

ARGONNE NATIONAL LABORATORY
9700 South Cass Avenue, Argonne, Illinois 60439

ANL/EES-TM--376

DE89 014007

ANL/EES-TM-376

EVALUATION OF COMPUTER-AIDED
SOFTWARE ENGINEERING TOOLS
FOR DATA BASE DEVELOPMENT

by

Mark A. Woyna and C. Robert Carlson*

Energy and Environmental Systems Division
Office of Advanced Computer Applications

February 1989

work sponsored by

U.S. DEPARTMENT OF DEFENSE
Organization of the Joint Chiefs of Staff
J-8 Directorate

*Illinois Institute of Technology, Chicago, Illinois.

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

&

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

CONTENTS

1	INTRODUCTION	1
1.1	Purpose	1
1.2	Outline of Report	1
1.3	Survey of Computer-Aided Software Engineering Tools	1
2	EVALUATION	4
2.1	Phase 1: Industry Survey	4
2.2	Phase 2: Application of Evaluation Criteria	4
2.3	Phase 3: Hands-On Evaluation	6
2.4	Phase 4: In-Depth Review	6
3	CONCLUSIONS AND RECOMMENDATIONS	14
3.1	CASE Summary	14
3.2	Evaluation Summary	14
3.3	Recommended Action	14
	APPENDIX A: List of CASE Tools and Results of Screening Process	17
	APPENDIX B: Evaluation Criteria	20

TABLES

1	Summary of Phase 3 of the Evaluation	7
2	Summary of Phase 4 of the Evaluation	13

EVALUATION OF COMPUTER-AIDED SOFTWARE ENGINEERING TOOLS FOR DATA BASE DEVELOPMENT

by

Mark A. Woyna and C. Robert Carlson

1 INTRODUCTION

1.1 PURPOSE

More than 80 computer-aided software engineering (CASE) tools were evaluated to determine their usefulness in data base development projects. The goal was to review the current state of the CASE industry and recommend one or more tools for inclusion in the uniform development environment (UDE), a programming environment being designed by Argonne National Laboratory for the U.S. Department of Defense Organization of the Joint Chiefs of Staff, J-8 Directorate. This environment gives a computer programmer a consistent user interface and access to a full suite of tools and utilities for software development.

In an effort to identify tools that would be useful in the planning, analysis, design, implementation, and maintenance of Argonne's data base development projects for the J-8 Directorate, we evaluated 83 commercially available CASE products. This report outlines the method used and presents the results of the evaluation.

1.2 OUTLINE OF REPORT

Section 1.3 provides an overview of CASE: its objectives, benefits, characteristics, components, and future direction. Chapter 2 describes a set of criteria for evaluating CASE tools for data base development support, as well as the results of our evaluation. Chapter 3 presents conclusions, makes recommendations regarding the suitability of available CASE tools for Argonne's data base development projects, and proposes future work.

1.3 SURVEY OF COMPUTER-AIDED SOFTWARE ENGINEERING TOOLS

A CASE tool is any software product that provides automated assistance in the creation or maintenance of software systems or the management of software development projects. Additionally, CASE supports methods and structured concepts for use in the development of software systems.

The objective of CASE is to improve the quality of software systems. By automating various phases of the software life cycle, many of the following benefits are realized: greater productivity, lower costs, faster development, software portability and reusability, project manageability, and standards enforcement.

Characteristics of CASE

Most CASE tools are microcomputer- or workstation-based software packages that integrate powerful graphics, used primarily to capture and display information, with a central data repository that promotes data sharing among users. In addition, CASE tools provide some form of automated methodology to assist in enforcing structured software development. Many CASE tools are integrated into larger tool sets, although some focus on only one area of CASE, such as diagramming or code generation.

A CASE environment will usually provide at least some of the following components and functions, either independently or as part of integrated tool kits:

- *Diagramming tools* -- to create and modify diagrams, facilitating the capture and display of design information for easy, interactive user analysis.
- *Information repository or data dictionary* -- to store and allow retrieval of design data, ensuring project stability by standardizing data access to all other tool sets. As a result, support is provided for the most important long-term asset: the data.
- *Analyzers and checkers* -- to provide automated analysis procedures to enforce specifications and identify inconsistencies in the design, often overlooked by users due to the large amount of code being developed.
- *Importer and exporters* -- to capture and disseminate data functions critical for the addition of new tools to the tool set.
- *Reengineering tools* -- to efficiently process existing information for design analysis.
- *Code-generation tools* -- to support systematic generation of system-specific "code" (i.e., data definition language) from system-independent design specifications.
- *Report-generation tools* -- to give the analyst a variety of perspectives on the design data.
- *Prototyping tools* -- to allow rapid system prototyping, which provides important end-user feedback early in the design process.

Current Issues in CASE

While CASE is a powerful concept, several issues need to be resolved before the products will be widely integrated into software development. First, although CASE should cover the entire software-development life cycle, it has proven more successful in

some areas (such as system design) than in others (such as requirements analysis). Second, interface standards need to be developed so that any number of tools, each focusing on a particular phase of the software-development life cycle, can be integrated into a single software-development environment. Solutions include the integration of data dictionaries, or the development of a dictionary standard, and the ability to transfer data between tools and dictionaries. Third, as the number of CASE users grows, user needs will be better defined: feedback from users will allow the tool vendors to focus on the functionality needed to support each phase of software development.

2 EVALUATION

Computer-aided software engineering tools were evaluated in four phases. The first three phases applied pass-fail screening criteria to each tool, and the fourth compared in detail the tools that survived the screening process.

2.1 PHASE 1: INDUSTRY SURVEY

We first compiled a list of 83 commercially available CASE tools, based on information collected from a variety of sources. The main selection criterion was that each tool have some functionality in data base development; functions of interest included data modeling, data analysis, and normalization.

We next obtained, from the various vendors, product information on the tools. Using this information, we verified each tool's functionality for data base development. Many of the tools were found to provide no support for data base development, but rather to be application builders or fourth-generation languages. Following this screening, the initial list was reduced to 32 tools (see Appendix A).

2.2 PHASE 2: APPLICATION OF EVALUATION CRITERIA

In the second phase of the evaluation, we developed a detailed set of selection criteria. The criteria focused on specific aspects of data base design as well as the more traditional aspects of CASE. The development of the criteria was an ongoing process: each phase of the evaluation revealed new information on the products' capabilities and limitations, and this information led to refinement of the criteria.

Criteria Definitions

The criteria were divided into nine major areas, defined as follows (see Appendix B for a complete list of the individual criteria):

- *Data dictionary.* The tool must have a central data dictionary for all information entering the system. System information (such as diagrams, data elements, entities, processes, and relationships between data) should be nonredundant and globally accessible, to ensure that data updates are instantly reflected wherever data are referenced.
- *Diagramming and graphics.* The tool should allow the user to create entity-relationship, data-flow, and structure diagrams. User-defined diagrams would also be useful. The diagramming facilities should be easy to use, be powerful enough to create and modify objects in the diagram, and provide a way to verify diagram correctness. The tool should be able to load the data dictionary

with information captured on diagrams and automatically generate diagrams from the data dictionary.

- *Methodologies.* The tool should support at least relational and entity-relationship models of the data in the system. Additional design methodologies are acceptable. The product should support at least one phase of data base development (planning, analysis, design, implementation, testing, or maintenance).
- *Multi-user capabilities.* The tool should allow concurrent access to the data dictionary, while ensuring data integrity at all times, by providing a single data repository with locking capabilities.
- *Environment support.* The tool should be supported on a personal computer, workstation, mainframe computer, or combination of the three. In each supported environment, the multi-user requirements must be met.
- *Import, export, and integration.* The tool should import and export data to and from other application packages (such as data base management systems, external data dictionaries, other CASE tools, or word processing packages). In addition, it should be possible to integrate the tool into an existing development environment.
- *User interface and usability.* The tool should be generally easy to learn and use. It should have a logical structure and provide a uniform working environment via a forms- or window-based interface.
- *Reporting.* The tool should provide a variety of summary and detail reports, including reports that check consistency, compare diagrams to the dictionary, list cross references, and track a project's progress. Automatic generation of design documentation is also desirable.
- *Miscellaneous.* The tool should be cost-effective in the context of Argonne's work for the J-8 Directorate, and should allow for future upgrades and customization to fit additional user needs. There should be evidence of the vendor's commitment to product quality, ability to provide the needed training and support for the product, and financial stability.

Application of Criteria to Products

Having developed the criteria for evaluating the tools, the next step was to determine how closely each of the 32 remaining tools satisfied the criteria. Although no attempt was made to quantitatively order the criteria, the nine categories were ranked

qualitatively. Data-dictionary, multi-user, and diagramming capabilities were weighed more heavily than the other six categories; however, a tool displaying serious deficiencies in any of the remaining categories was eliminated. After gathering additional information by conducting telephone interviews, analyzing product literature in more detail, examining tool documentation, observing limited demonstrations of tools, and reading product reviews in technical publications, we were able to reduce the list to 15 tools.

2.3 PHASE 3: HANDS-ON EVALUATION

The next phase of the evaluation was performed by observing additional product demonstrations and through limited hands-on analysis. We obtained full or partial versions of many of the products and attended a CASE symposium where the remaining products were being demonstrated. The symposium also allowed for discussions with the vendors concerning their tools' functionality, limitations, and future development; it provided important insights into the CASE industry.

The demonstrations proved extremely valuable in judging the actual capabilities of the tools. In many instances, the vendor's claims and the tool's performance were not closely related. Using this information, the list was reduced to three tools that best met the evaluation criteria. This phase of the evaluation, summarized in Table 1 and Appendix A, reduced the number of tools under consideration to three.

2.4 PHASE 4: IN-DEPTH REVIEW

The three remaining tools were then assessed in detail. A full working version of each product was obtained. At this point, the analysis began to focus on how the tool could be integrated into the Argonne/J-8 working environment, including its ability to support the data base design methodology being developed by Argonne. We used each tool in a sample design session and thoroughly tested its ability to capture and report design data via diagramming, the functionality of its data dictionary, and its general usability. The three tools, described in the following sections, are Data-Station, a product of Charles River Development; DesignAid, by Nastec; and Blue/60 Data Modeler, from Advanced Logical Software.

Data-Station, from Charles River Development

- *Data dictionary.* Data-Station is built around a central data dictionary implemented as a relational data base using the Oracle data base management system. The dictionary is loosely based on the American National Standard Institute's (ANSI's) information resource dictionary system (IRDS) standard X3.138. It contains more than 80 relational tables with more than 600 data columns of information, including information about attributes, configuration changes, data elements, data element groups, data flows, data

TABLE 1 Summary of Phase 3 of the Evaluation

Screening Status and Vendor	Tool	Summary of Findings
<u>Passed Phase 3</u>		
Advanced Logical Software	Blue/60 Data Modeler	Strong methodology support Excellent graphics
Charles River Development	Data-Station	Excellent dictionary True multi-user capability
Nastec Corp.	Case 2000 DesignAid	Adequate dictionary support Excellent graphics
<u>Failed Phase 3</u>		
Bachman Information Systems	Data Analyst	IBM mainframe only
Cadre Technologies	Teamwork/IM	Limited entity types in dictionary
Chen and Associates, Inc.	ER-modeler	No dictionary support Poor graphics
Computer Systems Advisors Ltd.	POSE	No dictionary support Single user only
Index Technology, Inc.	Excelerator	Limited entity types in dictionary
LBMS, Inc	Auto-mate Plus	Weak dictionary support Single user only
	Super-mate	Weak dictionary support Single user only
McDonnell Douglas Corp.	Prokit Workbench	Weak dictionary support Single user only
Optima, Inc.	Designmachine	Weak dictionary support
	Designvision	No dictionary support Weak graphics
Systemoid	Consoi-ERM	No dictionary support Single user only
Yourdon, Inc.	Analyst/Designer Toolkit	Weak dictionary support

stores, documents, drawings, entities, fields, keys, modules, network links and nodes, processes, programs, projects, project versions, records, relationships, systems, tables, and users. Each object in the dictionary contains varied descriptive information. The dictionary ensures integrity via triggers.

- *Diagramming and graphics.* Data-Station is based on AutoCAD, a professional computer-aided design system. The graphics system can be used to prepare entity-relationship, data-flow, structure, and presentation diagrams. Pull-down menus provide the user interface, and a mouse or digitizer can be used. Included are commands for creating, editing, moving, scaling, and copying graphic objects. Custom shapes can be created dynamically during drawing preparation and stored for later use. A drawing generation facility can be used to automatically produce diagrams from the contents of the dictionary. The graphic system is integrally linked to the dictionary. Objects, relationships, flows, and annotations on a drawing are checked for correct syntax and compared to the dictionary to assess impact. Under the user's control, the diagram can be loaded into the dictionary according to a set of configuration rules.
- *Multi-user capabilities.* The data dictionary can be established on a distributed network server, providing multi-user access and control over the network. Individual workstations can concurrently access local and host data. Transaction management, concurrency, and rollback facilities are provided by the data base management system.
- *Methodologies.* Data-Station can be used in conjunction with a life-cycle methodology. It supports and integrates the predominant methodology and modeling techniques, including entity-relationship, relational, Yourdon/DeMarco, and Gane and Sarson. Import and export facilities are provided to allow integration with additional tools, such as normalization tools.
- *Environment support.* Data-Station runs on any IBM-compatible personal computer with an 80286 or 80386 processor and at least 30 megabytes of disk space and 2 megabytes of extended memory. In addition, it can be run in single- or multi-user configurations on a Sun workstation with a Unix operating system.
- *Import, export, and integration.* A set of utilities provides for data transfer, including import and export of projects, versions, specific objects, and selected groups of objects. Data can be imported from or exported to other Data-Station dictionaries or to ASCII files (for import or export to other data dictionaries or tools).

- *User interface and usability.* Data-Station provides a menu-based interface to the data dictionary and tool utilities. Pull-down menus and mouse support are provided for graphic operations. The interface is easy to use and gives consistent, logical access to the product.
- *Reporting.* Various analysis reports are provided to check consistency, compare diagrams to the dictionary, balance process and data flows, and identify foreign keys and relationships. More than 60 predefined detail and summary reports are supplied, covering all aspects of the dictionary. In addition, an SQL*Plus utility provides formatted reports for user-defined SQL queries. The queries may be saved for later use.
- *Miscellaneous.* Data-Station incorporates SQL-based security, allowing the administrator to grant users privileges on the tables in any storage area. Utilities are provided to access the data dictionary and automatically generate data-structure definitions for a variety of host languages. In addition, SQL "create table," "create index," and "create permit" statements can be generated for target SQL environments.

The dictionary is defined as a project to itself. The structure of the data dictionary is stored in the dictionary and can be accessed via any screen, report, query, or utility. In addition, the on-line help facility can be used to access the dictionary project.

The cost of Data-Station is approximately \$50,000 for a five- to nine-user configuration on a Sun workstation. The cost includes licenses for Data-Station, Run-time Oracle, and AutoCAD. A single-user version for a Sun workstation can be purchased for about \$20,000, and the single-user version for a personal computer costs about \$15,000.

DesignAid, from Nastec

- *Data dictionary.* Information for all DesignAid object types is stored in a central data dictionary. Each object type is composed of identical attributes: object name, object type, alias, description, data type, data size, initial value, constraints, date defined, defined by, lock status, locked date, and locked by. Additional information can be stored in the extended description field, which is stored as an additional text file. Since this additional information is stored as text, reporting is possible but analysis can only be performed via user-written utilities.

- *Diagramming and graphics.* DesignAid can produce entity-relationship, data-flow, structure, context-level, transition, and user-defined diagrams. Facilities are provided to validate diagrams for syntax and data and process balancing. Diagram objects are automatically inserted into the data dictionary, and the data dictionary can be accessed directly from the graphics facility.
- *Multi-user capabilities.* DesignAid's data dictionary can be maintained on a network server, allowing access by individual workstations on the network. Each object is stored in the dictionary and can be simultaneously accessed by each user. Consistency problems may arise if more than one user attempts to update an object simultaneously.
- *Methodologies.* DesignAid supports and integrates the predominant methodology and modeling techniques, including entity-relationship, Yourdon/DeMarco, Warnier-Orr, and Gane and Sarson.
- *Environment support.* DesignAid can operate on an IBM-compatible personal computer or in a Digital VAX workstation environment. The system supports the IBM PC-Network, Novell, and DECNET local area network.
- *Import, export, and integration.* DesignAid provides no facilities for importing or exporting data from or to the data dictionary. However, through user-written utilities and DesignAid's base modules, data in ASCII format can be transferred to external files and data dictionaries.
- *User interface and usability.* DesignAid is a menu-driven, user-friendly product. A mouse is supported for all menu operations. Although a user can learn the fundamentals of the system in little time, considerable time may be required to use the product to its full potential.
- *Reporting.* DesignAid offers 10 standard reports as well as user-definable report formats. Each object type can be reported on, although the number of attributes that can be reported for each object type is limited. Integrated word processing, graphics, and documentation management allow the user to develop customized reporting and documentation.
- *Miscellaneous.* The cost of DesignAid is \$6,900. A copy must be purchased for each hardware platform.

Blue/60 Data Modeler, from Advanced Logical Software

- *Data dictionary.* Blue/60's data dictionary stores data elements, entities, and relationships. In addition, entity-relationship diagrams are stored and can be accessed from the data model editor. The dictionary contains various information for each object type, including name, description, aliases, functional dependencies, usage statistics, data format, and relationships.
- *Diagramming and graphics.* Blue/60 provides facilities for the creation of only entity-relationship diagrams, supporting Bachman notation. Diagram syntax is enforced during its creation. Entities and relationships on a drawing can be compared to the dictionary to assess impact. Under the user's control, the diagram can be loaded into the dictionary.
- *Multi-user capabilities.* Blue/60 is a single-user tool only. Although data can be copied to many disks and distributed among users, data consistency cannot be ensured.
- *Methodologies.* Blue/60 is one of the few products to support the design of data bases using both entity-relationship and relational methodologies. Initially, the user enters the complete set of attributes into the data dictionary along with the functional dependencies among the attributes. At this point, the user may wish to normalize the data model or use the attributes to define entities. Normalization to third normal form is accomplished through functional dependency analysis.* Inconsistencies between the two types of methodologies are reported during normalization, allowing the user to specify the overriding methodology. The result of normalization is a set of relational tables in third normal form.
- *Environment support.* Blue/60 is available only for the Apple Macintosh.
- *Import, export, and integration.* Blue/60 provides no utility to import or export data-dictionary information.
- *User interface and usability.* Blue/60's interface is consistent, user friendly, and based on menus and icons. A mouse is supported by the Apple Macintosh. A consistent interface is provided for both data-dictionary access and diagramming. Blue/60 is very easy to learn and use, and the user manual is excellent.

*See Codd, E.F., *A Relational Model of Data for Large Shared Data Banks*, Communications of the ACM, 13(6):377-387 (1970).

- *Reporting.* Blue/60 provides several reports for each object type (attribute, entity, or relationship) as well as normalization reports used to correct anomalies in the data dictionary prior to normalization.
- *Miscellaneous.* The cost of Blue/60 is \$1,875 per copy.

Summary of Detailed Analysis

Table 2 summarizes the detailed analysis of Charles River Development's Data-Station, Nastec's DesignAid, and Advanced Logical Software's Blue/60 Data Modeler. The most significant differences among these products are the extent of multi-user access and the cost. While the products are not equivalent in the other categories, the differences require too extended a treatment to present here.

TABLE 2 Summary of Phase 4 of the Evaluation

Category	Data-Station	DesignAid	Blue/60 Data Modeler
Data dictionary	Comprehensive relational data-base management system	Minimally adequate proprietary structure	Minimally adequate proprietary structure
Diagramming and graphics	Excellent, but some functions unavailable	Excellent	Excellent
Multi-use capabilities	Yes	No	No
Methodologies	Entity-relationship, relational, Yourdon/DeMarco, Gane and Sarson	Entity-relationship, relational, Yourdon/DeMarco, Gane and Sarson, Warnier-Orr	Entity-relationship, relational
Environment support	IBM PC, Sun	IBM PC, VAX	Macintosh
Import, export, and integration	Yes	Minimal	No
User interface and usability	Excellent	Excellent	Excellent
Reporting	Excellent	Excellent	Excellent
Miscellaneous	\$15,000-20,000/copy or \$50,000 for nine-user version	\$6,900/copy	\$1,875/copy

3 CONCLUSIONS AND RECOMMENDATIONS

3.1 CASE SUMMARY

The development of data bases is not an integral part of traditional software development. Most CASE tools have little or no functionality in data analysis or data base development; some are no more than sophisticated drawing tools. In other tools, data modeling capabilities appear to have been added as an afterthought. Many of the tools appear to be hastily marketed products that were initially developed for in-house projects. While some available tools offer limited functionality to single users who are developing small data bases, they do not support team design of medium-scale to large data bases. In general, CASE has made considerable strides in the areas of diagramming and support of traditional software methodologies; however, the products' proprietary dictionary structures limit their potential.

3.2 EVALUATION SUMMARY

Charles River Development's Data-Station is the only tool whose functionality approaches the needs of Argonne's current efforts to develop data bases for the J-8 Directorate. This conclusion is based primarily on four major areas of concern:

- Data-Station is the only product evaluated that is a true multi-user system, a direct result of its underlying relational data base.
- Data-Station is one of the few products that support both entity-relationship and relational design (Advanced Logical Software's Blue/60 Data Modeler also has this feature).
- Data-Station is by far the most complete and comprehensive data base development tool evaluated. Its dictionary and reporting capabilities far exceed those of its closest competitors, features that also can be attributed to its underlying relational data base.
- The vendor appears willing to discuss Data-Station's limitations and incorporate possible modifications and extensions.

3.3 RECOMMENDED ACTION

No data base development tools now on the market will meet Argonne's long-term needs in its work for the J-8 Directorate. Only one tool evaluated, Charles River Development's Data-Station, begins to approach the required functionality. However, Data-Station's high cost, projected availability, and limited expandability reduce its desirability.

It is possible that a useful, powerful, flexible, and inexpensive tool for data base development will not be available in the immediate future. Thus, while using an existing

tool in Argonne's current data base development efforts remains an option, it may not serve the J-8 Directorate's long-term needs.

The combination of the evaluation of CASE tools and ongoing data base development at Argonne clearly points to the need for a central, comprehensive data repository to support the development effort, a need met by none of today's commercial products. In addition, continued work on a data base development methodology is needed to help identify candidate areas for automation.

The key to CASE success lies in the ability to integrate the tools with a central data dictionary and support the development methodology. Having established both a common repository for all design data and a standard methodology, individual tools can be identified to meet specific needs in the data base development life cycle, eliminating the need for tools to support their own data repository (a need that is a major weakness of existing tools). We recommend the course of action described below.

Develop an Information Resource Dictionary System

Argonne should continue its work to develop an information resource dictionary system that conforms to ANSI's X3.138 standard for data dictionary systems. The ANSI standard is designed to support all phases of a software system's life cycle, as well as other tasks; it specifies that all conforming products share a common set of features that are independent of implementation.

The benefits of this approach include the ability to integrate multiple IRDS products from different vendors, increasing user efficiency and improving data sharing among products. This approach is a departure from that of available products, whose different data-repository structures and contents limit their integration potential. In addition, adherence to the ANSI standard provides flexibility for future growth by supporting schema extensibility, which allows the user to customize the dictionary system. Our proposed research would focus on the following two tasks:

- *Design and implement an IRDS.* The ANSI standard specifies a basic set of facilities to be supported in a core system plus additional, more-advanced facilities to be supported in optional system modules. Argonne's development of an IRDS for the J-8 Directorate must focus on the suitability of the ANSI standard in the context of our requirements and include consideration of extensions to the standard. Following this investigation, the design and implementation of an IRDS would commence.
- *Develop an information resource dictionary (IRD) schema.* An IRD schema presents a conceptual view of the allowable metadata in the dictionary, defining the basic structures to be supported. The ANSI standard provides a number of predefined schema structures. These include the minimal schema (part of any IRDS), which contains descriptors that provide critical schema structures used in every system, and the basic functional schema, which supplies commonly

used schema structures that may be useful in many IRD applications. The power of the ANSI standard lies in its extensibility, which allows for customization by an organization to suit its specific information requirements. Investigation would continue to determine the information requirements of the J-8 Directorate. From this investigation, we would determine whether the minimal and basic functional schema would provide the needed support for our information requirements or whether schema extensions would be necessary.

Develop Methodology

Argonne should continue to evolve a data base development methodology, focusing on stepwise refinement of the methodology as well as integration of CASE tools. Stepwise refinement would help identify candidate areas that may be automated or semi-automated. Integration of CASE tools would focus on how specific design steps can be automated and whether an existing tool can meet these needs.

Evaluate and Acquire CASE Tools

Once candidate areas for automation were identified, studies would be conducted to determine if the needs would be best suited by a commercial tool or one developed in-house. New versions of existing commercial CASE tools, as well as new products, should continue to be evaluated. Technical journals and other sources must be reviewed to maintain an up-to-date data base of information on CASE tools.

Prototype CASE tools being developed by both commercial firms and universities should be investigated, to gain insight into the future direction and timing of CASE development. In addition, it may be possible to direct research toward a usable product via technical or financial assistance through Argonne.

APPENDIX A:

LIST OF CASE TOOLS AND RESULTS OF SCREENING PROCESS

Vendor	Tool	Results, by Screening Phase		
		1	2	3
Adpac	Diagraphics	fail		
Adpac	PM/SS	fail		
Advanced Logical Software	Anatool	PASS	fail	
Advanced Logical Software	Blue/60 Data Modeler	PASS	PASS	PASS
Ags Management Systems	Multi-Cam	fail		
Al Corporation	Intellect/DB2	fail		
American Management Systems	Life-cycle Productivity System	fail		
Applied Data Research, Inc.	Ideal	fail		
Arthur Anderson and Co.	Design/1	fail		
Arthur D. Little, Inc.	Irma	PASS	fail	
Arthur Young and Co.	AY/IEW	fail		
Asyst Technologies, Inc	The Developer	fail		
Bachman Information Systems	Data Analyst	PASS	PASS	fail
Borland International	Reflex	fail		
Cadre Technologies	Teamwork/IM	PASS	PASS	fail
Cadware	Sylva/Rule Tool	PASS	fail	
Cascade Software Systems, Inc.	Lyddia	PASS	fail	
Catalyst	Pathvu/Retrofit	fail		
CGI Systems, Inc	Pacbase	PASS	fail	
Charles River Development	Data-Station	PASS	PASS	PASS
Chen and Associates, Inc.	ER-modeler	PASS	PASS	fail
Cincom Systems	Mantis	fail		
Cincom Systems	Normal	fail		
Cincom Systems	Supra/Ultra	fail		
Computer Associates	CA-universe/PT	fail		
Computer Engineering Consultants Ltd.	Analysis Workbench	fail		
Computer Sciences Corp.	Design Generator	fail		
Computer Systems Advisors Ltd.	POSE	PASS	PASS	fail
Cortex Corporation	Corvision	fail		
Dacom, Inc	IDEF/Leverage	PASS	fail	

Vendor	Tool	Results, by Screening Phase		
		1	2	3
Data Administration, Inc.	Data Expediter	fail		
Digital Equipment	Case for Information	fail		
Digital Research, Inc.	Gem Draw	fail		
Disus	DEFT	PASS	fail	
Etech	Softrobot	fail		
Hewlett-Packard	HP Teamwork	fail		
Holland Systems Corp	Design4data	fail		
I-Logix, Inc	Statemate	fail		
IBM	RDT	fail		
IDE	Software through Pictures	fail		
Index Technology, Inc.	Excelerator	PASS	PASS	fail
Index Technology, Inc.	Excelerator/RTS	PASS	fail	
Integrated Systems, Inc.	Autocode	fail		
James Martin Associates, Inc.	Information Engr Methodology	fail		
Jordan-Webb Info Systems Ltd.	Exsys Data Modeler	fail		
Knowledgeware, Inc.	Data Designer 2	PASS	fail	
Knowledgeware, Inc.	IEW	PASS	fail	
Language Technology, Inc.	Inspector/Recorder	fail		
LBMS, Inc.	Auto-mate Plus	PASS	PASS	fail
LBMS, Inc.	Super-mate	PASS	PASS	fail
Management Science America, Inc.	Information Expert	fail		
Manager Software Products	Designmanager	fail		
Manager Software Products	Managrview	fail		
M Bryce and Associates, Inc.	Pride-DBEM	PASS	fail	
M Bryce and Associates, Inc.	Pride-IRM	PASS	fail	
McDonnell Douglas Corp.	Prokit Workbench	PASS	PASS	fail
Meta Systems	PSL/PSA	fail		
Meta Systems	Structured Architect	fail		
Micro Focus, Inc.	Cobol/2 Workbench	fail		
Mitrol, Inc.	Mfg 2 Results Oriented Info Pak	fail		

Vendor	Tool	Results, by Screening Phase		
		1	2	3
Must Software International	Nomad	fail		
Nastec Corp.	Case 2000 DesignAid	PASS	PASS	PASS
Netron, Inc.	Netron/Cap	fail		
On-line Software International, Inc.	UFO	fail		
On-line/Database Software, Inc.	Tab	fail		
Optima, Inc.	Designmachine	PASS	PASS	fail
Optima, Inc.	Designvision	PASS	PASS	fail
Oracle Corp.	SQL Design Dictionary	PASS	fail	
Pansophic Systems, Inc.	Telon	fail		
Parameter Driven Software, Inc.	PDS-Adept	fail		
Popkin Software and Systems, Inc.	System Architect	fail		
Promod, Inc.	Promod	fail		
Sage Software, Inc.	APS Development Center	fail		
Softlab, Inc.	Maestro	fail		
Sun Microsystems	Schemadesign	PASS	fail	
Sun Microsystems	Sunsimplify	PASS	fail	
Syscorp International	Microstep	fail		
Systemoid	Consoi-ERM	PASS	PASS	fail
Texas Instrument	Information Engineering Facility	PASS	fail	
Transform Logic Corp.	Transform	fail		
Unicorn Systems Co.	VMCICS	fail		
Visible Systems Corp.	Visible Analyst Workbench	PASS	fail	
Yourdon, Inc.	Analyst/Designer Toolkit	PASS	PASS	fail

APPENDIX B:
EVALUATION CRITERIA

1 Data Dictionary

- 1.1 Is the dictionary based on a relational data base management system or some proprietary structure?
- 1.2 What object types does the dictionary support?
- 1.3 Are user-definable object types supported?
- 1.4 Can the user define rules for object types?
- 1.5 Does the dictionary enforce consistent usage of objects?
- 1.6 Does the dictionary support the definition of structures (i.e., relationships or data flows among data objects)?
- 1.7 Can the dictionary record where and when information about an object is referenced?
- 1.8 Does the dictionary support multimodel cross-referencing?
- 1.9 Does the dictionary support multilevel descriptions?
- 1.10 Does the dictionary support design versioning?
- 1.11 Can multiple dictionary designs be easily merged?
- 1.12 Is a dictionary change immediately available to other users and to all models referencing it?
- 1.13 Is a history of dictionary modifications maintained?
- 1.14 Does the dictionary support impact analysis of changes to dictionary entries?
- 1.15 Does the dictionary support SQL (or equivalent) user access?
- 1.16 Does the dictionary support view access?

2 Diagramming and Graphics

- 2.1 Does the tool support a graphical representation of the data (e.g., entity-relationship)?
- 2.2 Does the tool support automatic design layout?
- 2.3 Does the tool support integrated graphic and textual descriptions?
- 2.4 Does the tool provide locator assistance for objects on the diagram?
- 2.5 Does the tool support design highlighting?
- 2.6 Are graphic symbols user definable?
- 2.7 Can graphic symbols be easily moved, copied, or deleted?
- 2.8 Does the tool support automatic reconnection of lines?
- 2.9 Are undo facilities provided?
- 2.10 Does the tool support resizing (zooming) for text/graphics?
- 2.11 Can line routing be user controlled?

3 Methodologies

- 3.1 What data models does the tool support (entity-relationship, relational, others)?
- 3.2 Does the tool support an object-oriented modeling approach?
- 3.3 Does the tool support the following semantic concepts?
 - Functional/multivalued dependencies
 - Attribute aliasing
 - Role information
 - Integrity constraints
- 3.4 Does the tool support a "top-down" design approach?
 - Entity-relationship to relational
 - Relational view integration
 - Other? Specify.
- 3.5 Does the tool support a bottom-up design approach?
 - Relational to entity-relationship
 - Source code to relational? Entity-relationship?
 - Other? Specify.
- 3.6 Are the data and process design functions integrated?
- 3.7 Does the tool support each phase of the design cycle?
 - Information planning
 - Requirements collection
 - Enterprise/conceptual design
 - Logical design
 - Local physical design
 - Distributed physical design
- 3.8 Does the tool provide design review and verification functions?
- 3.9 Does the tool support automatic model-to-model conversion?
- 3.10 Does the tool support automatic normalization?
- 3.11 Does the tool perform consistency checking and conflict analysis?
- 3.12 Is the tool part of a larger CASE package?
- 3.13 Does the tool support retrieval/update supportable view analysis?
- 3.14 Does the tool use an expert system? How?
- 3.15 Does the tool support the following physical design concepts?
 - Primary storage structure selection
 - Index selection
 - Access path definition
 - Distributed data partitioning
 - Distributed query formulation
 - Interface to a data base management system
 - Other? Specify.

4 Multi-user Capabilities

- 4.1 Does the tool support parallel design specification?
- 4.2 Can the dictionary support multi-user access?
- 4.3 Does the dictionary provide locking capabilities and, if so, at what granularity?
- 4.4 Does concurrent dictionary access produce unacceptable file server delays?
- 4.5 Does the tool support integration of design specifications?

5 Environment Support

- 5.1 What hardware does the tool require?
 - Mainframes
 - Workstations
 - Personal computers
- 5.2 What operating systems does the tool require?
- 5.3 What are the tool's minimum memory/disk requirements?
- 5.4 Does the tool require any special hardware or software?
- 5.5 Are there performance limitations due to the following?
 - Tool limitations on design size
 - Memory availability
 - Processor speed
 - Relative speed of tool
- 5.5.5 Graphics subsystem
- 5.5.6 Dictionary/multi-user subsystem

6 Import, Export, and Integration

- 6.1 Does the tool support an import/export facility for establishing a bridge to a host dictionary?
- 6.2 Can designs be downloaded to a data base management system dictionary? Which one(s)?
- 6.3 Can SQL specifications be generated?
- 6.4 What is the format of import/export files?

7 User Interface and Usability

- 7.1 Is the interface easy to learn and use?
- 7.2 Are windowing or mouse functions provided?
- 7.3 Does the tool support interactive designs?
- 7.4 Does the tool support expert and novice design modes?
- 7.5 Can the full capabilities of the tool be learned quickly?
- 7.6 Is the tool's documentation easy to use?
- 7.7 Is the tool's documentation complete and well organized?

8 Reporting

- 8.1 Does the tool support the documentation needs of each phase of the design process?
- 8.2 Does the tool support automatic documentation generation?
- 8.3 Does the tool support user-definable report specifications?
- 8.4 Can the reports be edited using a word processor?
- 8.5 Can the reports be saved?

9 Miscellaneous

- 9.1 What is the tool's cost?
- 9.2 Is the tool expandable? How?
- 9.3 Does the vendor provide tool support and training?
- 9.4 Is the tool accepted in the user community?