

PATHFINDER BUILD COMMANDS DOCUMENTED

The Build/refresh process is an important step to gaining the benefits of many PATHFINDER options. Cross reference (X-ref) consists of the Object X-ref and the Field X-ref. The Object X-ref documents the relationships among objects in application software. The Field X-ref documents the relationships among files, fields and programs. The Object and Field X-ref files are separate sets of files, allowing them to be managed independently of each other. Each can be built, refreshed or removed without affecting the other.

A normal course for getting the X-ref in place is: 1) initial build, 2) refresh. The initial build may take a few runs to finalize. A refresh can then be setup to run on a regular basis to keep the X-ref up-to-date for any object changes. The build and refresh jobs are submitted using the same menu options:

“1. Build/refresh object X-ref” on the Object X-ref Menu.

“1. Build/refresh field X-ref” on the Field X-ref Menu.

Based on your responses, the menu options submit the BLDOBJ (Build/refresh Object X-ref) and BLDFLD (Build/refresh Field X-ref) commands with the appropriate parameters. For greater flexibility, especially for a refresh procedure, the commands can be placed in a CL program, SBMJOB (Submit Job) command or job scheduling system. This will allow the commands to be executed regularly, without having to go through the menu options. It also ensures that the same parameter values are used every time. The following is a typical build command:

```
HAWKEYE/BLDOBJ PGM(*DOCLIBL/*ALL) SAVE(*CLEAR)
                DOCL(*DEFAULTS) SRCL(*DEFAULTS) BLD(*USER)
                CALL(*USER) OMIT(*USER) XREFLIB(HAWKEYE)
                PRTFILE(QSYS/QSYSPRT)
```

The following is a typical refresh command:

```
HAWKEYE/BLDOBJ PGM(*DOCLIBL/*CHANGES) SAVE(*UPDATE)
                DOCL(*DEFAULTS) SRCL(*DEFAULTS) BLD(*USER)
                CALL(*USER) OMIT(*USER) XREFLIB(HAWKEYE)
                PRTFILE(QSYS/QSYSPRT)
```

The underlined values in the refresh command represent the only difference between the two commands. The parameters for DOCL (DOCLIBL), SRCL (Source List), BLD (Build Values), CALL (CALLING List) and OMIT (Omit List) correspond with the Setup Menu options. These parameters can be *USER (your values), USRPRF name (values for your user profile), *STANDARD (standard values) or alternate name (alternate values). The DOCL and SRCL

parameters can also be *DEFAULTS which will reference the user defaults for DOCLIBL and Source List. The values used for a refresh should be the same as those used for the original build process, ensuring that the same information and preferences are used to update the documentation.

The build/refresh execution time is typically around two hours per 1000 RPG or COBOL programs. The refresh is a much faster job because it only deals with the programs that have changed since the last refresh. Most sites are able to run the job nightly to keep the X-ref up-to-date.

Parameter Table

<u>Parameter</u>	<u>Description</u>	<u>Type</u>	<u>Length</u>
PGM	Libraries to document	Character	20
SAVE	Save option	Character	7
DOCL	Documentation library list	Character	10
SRCL	Source list	Character	10
BLD	Build values	Character	10
CALL	Calling list	Character	10
OMIT	Omit list	Character	10
XREFLIB	X-ref library	Character	10
PRTFILE	Printer file	Character	20

The following describes the individual command parameters and the valid values. The first valid value listed for each is the default.

PGM

Libraries to document – Character (20)

The first ten bytes represent the library, the second ten bytes represent the objects.

The “library” portion of the parameter directs PATHFINDER to a list of libraries to document. Using *DOCLIBL (documentation library list) is usually most efficient and most precise. Your list of libraries in the *DOCLIBL can be specific to your application thereby eliminating unnecessary libraries from the documentation. The value *DOCLIBL will reference the documentation library list in the DOCL parameter of this command.

Valid values are:

- *ALLUSR
- *ALL
- *LIBL
- *USRLIBL
- *CURLIB
- *DOCLIBL
- valid library name

The “objects” portion should be *ALL for a normal library method build. This will cause all objects in the libraries to document to be exploded. For an application or “top-down” build, the setting should be a valid program name, *CALLING (calling list) or *USRPRF (user profile). In the top-down build, PATHFINDER will follow the jobstreams to the end, starting with the programs provided. The option for *CALLING will reference the calling list provided on the CALL parameter of this command. The option for *USRPRF will reference the initial program to call and the initial library list from the user profiles on the system. For a normal refresh, regardless of how the initial build was created, the value should be *CHANGES (with changed programs). This will direct PATHFINDER to document only the objects that have changed since the last refresh was completed.

Valid values are:

- *ALL
- *CALLING
- *CHANGES
- *USRPRF
- program object name

SAVE ()

Save option – Character (7)

This value tells PATHFINDER what to do with the documentation generated. The initial build or a “rebuild” process should be *CLEAR to have the existing X-ref removed before replacing it with the new. A subsequent refresh should use the value of *UPDATE. Using the PGM (*CHANGES) with SAVE (*UPDATE) will allow PATHFINDER to locate and document only the objects that have changed and update the existing documentation. The values for *ADD and *REMOVE can be used to add or remove the documentation for objects being documented. For instance, to add or remove a library’s objects from the X-ref, run a build for that library specifying *ADD or *REMOVE as the save option. The option for *NONE will cause the documentation to be printed, not saved to disk.

Valid values are:

- *UPDATE
- *ADD
- *CLEAR
- *NONE
- *REMOVE

The next five parameters direct PATHFINDER to the specific setup values that the build/refresh process is to use. Refer to the Setup Menu for maintaining these values. It is important to note the default values for these parameters. The default for DOCL and SRCL is *DEFAULTS which will reference the job’s user’s default values for DOCLIBL and Source List. And the default for BLD, CALL and OMIT parameters is *USER which will reference the setup associated with the job’s user profile. Therefore, when automating the refresh procedure, these values should be

*STANDARD or a hardcoded name. This will avoid the potential of a different user profile executing the job and using the wrong setup values.

DOCL()

Documentation library list – Character (10)

This parameter refers the build/refresh to a documentation library list to use. It provides a list of libraries to qualify unqualified object reference.

Valid values are:

- *DEFAULTS
- *USER
- *STANDARD
- user profile name
- alternate documentation library list name

SRCL()

Source list – Character (10)

The source list directs PATHFINDER to program source files. This is required when program object descriptions source information is inaccurate. This is the case when the source has been moved since the object was compiled.

Valid values are:

- *USER
- *STANDARD
- user profile name
- alternate source list name

BLD ()

Build values – Character (10)

Build values are a set of preferences for the build process. This allows particular build functions to be turned on or off as needed.

Valid values are:

- *USER
- *STANDARD
- user profile name
- alternate build values name

CALL ()

Calling list – Character (10)

A calling list is a list of programs to document using the application or “top-down” method. The programs represent the entry points for the jobstreams to be documented. It is referenced when using *CALLING in the PGM parameter of this command.

Valid values are:

- *USER
- *STANDARD
- user profile name
- alternate calling list name

OMIT ()

Omit list – Character (10)

The omit list is a set of programs that will be completely ignored in the generated X-ref. This is helpful when a program is being called from virtually everywhere in the application. If omitted, there is a savings in documentation time and space.

Valid Values are:

- *USER
- *STANDARD
- user profile name
- alternate omit list name

XREFLIB ()

Library to store X-ref –Character (10)

For a build, enter an existing library in which you wish the X-ref to be stored. For a refresh or a rebuild, enter the library that contains the existing X-ref files. The library HAWKEYE can be used to hold X-ref data, as can any other library on the system. For backup and recovery purposes, you may find it helpful to store the X-ref in a library by itself.

Valid values are:

- *DEFAULTS
- valid library name

PRTFILE ()

Printer file to use – Character (20)

The first ten bytes represent the library, the second ten bytes represent the printer file object.

This is the printer file to use for the reports generated by the build/refresh process. It may be any valid printer file. This allows you to direct output to a particular OUTQ or to employ any of the special features of printer files such as page orientation.

Valid values are:

- *DEFAULTS
- valid printer file library/name

For further information, please contact our Technical Services department. We can be reached by email at info.hawkinfo.com or call us Monday-Thursday, 7 a.m. to 5 p.m. (MST) and Friday 7 a.m. to 3 p.m., VOICE (970) 498-9000 or FAX (970) 498-9096.