line spacing) and vertical spacing (column spacing).
- The capability to specify left or right justification of data within field limits and/or column limits.
- The capability to print two or more lines of detail per item.
- The capability to fit volume output to pre-printed forms. This should include the capability to inhibit or modify such features as page heading, page numbering and column heading.
- The capability to specify the order of columns across the page.
- The capability to override edit and decoding specifications in data definition. Note that this covers the capability to override such specifications and does not evaluate edit or decoding as such. Edit capabilities are covered in Parameter V.C; decoding in Parameter IV.D.2.
- The capability to inhibit printing of selected information (e.g., a column of data).
- The capability to specify the output of detail data only, summary data only, or detail and summary data.
- The capability to prepare a spread sheet format.

The spread sheet format capability may be of several types. Typically, it involves the preparation of an array for a multi-valued field. Perhaps the simplest example is the capability to list detail fields for an item across the page in several columns rather than down the page in a single column. Another example is the capability to list detail values for a field in one of several columns, with each column denoting a range of values. A variation of this is the capability to tally or sum the values for a field according to specified ranges and to list the results (rather than the detail fields) across the page by column.

4.5.3 Editing

The capability to edit data on output facilitates the preparations of output reports and displays. Output editing of a field can be designated in a data definition or in output specifications. Editing generally involves the addition of characters such as dollar signs and the removal of characters such as leading zeros for the purpose of improving readability or appearance. Examples of output editing functions are:

1) Suppression of leading zeros
2) Floating plus and minus signs
3) Inserting of fixed or floating dollar signs (a floating dollar sign represents a significantly greater capability than a fixed dollar sign)
4) Substitution of asterisks for leading zeros (check protection)
5) Insertion of punctuation such as commas and decimal points
6) Insertion of slashes or hyphens, as in dates: 17/11/66
7) Substitution of "DR" and "CR" (or other debit and credit symbols) for plus and minus signs

4.5.4 Page Numbering and Control

The capabilities relating to page number and control are evaluated in this parameter. The following capabilities should be considered:

1) Control of page breaks depending on length of output form

   - Specified page lengths
   - Any page length

2) Automatic control over page numbering

   - Specifications of starting page number (page 1 or page XXX)
   - Specification of the location of page number (e.g., upper right corner, upper center, lower center of page, etc.)

3) Capability to break page on change of

   - Major key
   - Any key

4) Capability to print major key in title or heading and break page on change of key, and optional capability to renumber from page 1 on each such change of key.

5) Capability to limit volume of output on:

   - Number of lines
   - Number of pages
   - Number of highest or lowest values retrieved or calculated.

6) Capability to print reports on pre-printed forms.

In some systems, certain page numbering functions are performed automatically in the absence of special specifications. The usefulness of such automatic functions, if any, should be considered in the evaluation for this parameter.

4.5.5 Output Media

This parameter is used to rate the availability and adequacy of output media. Although hardware characteristics are not to be evaluated explicitly, input/output media are considered since they interface with the user. Input/output media will be discussed in general terms; the distinction between an output device (e.g., a high-speed printer) and an output medium (e.g., a printed page) need not be made for rating purposes.

The rating for a medium should not be confused with the importance of the medium or its capability relative to other media. For example, if paper tape and punched cards are being evaluated, the rating for each should reflect the adequacy of each method for the requirement, and should not be a rating of paper tape vs. punched cards.

4.5.5.1 <u>On-Line Presentation</u>. These media are used when the results of a task are to be presented immediately after or during the execution of the task. The device is on-line and interacts directly with a person who is interested in the data output.

1) Typewriter or low-speed printer
2) High-speed printer
3) Cathode ray tube display
4) Electronic display
5) Plotter
6) Audio output device

4.5.5.2 <u>Off-Line Presentation</u>. These media are used to prepare output that is not immediately used; although the device may be operating on-line, it is considered off-line in the sense that there is not direct interaction with the user of the output. Consider and rate the following:

1) Typewriter or low-speed printer
2) High-speed printer
3) Plotter
4) Photographic device

4.5.6 Capability to Prepare Input for Another System

This parameter measures the capability of a system to produce output which will be read as input by another system. Measurement of this parameter will depend on whether compatibility requirements are known or may be anticipated. If a known requirement for a specific output exists, then the capabilities for the exact media, formats, etc., can be evaluated. If a general requirement exists (i.e., it is anticipated that output will be prepared for other systems, but the exact requirements are unknown), then the broader the capabilities for output, the higher the rating for this parameter.

The following capabilities should be considered:

1) Output media.
   - Punched cards
   - Paper tape
   - Magnetic tape
   - Disk packs
   - Communication links
2) File labels, etc.
3) Blocking factor (number of logical records per physical record)
4) Record length and identification
   - Fixed or variable
   - Size
5) Physical organization of records
   - Sequential
   - Random
6) Record structure
   - Hierarchic levels
   - Identification of subrecords

- Variability of subrecord formats within a record
- Sequence of subrecords within a record

7) Location of data definition
   - In master file
   - In separate file

8) Data coding (alphanumeric, binary, etc.)

4.5.7 Ease of Use

This parameter measures the ease of use of the GDMS for output functions. This measurement is effected by a consideration of the same factors as for the other ease of use measurements:

- Language Considerations
- Skill Level Required
- Ease of Learning

The distinction between language capabilities intended to enhance the power and flexibility of the system and those which contribute to ease of is a difficult one to determine. Unquestionably, many of the items listed in the foregoing sections do contribute to the ease of use. However, rather than considering an extensive list of items which would overlap to a large extent with features already enumerated, a composite judgment of the GDMS language for output specification is a preferable approach to rating this parameter.

4.5.8 Performance

This parameter is a measurement of performance in respect primarily to reports preparation but may include performance of on-line display devices and terminal devices. As with the other performance parameters, measurements may be categorized under three headings:

1) Total man hours — time required for preparation of task preparation for typical report generation.

2) Response time for completion of selected output tasks.

3) Machine time for execution of selected output tasks.

The first consideration, above, may include an analysis of the time required for formulation of an on-line request for output. The latter two items depend to a large extent on the characteristics of the output media such as:

- Printing speed of the high-speed printer
- Typewriter output rate of on-line remote inquiry station
- Computer time required for display generation
- Display unit response characteristics

The efficiency of report generation may be affected to a large extent by the basic design considerations of the GDMS, e.g.

- Can a large number of different reports be generated by one pass of the data?
- Can the format of the report be changed by reformatting without repeating a retrieval pass?

It may be difficult to measure output performance independent of other performance considerations. In some cases a composite measurement of performance for two or more of the functional divisions of the parameter organization may be preferable. For example, performance evaluation of data retrieval and output (Section 4.3.8 and 4.5.8) could be undertaken by measurement of selected task preparation and execution times. This would be appropriate for systems for which these functions were inextricably integrated.

PARAMETER WORKSHEET

Parameter Group: V. Output

Parameter: V.A Formats

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
V.A Formats						
1. Report format capabilities - automatic						
a. Capability to automatically calculate column width taking into account edit symbols such as dollar signs, extra digits in totals, and column headings						
b. Capability to adjust format to the width of form or to the output device						
c. Capability to automatically select a field name or title from data definition for column headings						
d. Capability to automatically select a report title from the task statement (e.g., task name)						
e. Capability to automatically apply edit and decoding specifications from data definitions						
f. Capability to order columns across the page according to predetermined rules or implied order of the task specifications						
g. Capability for automatic generation of more than one report from one set of retrieved data						
Cont'd.						

NOTES:

PARAMETER WORKSHEET

Parameter Group: V. Output

Parameter: V.A Formats (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
2. Headings						
a. Page and report headings						
1) Report titles						
2) Page titles						
3) Dates						
4) Security classification						
5) Page numbers						
b. Column headings						
1) Name of field printed						
2) Number of the field printed						
3) Title specified in data definition						
4) Title specified in output specifications						
5) Automatically supplied sequence number						
c. Trailer information						
d. Other						
1) Descriptive labels						
2) Line number for each page						
Cont'd.						

NOTES:

PARAMETER WORKSHEET

Parameter Group: V. Output

Parameter: V.A Formats (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
3. Graphical output						
a. Types of graphic output						
1) Cartesian graphs						
2) Pie charts						
3) Bar charts						
4) Time-series graphs						
5) PERT charts						
6) Maps and other pictorial						
7) Polar coordinates						
8) Other (list)						
b. Media flexibility						
1) CRT display						
2) Projection capability						
3) Photo copy						
4) 35 mm film						
5) Plotter						
6) Other (list)						
Cont'd.						

NOTES:

PARAMETER WORKSHEET

Parameter Group: V. Output

Parameter: V. A Formats (cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
3. Graphical output (cont'd.)						
c. Graphical output capabilities						
1) Symbol generation (e.g., alpha-numeric special symbols, pictorial building blocks)						
2) Scaling (from an input parameter, or from limits of data)						
3) Non-linear scales (e.g., logs, log-log, calendar dates)						
4) Background generation (grid pattern, map, etc.)						
5) Straight line capability						
6) Curved line capability beyond concatenation of straight lines						
7) Other (list)						
4. Audio output						

NOTES:

PARAMETER WORKSHEET

Parameter Group: V. Output

Parameter: V. B Formats — User Specified

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
V. B Formats — User Specified						
1. Capability to vary horizontal (line spacing) and vertical spacing (column spacing)						
2. Capability to specify left or right justification of data within field limits and/or column limits						
3. Capability to print two or more lines of detail per item						
4. Capability to fit volume output to pre-printed forms						
5. Capability to inhibit or modify page heading page numbering and column heading						
6. Capability to specify the order of columns across the page						
7. Capability to override edit and decoding specifications in data definition						
8. Capability to specify the output of detail data only, summary data only, or detail and summary data						
9. Capability to prepare a spread sheet format						
10. Capability for specification of more than one report from one set of retrieved data						
11. Capability for user to add own code for output preparation						

NOTES:

PARAMETER WORKSHEET

Parameter Group: V. Output

Parameter: V.C Editing

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
V.C Editing						
1. Suppression of leading zeros						
2. Insertion of plus and minus signs						
a. Fixed						
b. Floating						
3. Insertion of dollar signs						
a. Fixed						
b. Floating						
4. Substitution of asterisks for leading zeros						
5. Insertion of punctuation						
a. Commas						
b. Decimal points						
c. Slashes						
d. Hyphens						
e. Other (list)						
6. Other (list)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: V. Output

Parameter: V. D Page Numbering and Control

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
V. D Page Numbering and Control						
1. Control of page breaks depending on length of output form						
a. For specified page lengths						
b. For any page length						
2. Page numbering control						
a. Starting page number may be specified						
b. Location of page number on the page may be specified						
3. Capability to break page on change of						
a. Major key						
b. Any key						
4. Capability to print major key in title or heading and break page on change of key						
5. Capability to renumber from page 1 on each change of key						
6. Capability to limit volume of output on:						
a. Number of lines						
b. Number of pages						
c. Values retrieved or calculated						
7. Capability to print reports on preprinted forms						

NOTES:

PARAMETER WORKSHEET

Parameter Group: V. Output

Parameter: V.E Output Media

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
V.E Output Media						
1. On-line presentation						
a. Typewriter or low-speed printer						
b. High speed printer						
c. Cathode ray tube display						
d. Electronic display						
e. Plotter						
f. Audio output device						
g. Other (list)						
2. Off-line presentation						
a. Typewriter or low-speed printer						
b. High-speed printer						
c. Plotter						
d. Photographic device						
e. Other (list)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: V. Output

Parameter: V.F Capability to Prepare Input for Another System

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
V.F Capability to Prepare Input for Another System						
1. Output media						
2. File labels, etc.						
3. Blocking factor						
4. Record length and identification						
a. Fixed or variable						
b. Compatability of record length						
5. Physical organization of records						
a. Sequential						
b. Random						
6. Record structure						
a. Hierarchic levels						
b. Identification of subrecords						
c. Variability of subrecord formats within a record						
d. Sequence of subrecords within a record						
7. Location of data definition						
a. In master file						
b. In separate file						
Cont'd.						

NOTES:

PARAMETER WORKSHEET

Parameter Group: V. Output

Parameter: V.F Capability to Prepare Input for Another System (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
8. Data coding						
a. Alphameric						
b. Binary						
c. Differences in a character set of "another system"						
d. Other						

NOTES:

PARAMETER WORKSHEET

Parameter Group: V. Output

Parameter: V.G Ease of Use

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
V.G Ease of Use						
1. Language consideration						
2. Skill level required						
a. System specialist						
b. Programming specialist						
c. Other professional (specify)						
d. Clerical						
e. Other (specify)						
3. Ease of learning						
a. Training required						
b. Number of practitioners						
c. Tutorial capabilities						

NOTES:

PARAMETER WORKSHEET

Parameter Group: V. Output

Parameter: V.H Performance

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
V.H Performance						
1. Total man hours						
a. Specification for report generation						
b. Keypunch						
c. Operation and support						
2. Response time						
a. Response time for batched or scheduled output						
b. On-line output response characteristics						
3. Machine time for report generation and other output functions (List selected sample jobs and record timing results. Attach subsidiary analysis sheets.)						

NOTES:

4.6 ENVIRONMENTAL CONSIDERATIONS

The sections to follow depart from the organization used for the first five sections. The parameters in this grouping describe system resources, both hardware and software, and general systems characteristics. Many of these parameters are environmental in nature — they do not measure the GDMS, per se, but, rather, other aspects of the anticipated operational environment which may be affected by the choice of GDMS.

The background considerations and application requirements cannot be anticipated in detail; therefore, the configuration of parameters for each evaluation may vary. Some of the considerations listed, therefore, should be regarded as optional. The determination of whether a factor should be included in the evaluation analysis depends primarily on whether it is a differentiating one in respect to the candidate systems (i.e., is it a GDMS evaluation variable). Parameter selection may be effected as a part of the weighting procedure; (e.g., by assigning zero weights to those parameters to be deleted). However, as a technical consideration, it should be noted that parameter selection is not entirely a matter of importance assessment. It is possible that a parameter is unquestionably important in an absolute sense (e.g., hardware configuration), yet, since it is a constant relative to both systems, inclusion of the parameter would serve no purpose.

The GDMS may be considered as having a pool of resources which may be categorized as hardware, software, and human. The categorization of parameters does not emphasize this approach since the user is interested in the end products of the GDMS rather than method of achieving these capabilities. However, if "system capability" in the absolute sense is the only criteria for evaluation, then considerations of efficiency and cost effectiveness would be ignored. It would be possible that a profligate expenditure of resources could achieve a capability which, while impressive, would represent a very poor investment of such resources.

Exhaustive analysis of the hardware and software components is not feasible or desirable. On the other hand, such a question as "Does the system appropriately and effectively utilize the potential of the hardware and other system resources?" is a proper topic for consideration.

The wide variation in evaluation criteria, the latitude for discretionary selection of parameters which is recommended, and the widely disparate applications which may be anticipated — these factors and others suggest that for some evaluations a hardware/software dichotomy may provide a useful framework for analysis — for others, the system capability will be considered only from an integrated standpoint.

For those evaluations for which it is appropriate to consider individually and successively both hardware characteristics and software characteristics, considerable emphasis may be placed on parameters in this section. In particular, Section 4.6.1 discusses the hardware aspects of the GDMS computer systems and Sections 4.6.2 through 4.6.5 describe a number of software considerations which are not analyzed in the foregoing functional parameter groupings.

4.6.1 Computer System Characteristics

Characteristically, computing systems are divided into five basic elements: (1) input, (2) arithmetic, (3) control, (4) memory, and (5) output. The arithmetic, control, and internal memory elements are sometimes referred to collectively as the Central Processing Unit (CPU).

However, the categorization of the computer system sub-parameters selected and described to follow depart slightly from that organization to one which is more significant in user terms. A more appropriate classification of this parameter is therefore:

1) Memory Hierarchy
2) Communications

Measurement of performance characteristics in the first five parameter groupings provide a considerable emphasis to the evaluation of CPU performance; therefore only the latter two items are detailed in this discussion.

4.6.1.1 Memory Hierarchy. There is an increasing use of memory hierarchies in computer and computer system design. This is evident not only in the proliferation of the numbers of memory units but also in the varying capabilities and capacities of the individual units. The usual method of categorization is according to speed, but memories may also be classified according to size (capacity), and capability (e.g., random access, read-only, content addressable, etc.). The ideal hierarchy is usually described as one with a fine gradation of speed and capacity characteristics, from small amounts of very high-speed storage (registers, scratchpad storage), through high-speed memory main storage, through successively large amounts of lower speed storage and finally, large amounts of bulk storage. Ideally, the steps from one type of storage to the next should be somewhat equal in magnitude. Although it is sometimes suggested that there are gaps in this continuum of memory elements (e.g., between drums and discs), it is seen that the array of memory devices to suit the speed/size/cost requirements of the user is fairly complete.

Although memory hierarchies are characterized mostly by speed and size, other aspects are also noteworthy. Specialization of memory functions, topological relationship of memories with processors, and control hierarchy are also important considerations.

A possible application of memory hierarchy technology is the analysis of the frequency of the access of data. This would involve the recording of statistics on data retrieval with the purpose of shifting data files to appropriate levels in the memory hierarchy depending on the frequency of use.

Automatic data transfers among the elements of a memory hierarchy is an interesting possibility. Ideally, the current operands of interest should reside in the fastest most accessible memory elements.

Transfers may, of course, be accomplished as a function of the computer program but this is not without expense in terms of programmer effort and/or computer time. If such transfers could be accomplished automatically on the basis of pre-selected criteria (e.g., frequency of use, recency of use, preset index of importance, etc.) the effective computer speed would be increased. Several algorithms are possible to evaluate the chosen criteria; however, only the most simple methods would be amenable to hardware mechanization at a cost commensurate with the resulting time saving.

A number of memory considerations are arrayed on Parameter Worksheet VI.A.1. The measurement of these considerations should depend on whether the competing GDMS's require considerably different memory and auxiliary storage configurations. The storage characteristics would then constitute an important GDMS variable. The sub-parameter organization as outlined on that form is arranged under three headings:

- Memory hierarchy characteristics
- Method of access
- Specific capabilities

The types of memory and storage are many and diverse. Very little emphasis need be given to any analysis of memory devices except in check-list fashion, and to note salient differences between GDMS's. The performance of this hardware is measured by performance parameters elsewhere and so a thorough analysis here would result in a redundant measure. The capacity and access times are listed for information purposes rather than to be rated individually, in order to be able to identify any outstanding features or deficiencies. It is noted additionally that access times do not tell the whole story as far as time efficiency is concerned. For example, for discs, it is necessary to consider many time factors (e.g., rotational delay, head positioning, transfer time, etc.) as well as probabilistic considerations (e.g., average delay to wait for beginning of selected track, inter-zone switches, etc.).

The method of access to data located in various stores in the memory hierarchy varies from the case of direct access in core stores to the use of complex algorithms used to retrieve data from discs, to associative techniques. For program storage some systems may provide paging hardware or other features to enable the operation of floatable programs.

A number of specific capabilities are listed under Parameter VI.A.1.c. One of the most significant of these is the memory protection feature which assumes great importance in a multi-user environment.

4.6.1.2 <u>Communications Consideration.</u> Communications factors constitute an important part of the GDMS. Since the system is assumed to be on-line and is in many cases a time-sharing system, communication characteristics may play a prominent part in effecting overall system capability.

In some cases the competing GDMS's may have the same general hardware configuration, the same types of communication characterisitcs. In this case, the considerations discussed here and listed on Parameter Worksheet VI.A.2, not be considered as GDMS variables.

Communication sub-parameters are classified on Parameter Worksheet IV.A.2 in four categories:

- Types of communication services
- Transmission codes
- Compatibility with data sets and terminal devices
- Specific communication capabilities

The numbers, types, and characteristics of terminal devices, not included in this parameter are considered in Section 4.5.5.

4.6.2 Operating System

The operating system consists of a set of supporting programs designed to control the computer as it proceeds sequentially through a string of jobs. It may also perform priority scheduling, I/O services, allocation of system resources, and monitor the overall operation of the computer system. In general, it synchronizes the system operation.

In some instances this may prove to be a very important GDMS variable. Comparative evaluations of compiler languages have shown in some cases that the supposed advantages of one language over another is due to the respective operating system characteristics.

Objectives for an operating system include:

- Minimize system idle time
- Minimize need for human intervention
- Provide an input/output interface for user programs
- Make optimum use of storage, processing, and peripheral resources
- Maximize thruput
- Minimize turnaround time
- Provide accounting for computer usage
- Sequence jobs
- Perform scheduling functions
- Multiprogramming and multiprocessing control

It should be noted that although typically an operating system would be considered a GDMS asset in some instances it may constitute a barrier between the user and the GDMS capability. It may also reduce performance. If the operating system overhead reduces the GDMS performance characteristics substantially or if it competes for limited system resources (e.g., internal memory) it could be regarded as a liability and would then be accorded a suitably low rating.

The capability to record information about system operations provides a basis for analysis of system activity. Examples of recording functions are:

1) Recording of system activity such as task identification, time of receipt, time of completion, processing time, waiting time, etc.

2) Recording of the number of input transactions, input data errors, and records selected, retrieved, modified, added, deleted, etc.

3) Recording of equipment errors and failures and action taken by system.

4) Recording of a record before and after any processing of that record to provide an audit trail.

5) Recording of operational status.

6) Recording of running time and user accounting.

Some of the functions listed above would ordinarily be accomplished by the monitor program (or operating system) — others by the GDMS programs. This distinction is not particularly significant to the user, but may dictate that the evaluator look to different sources for measurement of the various sub-parameters.

4.6.3 Available Programming Languages

The GDMS definition includes provision for a problem oriented language to be used for file maintenance and data retrieval. In addition, particular GDMS designs may include provision for special user oriented query languages. However, in addition to the language considerations discussed in Sections 4.1 - 4.5, there may be the capability for use of other procedure oriented, assembly or macro languages. This type of capability is construed here to constitute a form of system support.

The method of evaluation is to list those languages which are available and useful to the GDMS user. In some cases, the same languages would be available for use with either GDMS since availability of languages is usually a function of systems environment rather than GDMS design.

The languages listed on Worksheet VI. C are indicative of the types of languages which may be available. The list is open ended and all programming methods of potential use should be listed. It should be emphasized that this parameter is not a measurement of the primary GDMS programming methods or languages which are rated elsewhere in the parameter organization.

4.6.4 Interfaces with Other Systems

One of the environmental considerations which the evaluator may need to consider is the availability of other software systems which are colocated or are available via communication links which may be called upon to perform needed services or auxiliary processing. It can be argued that this is an independent consideration, not a function of GDMS design, and if this is an appropriate viewpoint in terms of the particular evaluation then this factor can be ignored. However, in some cases the auxiliary capabilities afforded by other systems may augment powerfully capabilities of a GDMS.

This has been considered briefly under the compatibility considerations in Sections 4.1.6 and 4.5.6 as related to input and output. However, a larger consideration is whether the unique or powerful capabilities obtained may be brought to bear on the GDMS operation. The question is not simply one of compatibility of data bases but rather of augmented capability which might be obtained through the combinative resources of two or more large systems.

The linking of systems is an important development and the trend toward making the particular capabilities of one software system available to other systems is evident.

The interrelationships between various types of software systems is a complicated subject. The definition of a GDMS is suggested in an earlier section of this report, and implied by the selection of parameters, may admit certain systems which are composites of subsystems and languages. For example, a data management system which qualifies only marginally as a GDMS may achieve powerful GDMS capabilities when linked to a time-sharing system. Thus, systems which are described variously as inquiry systems, data collection systems, management control systems, etc., may achieve GDMS status via marriage with another system or by addition of a generalized capability, a new programming language, or additional hardware facilities.

The measurement of this parameter must be specific to each evaluation and subparameters are therefore not listed. A listing of the systems which might be employable in the GDMS environment and the nature of the system interfaces could be made on the worksheet form. These cannot be anticipated here. In general, a subsidiary analysis would be required to evaluate this parameter appropriately. It would not necessarily have to be extensive, however, since the judgement of the combinative aspects of systems would be largely a subjective judgement, in any case.

4.6.5 Systems Support

Several different support functions may be available from the equipment vendor, the software vendor, or a military agency that specializes in these services. A number of such support activities are outlined to follow:

1) Assistance in implementation and use of the system
    - Problem analysis
    - Data preparation
    - Program preparation
    - Operations

- Maintenance
- Diagnostics

2) Modification of system software

- Writing and incorporation of routines to meet special requirements
- Incorporation of GDMS improvements
- Maintenance

3) Documentation

4) Training

This parameter is not intended as a measurement of personnel (measured by Parameter VI.G), but rather a measurement of support services and support programs.

4.6.5.1 Programming and Software Support. Assistance in implementation and use of the GDMS may vary substantially with respect to programming and software support. In general, this capability may be classified as:

- Support services
- Support programs

4.6.5.2 Documentation. The measurement of this parameter should relate to user values. Therefore, the documentation of primary interest may include:

- System Operational Description
- User Manuals
- Operating Manuals
- User Guides

Of less concern is programming documentation such as design specifications, program specifications, flow charts, etc. These are of importance for purposes of systems maintenance and other support activities but are less directly related to user support. In case system modification is required, however (almost always), it is increasingly important.

This parameter is both qualitative and quantiative. It should be rated on the basis of the number and kinds of useful support documentation and on the basis of a qualitative rating; the latter should be from a standpoint of utility (e.g., conciseness and quality of expression) rather than esthetic appeal.

One aspect of documentation relates to the GDMS evaluation as a whole. The evaluator should be alert to the potential danger of judging a system by the quality of its documentation. The main difficulty would be in other areas (e.g., file maintenance, language considerations, etc.). If the evaluator relies for much of his information in the form of system descriptions, language descriptions, or other systems documents, the quality of the documentation may result in bias in either direction. A positive bias might result from the favorable impression of good documentation. Conversely, negative bias might be accorded a system which had complete documentation in which shortcomings were clearly evident — as compared with a vague description with weaknesses hard to diagnose. For this reason, the evaluator should consciously strive to separate the "apparent" from the "real" by discounting the documentation variable in rating all other parameters; and then rate it separately in the appropriate context (that of the user, not the evaluator).

4.6.5.3 Training. The training facilities which are made available to help the user acquire proficiency in task specification or system operation are evaluated by Parameter VI.E.3.

This parameter, though closely related to the Ease of Learning parameter, is nonetheless distinct. Ease of learning, discussed under the major heading "Ease of Use", is measured as a function of the time and/or effort required for a person of the requisite skill level to attain proficiency. The training parameter is intended to measure any training facilities that will be avialable to the user. Examples of such facilities might include a formal training course, a self-teaching manual, or self-teaching computer program.

4.6.6 Installation Planning

Installation planning involves the organization and scheduling of human and machine resources in order to effect the establishing of a data processing facility. This can be a very involved process and is sometimes solved with the use of PERT or other critical-path methods. The evaluation of this parameter will require a subsidiary analysis unless the required information has already been prepared. This may be regarded as an optional parameter depending on whether the implementation of the candidate GDMS involves the installation of a new computing system.

This parameter, along with others in this grouping, somewhat removed from the central capabilities of a GDMS, may be voided by the evaluator by assigning a zero weight, if appropriate. The evaluation of two systems which anticipate usage of the same computer configuration would not require a measurement of this parameter, for example.

This parameter should be rated if it represents a consideration which would tend to differentiate between the candidate systems. The term installation planning usually refers to the installation of a computer system; however, it may also refer to other specialized equipment (e.g., installation of terminal equipment, etc.). The major reason for inclusion in the parameter list is to provide a means for evaluation in case a particular problem is found to exist in this area which would pose a limitation upon the GDMS performance. The primary problem which a lack of installation planning might occasion would be the prevention of the availability of the installation facilities at the required operational date.

The items shown on Worksheet VI. F are intended to be used only as a checklist in order to bring to light any possible sources of difficulty which might otherwise be overlooked. Therefore, it is probably not appropriate for the evaluator to try to evaluate each sub-item with an individual standard scale rating. In particular, it is seen that the time scheduling activities (VI. F.2) listed are not capabilities, nor alternate items reflecting levels of capability, but rather a chronological listing of salient milestones in the installation planning. The actual or estimated dates of completion could be entered in the third column of the worksheets for information purposes to be compared against requirements information in the first and second columns.

4.6.7 Personnel

The availability of necessary personnel to provide the required skills for GDMS operation is a consideration which may in some cases constitute a variable which should be rated. This would be the case particularly if it appeared to be a limiting factor (e.g., if lack of needed personnel would suggest a degradation of performance).

The personnel parameter is analyzed from three standpoints on Worksheet VI. G:

1) Skill levels and job description
2) Sources of personnel
3) Other personnel considerations

This parameter is closely related to other parameters in this grouping (particularly VI.E), and may be regarded as an optional parameter depending on the scope and emphasis of the particular evaluation.

4.6.8 General Systems Characteristics

The evaluation of military systems is conducted by measuring the system performance according to various selected criteria such as:

- Reliability
- Availability
- Expansibility
- Compatibility
- Adaptability
- Survivability

Although the structure of GDMS parameters has not emphasized these traditional virtues of tactical or strategical military systems, it is appropriate to permit the evaluator the opportunity to consider that GDMS in this context. Certain of those listed may not be appropriate (e.g., Survivability), and others have been treated to varying degrees of thoroughness by the functional parameters.

Evaluation of this parameter will be dependent on the particular requirement and only the most general considerations are detailed in the parameter worksheet. It should be emphasized that although relatively little can be delineated in detail before the applications requirements information is known, evaluation of parameters of this type is likely to be crucial and in some cases the overriding factor for evaluation.

4.6.8.1 Reliability. The reliability of a GDMS operation may be difficult to assess except as a subjective estimate. Certain numerical data may be available relating to equipment failure, MTBF (Mean Time Between Failures), and computer "down time" expressed as a percentage, etc. However, it is important to translate these into user terms. The capability to perform unfailingly and without delay an important management information retrieval task should be accorded a higher rating than similar performance of less vital tasks.

A number of reliability considerations have been detailed in the functional parameters (e.g., File Security, Parameter III. F).

Operation Under Non-Optimum Conditions

Unforeseen events such as equipment or program failure can result in partial loss of computer equipment and communications services. The availability of backup equipment and the provision for backup or "grandfather" data files, as well as the feasibility of operations with manual files and procedures should be evaluated. Alternate means of communications also should be evaluated.

Partial loss of equipment does not necessarily mean total loss of capability. Some systems are capable of "graceful degradation" and provide limited, but still useful, operations under certain types of conditions. This mode of operation is preferable to a total loss of operational capability.

Restart and Recovery

The provision for restart and recovery procedures can save computer-rerun time caused by halts due to interruptions, equipment errors, or equipment failure. This capability is of greater importance when execution times are long and rerun times become significant.

Checkpoint procedures provide for program restart without complete or excessive repetition of processing performed prior to an unscheduled halt. The lack of such a capability would require that a job be started from the beginning in the event of a run halt. The procedures should also provide protection for updated records in a file on a random access device when restart is initiated after a halt.

The capability to reconstruct a file from a backup file and a transaction file provides protection against partial or total loss of a file. File loss can result from equipment failure, operator error, etc., and can be serious if file reconstruction is difficult or impossible.

4.6.8.2 Modification, Expansion, or Conversion. Most systems of any size or complexity may be expected to undergo modifications. This is due both to changing requirements and to increased understanding of user needs.

Ease of conversion of the system for use on another computer can be significant if a change of computers is anticipated. The language used to program the system and the compatibility of the old and new computers are factors to consider.

The ease with which new storage and input/output devices can be incorporated into the system should be evaluated. The system should also be adaptable to both increase and decrease in the number of devices used.

4.6.8.3 Availability. Any system being evaluated must, of course, be available for use when required for an application. A system that is operational has certain advantages over a system that is still under development. The characteristics of the operational system can be observed or tested and the opportunity to run actual or benchmark problems may exist. Further, implementation experience may be available and programming brings may have been largely eliminated in the operational system.

Predicted or promised delivery dates for a new system may prove to be unreliable. Experience with existing GDMS's has shown that program production, expecially system integration can fall badly behind schedule. Even when interim reports of system design and implementation progress show the programming effort to be on schedule, the danger of delays during final checkout and system test are significant. Therefore, assumptions as to future availability of a GDMS should be conservative.

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI. A. Computer System Characteristics

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
VI. A Computer System Characteristics						
1. Memory hierarchy						
a. Memory hierarchy characteristics (List capacity and access time characteristics for each)						
1) Internal memory						
• Main memory						
• Control memory						
2) Auxiliary memory and bulk storage						
• Magnetic tape						
• Drum						
• Disc						
• Magnetic cards						
• Data cell						
• etc.						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI. A. Computer System Characteristics (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
1. Memory hierarchy (Cont'd.)						
b. Method of access						
1) Data						
• Sequential access						
• Indexed sequential						
• Random						
2) Programs						
• Paging hardware						
• Floatable programs						
c. Specific capabilities						
1) Memory protection						
• Hardware protect						
• Software protect						
• File protection						
• Program protection						
– User programs						
– System programs						
2) User capability to balance access time/storage tradeoff						
3) Memory relocation capabilities						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI. A. Computer System Characteristics (Cont'd.)

Date: Evaluator.

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
c. Specific capabilities (Cont'd.)						
4) Revising storage of data files in the memory hierarchy depending on frequency of use						
5) Communication between memories and processors						
• Between memories						
• Between memories and processors						
• Alternate paths						
• Flexibility of switching network						
2. Communications capabilities						
a. Types of communication services available (list transmission speeds and other characteristics)						
1) Voice grade – private line						
2) WATS						
3) TELEX						
4) TWX						
5) TWX Prime (TWX')						
6) Other (list)						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI. A. Computer System Characteristics (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
2. Communications capabilities (Cont'd.)						
b. Transmission codes						
1) Codes						
• ASCII						
• EBCDI						
• Five-level Baudot						
2) Code translation provided prior to entry in CPU						
c. Compatibility with data sets and terminal device						
1) Teletype						
2) Typewriter terminal						
3) Display consoles						
4) Remote printers						
5) CRT						
6) Other (list)						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI. A. Computer System Characteristics (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
2. Communications capabilities (Cont'd.)						
d. Specific communication capabilities						
1) Station-to-station communication off-line						
2) Communication control recognizes authorized users, and protects shared files or private files from unauthorized access						
3) Concurrent two-way communication						
4) Simultaneity of communications and CPU						
5) Buffered transmission						
6) Externally specified index						
7) Error checking and correction in transmission facilities						
• Validity checking						
– Parity						
– Hamming codes						
– Character legality check						
• Corrective procedures						
– Interrupt generated when error detected						
– Retransmission						
8) Capability for stacking transmission						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI. B. Operating System

Date: **Evaluator:**

GDMS: **Data Source:**

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
VI. B Operating System						
1. Allocation of resources						
Optimum use of:						
a. Memory (space)						
b. CPU (time)						
c. Peripheral equipment						
d. I/O control						
1) Channel						
2) Device						
2. Monitor control — control by:						
a. Card						
b. Console						
c. Other (list)						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI. B. Operating System (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
3. Specific capabilities for:						
a. Single job runs without a monitor						
b. Job stacking under a monitor						
c. Time sharing operation						
d. Multi-programming						
e. Multi-processing						
f. Control of foreground/background requirements						
g. Priority control						
h. Dynamic memory allocation						
i. Automatic paging						
j. Other capabilities requiring further consideration						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI. B. Operating System (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
4. Recording						
a. GDMS recording of:						
1) Task identifications						
2) Number of input transactions						
3) Number of input data errors						
4) Records selected, retrieved, modified or deleted						
5) Input errors						
• Task specification errors						
• Data errors						
6) A record before and after processing						
b. Logging of:						
1) Run progress						
2) Running times						
3) Operational status						
4) Operator intervention						
5) User accounting information						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI. C. Available Programming Languages

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
VI. C Available Programming Languages						
1. Compiler languages						
a. JOVIAL						
b. FORTRAN						
c. COBOL						
d. PL-1						
e. Other (list)						
2. Assembly languages (list)						
3. RPG						
4. Macro library						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI. D. Interfaces with Other Systems

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
VI. D Interfaces with Other Systems* (List system and characteristics and capabilities of each.)						
1. Ability of GDMS to operate under the monitor currently in use						
2. Ability to communicate with a teleprocessing system						
3. Ability to operate as a time-shared job						
4. Ability to operate as a segment in a multiprogramming mode						
*This parameter is considered only if it is a differentiating factor between competing systems.						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI. E. System Support

Date: **Evaluator:**

GDMS: **Data Source:**

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
VI. E System Support						
1. Programming and software support						
a. Services						
1) Problem analysis						
2) Data preparation						
3) Program (task specification) preparation						
4) Operations assistance						
5) Program maintenance						
6) Incorporation of GDMS improvements						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI. E. System Support (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
1. Programming and software support (Cont'd.)						
b. Programs						
1) Utilities programs (list)						
2) Diagnostic programs						
• Trace programs						
• Selective file dump						
• Dynamic dump						
• Post-mortem dump						
• Other (list)						
3) Other support programs (list)						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI. E. System Support (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
2. Documentation						
a. Types of documentation						
1) Users manual						
2) Language description						
3) Functional specifications						
4) Systems operating description						
5) Systems design description						
6) Program design description						
7) Operating manual						
8) Maintenance manual						
9) Programming documentation						
• Program specification						
• Flow charts						
• Annotated program listings						
10) Training manuals*						
11) Others (list)						
b. Quality of overall documentation						
*Training Manuals may be rated alternately under Parameter VI. E. 3, Training						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI. E. System Support (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
3. Training						
a. Training courses (list duration)						
1) User						
2) Operators						
3) Programmers						
4) Other (list)						
b. Self teaching training manual						
c. On-the-job training						
d. Self-teaching computer program						
e. (Other)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI. F. Installation Planning

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
VI. F Installation Planning						
1. Support services for installation						
a. Consultants required						
1) Data processing						
2) Electrical						
3) Mechanical						
4) Architectural						
5) Other (list)						
b. Provided:						
1) Directly by computing system vendor						
2) Outside installation service must be solicited						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI. F. Installation Planning (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
2. Time scheduling (planning functions – list dates)						
a. Layouts approved						
b. Equipment on order						
c. Electrical and mechanical phases complete						
d. Delivery of equipment						
e. Checkout of equipment						
f. Equipment operation						
3. Installation service (maintenance) after installation						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI. G. Personnel

Date:

Evaluator:

GDMS:

Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
VI. G Personnel						
1. Skill levels and job descriptions						
a. Supervisory						
b. Senior system analyst						
c. Systems analyst						
d. Senior programmer						
e. Applications analyst						
f. Programmer						
g. Console operators						
h. Clerical, keypunch, etc.						
2. Sources of personnal						
a. Employees						
b. Military personnel						
c. Contractor (list by type and availability)						
d. Civil service personnel						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI.G. Personnel (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
3. Other personnel considerations						
a. Turnover factor						
b. Performance levels of personnel						
c. Management personnel ratio						
d. Organizational characteristics						
1) Functional organization						
2) Line organization						
e. Administrative services						
f. Recruitment and training potential						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI. H. General System Characteristics

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
VI. H General System Characteristics						
1. Reliability						
a. MTBF (Mean Time Between Failures) data (list equipment and pertinent statistics)						
b. Operation under non-optimum conditions						
1) Graceful degradation and fail soft capabilities (list features which contribute to these capabilities)						
2) "Back-up" provisions (e.g., for disc crashes, saving old master files) (list)						
c. Restart and recovery						
2. Modification, expansion, or conversion						
a. User capability to "step up" to larger system without large investment in:						
1) Equipment acquisitions						
2) Reprogramming						
(Cont'd.)						

NOTES:

PARAMETER WORKSHEET

Parameter Group: VI. Environmental Considerations

Parameter: VI.H. General System Characteristics (Cont'd.)

Date: Evaluator:

GDMS: Data Source:

PARAMETER DESCRIPTION	APPLICATIONS REQUIREMENTS		SYSTEM OBSERVATIONS		COMPUTATION OF SCORE	
	REQ.	DES.	OBS.	RATING	WEIGHT	SCORE
2. Modification, expansion, or conversion (Cont'd.) b. Modularity features which contribute to: 1) Ease with which GDMS can be expanded 2) Ease of conversion to another computer 3) Ease of addition of new storage devices 4) Expandability of program functions 3. Availability (List delivery dates for equipment and milestone dates for software development. Note contingency factors which may affect delivery.)						

NOTES:

Section V

OTHER APPROACHES

5.1 DEVELOPMENT OF PARAMETER LIST

The selection, development, and organization of parameters were major tasks of the study and were discussed in Section 3.1. Some background material that did not appear in Section 3.1 is presented here.

5.1.1 General Characteristics

The number of parameters, the number of hierarchical levels, the balance (e.g., number of major groupings, number of parameters per grouping, etc.), and the consistency (e.g., level of rating, rating subparameters individually or collectively, etc.) were not problems as such, but were nonetheless subject to specification by the project team. For a given amount of effort, it would have been possible to develop a small number of parameters described in great detail or to formulate a large number of parameters with only brief discussion; neither objective seemed superior. It was decided not to pre-specify (or force) these characteristics of the parameter list, and to permit them to develop naturally during the course of the study.

5.1.2 Major Parameter Groupings

The various interim lists of parameters that were developed and modified during the study are too long to present here. The evolution of the major groupings, however, can be discussed. The parameter list in the statement of work was, of course, the starting point for the development of the major groupings. Many basic organizations were considered, and it became apparent that the major functions of a GDMS should be incorporated into the parameter list. Three examples of organizations of major GDMS functions follow.

Richard G. Canning* suggests that file management systems consist of five basic functions:

1) File Creation
2) File Maintenance
3) Select-extract
4) Sort
5) Report

His estimate (as of October 1965) of what generalized file processing systems will look like in the future is:

1) Major Functions
   - Edit
   - Update
   - Select-extract
   - Sort/Merge
   - Report

2) Minor Functions
   - Accumulate
   - Compute
   - Code Conversion
   - Field Validity
   - File Search
   - Logical Selection
   - Own Code
   - Sequence Check
   - Summarize
   - Table Build/Change
   - Table Lookup

---

*"Generalized Processing Software," <u>EDP Analyzer</u>, October 1965, p. 4, 8-13.

3) Data Definitions

4) Operating Environment

- Operating System
- Multiprogramming

The developers of GIS (Generalized Information System)* describe its major functions as being:

- Data file design and creation
- File maintenance
- Selective retrieval and processing
- Document reference and full text indexing
- Control of task processing

The five basic functions of file management, according to John A. Postley,** are:

1) File creation and maintenance
2) Information retrieval
3) Report preparation
4) Sorting
5) Processing

As can be observed from the foregoing small sample, descriptions of the major functions of a GDMS can take on various forms. In addition, a comprehensive evaluation should include supporting software, hardware, and other environmental considerations. As a result, the final list of parameters consist of the detailed functions of a GDMS as well as many other areas that are related to the performance and availability of a GDMS but that are not functions of a GDMS. Examples of two intermediate groupings and the final organization are shown on Table 3.

---

*Bryant, J. H. and Semple, Parlan, Jr., "GIS and File Management," <u>Proceedings of the 21st National Conference</u>, ACM, 1966, p. 97.

**Postley, John A., "File Management Applications," <u>DPMA Quarterly</u>, July 1966, p. 22

Table 3.

PARAMETER GROUPINGS
Early List
Data base structure capability
Human and machine effort in data base preparation
On-line file maintenance capabilities
Batch process file maintenance capabilities
Efficiency of the file maintenance function
Capability of information retrieval and reporting systems
Ad hoc query capability
Output capabilities
System availability and growth capabilities
Costs
Input/output monitor capability
Supervisory system capability
Interim List
File structure/definition/organization
File generation
File maintenance
Queries
Processing
Output
Other
Final List
Data Definition and Data Organization
File Creation and Maintenance
Retrieval
Processing
Output
Environmental Considerations

5.1.3 Software Systems Surveyed

The parameter list was developed primarily from a GDMS capability point of view rather than from a requirements viewpoint. This was done since the description of GDMS capabilities was considered more tractable than trying to define a broad range of requirements. In order to consider the capabilities of various GDMSs and related systems, a survey of such systems was made. These systems were examined during the course of study in varying degrees of detail depending upon the documentation available. Although some of the systems are GDMSs, many are not. All, however, have some features or capabilities, such as file management and on-line query and display, which pertain to the study. See Table 4 for a list of systems surveyed. Examples of classifications and surveys of various systems can be seen in several documents.*

---

*(1) Advanced Programming Developments: A Survey, ESD-TR-65-171, Electronic Systems Division and Computer Associates Inc., February 1965.

(2) Report on Initial Planning for GENISYS: (Generalized Information System), ESD-TR-65-463, System Development Corp., July 1965, p. 79.

(3) "Generalizing File Processing Software," EDP Analyzer, October 1965.

Table 4.

SOFTWARE SYSTEMS SURVEYED		
Short Name	Full Name or Description	Developer
ADAM	Automated Data Management	MITRE Corp.
COLINGO-D	Compile On-Line and Go	MITRE Corp.
COLINGO-C-10	Compile On-Line and Go	MITRE Corp.
DEACON	Direct English Access and CONtrol	GE-TEMPO
DOCUS	Display Oriented Compiler Usage System	Informatics Inc.
GENISYS	GENeralized Information SYStem	SDC
GIS	Generalized Information System	IBM
GPDS	General Purpose Display System	SDC
---	Hospital Computer Project (a time-shared, remote access system)	Bolt Beranek & Newman Inc.
IDS	Integrated Data Store	General Electric
INTIPS	Integrated Information Processing System	RADC/Informatics
JOSS	Johnnaic Open Shop System	RAND Corp.
MANAGE	Generalized File Management System	SDS
Mark III	File Management System	Informatics Inc.
Mark IV	File Management System	Informatics Inc.
RPG	Report Program Generator (System 360)	IBM
TDMS	Time-Shared Data Management System	SDC
TSS-LUCID	Time-Shared Language Used to Communicate System Design	SDC

5.2 WEIGHTING

Several approaches for weighting were considered: two were developed in detail and are the main subject of this section. Some of the other approaches will be described in brief.

The derivation of weights from a ranking procedure was considered since it seemed to offer the advantage of simplicity. Parameters would be ranked and the ranking would be in effect an indirect assignment of weights. The conversion of ranks to weights, however, using a fixed formula means that the evaluator is weighting by ranking without knowing what the actual weights are. This is considered less desirable than a direct weighting method. Further, the ranking of a large number of parameters can be a cumbersome process, even with a provision for group ranking. Since a rank indicates only the relative order of importance, it does not reflect the relative degree of importance. The conversion of a rank to a weight, therefore, requires arbitrary, predetermined rules for assigning a weight that reflects degree as well as order of importance. In sum, the use of ranking offered few advantages while adding a conversion step in the evaluation procedure.

The notion of using monetary units such as dollars for weighting was considered. This idea was based on the premise that the evaluator (or user) would estimate the dollar value of each parameter. The dollar values would then be reduced to percentages of the total estimated value and the resultant percentages are used as weights. The use of dollars was considered since it might add realism to the weighting exercise, and it would provide a means for different users to reflect their estimates of importance of the capabilities that they used. This method was not used because the estimation of the value of each detailed parameter appeared to be impractical, and the use of value in this context was not necessarily a good measure of relative importance.

Personnel and marketing rating and weighting methods were analyzed in the hope of finding ways to remove personal biases and preferences. Much of what was surveyed was not directly applicable. However, some use was made of personnel rating schemes in developing the rating method in the study.

5.2.1 Confidence Factors

In Section 2.2, the use of weights to reflect confidence levels was rejected. A more detailed treatment of this subject follows. Portions of earlier text are repeated verbatim for comprehensiveness.

The degree of importance which is attached to a parameter is referred to as a weight. A basic assumption of the evaluation method is that weights are expressed numerically and will be used to adjust the parameter rating according to relative importance judgements.

The weights to be applied are subjective estimates of the importance of the parameter, sub-parameter, or group of parameters under consideration. There are other interpretations, definitions, and usages of the term "weight" than to signify relative importance, however. For example, for certain mathematical and statistical problems, it is appropriate to use weights as a measure of the confidence or reliability of an estimate or of a sample variable. This concept is worthy of consideration for our evaluation technique. It would be a conservative, and possibly more accurate, procedure for the overall method for the evaluator to assign a relatively low weight to a parameter for which he has little confidence in the basis of judgement or for parameters for which the judgement is strictly a matter of (perhaps contentious or divided) opinion.

To summarize this problem, it is possible that an important parameter could be rated with a low degree of assurance that the rating is accurate. In such a case, a high weight could cause a considerable distortion to the overall evaluation if the rating was indeed erroneous.

In general, there are several approaches to the problem of the confidence or reliability of the evaluator's estimates.

1) A confidence factor (in effect, an error estimate) could be introduced to reflect the degree of reliance the evaluator has in his own estimate. (This factor could also be introduced by someone other than the evaluator.)

2) The "confidence" consideration could be evaluated as a part of the weighting process. In this case, the "weight" would have a composite meaning of "importance" and "reliability of estimate."

3) The "confidence" factor could be disregarded entirely on the assumption that the educated opinion of the evaluator, even if only reflected as a slight inclination toward one position or another is likely to be better than "no opinion."

Computation of a Confidence Factor

A computation of a score for a particular parameter could be of the form:

$$R \times W_1 \times W_2 = S$$

where R is the rating, W_1 the Importance Weight, W_2 the Confidence Level Weight, and S the computed score. The steps relating to evaluation of each of the factors could be paraphrased by the evaluator as:

1) How well does the DGMS perform in respect to Parameter X?

2) How important is Parameter X relative to all other parameters?

3) How well founded are my estimates of a and b?

The introduction of a confidence factor is theoretically justified, since the evaluator should pass judgement on the system elements in only those areas for which he has sufficient information and knowledge. However, weighting confidence levels as a procedural step in the evaluation method is not recommended.

The Technique, as developed in the current study, is not sufficiently precise to permit the addition of yet another subjective measurement. The simplifying assumption is made that the factor, if a sufficiently compelling consideration, will be treated subjectively by the evaluator as a part of the weighting process and not as a separate procedural step in the evaluation.

5.2.2 Mechanics

The procedures for weighting a rating and for aggregating a total system score are interrelated and are logically developed as a combined subject. Although several methods were considered, two general techniques were investigated in detail for weighting parameters and accumulating system scores. These are described in this section and illustrated in Tables 1 and 5. These methods involve the mechanics for arriving at a total system score by totaling weighted scores of the system parameters; they do not address the problem of methodology for determining weights for specific parameters.

An analysis of both methods is of sufficient interest to warrant presentation in the report. Method A, which uses percentage weights, is the technique described in Section 2.2. Most of the details of this method are repeated verbatim in this section for comparison purposes with Method B, which uses integral weights. Ratings, scores, etc., have been defined in previous sections and will not be repeated here.

5.2.2.1 Method A: Weighting Method Using Percentage Weights — This method was described in detail in Section 2.2. In this method, weights denoting the relative importance of parameters are apportioned on a percentage basis. The total weight for each group of related parameters at a given hierarchical level is unity. The weighting can be accomplished in any order: bottom-up, top-down, or inside-out. When the ratings and weights are extended and summed from the bottom up, the resulting score for each hierarchical level retains the value significance of the standard scale. (See Table 1.)

5.2.2.2 Method B: Weighting Method Using Integral Weights — This method is based on the assignment of integral values to the parameters on a relative basis throughout the list. Since the smallest weight permitted is a "1" and such a weight will be given only to the least significant items, it is appropriate to effect this method in a "bottom-up" order.

The method may be described as composed of the following steps:

1) By inspection, the analyst will select those item(s) which he regards as the least significant and assign a weight of "1" to it (them).

2) Proceeding in any convenient order, he will assign weights for the more important items. (The most probable order would be to proceed to associated items, i.e., those which could be compared more readily with the parameters already weighted.)

3) Frequent cross checks between parameters at both local and remote sections of the parameter list are made to assure consistency. Adjustments may be made to improve the balance of the weights throughout the list.

4) The weights are aggregated and totals carried forward. Comparisons are made between the totals of the parameter groups to assure that they correctly reflect their relative importance.

Table 5. Weighting Method "B": Integral Weights and Cumulative Computation (Sheet 1 of 2)

Line (1)	Parameter Hierarchy (2)	Rating (3)	WEIGHTS II.C.2.a, etc. (4)	I.A.1, etc. (5)	I.A, I.B, etc. (6)	I, II, III (7)	Percent of Total (8)	Weighted Scores (9)
1	I					40	50	
2	A				28		35	
3	1	7		16			20	112
4	2	9		12			15	108
5				28				
6	B				12		15	
7	1	6		6			7.5	36
8	2	4		1			1.25	4
9	3	2		1			1.25	2
10	4	9		4			5	36
11				12				
12					40			
13	II					24	30	
14	A	9			4		5	36
15	B				6		7.5	
16	1	9		5			6.25	45
17	2	5		1			1.25	5
18				6				
19	C				14		17.5	
20	1	6		6			7.5	36
21	2			4			5	
22	a	4	1				1.25	4
23	b	9	2				2.5	18
24	c	7	1				1.25	7
25			4					
26	3	8		3			3.75	24
27	4	7		1			1.25	7
28				14				
29					24			
30	III					16	20	
31	A				8		10	
32	1	10		4			5	40
33	2	0		1			1.25	-
34	3	10		1			1.25	10
35	4	10		2			2.5	20
36				8				
37	B	9			4		5	36
38	C	4			2		2.5	8
39	D				2		2.5	
40	1	9		1			1.25	9
41	2	5		1			1.25	5
42				2				
43					16			
44						80		608

Table 5. Weighting Method "B": Integral Weights and Cumulative Computation (Sheet 2 of 2)

Line	Method of Computation Used by GDMS Evaluator											
	Eval. of II.C.2			Eval. of I.A, etc.			Eval. of I, II, III			Norm. Scores	System Score	Norm. Scores
	Rating	Weight	Score	Rating	Weight	Score	Rating	Weight	Score			
(10)	(11)	(12)	(13)	(14)	(15)	(16)	(17)	(18)	(19)	(20)	(21)	(22)
1											298	7.5
2									220	7.9		
3				7	16	112						
4				9	12	108						
5				Total I.A		220						
6									78	6.5		
7				6	6	36						
8				4	1	4						
9				2	1	2						
10				9	4	36						
11				Total I.B		78						
12							Total I		298	7.5		
13											182	8.4
14							9	4	36			
15									50	8.3		
16				9	5	45						
17				5	1	5						
18				Total II.B		50						
19									96	6.9		
20				6	6	36						
21						29						
22	4	1	4									
23	9	2	18									
24	7	1	7									
25	Total II.C.2		29									
26				8	3	24						
27				7	1	7						
28				Total II.C		96						
29							Total II		182			
30											128	8.0
31									70	8.8		
32				10	4	40						
33				0	1							
34				10	1	10						
35				10	2	20						
36				Total III.A		70						
37							9	4	36			
38							4	2	8			
39									14	7.0		
40				9	1	9						
41				5	1	5						
42				Total III.D		14						
43							Total III		128			
44									System Total		608	7.6

5) After the parameters have been "sized" by an initial weighting process, as described in steps 1 through 4, the process may be iterated to the satisfaction of the analyst. Although the initial weighting will be undertaken in a "bottom-up" sequence, the iteration may be in either "bottom-up" or "top-down" order.

The weights for the sample hierarchy for Method B, shown in Table 5, were selected to bear the same relative importance as for Method A. (It is noted that III. D. 2 was (Line 41) of such slight significance for Method B that it was modified (rounded up) to conform with its sibling parameter and was given a weight of 1. This illustrates one of the disadvantages of using integrals. If the least important item is extremely insignificant and is given a weight of "1", the most important items may be accorded weights so large as to be unwieldy.

It is assumed that the weights shown in Columns 4-7 were derived by the procedure outlined in the 5 steps shown above. Column 8 shows the value of each selected weight expressed as a percentage of the system total. The weighted score for each rated parameter is shown in Column 9. The total of 608 has no "absolute" significance. It will have a relative significance to a total computed for another GDMS. It is noted that for Method B there is no attempt to predetermine either the total of weights for the parameters or the total score.

The method of computation of scores at each parameter level is shown in more detail in Columns 11-22. Subtotals derived using Method A have a significance which may be interpreted on the standard rating scale (e.g., a subtotal of 8.64 would be interpreted as a good score). For Method B no absolute significance can be attached to any total. However, such a score may be derived by dividing the computed score by the total of the weights which were used to compute that score. For example, a score (called here a normalized score) of 7.9 was computed for Parameter I. A (see Colume 20, line 2) by dividing the accumulated score for I. A of 220 by the total weights, 28. Normalized scores are shown in Columns 20 and 22.

The advantages and disadvantages for both Method A and Method B are described as follow:

5.2.2.3 Advantages and Disadvantages

Method A — Advantages

- The weighting may be undertaken in any order. (Bottom-up or top-down)
- Any set of sibling parameters may be weighted independently, without consideration of other weights assigned or to be assigned.
- Computed scores at all hierarchical levels retain a numerical significance compatible with the grading assumptions used for rating on the standard scale.

Method A — Disadvantages

- The use of decimal values may appear more complex and suggest a precision which is not matched by genuine accuracy (the precision is certainly greater than the accuracy).
- Weights may not readily be compared between remote parts of the parameter list without a conversion process to determine the system weight (as shown in Column 9, Exhibit A).

Method B — Advantages

- Use of integers gives appearance of greater simplicity and a slight advantage in computation ease.
- Weights at all levels may be compared (e.g., the weight for II.C.2.b is readily seen to be equivalent to that of III.A.4).
- The effect in any weight change would be apparent in terms of total system score (e.g., a change of the weighting of II.A from 4 to 6 would easily be seen to affect the total score given the system by 18 points (9x2). However, this is true only if rating has already been done. Weighting will ordinarily precede rating.

Method B — Disadvantages

- The use of integers only may cause distortion due to the use of discrete rather than continuous weighting levels. This was noted in the discussion of Method B.
- The order of weighting must initially proceed from the consideration of the least important parameters (in order to fix a certain level of parameter detail with a weight of "1".

In general, it is seen that the disadvantages of Method B are contra of the advantages of Method A and vice versa. It is further noted that as these methods were refined they tended to be more nearly alike. The computations of the system weights of Method A (Column 9) are designed to provide a comparative measure similar to that obtained for weights of Method B. Similarly, the normalized scores shown in Columns 20 and 22 illustrate a method to convert the scores obtained in Method B to the same basis as those obtained for Method A.

The two methods are computationally similar, if not identical, but involve procedural differences of significance.

Conclusions

The method adopted by the project team is Method A. The primary reasons for this selection are indicated in the foregoing discussion. To summarize these in different language, we have concluded that:

- By weighting on a "local" level, i.e., by weighting sibling parameters, the need for the evaluator to adjust or manipulate totals is removed. In Method A, the evaluation has one consideration: What is the relative importance of these sub-items as expressed in percentages? For Method B, the problem is compounded. In addition to the consideration of relative importance, the evaluator must take care that as he selects weights for individual parameters his total for that parameter group is neither over- nor understated.

An example, using the model hierarchy, illustrates this point. In Table 5, the evaluator, in considering Parameter I. B. 4, may decide that a weight of 4 is appropriate, but note at the same time that the total of I. B. will then be 12. He may feel, however, that it should be 14 (perhaps to reflect one-half of the weight of the weight of the I. A parameter — 28). He may be unable to adjust the weights of I. B upwards or of I. A downwards to obtain the desired rates. By contrast, it is seen that for Method A, the total of any sibling group is 100% (e.g., the 0.70-0.30 ratio of I. A and I. B could be readily adjusted to 0.65-0.35 without affecting any other parameter group.

- A second advantage, which has motivated the selection of Method A, is the retention of normalized scores at each hierarchical level (parameter grouping, parameter, and sub-parameter). For Method A, the total for II. B is shown to be 8.6, connoting a "good" score. The recognizable significance of these scores may provide a reasonableness check for the evaluator at all levels. This is not true of the score of "50" derived for II. B in Method B.

- A final reason for the selection is that the parameters may be weighted in any order and any sibling group may be weighted independently. This may have particular significance if the weighting is to be undertaken in more than one procedural step, or by more than one evaluator. Partition of the weighting tasks among analysts according to particular background knowledge, expertise, or specialty is facilitated.

5.3 RATING METHODS

Several different rating approaches were explored during the course of the study; some of these will be described in this section. In all of the approaches, it is assumed that measurements of capability have been obtained for each of two GDMSs (System A and System B), and that the approach described is used to obtain a rating. Except where noted, these ratings can be weighted and accumulated using the techniques presented in the body of the report.

The advantages and disadvantages of each approach are summarized in Table 7. Items of special importance are discussed with each approach.

5.3.1 Approach "W": Capability Ratios

In this approach, the capability measurement for System A is divided by the capability measurement for System B. The resultant ratio is considered to be a measurement of effectiveness of System A relative to System B. The ratio is computed for each parameter evaluated: hypothetical examples are shown in Table 6.

Comment

- The computation of a meaningful ratio is impossible when the capability for either system is 0.
- Some ratios are likely to be very large or very small and will distort an overall score.
- Using the system (whether A or B) with the "best" capability as the divisor in the ratio, and computing the ratio for each system, will result in ratios that fall between 0 and 1. Although this variation is, in a sense, normalization, and is somewhat of an improvement over the single A ÷ B ratio approach, it still is not as good as the other approaches.

Table 6. Examples of Ratings for Various Approaches

Parameter**	Requirement		Capability		Ratings					
	Minimum	Desired	Syst. A	Syst. B	"W"	"X"	"Y"		"Z"	
					A/B	A-B	A	B	A	B
1. Number of input devices	10	20	10	10	0.56	-2	50%	90%	3	8
2. Maximum size	500	1,000	1,250	2,500	0.5	0	100 or 125	100 or 250	10	10
3. Ease of use	--	easy	easy	hard	*	+3	*	*	9	4
4. Availability of certain features	--	yes	yes	no	*	+1	*	*	10	0
5. Output device	--	yes	yes	yes	*	-1	*	*	7	9

* Cannot be rated directly with this method

** Hypothetical for illustration purposes

5.3.2 Approach "X": Rating of Capability Differences

This rating method is based on the usefulness of the difference between the capabilities of two systems in fulfilling requirements. The capability of System A is compared to that of System B on a parameter-by-parameter basis, and the value, if any, of the difference, if any, is assigned a rating. A scale of any range can be used; the one shown was arbitrarily chosen for illustrative purposes. Note that the difference must be useful in terms of requirements; a difference can exist and be rated 0 if it is not effective.

Effectiveness of the difference between Systems A and B in meeting requirements:

	Rating
• High, in favor of System A	+3
• Moderate, in favor of System A	+2
• Low, in favor of System A	+1
• None, or not important	0
• Low, in favor of System B	-1
• Moderate, in favor of System B	-2
• High, in favor of System B	-3

If System A is more effective than System B, a positive point rating is assigned to a parameter, and a negative rating when the opposite is true. Hypothetical examples are shown in Table 6.

The ratings are then weighted and aggregated. A net positive total score indicates that System A is more effective than B for the problem requirements. A negative total score means System B is better than A; a 0 total score implies A and B are equally effective.

Comment

- The technique forces a direct comparison of two systems for each parameter evaluated. This prevents (or tends to prevent) incorrect relative ratings that can occur when each system can be rated individually.
- In some cases it appears to be more feasible to assess the difference between two systems in meeting a problem requirement rather than evaluating each system parameter independently against a problem requirement. For example, a quantitative requirement for ease of use of query language is difficult to state, yet it could be desirable to assign more credit to the system with the easier language. In such a case, a direct comparison of the two systems is necessary in order to assess relative simplicity.
- The techniques measure the relative effectiveness of two systems in meeting requirements; there is no measure of how well either system fulfills requirements.
- Two (and only two) systems must be evaluated at a time, and a single score is developed for the pair of systems. When more than two systems are evaluated for the same problem-mix, every possible combination of pairs of systems must be evaluated. The comparison of the scores for pairs of systems is not straightforward.

5.3.3 Approach "Y": Percentage of Requirement

In Approach Y, the capability of each GDMS for each parameter is expressed as a percentage of the requirement. A rating of 100% is assigned if a requirement is fully met, and 0% if no useful capability (in terms of requirements) exists. If a requirement is partially met, a rating between 0 and 100% is selected, depending on the degree of fulfillment. Table 6 illustrates this method.

The percentage ratings are then weighted and aggregated, and an overall score is developed for each system evaluated. The overall scores have significance in that they indicate the overall weighted percentage of requirements fulfilled, and the scores can be directly compared among two or more systems.

When a system capability exceeds a problem requirement, and the excess capability is useful, a percentage value over 100 can be assigned. This presents additional estimating difficulties, however, and does not result in all parameters being quantified in the same normalized range.

Comment

Since the computation rule is simple division, it is impossible to reflect non-linear functional relationships between capability and effectiveness. The evaluator has no flexibility in deciding how effective or useful a capability is in terms of requirements.

5.3.4 Approach "Z"

This approach is the one that was chosen and developed for the evaluation technique and is described in detail in Section 2.4. In brief, the capability of a system is compared against the requirement, and the parameter is rated by the evaluator using the standard scale. Table 6 illustrates ratings based on this approach for comparison purposes.

Comment

The comparison shown in Table 6 portrays the superiority of this method. It is flexible, provides sufficient sensitivity, reflects the evaluator's judgement, and yields scores that are relatively easy to understand.

Conclusion

Approach "Z", which is used in PEGS, is superior to the other methods; Table 7 summarizes the main features of this approach.

Table 7. Comparison of Rating Methods

	"W" (A/B)	"X" (A-B)	"Y" %	"Z" PEGS
Rated against requirements	No	Yes	Yes	Yes
Ratings indicate effectiveness	No	Yes	No	Yes
Ratings are normalized	No	Yes	Yes	Yes
Independent overall score for each system	No	No	Yes	Yes
Does not require relative rating of systems in pairs with one score per pair	No	No	Yes	Yes
Comparison of scores for more than two systems straightforward	No	No	Yes	Yes
Score has significance in terms of requirements	No	No	Yes	Yes
Overall score can be normalized and interpreted using basic rating scale	No	Yes	Yes	Yes
Handle all types of parameters	No	Yes	No	Yes
Flexibility in rating such as using functional relationships	No	Yes	No	Yes
Permits direct rating of capability without intermediate quantification step	No	Yes	No	Yes

Section VI

CONCLUDING REMARKS

There are two topics which deserve summarization in this section of the report: the features of PEGS, and the problems with quantitative evaluations. In addition, suggestions for the future use and development of PEGS are presented.

6.1 FEATURES OF PEGS

The features of PEGS have been discussed in detail throughout the report; they will now be summarized in outline form with references to the appropriate sections shown in parentheses.

1) General

- Provides a systematic technique for evaluating complex systems.
- Can be used for a variety of objectives and purposes (Section 2.1).
- Computations are simple; a computer program is not required as is usually the case with simulations (Sections 2.2 and 2.5).
- Basic procedure is easy to revise.
- Provides an independent overall score for each system evaluated (Section 2.2 and 2.5).
- Comparison of overall scores for two or more systems is straightforward (Sections 2.2, 2.4, 5.2, and 5.3).

2) Parameters

- A comprehensive list of parameters is provided (Section IV).
- Parameter list is open-ended; new parameters can be added easily (Sections 2.1.7 and IV).
- Parameters to be used for a given evaluation are selected by evaluator (Section 2.1.7).

3) Weighting (Sections 2. 2 and 5. 2)

- The parameters in a group of parameters may be weighted independently of all other parameters.
- The computed scores at each hierarchical level retain their significance in terms of the standard weighting scale.
- Weighting method may be undertaken in any order, bottom-up or top-down.
- The use of percentages provides any degree of precision desired and is a familiar concept.
- Weights are not pre-specified and are assigned by the evaluator thus providing complete weighting flexibility.

4) Rating (Sections 2. 4 and 5. 3)

- Capabilities are rated against requirements.
- Ratings for capabilities are not pre-specified and are based on evaluator's judgement of effectiveness.
- Ratings are normalized by using a standard scale.
- Rating method can handle all types of parameters; parameters can be rated directly, if necessary, without going through an intermediate quantification step.
- The evaluator can use the rating descriptors provided or he can formulate his own set.
- Ratings and scores are easy to understand (above sections and 5. 2 and 5. 3).
- The evaluator has complete freedom in formulating the functional relationship between capability and effectiveness for every parameter (above sections and 3. 2. 2).

6.2 PROBLEMS WITH QUANTITATIVE EVALUATION

A number of problems with the development and use of quantitative evaluation techniques have been raised in various sections of the report. These problems are not peculiar to PEGS and they would prevail for many other techniques as well. They are:

1) Information about GDMS's is often incomplete and varies in level of detail among systems. This is especially true for systems under development.
2) Difficulty in determining future requirements or unknown general requirements of an application mix.
3) Selection of proper criteria for measuring effectiveness.
4) Difficulty of avoiding overlap in a comprehensive list of parameters.
5) Danger of assigning too much importance to parameters that are easily quantified.
6) Evaluation results cannot be validated; results are valid only in a subjective sense.

The last point is particularly important, since the development and application of the technique involves a good deal of judgement and opinion. As a consequence, there is no known method that can be used to ascertain the validity of an evaluation based on a technique such as PEGS. The technique does not predict or simulate a result that can be physically measured or observed. For example, the accuracy of a method to estimate sorting times can be validated by performing sorts and comparing actual results with estimates. Overall system evaluation, on the other hand, predicts overall usefulness of the system, which in itself cannot be measured. This does not imply that the technique is not useful; it does imply that the results of the technique are not conclusive and should be used with caution.

6.3 FUTURE USE AND DEVELOPMENT

6.3.1 Evaluator

The final procedure that emerged from the study places great importance on the skill and experience of the evaluator as described in Section 1.3.4. There are many areas, such as the selection of rating descriptors, where a number of alternative approaches are suggested, and the evaluator must exercise judgement as to which alternative provides the best basis for an evaluation. Although it would be easier for the evaluator to use (and for the project team to develop) a more rigid evaluation procedure, it was felt that the technique is more sensitive and adaptable to more uses by being flexible in many areas. The future use of PEGS, therefore, should be undertaken only with qualified evaluators in order to realize the full potential usefulness of PEGS.

6.3.2 Further Work

The development of PEGS indicates several logical areas for future work in order to fully utilize and exploit the results of the study. In many respects, PEGS is similar to a computer program, in that it requires debugging, use and refinement in order to realize its full potential. The details of a plan for further work are the proper subjects of a proposal and will not be outlined here. The two major areas of use and refinement will be discussed briefly.

The use of PEGS for a variety of evaluation objectives, application environments, and GDMS's would provide the basis for analyzing its usefulness. There is no substitute for evaluation experience with Air Force applications and specific systems; field work is strongly recommended. Both conventional and test evaluations should be made in order to learn more about the time and effort required to evaluate system, and to determine the usefulness of the scores resulting from the evaluations. The consistency of results; variations of scores among systems and evaluators; and the evaluation of proposed versus existing systems would be among the topics analyzed.

The evaluator's task is not mechanical or clerical in nature; he must analyze requirements, determine weights, become knowledgeable about two or more GDMS's and then make a large number of judgements in order to arrive at overall scores. However objective an evaluator strives to be, his judgements will be based, to some extent, on his personal preferences as well as on his background and experience. It is unlikely that two evaluators independently would arrive at identical scores for a given situation. The consistency of results of parallel evaluations should be analyzed during a field checkout of PEGS.

The refinement of PEGS would either be done after the use of PEGS or it could be a parallel effort. Improvements would probably be made in the parameter list, analysis of GDMS's rating methods, weighting techniques, and summarization schemes, as well as in other areas. The identification, definition, organization, selection, and measurement of parameters would be one of the major areas undergoing further development. Although it is impossible to predict what changes will be made in PEGS, it is nonetheless inevitable that PEGS will undergo continual refinement as it is used.

The analysis and refinement of PEGS would yield other benefits as well. A better understanding of the optimum use of GDMS's both in terms of the selection of applications and of GDMS's should result. Further use and development of PEGS would provide a sound basis for the specification and design of new generalized systems.

BIBLIOGRAPHY

Abrahams, Paul W., et al, Quantitative Methods of Information Processing System Evaluation, Technical Report No. 5201-TR-0072 (AD 433 220), ITT Data Processing Center, Paramus, New Jersey, October 1963.

Advanced Naval Tactical Command and Control Study, TR-65-58-2, Volume IV, Methodology, Informatics Inc., January 1965.

Advanced Programming Development: A Survey, ESD-TR-65-171, Electronic Systems Division and Computer Associates Incorporated, February 1965.

Arbuckle, R. A., "Computer Analysis and Thruput Evaluation," Computers and Automation, January 1966, pp. 12-19.

Baker, C. L., JOSS: Introduction to a Helpful Assistant, Memorandum RM-5058-PR, The RAND Corporation, July 1966.

Bauer, Walter F., "On-Line Systems — Their Characteristics and Motivations," On-Line Computing Systems (American Data Processing, Inc.), 1965.

Bauer, Walter F, and Frank, Werner L., "DODDAC — An Integrated System for Data Processing, Interrogation, and Display," Eastern Joint Computer Conference Proceedings, Volume 20, 1961, pp. 17-32.

Borko, H., Evaluating the Effectiveness of Information Retrieval Systems, SP-909/000/00, System Development Corporation, August 1962.

Borko, H., The Conceptual Foundations of Information Systems, SP-2057, System Development Corporation, May 1965.

Bourne, C. P., et al, Requirements, Criteria, and Measures of Performance of Information Retrieval Storage and Retrieval Systems, Stanford Research Institute, December 1961.

Bryant, J. H., and Semple, Parlan, Jr., "GIS and File Management," Proceedings of the 21st National Conference, ACM, 1966, pp. 97-107.

"Coming Changes in Systems Analysis and Design," EDP Analyzer, November 1965.

Connors, T. L., An ADAM Presentation, MTP-16, The MITRE Corporation, October 1965.

Corbin, Harold S., and Frank, Werner L., "Display Oriented Computer Usage System," Proceedings of the 21st National Conference, ACM, 1966, pp. 515-526.

Display Oriented Computer Usage System (Interim Technical Documentary Report), TR-66-75-1, Informatics Inc., April 1966.

Edwards, N. P., "On the Evaluation of the Cost-Effectiveness of Command and Control Systems," AFIPS Conference Proceedings, Volume 25, 1964, Spring Joint Computer Conference, pp. 211-218.

"File Management and Data Retrieval," EDP Analyzer, November 1964.

Franks, E. W., A Data Management System for Time-Shared File Processing Using a Cross-Index File and Self-Defining Entries, SP-2248, System Development Corporation, April 1966.

"Generalized File Processing Software," EDP Analyzer, October 1965.

Generalized Information System, Application Description, International Business Machines Corporation, 1965.

Gradwohl, Alan J., et al, Use of Air Force ADPS Experience in Judging Proposals for New Automation, ESD-TR-65-276 (AD 618 075), Planning Research Corporation, March 1965.

Grant, E. E., The LUCID Users' Manual, TM-2354/001/00, System Development Corporation, June 1965.

Guillebeaux, C., and Grant, E. E., The General Purpose Display System (GPDS), Users' Manual -- Part I, TM-2722/001/00, System Development Corporation, March 1966.

Head, Robert V., "Real Time Systems: A Complexity Checklist," Computers and Automation, May 1965, pp. 27-30.

Henderson, Madeline M., Tentative Bibliography on Evaluation of Information Systems, Research Center and Advisory Service on Information Processing, National Bureau of Standards, 1965.

Henderson, Madeline M., et al, Corporation Convertibility and Compatibility Among Information Systems: A Literature Review, National Bureau of Standards Miscellaneous Publication 276, June 1966.

Hillegass, John R., "Standard Benchmark Problems Measure Computer Performance," Computers and Automation, January 1966, pp. 16-19.

Hillegas, J. R., et al, "Generalized Measures of Computer System Performance," 1962 ACM National Conference Digest of Technical Papers, September 1962, pp. 120-121.

Hitch, Charles J., and McKean, Roland N., The Economics of Defense in the Nuclear Age, R-346, The RAND Corporation, March 1960.

Hospital Computer Project, Memorandum Nine, Progress Report, The Massachusetts General Hospital, February 1966.

Information System Sciences: Preprints and Supplementary Papers for the Second Congress, Sponsored by the Air Force Electronics Systems Division and the MITRE Corporation, November 1964.

Introduction to Integrated Data Store, General Electric, April 1965.

Joslin, Edward O., "Cost-Value Techniques for Evaluation of Computer Systems Proposals," AFIPS Conference Proceedings, Volume 25, 1964, Spring Joint Computer Conference, pp. 367-381.

Joslin, Edward O., and Aiken, John J., "The Validity of Basing Computer Selections on Benchmark Results," Computers and Automation, January 1966, pp. 22-23.

Kosakoff, Martin, and Buswell, D. L., "Experience with a Generalized Information Processing System," AFIPS Conference Proceedings, Volume 24, 1963, Fall Joint Computer Conference, pp. 183-192.

Markel, Gene A., Toward a General Methodology for Systems Evaluation, 352-R-13 (AD 619 373), HRB-Singer, Incorporated, July 1965. [This report contains an annotated bibliography of literature on systems evaluation and includes a cross-referenced topical listing.]

McCloy, Edward, "Some Thoughts on EDP Equipment Selection," Journal of Data Management, April 1966, p. 20.

McGee, W. C., "Generalization: Key to Successful Electronic Data Processing," Journal of the Association for Computing Machinery, January 1959, pp. 1-23.

McGee, R. C., and Tellier, H., "A Re-Evaluation of Generalization," Datamation, July/August 1960, pp. 25-29.

Miller, James C., "Conceptual Requirements for Determining Information Requirements," AFIPS Conference Proceedings, Volume 25, 1964, Spring Joint Computer Conference, pp. 609-620.

Opler, Ascher, "Measurement of Software Characteristics," Datamation, July 1964, pp. 27-30.

OTC Query Language, ESD-TDR-64-443, Volume XXII, American Institute for Research, June 1964.

Patrick, Robert L., "Measuring Performance," Datamation, July 1964, pp. 24-27.

Postley, John A., "File Management Applications," DPMA Quarterly, July 1966.

Postley, John A., Informatics' Urban Management Data System, AIS-65-006, Informatics Inc.

Postley, John A., et al, Report on the Study of Behavioral Factors in Information Systems, Hughes Dynamics, Incorporated.

Postley, John A., and Buettell, T. Dwight, "Generalized Information Retrieval and Listing Systems," Datamation, December 1962, pp. 22-25.

Proceedings of the Second Symposium on Computer-Centered Data Base Systems, TM-2624/100/00, System Development Corporation, December 1965.

Rees, Alan M., The Evaluation of Retrieval Systems, Paper presented at the 2nd Annual Conference on Technical Information Center Administration organized by Drexel Institute of Technology, Philadelphia, Pennsylvania, June 14-17, 1965.

Report on Initial Planning for GENISYS (Generalized Information System), ESD-TR-65-463, System Development Corporation, July 1965.

Rosenthal, Solomon, "Analytical Technique for Automatic Data Processing Equipment Acquisition," AFIPS Conference Proceedings, Volume 25, 1964, Spring Joint Computer Conference, pp. 359-366.

Sable, J., et al, Design of Reliability Central Data Management Subsystem, RADC-TR-65-189, Volume I (AD 620 025), Volume II (AD 469 269), Auerbach Corporation, July 1965.

SDS MANAGE Reference Manual, 90-10-46A, Scientific Data Systems, Inc., May 1966.

Shaw, C. J., A Bibliography for Generalized Information System Designers, TM-2289, System Development Corporation, March 1965.

Shaw, J. C., JOSS: Experience with an Experimental Computing Service for Users at Remote Typewriter Consoles, P-3149, The RAND Corporation, May 1965.

Simmons, R. F., "Answering English Questions by Computer: A Survey," Communications of the ACM, January 1965, pp. 53-69.

Snyder, Monroe B., et al, Methodology for Test and Evaluation of Document Retrieval Systems: A Critical Review and Recommendations, HSR-RR-66/6-Sk, Human Sciences Research Incorporated, January 1966.

Statland, N., et al, An Approach to Computer Installation Performance Effectiveness Evaluation, ESD-TR-65-276 (AD 617 613), The Auerbach Corporation, June 1965.

Taylor, Alan, et al, Quantitative Methods for Information Processing Systems Evaluation, ESD-TR-64-194, Auerbach Corporation, January 1964.

Three COLINGO-Like Solutions to the Data Base Problem, The MITRE Corporation, October 1965.

Unclassified

Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Informatics, Incorporated 5430 Van Nuys Boulevard Sherman Oaks, California 91401	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP N/A

3. REPORT TITLE

A METHODOLOGY FOR COMPARISON OF GENERALIZED DATA MANAGEMENT SYSTEMS: PEGS (PARAMETRIC EVALUATION OF GENERALIZED SYSTEMS)

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Final Report

5. AUTHOR(S) (Last name, first name, initial)

Dowkont, Anthony J.
Morris, William A.
Buettell, T. Dwight

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
March 1967	286	6

8a. CONTRACT OR GRANT NO. AF 19(628)-5936 b. PROJECT NO. 2801 c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) ESD-TR-67-2 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) None

10. AVAILABILITY/LIMITATION NOTICES

This document may be further distributed by any holder only with specific prior approval of Hq ESD (ESTI).

11. SUPPLEMENTARY NOTES None	12. SPONSORING MILITARY ACTIVITY Directorate of Computers, Electronic Systems Division, Air Force Systems Command, USAF, L. G. Hanscom Field, Bedford, Mass. 01730

13. ABSTRACT

The objective of this research study contract was to develop a practical technique for evaluating generalized data management systems. This report describes the technique that was developed for the quantitative evaluation of the relative effectiveness of large on-line generalized data management systems.

Parametric Evaluation of Generalized Systems (PEGS) is a procedure based on analysis of user-orientated system parameters. The utility of a system is measured in terms of its usefulness in a particular application environment. The overall effectiveness of the system is evaluated, rather than any individual hardware or software component.

A large number of system parameters is described. Each parameter is a value attribute of a data management system, with respect to its capability, its ease of use, or its performance.

Techniques are specified for measuring the utility of a system to the user in terms of each parameter. These measurements of individual parameter utility are expressed as ratings based on a standard scale. Each rating is weighted by a measure of its relative importance in a particular application. Finally, a single numeric figure-of-merit is computed for each generalized data management system evaluated.

DD FORM 1 JAN 64 1473

Unclassified

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Generalized data management systems						
File management						
Software evaluation techniques						
Computer programming						
PEGS						
Methodology						
Computers						
Software						
Evaluation						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through
____."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through
____."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through
____."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, or *(U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, rules, and weights is optional.