

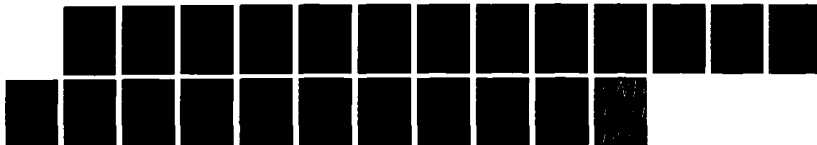
AD-A185 951

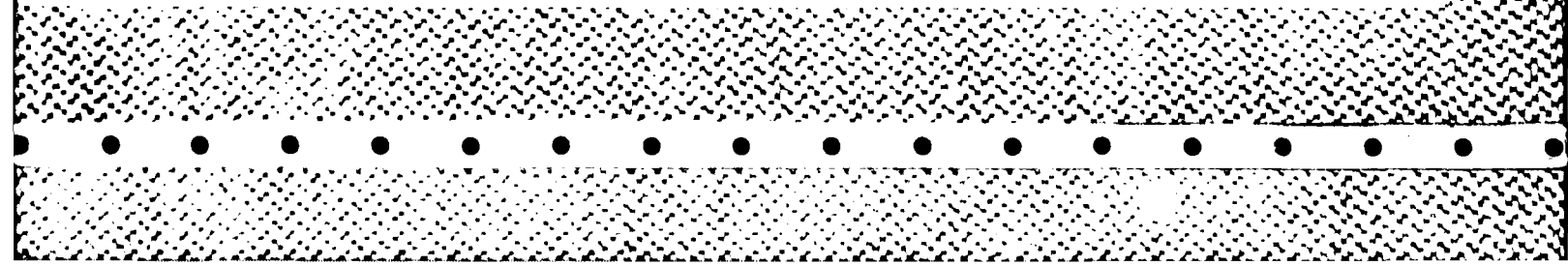
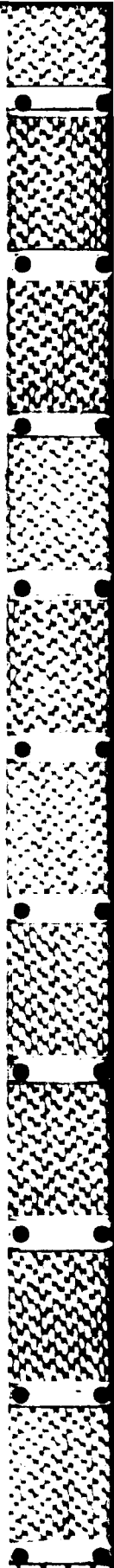
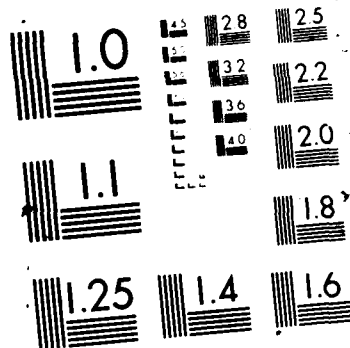
DOD GATEWAY INFORMATION SYSTEM (DGIS): COMMON COMMAND
LANGUAGE MAPPING(U) DEFENSE TECHNICAL INFORMATION
CENTER ALEXANDRIA VA OFFICE OF I. R L BIXBY OCT 87
DTIC/TR-87/20 F/G 12/5

1/1

UNCLASSIFIED

ML





①

AD-A185 951

**DOD GATEWAY INFORMATION SYSTEM (DGIS):
COMMON COMMAND LANGUAGE MAPPING**

Randy L. Bixby

October 1987

**D
T
I
C** Defense
Technical
Information
Center

DTIC CCL Report No. 2

Office of Information Systems and Technology

Cameron Station, Alexandria, VA 22304-6145

AD-A185 951

ON PAGE

DTIC MARKINGS

DTIC FILE COPY

1a. REPORT SECURITY CLASSIFICATION
Unclassified/Unlimit

2a. SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE

3. DISTRIBUTION/AVAILABILITY OF REPORT

Approved for public release; distribution unlimited

4. PERFORMING ORGANIZATION REPORT NUMBER(S)

DTIC/TR-87/20

5. MONITORING ORGANIZATION REPORT NUMBER(S)

6a. NAME OF PERFORMING ORGANIZATION
Defense Technical Information Center

6b. OFFICE SYMBOL
(if applicable)
DTIC

7a. NAME OF MONITORING ORGANIZATION

6c. ADDRESS (City, State, and ZIP Code)

Cameron Station
Alexandria, VA 22304-6145

7b. ADDRESS (City, State, and ZIP Code)

NOV 23 1987

8a. NAME OF FUNDING/SPONSORING ORGANIZATION

8b. OFFICE SYMBOL
(if applicable)

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

8c. ADDRESS (City, State, and ZIP Code)

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.
65801S

PROJECT NO.

TASK NO.

WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)

DoD Gateway Information System (DGIS): Common Command Language Mapping

12. PERSONAL AUTHOR(S)
Randy L. Bixby

13a. TYPE OF REPORT

13b. TIME COVERED
FROM TO

14. DATE OF REPORT (Year, Month, Day)
8710

15. PAGE COUNT
20

16. SUPPLEMENTARY NOTATION

DTIC CCL Report No. 2

17. COSATI CODES

FIELD	GROUP	SUB-GROUP
5	2	
12	5	

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)

Common Command Language, CCL, C language, PROLOG, Artificial Intelligence, DoD Gateway Information System, DGIS

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

The DoD Gateway Information System (DGIS) Common Command Language (CCL) project was initiated to provide a single standard command language for searching any online database system. This report relates the requirements analysis for the initial CCL prototypes in C language. Requirements were determined for six online database systems: DROLS, DIALOG, NASA/RECON, BRS, ORBIT, and MATRIS, using the National Information Standards Organization (NISO) draft standard for Common Command Language as a basis. Four C prototypes were completed: DROLS, DIALOG, BRS, and NASA/RECON. At that time the decision was made to incorporate Artificial Intelligence Capabilities into CCL, and the programming of the CCL translators was converted to PROLOG.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT

UNCLASSIFIED/UNLIMITED SAME AS RPT. DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION

Unclassified/Unlimited

22a. NAME OF RESPONSIBLE INDIVIDUAL
Randy L. Bixby

22b. TELEPHONE (Include Area Code)
202/274-5367

22c. OFFICE SYMBOL
DTIC-EB

DOD GATEWAY INFORMATION SYSTEM (DGIS): COMMON COMMAND LANGUAGE MAPPING

EXECUTIVE SUMMARY

One of the major problems in accessing multiple diverse online database systems is the necessity to learn the native command language for each system. DTIC's Common Command Language (CCL) project, one facet of DoD Gateway Information System (DGIS), will provide a single standard language for searching any database system, thus eliminating the need to learn multiple command languages. This second report on CCL relates the requirements determination for the initial CCL prototypes in C language.

CCL adopted the standard command language proposed by the National Information Standards Organization (NISO). The NISO commands were divided into functional groups and corresponding native commands from six database systems were determined. Requirements for the native commands were mapped and the information applied to translations of those commands into CCL via C language. Database systems prototyped were DROLS, DIALOG2, BRS, and NASA/RECON.

The prototypes written in C language proved inadequate for many of the desired CCL capabilities. Simple command substitution was not enough: CCL needed to understand responses from the host system in order to conduct coherent conversations between CCL and the host. A more human-like relationship between the searcher and the database systems was required. This led to the decision to apply Artificial Intelligence (AI) capabilities to CCL. Therefore, the translation prototype effort shifted from C language to Prolog, a fifth generation relational language based on logic. C continues to support the interaction between UNIX (system and user) and Prolog (CCL).

The introduction of AI to CCL will enhance CCL by providing increased flexibility and the capability to accommodate search functions not available in all database systems, such as saving search strategies, estimating search costs, and reviewing entire session histories. CCL will thus provide true universality across diverse database systems.

Contents

Executive Summary	i
Introduction	1
Background	1
Method	2
The Future	6
Conclusions	8
Acknowledgement	8
References	9
Appendix A - Table 1, American National Standard For Information Sciences--Common Command Language for Online Interactive Information Retrieval	A-1
Appendix B - Common Command Language Mapping Formats	B-1

The DoD Gateway Information System (DGIS): Common Command Language Mapping

Randy L. Bixby

Introduction

The problem of multiple command languages has plagued online searchers since the creation of the second online database search system. Although most database searching functions are comparable from one system to another, each database system has developed its own native commands, either from a desire to be unique, or in the belief that it has truly found the best way to search. The ever-expanding number of online database systems has resulted in a plethora of command languages. As a result, online searchers have often had to limit their searching to only one or a few database systems, not because of expense or lack of access, but because they simply do not have time to become proficient in the many diverse native command languages.

The Defense Technical Information Center (DTIC) is doing something to remedy this problem. In April 1986, DTIC began working on the design of one common command language (CCL) for online database searching. The purpose of CCL is to allow searching of multiple, diverse databases using a single set of commands. This paper outlines the mapping of requirements for the CCL command translator.

Background

DTIC is charged with providing information services to the Department of Defense (DoD) scientific and technical community. These services range from collecting and disseminating bibliographic information to sponsoring and directing research into innovative information handling techniques. One of these efforts is the DoD Gateway Information System (DGIS), a long-term, ongoing project to provide online, streamlined methods for identifying, accessing, searching, and analyzing data from heterogeneous databases of interest to the Department of Defense. CCL is one component of DGIS. Other components include remote database connection routines, a directory of resources, post-search data

processing and analysis routines, and simultaneous search capabilities. Considerable progress had already been made on other DGIS components when work began on CCL. As CCL becomes functional, it will be incorporated with the other DGIS capabilities.

The CCL project team consists of three people. The project officer is a program analyst at DTIC. The requirements analyst is a technical information specialist with a library science background, also at DTIC. And the programming expertise is provided by a technical expert consultant. This team was established to bring together three different viewpoints and backgrounds in the area of computerized information. The program analyst had extensive experience with other components of the DGIS project, so could provide links with those efforts, historical perspective, and a broad view of DGIS as a whole. The technical expert had worked extensively in the C programming language, used in the UNIX operating system on which DGIS runs. The requirements analyst had hands-on experience with several online database systems, and had encountered the problems associated with multiple command languages. Experts in other fields, both within and outside DTIC, are consulted when the need arises.

The CCL project team decided at the outset that the primary approach was to be a practical one - in the shortest possible time, a working prototype CCL system, no matter how simple, was to be functional. Refinements would be added later, after the basic problems were solved. The first critical component of this process was command language requirements and mapping.

Method

The first step in the CCL requirements mapping process was a literature search, to see what had already been done in the field. However, in keeping with the project team's desire to have something up and running in the shortest possible time, no effort was made for the search to be comprehensive. Emphasis was placed on getting enough basic information together to get started.

The search revealed that many authors in the information retrieval field were concerned about the lack of standardization in the multitude of database systems available, and the apparent need for a common command language to alleviate the difficulties in searching multiple database systems. Compilations of command language comparisons had been

prepared by several researchers, the most extensive being Lucinda Conger's Online Command Chart. The comparison chart format used by Conger and others was initially adopted as a working model for the CCL project. But it was soon determined that more than simple command comparisons was needed for what the CCL project team wanted to do. And the question of which command language should be used as the standard had to be resolved before further work could be done. The CCL project team decided, in the interest of adherence to U. S. standards, to make use of a parallel concurrent effort by the National Information Standards Organization (NISO).

On 20 February 1986 NISO Committee Z39 issued a draft standard, American National Standard--For Information Sciences Common Command Language for Online Interactive Information Retrieval. The NISO draft standard "specifies the vocabulary, syntax, and operational meaning of commands in a command language for use with online interactive retrieval systems." The NISO commands were adopted as the standard for the CCL project, and all command mapping efforts were based on them. Appendix A shows Table 1 of the NISO Draft Standard, with a brief synopsis of the adopted commands.

The NISO commands were divided into four functional groups. These groups were:

Group1 START
 CHOOSE
 FIND
 DISPLAY
 STOP

Group 2 SCAN
 MORE
 BACK
 RELATE

Group 3 PRINT
 REVIEW
 SORT
 SAVE
 DELETE

Group 4 EXPLAIN
 HELP
 SHOW
 SET
 DEFINE

Group 1 consists of the minimum command set required for online searching: logging on to a system, choosing the database or file to be searched, identifying the search terms, displaying search results on the user's terminal, and logging off the system.

Group 2 adds more searching capabilities and the relationship between terms. Group 3 provides commands for manipulating result sets, and Group 4 consists of user-oriented commands.

The CCL project team also identified three command functions not defined in the NISO Draft Standard which were deemed to be either necessary or extremely useful. These commands were:

COMBINE - to create new sets from various combinations
 of already existing sets
EXECUTE - to execute a previously saved search strategy
COST - to display estimated expense of a search

These commands were added to groups 2, 3, and 4, respectively.

The project team next chose the database systems for the initial prototyping of CCL. The team wanted systems already familiar to DoD information seekers, or systems likely to be useful to them. A range of system capabilities and enhancements, or lack thereof, was also desired. The database systems needed to be readily available to the Defense Community. And the systems had to offer at least some files constructed with tagged formats, a requirement for the DGIS post-processing function. With these criteria in mind, the following database systems were selected as the first group to be implemented:

1. DROLS (Defense RDT&E OnLine System), developed and maintained by Defense Technical Information Center, Alexandria, VA. This database system contains bibliographic citations and other information on technical reports, work unit summaries, and planning data on defense-related research. DROLS is available to eligible DoD sites who register with DTIC.

2. DIALOG2. This database system includes over 200 files in a wide range of subject areas. DIALOG2 is commercially available through DIALOG Information Services, Inc., Palo Alto, CA.

3. BRS. This database system includes nearly 100 files in the life sciences, medicine and pharmacology, physical and applied sciences, and business. BRS is commercially available through BRS Information Technologies, Latham, NY.

4. ORBIT includes nearly 70 files in the sciences, business, patents, statistics, engineering, government, education, and library and information science. It is commercially available from Pergamon ORBIT, Inc., McLean, VA.

5. NASA-RECON. This system is maintained and made available through the NASA Scientific and Technical Information Facility at BWI Airport in Maryland. NASA-RECON consists of numerous files dealing primarily with aeronautics, astronautics, geosciences, mathematics, computer sciences, and engineering. The database system is available to qualified government users and NASA contractors who register with NASA.

6. MATRIS (Manpower & Training Research Information System). MATRIS is administered by the Defense Technical Information Center in Alexandria, VA, and is jointly sponsored by the DoD Office of the Under Secretary of Defense/Research and Engineering (OUSD/R&E) and the Office of the Assistant Secretary of Defense/Force Management and Personnel (OASD/FM&P). MATRIS contains information about planned, ongoing, and recently completed DoD research in the areas of Human Factors, Manpower and Personnel, Education and Training, and Simulation and Training Devices. Those interested in MATRIS information must register with DTIC.

DOE/RECON was considered for CCL prototyping. However, DOE no longer offers its unclassified databases directly. Because DGIS is currently unclassified only, and DIALOG's DOE ENERGY file contains all unclassified information processed at DOE's Technical Information Center, the CCL project team postponed mapping of DOE/RECON.

This group of six selected database systems presented a wide variety of characteristics and levels of sophistication to be incorporated into CCL. Together, the systems would provide a challenge for initial CCL translator

programming efforts, and a valuable learning experience leading to succeedingly sophisticated CCL versions.

The CCL mapping effort involved much more than a comparison of command words. Quick-reference command comparison charts such as used by Congers and others proved too limited for what the project team needed. Translator programming requirements included information on command syntax, alternate command input such as abbreviated forms, characters allowed and prohibited, maximum length of input, any system idiosyncrasies, and the system response to each command. The mapping format evolved several times, as programming requirements for the CCL translators became more defined. The final format, besides providing the information needed to program the translators, is to be used as the basis for both an online help system and a printed CCL manual. Appendix B shows the evolution of the command language mapping format.

Commands were mapped using database system user guides and reference manuals, conversations with database system "help desk" personnel, and actual searching of the various database systems.

Information was gathered for the Group 1 commands for each of the six selected database systems. CCL prototypes were implemented in C programming, using two UNIX tools, LEX and YACC, for command translation. The advantage of this approach was that it is based on the standard language and tools of UNIX, thus making CCL highly portable within the UNIX world. In September 1986, DIALOG2 became the first working CCL prototype, followed by DROLS, BRS, and NASA/RECON. When these four prototypes were completed, commands from the other Groups were added. The C prototypes also provided online help - the user could access a manual page explaining the CCL command, its translation, and its usage with respect to a host database. UNIX shell commands could also be executed.

The Future

The CCL prototypes written in C brought to light inadequacies in the C language-based CCL. The prototype translators revealed that each database system is individualistic and must be treated as such. The translator programming is totally dependent on the mapping requirements stated for each system. The basic problem was that simple command substitution, provided by algorithmic programming as in C, was not sufficient for CCL. It was very difficult to support CCL commands for

which there was no counterpart in the host system (such as COMBINE in DROLS). When using CCL, a searcher is conversing with a system, not retrieving from it. The user tells CCL what to do, and CCL tells the target system what to do. CCL also needs to understand the response from the host system: returning results, outputs, and messages from the database. This need is critical if there is to be a coherent conversation between the database and the user through CCL. Third generation programming languages have problems handling such conversations.

With this concept in mind, therefore, the decision was made in April 1987 to apply Artificial Intelligence (AI) capabilities to CCL. The purpose of AI was to make the human/machine relationship and interaction more human-like. AI is also the means by which CCL can provide universality across database systems by accommodating functions not available in all systems, such as saving search strategies and reviewing session history.

The fifth generation tool chosen for the AI applications was Prolog, a simple but powerful relational language based on logic. At the time the decision to go AI was made, prototype efforts in C were discontinued, except to support the interaction between UNIX (system and user) and Prolog (CCL).

Work is currently (September 87) being done on the Prolog versions of CCL-DROLS and CCL-DIALOG. All command Groups are being programmed simultaneously. A number of knowledge bases are being established, one of which will be common to all database systems on CCL, others unique to each particular system. When the DROLS and DIALOG modules are completed, BRS, ORBIT, MATRIS and NASA/RECON will be added. A decision will be made at that time as to the direction and scope the project will take, and the protocol for adding more database systems.

CCL is anticipated to produce a multi-mode system in which searches may be performed on any database system using CCL commands, and the response received in either native mode or, for functions not available in the native system, in CCL. Also planned is the capability to search any database system using any command language, e.g., searching DROLS using DIALOG commands. Searching a system in its own native language will, however, always be retained as an option. This approach allows tremendous flexibility, and makes DGIS a much more useful information tool, especially for DoD information seekers who are not online searching experts.

Conclusions

DTIC's Common Command Language project has made definite, tangible progress in the development of one command set capable of searching multiple online database systems. AI technology is now being applied to CCL development. Prolog-driven CCL-DROLS is already in prototype testing, with DIALOG, BRS, ORBIT, MATRIS, and NASA-RECON soon to follow. DTIC's goal for CCL is to make a reality the online database searcher's dream for an easier way to access multiple diverse databases.

ACKNOWLEDGMENT

Mr. Allan D. Kuhn is the Project Officer for DGIS Common Command Language Design. Mr. Duc Tien Tran, Consultant, Control Data Corporation, is the programming technical expert.

References

Auger, Peter R., "My Wish is Your Command?", 5th International Online Information Meeting, London, 8-10 December, 1981, p. 433-445.

Cochrane, Pauline A., "Can a Standard for an Online Common Command Language be Developed?", Online, January 1983, p. 36-37.

Conger, Lucinda, Online Command Chart, Second Edition, Online, Inc., Weston, CT, 1982.

Cotter, G. A., The DoD Gateway Information System. October 1985, ADA-161-701, Defense Technical Information Center, Alexandria, VA.

Cotter, G. A., The DoD Gateway Information System: Prototype Experience. April 1986, ADA-166-200, Defense Technical Information Center, Alexandria, VA.

Cotter, G. A., and Kuhn, A. D., The DoD Gateway Information System (DGIS): User Interface Design. September 1986, ADA-174-150, Defense Technical Information Center, Alexandria, VA.

Harter, Stephen P., and Peters, Anne Rogers, "Heuristics for Online Information Retrieval: A Typology and Preliminary Listing," Online Review, Vol. 9, No. 5, October 1985, p. 407-424.

Kuhn, Allan D., with Bixby, Randy L. and Tran, Duc Tien (consultant), DoD Gateway Information System (DGIS) Common Command Language: The First Prototyping and the Decision for Artificial Intelligence. August 1987, Defense Technical Information Center, Alexandria, VA.

National Information Standards Organization, American National Standard--For Information Sciences Common Command Language for Online Interactive Information Retrieval (DRAFT), February 1986.

Negus, A. E., "Why Standards for Command Languages?", 5th International Online Information Meeting, London, 8-10 December, 1981, p. 425-431.

Negus, Alan E., "Development of the EURONET-DIANE Common Command Language," 3rd International Online Information Meeting, London, 4-6 December, 1979, p. 95-98.

Verheijen-Voogd, C., "Is EURONET C.C.L. a 'Common' Command Language?", Online Review, Vol. 5, No. 5, October 1981, p. 399-401.

Wells, Christine A., "Quick Reference Command Languages Chart," Online Review, Vol. 7, No. 1, January 1983, p. 45-49.

APPENDIX A

TABLE 1

AMERICAN NATIONAL STANDARD
FOR INFORMATION SCIENCES

COMMON COMMAND LANGUAGE
FOR ONLINE INTERACTIVE INFORMATION RETRIEVAL

(DRAFT)

Common Command Language for Online Interactive Information Retrieval

Table 1
Primary Command Words and Abbreviations

Command Word	Abbreviation	Function
BACK	BAC	To view data preceding displayed data or items on a list.
CHOOSE	CHO	To select file(s) or database(s) to be searched.
DEFINE	DEF	To create user macros, to rename a command word, or to name a command expression with a word.
DELETE	DEL	To delete searches, saved items, PRINT requests, or defined commands.
DISPLAY	DIS	To view online the results of searches of the database(s).
EXPLAIN	EXP	To obtain information about the system, its use, and its database(s).
FIND	FIN	To enter a search statement.
HELP	HEL	To directly obtain online assistance or instruction specific to the context of the interaction.
MORE	MOR	To view continuing data, or data following displayed data or items on a list.
PRINT	PRI	To request offline printing.
RELATE	REL	To view terms logically related to search term.
REVIEW	REV	To view search history.
SAVE	SAV	To save search strategies for subsequent use.
SCAN	SCA	To view an ordered list of index terms.
SET	SET	To set or override default parameter values.
SHOW	SHO	To view session parameter default values and non-instructional system or session information.
SORT	SOR	To arrange search results by specified field(s).
START	STA	To initiate a session.
STOP	STD	To terminate a session.

APPENDIX B
COMMON COMMAND LANGUAGE
MAPPING FORMATS

Format 1

GROUP COMMANDS

<u>Command</u>	<u>Function</u>	<u>Vendor</u>	<u>Most Common Syntax</u>	<u>Optional Syntax</u>
CHOOSE	To select a file or database once user is logged onto a system	DROLS	@SXX@	@STR@ @SWU@ @SCF@
		NASA-RECON	BB A	BB 1 B A
		DIALOG2	B _	Begin _
		BRS	..change/xxxxx	..change <CR> (prompts for name)
		ORBIT	File xxxxx	

COMMON COMMAND LANGUAGE

COMMAND

bb__

SYNOPSIS

Selects a file to be searched.
Clears from the computer memory and disk storage all unsaved data from any previous search.

DATABASE SYSTEM

NASA/RECON

NISO CORRESPONDING

COMMAND

CHOOSE

NISO DESCRIPTION

Selects file(s) or database(s) to be searched.

NOTES

Blank can be filled with either an alpha file designator or the corresponding numeric designator at file collection.

SEE ALSO

(in target database)

REFERENCES

NASA/RECON User's Reference Manual p. 3-3.

CCL

Syntax

FIND

Description

To enter a search statement

Notes

The FIND command invokes a search of one or more of the system's databases. The FIND command requires a specification, which may consist of one or more search elements.

CL - DIALOG2

Syntax

SELECT <search term>

Description

The SELECT command is used to set aside index terms or groups of terms in numbered sets for later manipulation with Boolean operators.

Notes

- o SELECT can be abbreviated S.
- o <search term> may consist of one or more terms.
- o The DIALOG2 system accepts search terms with or without a space between the command and the term, e.g.,

SELECT LASER = SELECTLASER

(If use of the abbreviated SELECT command S in combination with the search term will inadvertently spell out a command, there must be a space between the command and the term:

```
?S ELECTRONIC
  2 2828 ELECTRONIC
?SELECTRONIC
  3   0 RONIC
```

CCL

Syntax
FIND

Description

To enter a search statement.

Output

The system response shall include a result set identifier and the number of "hits" or retrievals.

Notes

The FIND command invokes a search of one or more of the system's databases. The FIND command requires a specification, which may consist of one or more search elements.

CL - DIALOG2

Syntax
SELECT <search term>

Description

The SELECT command is used to set aside index terms or groups of terms in numbered sets for later manipulation with Boolean operators.

Output

When a search has been completed, the system responds with a line, e.g.,

1 8918 TELEVISION

where 1 = the automatically assigned set number
8918 = the number of records (postings or hits) in the database being searched that include the word TELEVISION.

Notes

- o <search term> may consist of one or more terms.
- o The DIALOG2 system accepts search terms with or without a space between the command and the term, e.g.,

?SELECT LASER = SELECTLASER

END

12-87

DTIC