Evaluating database management systems

by EDWARD DAVIDSON
General Electric Company
Philadelphia, Pennsylvania

ABSTRACT

This report documents a methodology developed and used for the evaluation and selection of database management software. The basic methodology can be used in the evaluation of other types of software. The report describes the step-by-step process and provides an extensive discussion of the definition of evaluation and selection criteria. The report will be of most use to first-time evaluators, but may also be of use to more experienced personnel.
BACKGROUND AND SCOPE

In the course of conducting an evaluation of several candidate database management software packages with the objective of selecting one of them for a particular application, it was noted that there were no clear and concise guides available to assist novice evaluators in this type of task. With the rapid changes in computer technology and the decline of hardware costs, more and more applications will be turning to database management systems. It was felt that a guide to the evaluation of database management systems would be of great use to the many organizations that will be faced with the task of selecting a software package for a particular environment or application.

This guide has been written for first-time evaluators; however, more experienced personnel may find some useful information presented. It is not meant to be the sole source of information needed to conduct thorough evaluations. Rather, it presents a sequence of events, suggestions of sources of information, and a set of evaluation criteria that should be augmented or amended as necessary to suit particular environments or applications. In some cases, the evaluation criteria present alternative conditions and possible implications of each alternative.

ENVIRONMENT IDENTIFICATION

Prior to selecting candidate database management systems for evaluation, a determination must be made as to the operating environment in which the software will be used. This environment includes both the computer hardware and the specific operating system installed or to be installed. Consideration must also be taken of possible subsequent installations of the system (i.e., second-site or distributed processing environments). Will all installations involve the same hardware and operating system? Some database management software packages, especially those marketed by the hardware vendors themselves, may be limited to one specific type of hardware or operating system, while other packages will run on a wide variety of systems. Although the single-environment package may solve the current need, it may limit future growth or dispersion of the application.

CANDIDATE SELECTION

Once the operating environment has been established, it is possible to select one or more database management software packages for evaluation. There are several sources of information that can be used to determine which packages are available for the appropriate operating environment.

1. Datamation is a monthly magazine on data processing that contains excellent articles on database technology as well as other computer-related topics. Each issue has a particular theme and some issues are devoted to database management systems. This publication frequently has product surveys (software in general, including database management systems, or specific types of software only, such as only database management systems). About two years' worth of the publication should be scanned for surveys and other pertinent articles. This magazine is popular with data processing personnel and should be widely available.

2. Computerworld is a weekly newspaper of data-processing information. In addition to product advertising, this publication has articles describing new product releases and enhancements or problems affecting other products as well as articles of general interest related to database technology.

3. Datapro Reports, found in many data-processing shops or technical reference libraries, contain articles on database techniques and general database topics, as well as articles on or reviews of specific systems. In addition to synopsis reviews on specific software packages, there are charts of comparisons between several similar systems. The product reviews are the results of Datapro surveys of product users and, depending on the number of respondents, pertinent or meaningful information may or may not be complete. Not all products are included; however, most of the popular, established packages are represented.

4. Interface is a quarterly publication, primarily available, and used, for vendor advertising. There are several editions, one of which contains database management systems. There is also an edition devoted to minicomputer software. With this publication, most of the popular database management systems can be compared briefly on relative features and price, vendors' addresses and other contact information can be determined, and promotional literature can be requested through reader-service cards.

5. Micro-Mini Systems is a monthly publication of interest to micro and minicomputer environments. During the past year, they have reviewed a number of database management systems for the mini and micro environments.

In addition to utilizing the publications described above, information on possible candidates may be provided by the personal experiences of the organization's staff members or from the hardware vendors, who may be able to provide a list of software which runs on their equipment.
DETERMINING SOFTWARE REQUIREMENTS

Before the evaluation process can begin, operational and functional requirements that the database management software must satisfy should be documented. The requirements will be used to select, categorize, and order according to priority the specific evaluation criteria that will be used to review each candidate package.

The requirements may cover one or more of the following categories (some specific requirements for a particular application may not be covered in this document but should be included as necessary):

1. Portability—on what computer hardware and under what operating systems must the software run? What is the timeframe for implementation on different hardware or operating systems? On what other hardware/operating system might the software run in the future?

2. Flexibility—what flexibility features must the software have? Integrated data dictionary? Integrated query language? Dynamic generation or deletion of keys, user views, access privileges? Variable-length records? Must the software support multiple users at one time? Multiple databases? Distributed processing? Batch and online users concurrently?

3. Security—what level of data security is required? File? Field? Will it be necessary to restrict access to specific data according to the values of one or more fields (i.e., user A is limited to data for his department only)? Some database management systems provide that level of security—"security by value"—in the definition of user views, while other packages restrict access to the field level, so that if a user is granted access to a field, he has access to all values of the field. Certain applications, such as personnel or payroll, may need to restrict some users to specific portions of the database. Most applications will not need this level of security.

4. Recoverability—is it required, or desired, that the database management software provide the capability for transaction journaling, before-image or after-image journaling, or system checkpointing? Must the software provide automatic recovery (automatically restore the database to a certain level based on a checkpoint file or journal file when the system is brought back up) or is it acceptable to require the intervention of programmers or operators to restore the database? Some DBMS software packages do not provide recovery features at all, relying only on system or user recovery procedures.

5. High Order Language (HOL) Interface—will the application require access to the database from HOLs (i.e., COBOL, FORTRAN, PL-1, C, etc.)? Which languages? How complex will the access keys be (i.e., multiple fields, etc)? The HOL call procedures for some database management systems are more restrictive or less flexible than others. Some applications may have access requirements that include accessing data in more than one file at one time.

6. Performance—is software response time critical? What is the maximum desired response time for a query? Update? Will the application be primarily used for database maintenance (update, preplanned reporting) or will it be used mostly for ad hoc queries or random retrievals? Is data storage critical? Some database software systems will make optimum use of online storage by compressing multiple blanks, zeros, or empty fields. For extremely large databases, depending on the operating environment, space utilization may be critical.

After the specific requirements have been determined, they should be prioritized according to their importance to the success of the application. If some requirements have alternative implementation possibilities (i.e., good recovery procedures are required, preferably automatic but we could live with semi-automatic procedures as long as . . . ), the alternatives should be prioritized. When the prioritized requirements have been documented, they should be used to develop a prioritized list of criteria which will be used in the evaluation of the candidate systems. One requirement may result in several specific criteria. The following section lists some possible criteria with potential implications of alternative implementations.

EVALUATION CRITERIA

The items listed below represent possible areas of concern in the evaluation of database management systems. These criteria are not intended to be a complete list and are not presented in any particular order. These items, and any others that organization staff members may suggest, and any that may be imposed directly by the requirements, should be reviewed for applicability to the system requirements. Where alternative implementations are possible, a determination should be made as to which implementation is appropriate for
the application, or the order of preference if more than one might be appropriate. Items that have no bearing on the specific requirements may be discarded, or may be used to further evaluate or compare several possible candidates that satisfy the required criteria. These optional criteria might be concerned with vendor reputation and support, level and quality of documentation, or features that are not currently required but may be of concern for future, as yet unplanned, application development activities.

**Performance Features**

1. Does the software optimize multikey retrievals? Some database management software will determine the shortest path to the data described by the combined keys by analyzing the number of records qualifying for each key and selecting the shortest access path. Other software packages will analyze records based on the order of the keys expressed in the query or call statement. Depending on the number of records qualifying for each key, one technique may result in better performance than the other. These concerns apply mainly to relational database management systems. Network-type software depends on imbedded pointers in the data, although some network systems are developing relational-type query languages that allow some level of optimization.

2. Does the software run in the "native" mode or in the "compatibility" mode on the particular computer system? Native mode processing is the most efficient, taking advantage of the hardware and system software features of the host system. It is usually written or compiled in the assembler language of the host system. Under "compatibility" mode the host system emulates another operating system in order to run the software. Emulation can have a significant detrimental impact on performance.

3. If two or more applications or users attempt to access the same record at exactly the same time, what level or type of lockout occurs, if any? Can two or more users access the same record simultaneously for "read-only" or "browsing"? Some software allows only one user access to a record, regardless of the type of operation performed. Other packages will allow a user to hold a record during an update operation to prevent any other user’s accessing it for updating but will permit other users to read or browse the held record. In applications that may have many simultaneous access attempts for browsing and updating the same record, this could affect the response time for some of the users. Some vendors use the term “multi-threading” to describe this feature.

4. Does the database management software allow simultaneous batch and interactive applications? Is there a software limit to the number of simultaneous users or applications?

5. What are the software limitations, if any, with regard to the number of databases supported by one copy of the software? Are there any limitations to number of files per database, number of records per file, number of bytes per record, number of bytes per database, and so on? Certain applications with large data requirements may find specific software packages too restrictive or limited with regard to the amount of data that can be handled.

6. Will the vendor provide a copy of the database management software, or allow access to a copy, for benchmarking? Will there be a charge for this? Will the vendor provide any assistance in the benchmarking set-up or execution? If benchmarking of the software is not possible or feasible for an organization, attempts should be made to obtain benchmark results or reports from other customers who have performed benchmarks. Although the specific controls of the current evaluators cannot be tested, pertinent performance information may be derived from these reports.

7. Must the database be taken off-line or out of productive use to perform any or all database administration functions such as defining a new field, expanding a field’s size, deleting a field, assigning a new key, or granting or revoking access privileges? Some database management software allow many of the database administration functions to be dynamic and have no impact on executing applications, except for those directly affected by the specific changes. Other packages require database utility programs to run with exclusive access and control of the database, some requiring the database to be dumped and restored prior to restarting the applications. This could have significant impact on environments where there is constant, heavy activity on the system. Alternative solutions to this problem could be to have database modifications performed during periods of little or no application usage (nights, weekends, or holidays) or prescheduled downtime for database maintenance.

8. If the software provides for the compression/decompression of repeated blanks, zeroes, and null-value fields, is the feature optional? How is the performance affected by invoking the option?

**Data Dictionary**

1. Does the software have an integrated data dictionary? If so, is the data dictionary active or passive? Most database management systems provide some sort of data dictionary. The more sophisticated packages have dictionaries that control all access to the data for the users and database administrator. This arrangement allows all access to a particular piece of data to view the data consistently, eliminates the need to have data definition statements in each program, standardizes the names of each field, and controls access to the database, file, and/or field level. Some of these also are used to store preencoded queries and user views. Passive data dictionaries record some information about the data for documentation purposes but exert no control over access to the data.

2. Can data dictionary information be modified without affecting active users? Some data dictionary software
allow fields to be added to the file or database definition, or information about currently defined fields to be modified, without restricting the use of the database to active users. The changes are made dynamically. Other data dictionary software provides database maintenance utilities that must be run with complete control of the database. Some database management systems require that the database be unloaded and reloaded after certain types of changes in or additions to the data dictionary. Other packages can handle changes with little or no impact on the physical database or existing data.

3. Is the data dictionary used to control the compression of multiple blanks, zeroes, and/or null-value fields? Some database management systems that offer data-compression features allow the data-compression option at the field level to be controlled by parameters in the data dictionary. In other software, compression is mandatory for all data or not allowed at all.

Recovery Features

1. Does the database management software contain any integrated recovery features at all? Some database software rely entirely on the operating system's backup and recovery features to protect the database. The more sophisticated systems provide some level of checkpoints, transaction journaling, before-image journaling, or after-image journaling, or a combination of these options.

2. If recovery is provided for, are the recovery procedures automatic? Following a crash, some database software will automatically restore the database through the last successfully completed transaction as soon as the system is brought back up. Other database systems will require operator or programmer action to restore the database, possibly application by application, from journal and checkpoint files. If specific personnel are required to restore the database, delays to production could be experienced if the personnel are not immediately available following a crash.

Flexibility Features

1. Does the software support concurrent processing by multiple interactive users or by interactive and batch applications? Some packages do not allow more than one application or user at a time. Others may have a limit to the number of active users or applications that can process concurrently. The design and potential usage of an application may require the capability to support several, if not many, concurrent users.

2. What is the format of the High-Order Language (HOL) interface ("call") and what languages does the software interface with? Does the software provide preprocessors for the applicable languages to translate and create the specific calls to imbed in the high-order language programs or does the programmer have to code each call directly in the program? Different packages provide different facilities. The most sophisticated packages allow the coding of English-like statements in the application program with subsequent preprocessors converting the query-type statements into the appropriate calls. This permits more productive coding by the programmers. Other packages allow query-type statements to be passed in call statements. The remainder of the packages require some type of control blocks to be defined, and control block parameters to be coded by the programmer. Of the last category, some packages will allow more complex key structures to be used in the access calls than in other database systems. As it can be seen, the method and format of the call structure may have an impact on the complexity, flexibility, and efficiency of the calls.

3. Does the software allow complex or concatenated keys in queries and HOL calls? Some packages allow the use of complex keys in both interactive queries and through calls from HOL programs. Other software packages are very limited in the access keys allowed in either query or call modes, although usually the call mode is the more restrictive. The ability to code very explicit access keys may be important in some applications. The use of complex keys may be limited by the database architecture employed by specific packages; many hierarchical and network database systems use simpler keys requiring the programmers to "navigate" the database for the desired data.

4. Can key fields and user views be created and deleted dynamically? Can new data items be defined to the database dynamically? In environments where the data requirements are continually changing, it may be advantageous to be able to expand the database, create and delete keys, and create, modify, and delete user views without having to restrict the use of the database or require the database to be unloaded and reloaded. Other applications, where the data requirements are static, may not need these features. Some of the newer packages allow for these dynamic operations and some of the more established packages are developing these capabilities.

5. Does the software permit the accessing of multiple databases and/or files in one access attempt (commonly called "joining")? Some software can only access one database at a time. Some packages restrict queries and calls to one database file at a time, requiring multiple queries or calls to satisfy complex retrieval requests. Other packages permit multiple-file access in a single query or call, simplifying the access procedure and increasing the flexibility of the system.

6. On what hardware does the software run? Some packages, especially those developed or marketed by a specific hardware vendor, may be restricted to the vendor's own hardware. Packages developed by software houses that do not construct hardware may run on almost every computer system available in certain classes of system. The current and future hardware configurations on which an application may run place constraints on the specific packages that should be considered for a specific
evaluation. A related question is whether the software runs in native mode or compatibility (emulation) mode on the hardware on which it runs.

7. Does the software have currently, or have plans for future implementation, features that will enhance future application development by the organization? Has the vendor demonstrated a capacity for, or is the vendor committed to, keeping the product current with advances in database technology? Some of the database management software packages stabilize as they mature, and although they may satisfy current requirements they may not be able to support substantial future growth. Other products, as they mature, implement features that reflect the advances in technology since the original implementation. Organizations that expect substantial increases in the number, size, and complexity of database applications using the same software, may want to consider the vendor’s future plans for the software.

8. What “user-friendly” features does the software provide to reduce the application-development effort and permit less skilled and nontechnical personnel to make effective use of the data? Some features to be considered include screen generation facilities, integrated query languages, integrated report writers, “HELP” commands accessible interactively, and meaningful error and diagnostic messages. While these features may not be used for initial selection and rejection of candidate packages, they may be considered when comparing closely similar systems to determine which is more “friendly” and easy to use.

**Security Features**

1. At what level is data security provided by the software? Some products provide security to the database of file level only. If you have access to the file, you have access to all data in the file. Other products provide access restrictions to the field level through user views. Still other products can provide security based on the value of specific fields. This is called “security by value.” An example would be to restrict a user to viewing data for other products can provide security based on the value of specific fields. This is called “security by value.” An example would be to restrict a user to viewing data for one department, or to prevent a manager from deleting on short notice or on a frequent basis. For those important in some applications.

2. Can access privileges be granted and/or revoked dynamically? In some environments, it may be advantageous to be able to modify access privileges instantaneously, without interfering with active users. Some applications may require access authorizations to be established or deleted on short notice or on a frequent basis. For those environments, this feature would be useful.

**Vendor Support and Reputation**

The items listed below, while most likely not constituting reasons to initially select or reject a particular candidate package, may be used to rank otherwise similar packages and can be used to identify potential areas of concern for products that otherwise fully meet all other selection criteria.

1. Does the vendor have a good reputation for responding to requests for information, dealing with reported problems with the software, and providing technical support of the software? This information can be obtained by interviews with current or past users of the software or through software surveys conducted by data-processing periodicals such as Datapro, Datamation, and Computerworld.

2. Are current users of the software satisfied with the performance of the software? Are they satisfied enough with the product to acquire other software produced by the same vendor? During interviews with the current users, comments may be made or solicited that will indicate whether the product is worth recommending to others, whether the product has lived up to its expectations, and whether the customer has enough faith in the quality of the vendor’s product to use other software from the same vendor. Some users, after acquiring a product, find out that it does not function as expected but have gone too far into development to discontinue use of the product; they have to modify their design to fit the product, which is not the ideal way to go. An isolated complaint should not cause undue alarm, but similar comments from several users about the same product should be considered significant.

3. Does the vendor provide a “hot line” for reporting problems with the product operation? Many vendors provide this service. Some have toll-free numbers, some operate twenty-four hours a day, including weekends. For applications that operate during nonbusiness hours, a twenty-four hour hot line may be an important consideration.

4. Are the software technical and user manuals clearly written, with adequate examples and illustrations? One significant area where products vary is in the quality of the documentation. Some otherwise excellent products have poorly written documentation, which greatly increases the effort required to learn and use the software effectively. Many application-development staffs do not have large training budgets and work under tight time constraints. To maximize the development effort, the staff should be able to quickly find and understand the information needed in the documentation, without having to constantly call the vendor’s technical staff for interpretation or explanation. Sample programs and routines, as well as examples of individual instructions, provided where the instructions are described, greatly improve the usefulness of the documentation.

5. Does the vendor supply small test databases with the installation of the product? Some vendors supply one or several test files with a small number of records to be used by the acquiring organization for training and experimenting with the data without having to define and load its own test data. This arrangement greatly speeds up training in the use of the product and allows a rapid review and evaluation of many software features and procedures. Some vendors also provide canned routines.
or scenarios to be used with the test database to demonstrate features.

6. Does the vendor have a procedure for notifying all users, on a regular basis, of reported problems and solutions or of the status of unresolved items? Some vendors conscientiously keep all users informed about the problem status of their product. Other vendors will respond to a reported problem directly to the reporting organization so that many organizations may experience the same problem without knowing that the problem may be global in nature and without knowing a workaround that might avoid the problem altogether.

7. Does the vendor provide installation support? If so, is it necessary, and at what cost? Some database management systems can be installed by the user by simply loading one or more files from a tape supplied by the vendor. Other packages require more technical support to install. Some vendor installation support is provided in the cost of the package, and some is available on a fee basis on top of the purchase price.

8. Does the vendor supply, or provide for, customer education and training in the use of the product? Is there any other training available that is related to the product? Some vendors have excellent training and education programs, not only in the elementary use of the software but in advanced techniques, internals, design, and so on. For other software, all education beyond a brief introduction to the software comes from manuals on a do-it-yourself basis. For organizations with small staffs and a need for rapid productivity, the customer training programs may be a critical factor.

9. Are enhancements and modifications to the products, either for product error correction or for the addition of new features, accomplished by user-installed zaps or patches, or by a complete replacement of the software? There is potential in user-installed zaps, especially where routines have to be coded in, for errors to be made because of poor documentation or human error. There is faster installation and less likelihood for error if a whole module is replaced with a new copy provided on tape directly from the vendor.

10. What is the cost of the various acquisition arrangements (buy/lease)? Are quantity discounts available? Are execute-only copies available? The acquiring organization should review the prices for the various options with regard to the features offered by the product, the support available, and the reputation of the vendor and product. A more expensive product is not necessarily a better product.

11. What is the cost of a maintenance agreement? What is the term of the maintenance agreement? Exactly what is provided for in the maintenance agreement? Different vendors provided different periods and levels of support.

12. Does the vendor provide additional features and capabilities to current users as enhancements to the existing software under the maintenance agreement, or are the features released as new products at additional cost? Some vendors are beginning to release additional features as add-on, extra-cost products. Before the purchase/lease/maintenance agreements are signed, it should be fully understood what the vendor’s position is on this.

13. Is there any indication of problems, including litigation, that might affect future use of the product? This may be discernible from the company’s annual report or may be asked directly of the vendor. Make sure you get all of the details, including the vendor’s position, and follow up where possible to check the results of the action. It might be wise to have an alternative plan in the event the software is affected after acquisition.

EVALUATING CANDIDATE SYSTEMS

When the system requirements and evaluation criteria have been defined and prioritized, and upon receipt of the literature and documentation requested of the vendors, the evaluation and review of specific products can begin.

It might be advantageous to develop some form of documentation that summarizes the specific requirements, evaluation criteria, and features desired of the software so that appropriate notes and comments can be made as information on each product’s capabilities becomes known. Also, a form may be developed for use when contacting users of each system, specifying questions or points of concern to be discussed with the user, and providing a place for recording the user’s responses. These types of forms will standardize the information gathered during the evaluation and simplify the subsequent decision. Examples of these types of forms are shown in Figures 1, 2, and 3.

There is no specific method of evaluating the vendor literature and documentation. Since vendors do not stress the features or capabilities they do not provide, it will be necessary to identify those features and capabilities they do provide and infer that others are not provided. If in doubt, contact the vendor representative. Many vendors provide a brief technical summary of the software features in addition to the more detailed technical material. The summary will indicate what the software can do, but may not explain how it does it. The summary is frequently a sales brochure; it should not be used as the sole source of information about a product unless the vendor is unable or unwilling to provide additional information, in which case that product should be eliminated from further consideration.

The documentation for each product should be reviewed for information relating to the specific requirements or criteria. This review may be accomplished in several passes of the literature, each delving deeper and deeper into the technical aspects of the data structure, commands, command sequence and structure, utility functions and features, and so on.

The first pass should attempt to identify those products that clearly are lacking in one or more criteria or requirements. A decision can be made to eliminate those products entirely or make allowances or exceptions for the lack of compliance. The object of these reviews is to gradually weed out software products that clearly do not satisfy the requirements. It may be that none of the candidate systems can match all of the requirements, and a decision may have to be made to relax or revise the requirements, develop an in-house system entirely,
Evaluating Database Management Systems

or develop a workaround for those requirements which cannot be satisfied by the software.

Through the increasingly detailed review of the documentation, it may be possible to eliminate some products and group others together by the features provided or techniques employed. It is then necessary to evaluate the competing products on how they handle certain conditions or situations and on their vendor support, quality of documentation, and other subjective criteria. Where one product does not surpass all the others on these subjective criteria, a decision should be made about how to rate the criteria.

Unless one product clearly surpasses all others in the majority of the criteria, there may be several candidates remaining after this review. At this point, it may be advantageous to arrange for demonstrations of the software by the vendor, or, if it can be arranged, for benchmarking the candidates.

Although benchmarking is the best method, it may not be feasible. Some vendors will provide a copy of their software free of charge for a limited time for this purpose, while other vendors may impose a fee. Other vendors may give the organization access to the software at the vendor’s site or at another customer’s site at a time when the software is not in active use by other users. One must define the benchmark criteria, develop and package tests for use on the benchmark system, and schedule the time and personnel required for the tests. In some cases the vendor will conduct or assist in the testing. The results of the benchmark tests can be used to rank the candidates tested.

If benchmarking is not feasible, owing to time, cost, or personnel availability, it may be possible to evaluate the software by attending a demonstration of the software by the vendor. Although not as precise as benchmarking, the demonstrations may illustrate some interesting features of, or facts about, the system. What is the response time exhibited during the canned demonstrations? Although subject to the type, size, and load of the machine during the test, certain trends may be noticed. Did the demonstrator need reference manuals in order to conduct the demonstration? If so, it may indicate that the system is complex or difficult to use, since usually the vendor puts its best available personnel on the system to make a good impression.

In addition to observing the canned demonstration, it would be advantageous to be prepared with a small database definition (perhaps five fields) that you can ask to have defined and installed while you watch. This will let you see exactly what is required to accomplish the task, and will allow you to create some of your own data to test on the system. The same database definition can be used with each vendor and a measure of comparative complexity may be determined.

The vendors of the leading candidate packages should be
asked to provide the names and telephone numbers of several customers of their products. Some vendors may be unwilling or unable (because of customer requests for confidentiality) to divulge the information. In any case, remember that the names supplied will most likely be those of their most satisfied customers and meaningful or pertinent negative comments may not be made. These users should be asked to comment on the performance of the product, whether the product has performed as expected, the effort required and problems experienced in installing the product, feature performance, and the support of the vendor in answering questions and resolving problems. Even the most satisfied customer may have some negative comments about specific features. Similar comments from several customers may indicate inherent problems, which can then be discussed with the vendor. The customers' responses should be documented for future reference.

At this point, sufficient information should be available for the selection of one database management system for the application. If selection is still not possible, the evaluating criteria should be reviewed, made more explicit, or expanded to allow for further refinement and elimination of candidates until one product is considered suitable for acquisition. The Evaluation Procedure Flow is illustrated in Figure 4.

**POST-SELECTION TASKS**

Following the selection of a particular product, materials developed or acquired during the evaluation process should be reviewed, discarded, or filed, as appropriate, for further reference. If possible, the results of the study and supporting materials should be centrally filed and made available to other internal users as the need arises. Any materials, documentation, and manuals on temporary loan from a vendor should be returned in a condition suitable for resale, or should be purchased from the vendor.