

SYSMAIN Utility

The SYSMAIN utility, which is available online and in batch mode, is used to transfer objects within the Natural system from one environment to another.

- Libraries
- Maintaining Libraries
 - List Function
 - Find Function
 - Copy, Move and Rename Functions
 - Delete Function
 - Import Function
- Invoking SYSMAIN by a Subprogram
 - Direct Commands
 - Additional Keywords for Direct Commands

Libraries

Libraries are used to store Natural objects such as programs, subprograms, subroutines, help routines, and text.

Some libraries are delivered with Natural, such as libraries beginning with the prefix "SYS". The library SYSTEM, for example, contains system-related programs, error messages, and DDMs (data definition modules).

When you first logon to Natural, your default library will in most cases be "SYSTEM". Because this library is generally full of system-related objects, you will probably want to use a different library to store your programs. You can add your own libraries to the Natural environment or use libraries set up for this purpose.

Note:

You can alter your default library from "SYSTEM" to any other library by using the Natural profile parameter INIT-LIB (see Installing and Setting Up Natural on UNIX or Installing and Setting Up Natural on OpenVMS).

To log on to a library, select "Library" on the Main Menu; a list appears showing the function <LOGON> followed by a list of all existing libraries:

```

2002-02-15                NATURAL                Library: SAG
14:34:19                  V 5.1.1 Software AG 2002      Mode   : STRUCTURED
User: SAG                  Work Area   : empty
.....
·Library      Direct      Services      OS          Fin      ·
.....
.....
· <LOGON>    ·
· DEMO1     ·
· DEMO2     ·
· DEMO3     ·
· SAMPLES   ·
· SYSEXPG   ·
· SYSEXRM   ·
· SYSTEM    ·
.....

Select Library

```

To create a new library, you select <LOGON>. A window is then displayed in which you enter the name of the library to be created. For information on naming conventions for libraries, see the system command LOGON.

To use an existing library, you either select the library from the list, or select <LOGON> and then enter the name of the desired library. The contents of the library are then displayed:

```

..... List * * .....
· Cmd Name      Type          SM S/C User ID   SRC Date      GP Date      ·
·-----
· <DIRECT COMMAND>
·   ARRAYD     Program      S  S   SAGPC    14:35 02-02-15
·   ARRAYE     Program      S  S/C SAGPC    14:35 02-02-15 11:25 02-02-18
·   BREAK1     Program      S  S/C SAGPC    14:35 02-02-15 11:25 02-02-18

```

Field	Explanation
Cmd	<p>In this field, you can enter a command code. To get a list of the possible commands, you press F2. Only the commands valid for the object selected are shown.</p> <p>C - Check: Checks source program for syntax errors.</p> <p>D - Read: Reads object's source program into work area.</p> <p>E - Edit: Reads object's source program into work area.</p> <p>L - List: Lists object's source program.</p> <p>H - Hardcopy: Sends copy of object to printer.</p> <p>R - Run: Compiles and executes source.</p> <p>S - Stow: Stores object in source and object form.</p> <p>U - Scratch: Deletes object in source and object form.</p>
Name	Object name.
Type	Object type.
SM	Programming mode (R = reporting mode, S = structured mode).
S/C	<p>S - The object exists in source form.</p> <p>C - The object exists in cataloged form.</p>
User ID	The ID of the user who saved/cataloged the object.
SRC Date	Source Program Date: The time and date when the object was last saved.
GP Date	Generated Program Date: The time and date when the object was last cataloged.

Maintaining Libraries

The SYSMAIN utility is used to perform various maintenance functions on Natural objects within a library and across different libraries. The following objects can be maintained: Program, Subprogram, Subroutine, Map, Data Area (local, parameter, and global), Copycode, Help routine, Recording, Dialog, ExpertModel, Text.

To invoke SYSMAIN, you select "Services" on the Natural Main Menu. A selection window will be displayed:

```

2002-02-15                NATURAL                Library: SAG
14:38:45                  V 5.1.1 Software AG 2002      Mode  : STRUCTURED
User: SAG                                Work Area : empty
.....
·Library      Direct      Services      OS      Fin      ·
.....
                        .....
                        ·D DDM Services      ·
                        ·S SYSMAIN          ·
                        .....
    
```

From this window, you select "SYSMAIN". A selection window with the following functions will be displayed:

Function	Description
List	Display a range of objects within a specific library.
Find	Locate a single object within a specific library.
Copy	Copy object from one library to another library.
Move	Transfer object from one library to another.
Delete	Delete object from a specific library.
Rename	Give an object a new name and optionally transfer it to a new library.
Import	Copy object from an external source into a Natural library.
Terminate	Leave SYSMAIN.

On the first screen displayed with each of the above functions (except Import), you can select either Programming Objects or DDMs for processing:

- To process DDMs, you mark the Views option with an "X" and press ENTER.
- To process another type of object, you mark the Programming Objects option with an "X" and press ENTER.

All functions are explained in the following sections and assume the selection of Programming Objects.

The view or programming object environment specification must always correspond to the relevant database ID (DBID) and file number (FNR) of an FNAT or FUSER system file.

List Function

The List function is used to display the source code of a specific object. This object can be selected from a range of objects contained within a specific library in a specific environment.

If you select first the List function and then the Programming Objects option, the following window is displayed:

```

..... LIST .....
.
. OBJECT: *
. LIBRARY: SYSTEM          CODE: X (S)ource
. DBID:    99   FNR:    42      X (C)ataloged
. USER ID: SAGEX
. DATE:    -   -           :
.....
    
```

In this window, you can specify the following items:

Object	The name of the object to be listed. You can use asterisk notation (*) to find all objects or those objects whose names begin with a specific character string.
Library	The name of the library containing the object to be listed. You can use asterisk notation (*) to list for selection all libraries or those libraries whose names begin with a specific character string. For example, for a list of all libraries whose names begin with "SY", you enter "SY*".
DBID	The number (ID) of the database which contains the object(s) to be listed.
FNR	The number of the Natural system file which contains the object(s) to be listed.
User ID	The user ID under which the object was most recently cataloged and/or saved. All objects cataloged and/or saved by the specified user are selected.
Date	The search begin date and time for cataloged/saved objects. All objects cataloged/saved after this date and time are selected.
Code	The object form (S = source, C = cataloged). Because it is possible to list the source code of an object, but not its cataloged code, the field "Source" must be selected. If only "Cataloged" is selected, all cataloged objects are listed, but none can be displayed.

If asterisk notation is used to specify the library and more than one library is found, then all libraries found are displayed for selection:

```

..... LIST .....
.  Library   DBID/FNR  .
. -----  .
. SYSEXP   99/42     .
. SYSEXR   99/42     .
. SYSTEM   99/42     .
. -----  .
. *** ENTER==>list *** .
. ***   ESC==>exit  *** .
.....
    
```

Select one library from the list and press ENTER. If asterisk notation is used to specify the object and more than one object is found, then object types are displayed for selection:

```

.....OBJECT TYPE.....
.X ==> select ALL .
. Program .
. Subroutine .
. Copycode .
. Map .
. Text .
. Helproutine .
. Subprogram .
. Global Data .
. Local Data .
. Parameter Data .
. Recording .
. Dialog .
. Class .
. ExpertModel .
.....

```

If you mark "ALL" or select the desired object type from the list, a list of either all objects or of all objects of the selected type contained in the previously selected library is displayed:

```

..... 10 Object(s) in Lib: SYSEXP.....
. Object   Type       S/C   User ID .
. -----  -
. BREAK1   Program      S/C   SAGEX .
. EMPL-LDA Local       S     SAGEX .
. IMP1     Program      S     SAGEX .
. IMP2     Program      S/C   SAGEX .
. IMP3     Program      S/C   SAGEX .
. LINKDEMO Program      S     SAGEX .
. LOGONQ-G Subprogram  S     SAGEX .
. NL-GDA   Global       S     SAGEX .
. -----  -
. *** ENTER==>list *** ESC==>exit *** .
.....

```

For each object listed, the object name, object type, object form (S = source, C = cataloged), and user ID under which the object was saved/cataloged are displayed.

To display the source code of an object, select the desired object from the list.

```

..... List: BREAK1 .....
. 0010 ***** .
. 0020 * EXAMPLE: 'BREAK1': AT BREAK STATEMENT .
. 0030 * .
. 0040 * PURPOSE: DEMONSTRATE NATURAL SYSTEM FUNCTIONS WITH AT BREAK .
. 0050 * CONDITION. INCLUDE USER-SUPPLIED TEXT. .
. 0060 * .
. 0070 * HIGHLIGHTS: AT BREAK STATEMENT, NATURAL SYSTEM FUNCTIONS OLD, MIN, .
. 0080 * AVER, MAX, SUM, TOTAL, COUNT .
. 0090 ***** .
. 0100 DEFINE DATA .
. 0110 LOCAL .
. 0120 1 EMPLOY-VIEW VIEW OF EMPLOYEES .
. 0130 2 NAME .
. 0140 2 CITY .
. 0150 2 SALARY (1) .
.....

```

To leave this function, you press ESC.

Find Function

The Find function is used to determine in which library of which environment a specific object is contained. Once found, the object's source code can be displayed.

If you select the Find function first and then the Programming Objects option, the following window is displayed:

```

..... FIND .....
.
. OBJECT:  B*
. LIBRARY: SYSEXP          CODE: X (S)ource
. DBID:    99   FNR:   42      X (C)ataloged
. USER ID: SAGEX
. DATE:    -   -   :
.....
    
```

In this window, you can specify the following items:

Object	The name of the object you are looking for. You can use asterisk notation (*) to find all objects or those objects whose names begin with a specific character string. For example, for a list of all objects whose names begin with "B", you enter "B*".
Library	The name of the library in which you want to look for the object to be found. You can use asterisk notation (*) to search for all libraries or only those libraries whose names begin with a specific character string.
DBID	The number (ID) of the database which contains the object(s) you are looking for.
FNR	The number of the Natural system file which contains the object(s) you are looking for.
User ID	The user ID under which the object to be found was most recently cataloged and/or saved. All objects cataloged and/or saved by the specified user are selected.
Date	The search begin date and time for cataloged/saved objects. All objects cataloged/saved after this date and time are selected.
Code	The object form (S = source, C = cataloged). You can find objects which are available in either source or cataloged form only, or objects which are available in both source and cataloged form (default). Note: Only source objects can be displayed.

A list of objects is then displayed:

```

..... 4 Object(s) in Lib: SYSEXP.....
. Object   Type           S/C  User ID  .
. -----  - - - - -
. BREAK1   Program          S/C  SAGEX   .
. BREAK2   Program           S    SAGEX   .
. BREAK3   Program           S    SAGEX   .
. BREAK4   Program          S/C  SAGEX   .
. -----  - - - - -
. *** ENTER==>list *** ESC==>exit ***
.....
    
```

For each object found, the object name, object type, object form (S = source, C = cataloged), and user ID under which the object was saved/cataloged are displayed.

To display the source code of an object, select the desired object from the list.

```

..... List: BREAK1 .....
. 0010 ***** .
. 0020 * EXAMPLE: 'BREAK1': AT BREAK STATEMENT .
. 0030 * .
. 0040 * PURPOSE: DEMONSTRATE NATURAL SYSTEM FUNCTIONS WITH AT BREAK .
. 0050 * CONDITION. INCLUDE USER-SUPPLIED TEXT. .
. 0060 * .
. 0070 * HIGHLIGHTS: AT BREAK STATEMENT, NATURAL SYSTEM FUNCTIONS OLD, MIN, .
. 0080 * AVER, MAX, SUM, TOTAL, COUNT .
. 0090 ***** .
. 0100 DEFINE DATA .
. 0110 LOCAL .
. 0120 1 EMPLOY-VIEW VIEW OF EMPLOYEES .
. 0130 2 NAME .
. 0140 2 CITY .
. 0150 2 SALARY (1) .
.....

```

To leave this function, you press ESC.

Copy, Move and Rename Functions

The **Copy** function is used to copy objects from a library in a source environment to a library in a target environment. The objects in the original library are not deleted.

The **Move** function is used to move objects from a library in a source environment to a library in a target environment. A move is basically a copy operation in which the objects are deleted from the source library.

The **Rename** function is used to rename objects. You can either rename the object in the source environment or rename the object and transfer it to another (that is, target) environment. Since renaming implies deletion of the original object in the source environment, you are prompted with an option to retain it. If you decide to retain it, the original object is not modified.

For the copy, move, and rename functions, if the target environment already contains an object with the same name as the object to be copied, moved or renamed, the specified object is not copied, moved or renamed.

Note:

Because of the similarities among the copy, move and rename functions, this section describes only the copy function.

Select the Copy function and then the Programming Objects option. The following window is displayed:

```

..... COPY .....
.
.           - Source -
. OBJECT:  *
. LIBRARY: SYSEXP          CODE: X (S)ource
. DBID:    99   FNR:   42      X (C)ataloged
. USER ID: SAGEX          XREF: n
. DATE:    - -      :
.
.           - Target -
. OBJECT:
. LIBRARY:                REPLACE:
. DBID:          FNR:
.....
    
```

In this window, you can enter the following source specifications:

Object	The name of the object to be copied. You can use asterisk notation (*) to list all objects or those objects whose names begin with a specific character string. You can then select specific objects from the resulting list for copying.
Library	The name of the library which contains the object(s) to be copied. You can use asterisk notation (*) to list all valid libraries or those libraries whose names begin with a specific character string. You can then select a specific library from the resulting list.
DBID	The number (ID) of the database which contains the object(s) to be copied.
FNR	The number of the Natural system file which contains the object(s) to be copied.
User ID	The user ID under which the object to be copied was most recently cataloged and/or saved. All objects cataloged and/or saved by the specified user are selected.
Date	The search begin date and time for cataloged/saved objects. All objects cataloged/saved after this date and time are copied.
Code	The object form (S = source, C = cataloged). You can copy objects which are available in either source or cataloged form only, or objects which are available in both source and cataloged form (default).
XREF	Indicates whether SYSMAIN is to support cross-reference data stored on Predict system files. NO - Cross-reference data are not processed, except when using the DELETE function. If a cataloged object is deleted, SYSMAIN always deletes any existing XREF data for this object. YES - All cross-reference data are processed. FORCE - All cross-reference data are processed and the object must be documented in Predict. (Not yet implemented) DOC - All cross-reference data are processed and the object must be documented in Predict but XREF data are not copied. (Not yet implemented)

If you either specify the desired library or select it from the list and use asterisk notation (*) in the Object field, a window is displayed, which lists all available object types:

```

.....OBJECT TYPE.....
.X ==> select ALL .
. Program .
. Subroutine .
. Copycode .
. Map .
. Text .
. Helproutine .
. Subprogram .
. Global Data .
. Local Data .
. Parameter Data .
. Recording .
. Dialog .
. Class .
. ExpertModel .
.....
    
```

Mark "ALL" to process all object types or mark a specific object name to process only objects of this type. In the window below, all objects have been listed:

```

..... Select for COPY in Lib: SYSEXPG.....
. X Object      Type          S/C  User ID .
. - - - - - - - - - - - - - - - - - - - - - - .
.   ARRAYD     Program        S/C  SAGEX   .
.   ARRAYE     Program        S    SAGEX   .
. X  BREAK1    Program        S/C  SAGEX   .
. X  BREAK2    Program        S    SAGEX   .
. X  BREAK3    Program        S    SAGEX   .
.   BREAK4     Program        S/C  SAGEX   .
.   EDITWORK   Subprogram    S    SAGEX   .
.   GDA01      Global          S    SAGEX   .
.   HELP01     Map            S    SAGEX   .
.   HOLDLOG    Program        S    SAGEX   .
.   LDA01      Local          S    SAGEX   .
.   MAP001X    Map            S    SAGEX   .
.   MAP002X    Map            S    SAGEX   .
.....
    
```

For each object listed, the object name, object type, object form (S = source, C = cataloged), and user ID under which the object was saved/cataloged are displayed.

You can select objects by marking them with an "X".

Once you have selected the objects, the following screen is displayed (for Copy and Move functions only; not applicable for Rename function):

```

.....
.X Select the specified Object(s) for copy
. Copy ALL specified Object(s)
.....
    
```

If you select the first function, a window is displayed for each selected object in which you can specify the new name to be used for the target object.

If you select the second function, all selected objects are copied/moved using the same object name for both the source and target object.

Specifying the Target Environment for Copy/Move

```

..... COPY .....
.
.   3 Object(s) selected
.
.           - Source -
. OBJECT:  *
. LIBRARY: SYSEXP          CODE: X (S)ource
. DBID:   99   FNR:   42      X (C)ataloged
. USER ID: SAGEX          XREF: n
. DATE:   - -      :   TYPE: all
.
.           - Target -
. OBJECT:  *
. LIBRARY: SYSTEST        REPLACE: n
. DBID:   30   FNR:   20
.....
    
```

Enter the following target specifications; the objects are then copied/moved accordingly:

Object	Name(s) for the target object(s).
Library	The name of the library which is to contain the copied or moved object(s).
Replace	<p>Replace option for objects moved or copied. It can be used to overwrite the object in the target environment; the following values can be specified:</p> <p>Y - An object with the same name which is already present in the target library is to be replaced.</p> <p>N - An object with the same name which is already present in the target library is not to be replaced; "N" is the default.</p> <p>Note: If a programming object is replaced, it is also deleted from the Natural buffer pool.</p>
DBID / FNR	The number (ID) of the database and the file number of the Natural system file into which the copied or moved object is to be placed.

Specifying the Name and Target Environment for Rename

```

..... RENAME .....
.
.           - Old object name -
. OBJECT:  BREAK1
. LIBRARY: SYSEXP          CODE: X (S)ource
. DBID:   99   FNR:   42      X (C)ataloged
. USER ID: SAGEX          XREF: n
. DATE:   - -      :
.
.           - New object name -
. OBJECT:  PROG02
. LIBRARY: SYSEXP          REPLACE: n
. DBID:   99   FNR:   42
.....
    
```

To rename one or more objects, enter the following specifications; the object is then renamed:

Object	New object name. By using asterisk notation (*), you can rename multiple objects whose names begin with the same character(s). For example, if you have selected objects whose names begin with "ABC" and want them all to begin with "XYZ", you enter "XYZ*" as new name. The remainder of each name (after the first three characters) is retained.
Library	The name of the library to contain the renamed object(s).
Replace	Replace option for renamed object which is being renamed. It can be used to overwrite the object in the target environment. Values: Y - An object with the same name which already exists in the target library is to be replaced. N - An object with the same name which already exists in the target library is not to be replaced; "N" is the default. Note: If a programming object is replaced, it is also deleted from the Natural buffer pool.
DBID / FNR	The number (ID) of the database and the file number of the Natural system file into which the renamed object is to be placed.

In the bottom half of the window, specify the new object name(s) and target environment (if applicable).

Delete Function

The Delete function is used to delete one or more objects from a library.

If you select the Delete function first and then the Programming Objects option, the following window is displayed:

```

..... DELETE .....
.
. OBJECT:  BR*
. LIBRARY: SYSEXP          CODE: X (S)ource
. DBID:    99   FNR:    42      X (C)ataloged
. USER ID: SAGEX          XREF: n
. DATE:    -   -      : CONFIRM: n
.....

```

Enter the following specifications for the object(s) to be deleted:

Object	The name of the object to be deleted. You can use asterisk notation (*) to list all objects or those objects whose names begin with a specific character string. You can then select specific objects from the resulting list for deleting.
Library	The name of the library which contains the object(s) to be deleted. You can use asterisk notation (*) to list all valid libraries or those libraries whose names begin with a specific character string. You can then select a specific library from the resulting list.
DBID	The number (ID) of the database which contains the object(s) to be deleted.
FNR	The number of the Natural system file which contains the object(s) to be deleted.
User ID	The user ID under which the object to be deleted was most recently cataloged and/or saved. All objects cataloged and/or saved by the specified user are selected.
Date	The search begin date and time for cataloged/saved objects. All objects cataloged/saved after this date and time are deleted.
Code	The object form (S = source, C = cataloged). You can delete objects which are available in either source or cataloged form only, or objects which are available in both source and cataloged form (default).
XREF	Indicates whether SYSMAIN is to support cross-reference data stored on Predict system files. NO - Cross-reference data are not processed, except when using the DELETE function. If a cataloged object is deleted, SYSMAIN always deletes any existing XREF data for this object. YES - All cross-reference data are processed. FORCE - All cross-reference data are processed and the object must be documented in Predict. (Not yet implemented) DOC - All cross-reference data are processed and the object must be documented in Predict but XREF data are not deleted. (Not yet implemented)

A window appears and you can then delete all objects or invoke a list displaying the objects to be deleted. You can then select specific objects from the list for deletion.

You select objects by marking them with an "X".

Once you have selected the objects, you are prompted to confirm the deletion.

Import Function

The Import function is used to import objects which are currently located in another OpenVMS or UNIX directory to a Natural library.

Also, an object may be located in the correct directory, however is not recognized by Natural because it is not included in the library information file "FILEDIR.SAG". In this case, the object has to be imported.

When you invoke the Import function, the following window is displayed:

```

..... IMPORT .....
.
.           - Source -
.
. PATH:
. OBJECT:*          CODE: X (S)ource
.                   X (C)ataloged
.
.           - Target -
.
. LIBRARY:          USER ID:          REPLACE:
. DBID:            MODE:
. FNR:
.....
    
```

Enter the following Source specifications for the object(s) to be imported:

Path	The complete RMS or UNIX path name of the directory from which the Import function is to be executed. Note for OpenVMS: The path name may contain logicals. Note for UNIX: The path name may start with a UNIX environment variable; press ENTER and the environment variable is replaced by the full path name.
Object	The name(s) of the object(s) to be imported. You can use asterisk notation (*) to list all objects or those objects whose names begin with the same character(s). The objects to be imported can then be selected from the resulting list. Note: The object must have a valid extension to be recognized by SYSMAN.
Code	The object form (S = source, C = cataloged). You can import source code or cataloged code for an object or both source code and cataloged code (default).

A window is then displayed, from which you select the desired objects.

Once you have selected the objects to be imported, the following window is displayed, in which you specify the target information:

```

..... IMPORT .....
. 29 Object(s) selected
.           - Source -
.
. PATH:    $HOME
. OBJECT:  *          CODE:
. TYPE:
.
.           - Target -
.
. LIBRARY: SYSEXP    USER ID: SAGEX    REPLACE: n
. DBID:    99        MODE:    X Structured
. FNR:    42         Report
.....
    
```

Enter the following Target specifications:

Library	The Natural library into which the objects are to be imported. If the library does not already exist, a new library is created. You can use asterisk notation (*) to list all existing libraries.
User ID	The user ID under which the imported object is to be saved and/or cataloged.
Replace	<p>Replace option for an object which is being imported. It can be used to overwrite an object in the Target environment; the following values can be specified:</p> <p>Y - An object with the same name which is already present in the target library is to be replaced.</p> <p>N - An object with the same name which is already present in the target library is not to be replaced; "N" is the default.</p> <p>Note: If a programming object is replaced, it is also deleted from the Natural buffer pool.</p>
DBID / FNR	The number (ID) of the database and the file number of the Natural system file into which the imported object is to be placed.
Mode	The programming mode. For Natural source objects, the options Structured (structured mode) and Report (reporting mode) are available. If you specify Structured, the programming mode of the objects being imported is set to structured mode.

Once you have entered the above information, the objects are imported into the specified library.

Invoking SYSMAIN by a Subprogram

MAINUSER is a subprogram which allows you to perform the various SYSMAIN functions directly from any user-written object (subroutine, program or subprogram) without going through the normal steps of invoking SYSMAIN. Upon completion of processing of the SYSMAIN functions, the utility is terminated and control is returned to the object from which the request was issued. MAINUSER can be used in either online or batch mode.

MAINUSER is invoked as follows:

CALLNAT 'MAINUSER' *command error message library*

The parameters are:

<i>command</i> (A250)	The direct command string to be executed by SYSMAIN.
<i>error</i> (N4)	The return code issued by SYSMAIN at the end of processing to indicate a normal end of processing or an error.
<i>message</i> (A72)	The message corresponding to the error given online.
<i>library</i> (A8)	The name of the library containing the utility SYSMAIN; by default, this is the library SYSMAIN. (Under UNIX, OpenVMS and Windows, this parameter does not apply and is provided for compatibility reasons only.)

An example of a callable routine is program MAINCALL in library SYSMAIN.

Direct Commands

SYSMAIN functions can be executed using direct commands issued as a parameter of the MAINUSER subprogram.

Direct commands consist of keywords and parameters. The sequence of the direct command syntax is not completely fixed. The rules which apply are:

- Function, object type and object name must be the first three parameters of the command string.
- The library or path name must be specified immediately after the FROM, IN and TO keywords. (If the optional keyword LIBRARY or PATH is used, it must be entered between the FROM, IN or TO keyword and the library or path name).
- The WHERE clause must always follow the FROM, IN or TO keyword and the library name; the sequence of the keywords and values within the clause can be specified in any order.
- The keywords and values of the WITH clause can be specified in any order, but the WITH clause must always be placed at the end of the command string.

Note:

In the syntax diagrams below, FM is shown instead of FROM to make the diagrams easier to read; however, FROM can always be used as a synonym for FM and vice versa.

FIND and LIST Direct Command Syntax

The direct command syntax of the FIND and LIST functions is:

```

{FIND}
{LIST}
[
  ALL
  CATALOGED
  SAVED
  STOWED
  VIEW
]
name IN [LIBRARY] lib-name [where-clause] [with-clause]
    
```

The *where-clause* is optional. The syntax is:

```

[ WHERE ] [DBID dbid] [FNR file-nr]
    
```

The *with-clause* is optional. The syntax is:

```

[WITH] [TYPE type] [USER user-id] [XREF xref] [REPLACE] [RCOP] [NOPROMPT]
    
```

Since the WHERE clause and WITH clause syntax are identical for each function, they are only shown once with the FIND and LIST command syntax above.

Examples:

```

FIND PROG1 IN DBID 1 FNR 6
FIND STOWED MAINMENU IN SYS* WHERE DBID 1 FNR 5
FIND ALL PROG2 IN PROD* FNR 27 DBID 1
    
```

```

LIST VIEW * IN lib-name
L SAVED TEST* IN lib-name TYPE PNS FNR 6
L SA TEST* TYPE PM IN lib-name FNR 6 DBID 2
    
```

COPY and MOVE Direct Command Syntax

The direct command syntax of the COPY and MOVE functions is:

```

{COPY}
{MOVE}
[
  ALL
  CATALOGED
  SAVED
  STOWED
  VIEW
]
name FM [LIBRARY] lib-name [where-clause]
      TO [LIBRARY] lib-name [where-clause] [with-clause]
    
```

Examples:

```

COPY PROG1 FM TESTORD TO ORDERS DBID 1 FNR 6 REP
C PGM* WITH REP TYPE PNS FM TESTLIB TO PRODLIB
COPY VIEW PERS FM OLDLIB FNR 10 TO NEWLIB FNR 16 REPLACE
    
```

```
M VIEW PERSONNEL FM OLDLIB FNR 20 TO NEWLIB FNR 24
M PROG1 TO NEWLIB
MOVE STOWED * FM OLDLIB WITH XREF Y TO NEWLIB WHERE DBID 100 FNR 160
```

DELETE Direct Command Syntax

The direct command syntax of the DELETE function is:

DELETE	<table border="1"> <tr><td>ALL</td></tr> <tr><td>CATALOGED</td></tr> <tr><td>SAVED</td></tr> <tr><td>STOWED</td></tr> <tr><td>VIEW</td></tr> </table>	ALL	CATALOGED	SAVED	STOWED	VIEW	<i>name</i> [IN [<u>LIBRARY</u>] <i>lib-name</i> [<i>where-clause</i>]] [<i>with-clause</i>]
ALL							
CATALOGED							
SAVED							
STOWED							
VIEW							

Examples:

```
D SA * IN LIBTEST TYPE GLA
DEL * TYPE PM IN TESTORD
DEL VIEW FINANCE IN TESTLIB DBID 12 FNR 27
```

RENAME Direct Command Syntax

The direct command syntax of the RENAME function is:

RENAME	<table border="1"> <tr><td>ALL</td></tr> <tr><td>CATALOGED</td></tr> <tr><td>SAVED</td></tr> <tr><td>STOWED</td></tr> <tr><td>VIEW</td></tr> </table>	ALL	CATALOGED	SAVED	STOWED	VIEW	<i>name</i> AS <i>new-name</i> FM [<u>LIBRARY</u>] <i>lib-name</i> [<i>where-clause</i>] TO [<u>LIBRARY</u>] <i>lib-name</i> [<i>where-clause</i>] [<i>with-clause</i>]
ALL							
CATALOGED							
SAVED							
STOWED							
VIEW							

Examples:

```
RENAME PGM1 AS PROG1
REN PGM1 AS PROG1 FM TESTLIB DBID 1 FNR 5 TO PRODLIB DBID 2 FNR 6
```

IMPORT Direct Command Syntax

The direct command syntax of the IMPORT function is:

IMPORT	<table border="1"> <tr><td>ALL</td></tr> <tr><td>CATALOGED</td></tr> <tr><td>SAVED</td></tr> <tr><td>STOWED</td></tr> <tr><td>VIEW</td></tr> </table>	ALL	CATALOGED	SAVED	STOWED	VIEW	<i>name</i> { FM IN } [<u>PATH</u>] <i>path-name</i> TO [<u>LIBRARY</u>] <i>lib-name</i> [<i>where-clause</i>] [<i>with-clause</i>]
ALL							
CATALOGED							
SAVED							
STOWED							
VIEW							

Examples for OpenVMS:

```
IMPORT PGM1 FM HOME TO USERLIB
```

```
I PGM2 IN DISK:[USER.Natural.SRC.example.test] TO LIB TESTUSER
```

Examples for UNIX:

```
IMPORT PGM1 FM $HOME TO USERLIB
```

```
I PGM2 IN /usr/natural/src/example/test TO LIB TESTUSER
```

Additional Keywords for Direct Commands

In addition to the keywords shown with the parameters above, the following keywords can also be used with direct commands to specify selection criteria:

Keywords	Explanation
ALL	All saved and/or cataloged programming objects are selected for processing.
CAT	All cataloged programming objects are selected for processing. (Any corresponding saved programming object is not processed.)
HELP	Activates online help.
IN/FM	Refers to a source environment.
<u>N</u> OPROMPT	Suppresses all prompts.
<u>R</u> COP	Used with direct commands to specify that a copy of the object being renamed is to be made.
<u>S</u> AVED	All saved programming objects are selected for processing. (Any corresponding cataloged object is not processed.)
STOWED	All programming objects which are both saved and cataloged are selected for processing.
TO	Refers to a target environment.
WITH	Optional keyword to indicate the start of a <i>with-clause</i> .
WHERE	Optional keyword to indicate the start of a <i>where-clause</i> .
.	End of command. If this character is detected anywhere within a command string, all subsequent data are ignored.