



@server®

# **iSeries. mySeries.**

## **CL Documentation Enhancements**

*Guy Vig*

*Senior Software Engineer*

*([gwvig@us.ibm.com](mailto:gwvig@us.ibm.com))*

© Copyright IBM Corporation, 2004. All Rights Reserved.  
This publication may refer to products that are not currently  
available in your country. IBM makes no commitment to make  
available any products referred to herein.

**iSeries. mySeries.**



# iSeries Navigator Prompter



# Access for Web Prompter

The screenshot shows a Microsoft Internet Explorer browser window titled "Run Command - CRTLIB [LP016AB.RCHLAND.IBM.COM]". The address bar contains the URL: `http://lp016ab:2030/webaccess/iWACCommandPrompter?cmd=CRTLIB&successURL=http%3a%2f%2flp016ab%3a2030%2fwebaccess%2`. The browser's navigation bar includes "Back", "Forward", "Home", "Search", "Favorites", "Media", and "Go" buttons. The search bar contains "Google". The browser's toolbar shows "Search the Web with Lycos", "IBM Business Transformation", "IBM Internal Help", "IBM Standard Software Installer", "PageRank", "25 blocked", "AutoFill", and "Options".

The main content area displays the "iSeries Access for Web" interface. The page title is "iSeries Access for Web" and the URL is "LP016AB.RCHLAND.IBM.COM". The page content is titled "Create Library (CRTLIB)".

The form contains the following fields and options:

- Library:
- Library type:
- ASP number:
- ASP device:
- Text 'description':
- Authority:
- Create authority:
- Create object auditing:
- Output:

At the bottom of the form are four buttons: "Run Command", "Submit Batch Job", "Save", and "Cancel".

The left sidebar contains a navigation menu with the following items:

- My Folder
- My Home Page
- Print
- Messages
- Jobs
- 5250
- Database
- Files
- Command
  - My commands
  - Run command
  - Search
- Download
- Customize
- Other
- Related Links:
  - iSeries Access for Web
  - iSeries Access
  - iSeries Navigator
  - WebSphere Host
  - Publisher
  - iSeries Information



# Remote System Explorer Prompter

**Create Physical File (CRTPF)**

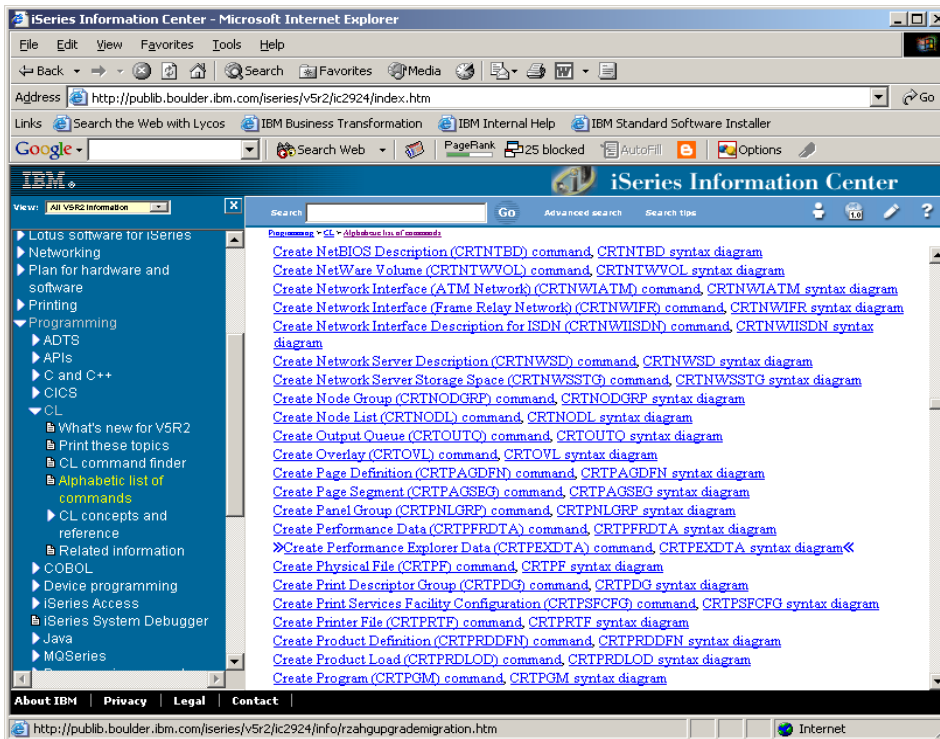
File:		Name
Library:	*CURLIB	Name
Source file:	QDSSSRC	Name
Library:	*LIBL	Name
Source member:	*FILE	Name
Record length, if no DDS:		Number
Generation severity level:	20	0-30
Flagging severity level:	0	0-30
File type:	*DATA	Name
Member, if desired:	*FILE	Name
User specified DBCS data:	*NO	Name
Text 'description':	*SRCMBRTXT	Character value...
Source listing options:	Advanced Parameters	
		Add
		Remove
		Move up
		Move down
System:	*LCL	
Expiration date for member:	*NONE	Date, MMDDYYYY

## Underlying technologies

- New API (QCDRCMDD) returns XML which provides command definition for a \*CMD object
- New API (QUHRHLPT) returns XHTML for online help text for one or more online help modules stored in panel group (\*PNLGRP) objects
- Java code invokes APIs, parses command XML to provide prompt and command help panels



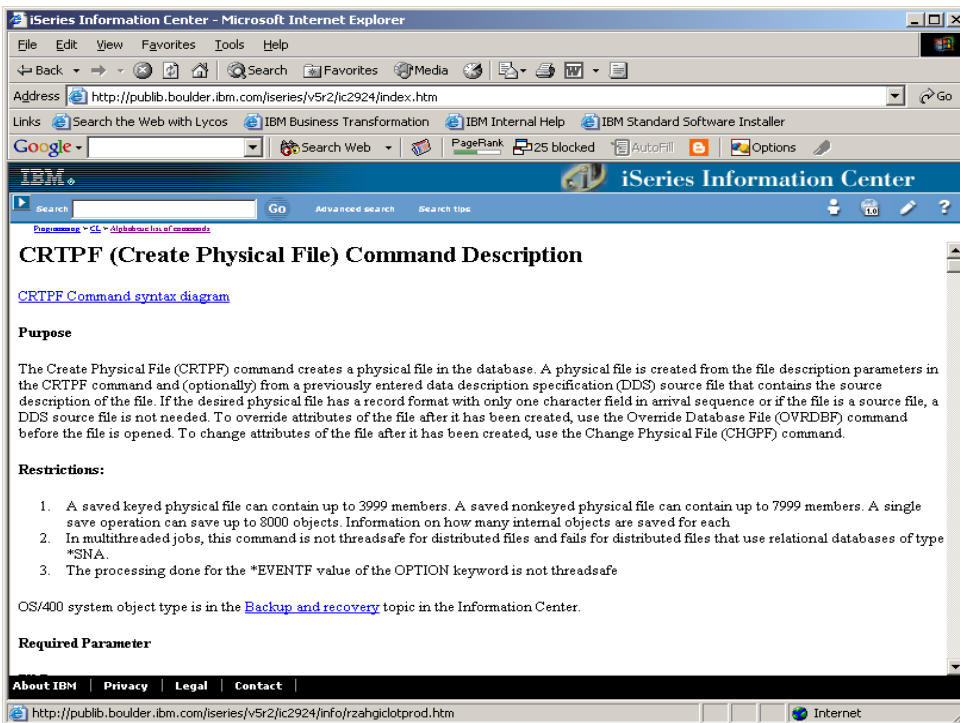
# V5R2 Information Center



- Information for each command takes two files
- Commands for some products, but many products' CL commands are missing (e.g. SQL, RPG, COBOL, C/C++)

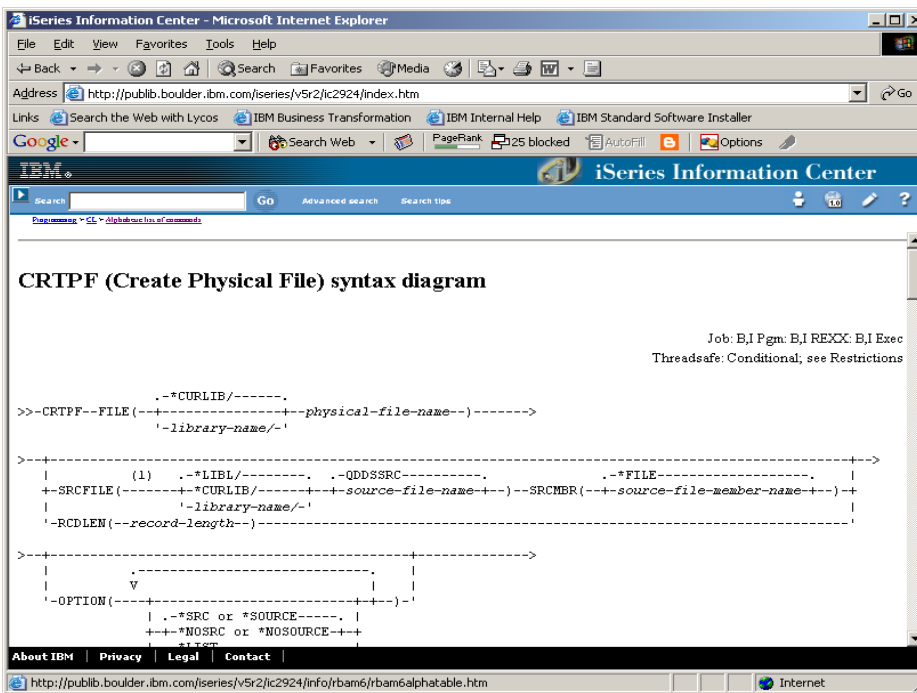


# Old Command Documentation



- Link to syntax summary (in a separate file)
- Very few other text links
- Text shows its “CL Ref” heritage
- Parameters may not be in prompt order

# Old CL Syntax Diagram



- Information not accessible
- Difficult for anyone to read
- Generated diagram may be too wide to print correctly
- No linkage to parameter textual help

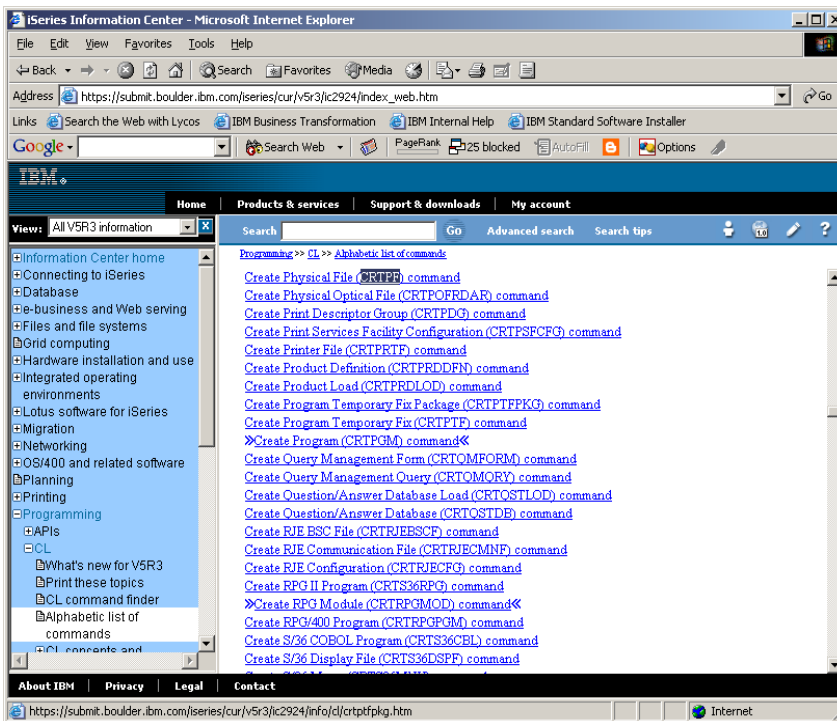


# Documentation Problems

- Three separate documentation formats requiring three different tag languages (UIM,SGML,HTML)
- Conflicts between online help & CL Ref style
- Developer nightmare to keep all three in synch
- Syntax diagrams failed accessibility requirements
- Not all products have commands in the Info Center
- Navigation within CL HTML files is poor



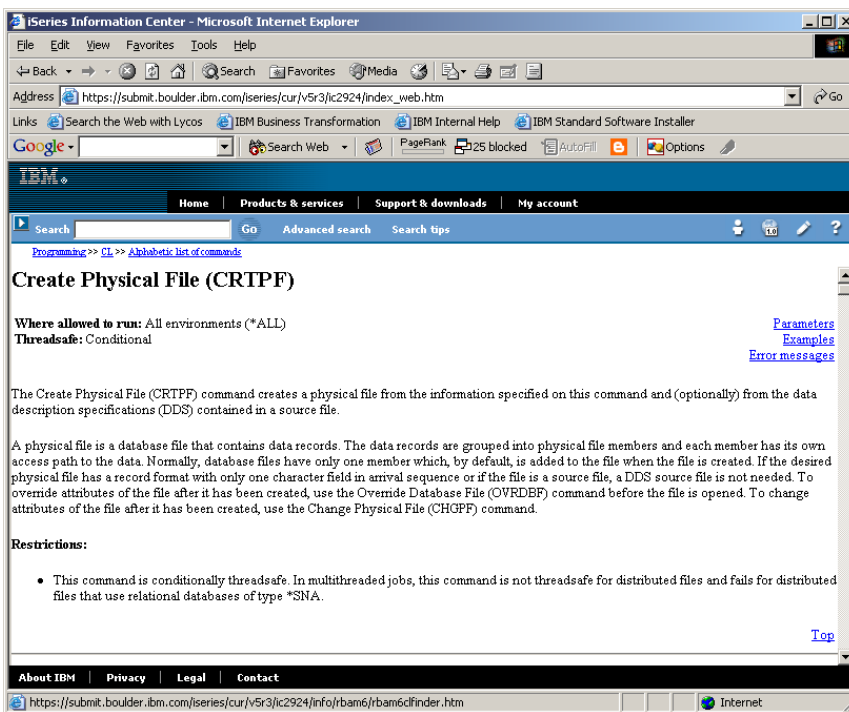
# V5R3 Information Center



- Single file for each CL command
- Commands for all iSeries licensed program products (i.e. products with product IDs that have 5722 prefix)
- New function in CL command finder to find CL commands by product



# New Command Documentation



- Looks familiar, but contains links to major sections
- All links are contained within the single file
- Title heading, Where allowed and Threadsafe from \*CMD XML
- All text from online help XHTML



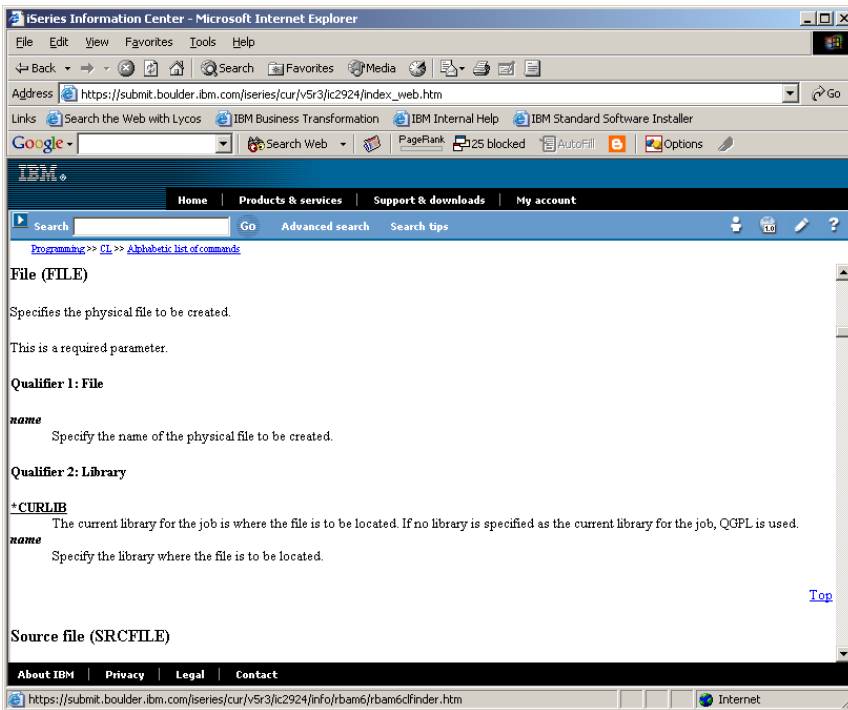
# New CL Syntax Table

The screenshot shows a web browser window displaying the IBM iSeries Information Center. The page title is "iSeries Information Center - Microsoft Internet Explorer". The address bar shows the URL: "https://submit.boulder.ibm.com/iseries/cur/v5r3/ic2924/index\_web.htm". The page content includes a search bar and a table of parameters. The table has four columns: Keyword, Description, Choices, and Notes. The parameters listed are FILE, SRCFILE, SRCMBR, RCDLEN, GENLYL, FLAG, and FILETYPE.

Keyword	Description	Choices	Notes
<a href="#">FILE</a>	File	<i>Qualified object name</i>	Required, Positional 1
	Qualifier 1: File	<i>Name</i>	
	Qualifier 2: Library	<i>Name, *CURLIB</i>	
<a href="#">SRCFILE</a>	Source file	<i>Qualified object name</i>	Optional, Positional 2
	Qualifier 1: Source file	<i>Name, QDDSSRC</i>	
	Qualifier 2: Library	<i>Name, *LIBL, *CURLIB</i>	
<a href="#">SRCMBR</a>	Source member	<i>Name, *FILE</i>	Optional, Positional 3
<a href="#">RCDLEN</a>	Record length, if no DDS	<i>Integer</i>	Optional, Positional 4
<a href="#">GENLYL</a>	Generation severity level	0-30, <b>20</b>	Optional
<a href="#">FLAG</a>	Flagging severity level	0-30, <b>0</b>	Optional
<a href="#">FILETYPE</a>	File type	<b>*DATA</b> , *SRC	Optional

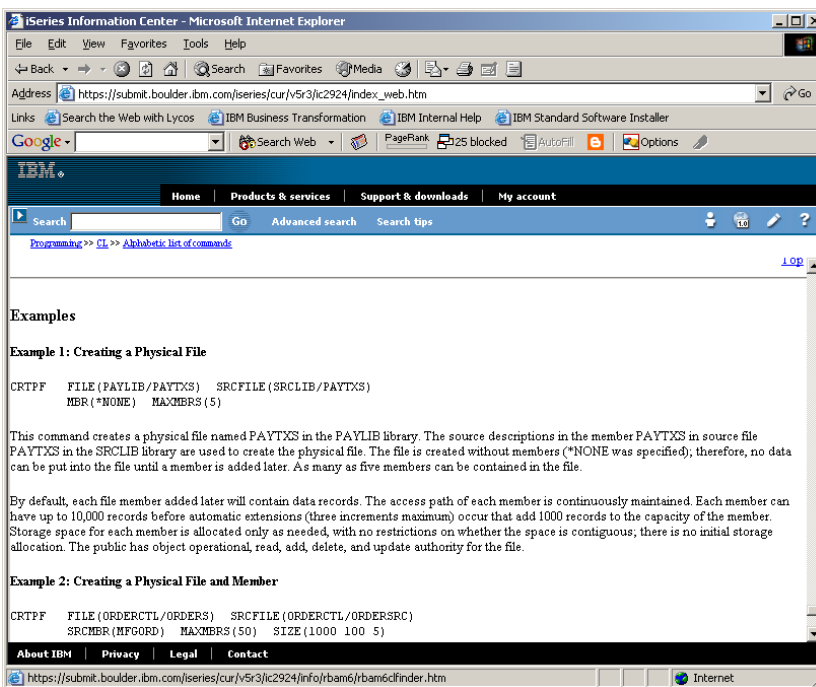
- Information much more accessible *for everyone*
- Table is generated from \*CMD XML
- Hypertext links to each parameter's textual help section
- Parameters listed in prompt order
- Table prints right

# New Parameter Documentation



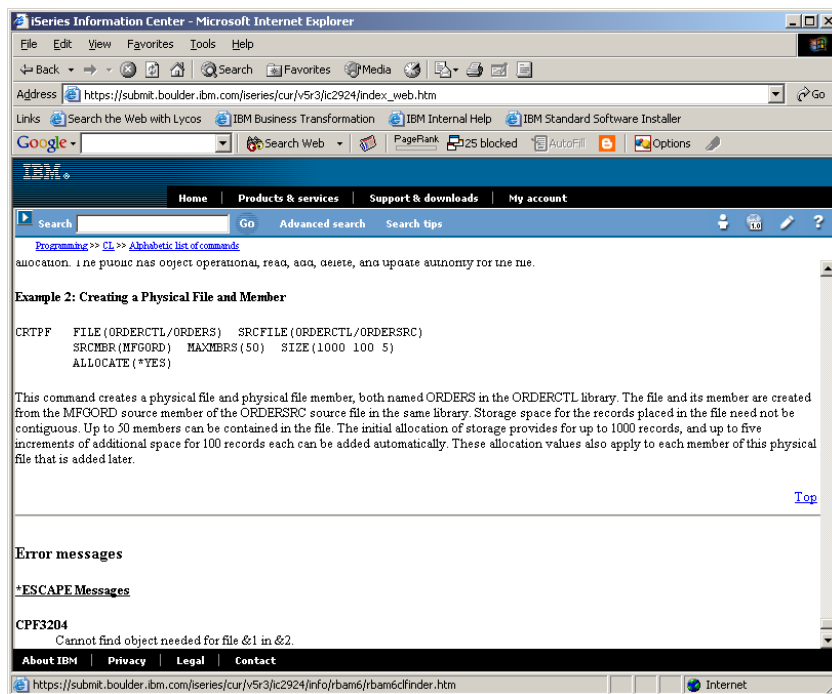
- Headings and all parameter help text comes from XHTML retrieved from online help panel groups
- Link provided after each parameter help section to jump back to beginning of file
- Can use browser 'Back' button or 'Alt + ←' to navigate

# New/Old Command Examples



- Examples were copied/converted from the “CL Ref” HTML files to UIM
- Examples will show up in online help too (after last parameter)
- Can jump directly to **Examples** from link at top of file
- All IBM commands *should* have

# New/Old Error Messages



- Monitorable error message info added to online help in V4
- Error messages will show up in online help after Examples
- Can jump directly to **Error messages** from link at top of file
- All IBM commands *should* have

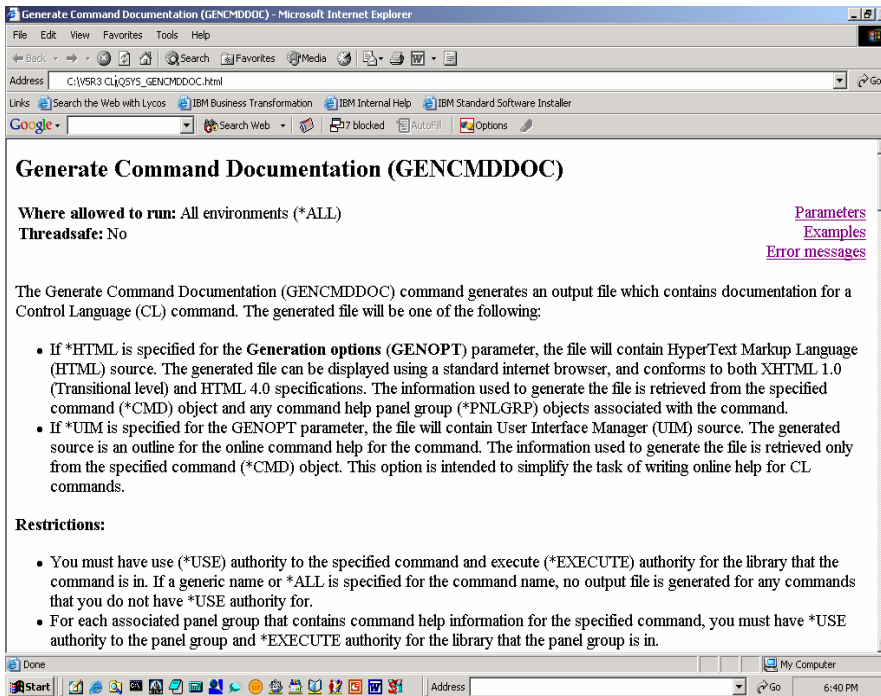
# Summary

- Providing GUI support for CL prompting & help
  - iSeries Navigator
  - iSeries Access for Web
  - Remote Systems Explorer
- Providing consistent and complete command help online and in the Info Center, using tools shipped in the operating system which  
**YOU can use too**





# Using the New Tools



• Java and XSLT in jt400.jar that ships in operating system

• Use the Java method if you want, but CL command provided too

• Can generate the same HTML found in the Info Center

• Can generate UIM for your own \*CMD

## Display Object List (DSPOBJL)

Where allowed to run: All environments (\*ALL)

Threadsafe: No

[Parameters](#)  
[Examples](#)  
[Error messages](#)

### Parameters

Keyword	Description	Choices	Notes
OBJ	Objects	<i>Qualified object name</i>	Required, Positional 1
	Qualifier 1: Objects	<i>Generic name, name, *ALL</i>	
	Qualifier 2: Library	<i>Name, *LIBL, *USRLIBL, *CURLIB, *ALL, *ALLUSR</i>	
OUTPUT	Output	<i>*, *PRINT, *OUTFILE</i>	Optional
OUTFILE	File to receive output	<i>Qualified object name</i>	Optional
	Qualifier 1: File to receive output	<i>Name</i>	
	Qualifier 2: Library	<i>Name, *LIBL, *CURLIB</i>	
OUTMBR	Output member options	<i>Element list</i>	Optional
	Element 1: Member to receive output	<i>Name, *FIRST</i>	
	Element 2: Replace or add records	<i>*REPLACE, *ADD</i>	

[Top](#)

### Examples

None

[Top](#)

### Error messages

Unknown

[Top](#)

# Why Do Online Command Help?

- For you (the command writer/developer)
  - Better looking
  - Keeps you from being the “help desk”
- For your company
  - Able to “hand off” responsibility to another developer
- For your “customer”
  - Not stuck if command developer leaves
  - More seamless extension of CL value proposition

## Learning <a little> UIM

- Developed for AS/400 as SAA follow-on to DDS
  - Syntax chosen was based on IBM BookManager
- Will only be discussing online help part of UIM
  - Appendix A of “Application Display Programming” book
- CRTPNLGRP compiles UIM source into \*PNLGRP
  - Fairly rigid tag language conformance required
- UIM tag and attribute names **not** case-sensitive
  - I’ll use lowercase in my examples (like :p instead of :P)

# UIM Help Tag Cheat Sheets

- **:pnlgrp** is very first tag ... **:epnlgrp** is the very last
- **:help** starts a help “module” ... **:ehelp** ends a module
  - The **name** attribute’s value identifies the help module
- **:xhn** for extended help headings, **:xh3** particularly (no **:exhn** tag)

```
:pnlgrp.  
:      <- other help modules  
:help name='DSPOBJL/OUTPUT'.  
Output (OUTPUT) - Help  
:xh3.Output (OUTPUT)  
:      <- body of this help module  
:ehelp.  
:      <- more help modules  
:epnlgrp.
```

- **:p** for each new paragraph (no **:ep** tag)
  - **:nt** and **:ent** for adding a note
  - **:hpn** for highlighted phrases, **:hp2** and **:hp1** especially
  - Types of lists: unordered lists (**:ul/:eul**), ordered lists (**:ol/:eol**), parameter lists (**:parml/:eparml**), definition lists (**:dl/:edl**).
    - **:li** for each list item in unordered and ordered lists (no **:eli** tag)
    - **:pt** for each parameter term, **:pd** for each parameter description (no **:ept** or **:epd**)
    - **:dt** for each definition term, **:dd** for each description (no **:edt** or **:edd**)

**:p.Specifies where the output from the command is sent.**

**:parml.**

**:pt.:pk def.\*:epk.**

**:pd.The output is displayed (if requested by an interactive job) or printed (if requested by a batch job).**

**:pt.:pk.\*PRINT:epk.**

**:pd.The output is printed with the job's spooled output.**

**:pt.:pk.\*OUTFILE:epk.**

**:pd.The output is directed to the database file specified for the :hp2.File to receive output (OUTFILE):ehp2. parameter.**

**:eparml.**

- **:xmp/:exmp** for unformatted text, most often examples

```
:xmp.
```

```
SAVCHGOBJ   OBJ(ORD*)   LIB(DSTPRODLB)  DEV(TAP01)  
              OBJTYPE(*FILE)  REFDATE(*SAVLIB)
```

```
:exmp.
```

```
:p.This command saves all files with names that start  
with the characters ORD in the library named DSTPRODLB  
that have changed since the last time the whole library  
was saved.
```

- **:imhelp** to imbed a help module in another help module

```
:help name='DSPOBJL/OUTPUT'.
```

```
Output (OUTPUT) - Help
```

```
:xh3.Output (OUTPUT)
```

```
:imhelp name='COMMON/OUTPUT/3'.
```

```
:ehelp.
```

## \*CMD to \*PNLGRP connections

- From the \*CMD side
  - Help identifier (HLPID) parameter on CRTCMD
  - Help panel group (HLPPNLGRP) parameter on CRTCMD
- From the \*PNLGRP side
  - Command-level help module (**name**='hlpid')
  - Parameter-level help modules (**name**='hlpid/kwd')
  - Examples module (**name**='hlpid/COMMAND/EXAMPLES')
  - Error messages module (**name**='hlpid/ERROR/MESSAGES')



# Command Online Help Flow

- Press F1 with cursor in Prompter parameter input field
  - Prompter determines keyword name of parameter and builds the help module name for the parameter
  - Help module is located in the \*PNLGRP associated with the command; help text is formatted and displayed in help “window”
- Press F2 from help window or F1 from command line
  - Prompter determines keyword names of all parameters and builds the help module names for the command-level help, each parameter-level help, examples help, and error messages help
  - Help modules are located in the \*PNLGRP and the associated help text is formatted and displayed in specified help module order

# Writing UIM help source <or not>

- The “old” way
  - Use editor of choice (like SEU), no UIM syntax checker
  - Start with **:pnlgrp**
  - Try to remember when matching end tags are required
  - Have a nice day!
- The “new” way (available starting in V5R3)
  - Use GENCMDDOC command with GENOPT(\*UIM)
  - Use editor of choice to replace <...> markers with descriptive text

# Adding Online Help (Walkthru)

- Write the command definition first
  - Can take the CRTCMD defaults of HLPID(\*NONE) and HLPPNLGRP(\*NONE) initially when creating command
  - Make sure you're happy with command name and parameter keyword names
  - Prompt text can be literal or can specify message IDs
  - Parameter order and conditional prompting can be changed after online help generated
  - Decide which parameters will always be required



## Sample Command Definition Source (part 1)

```
DSPOBJL: CMD  PROMPT('Display Object List')

      PARM  KWD(OBJ) TYPE(Q1) MIN(1) MAX(1) +
            FILE(*NO) PROMPT('Objects')

      PARM  KWD(OUTPUT) TYPE(*CHAR) LEN(1) RSTD(*YES) DFT(*) +
            SPCVAL(* (*PRINT P) (*OUTFILE F)) +
            EXPR(*YES) PROMPT('Output')

      PARM  KWD(OUTFILE) TYPE(Q2) FILE(*OUT) +
            PMTCTL(OUTFILE) PROMPT('File to receive output')

      PARM  KWD(OUTMBR) TYPE(E1) +
            PMTCTL(OUTFILE) PROMPT('Output member options')

E1:    ELEM  TYPE(*NAME) LEN(10) DFT(*FIRST) +
            SPCVAL(*FIRST) EXPR(*YES) PROMPT('Member to receive output')
      ELEM  TYPE(*CHAR) LEN(1) RSTD(*YES) +
            DFT(*REPLACE) SPCVAL((*REPLACE R) (*ADD A)) +
            EXPR(*YES) PROMPT('Replace or add records')
```



## Sample Command Definition Source (part 2)

```
Q1:    QUAL TYPE(*GENERIC) LEN(10) SPCVAL(*ALL) +
        MIN(1) EXPR(*YES)
        QUAL TYPE(*NAME) LEN(10) DFT(*LIBL) +
        SPCVAL(*LIBL *USRLIBL *CURLIB *ALL *ALLUSR) +
        EXPR(*YES) PROMPT('Library')

Q2:    QUAL TYPE(*NAME) LEN(10) MIN(1) EXPR(*YES)
        QUAL TYPE(*NAME) LEN(10) DFT(*LIBL) +
        SPCVAL((*LIBL) (*CURLIB *CURLIB)) +
        EXPR(*YES) PROMPT('Library')

DEP    CTL(&OUTPUT *EQ F) PARM((&OUTFILE *NE ' ') +
        NBRTRUE(*EQ 1) MSGID(CPD9861)
DEP    CTL(&OUTMBR *NE *FIRST) PARM((&OUTFILE *EQ ' ') +
        NBRTRUE(*EQ 0) MSGID(CPD9867)
DEP    CTL(&OUTPUT *NE F) PARM((&OUTFILE *EQ ' ') +
        NBRTRUE(*EQ 1) MSGID(CPD9862)

OUTFILE: PMTCTL CTL(OUTPUT) COND((*EQ F))
```

## 1. Create the \*CMD from command definition source

```
CRTCMD  CMD(VIG/DSPOBJL)  PGM(VIG/DSPOBJLCP)  
        SRCFILE(VIG/QCMDSRC)  SRCMBR(DSPOBJL)  
        MAXPOS(1)  THREADSAFE(*COND)  TEXT(*CMD)
```

## 2. Generate UIM source from the \*CMD object

```
GENCMDDOC  CMD(VIG/DSPOBJL)  
           TODIR(`/QSYS.LIB/VIG.LIB/QPNLSRC.FILE')  
           TOSTMF(*CMD)  GENOPT(*UIM)
```

## 3. Edit the generated UIM source

```
STRSEU  SRCFILE(VIG/QPNLSRC)  SRCMBR(DSPOBJL)  
        TYPE(TXT)  OPTION(2)
```

# Editing the Generated UIM

- Command-level help module
  - :help and :ehelp tags generated for you
  - :p tag generated for 1<sup>st</sup> sentence and sentence started
  - Replace <...> marker with one or more sentences that describe the function provided by the command
  - Edit or delete the sample Restrictions list with command-level restrictions like authorities required to run the command, prerequisite commands, threadsafe restrictions (if conditionally threadsafe), etc.



## Before

## After

```
0001.00 :pntgrp.
0002.00 :*****
0003.00 :* Help for command DSPOBJL
0004.00 :*****
0005.00 :help name='DSPOBJL'.
0006.00 Display Object List - Help
0007.00 :p.The Display Object List (DSPOBJL) command <...>
0008.00 :* Describe the function provided by the command.
0009.00 :p.:hp2.Restrictions::ehp2.
0010.00 :ul.:li.
0012.00 You must have <...>
0013.00 :* List all special authorities required to run the command.
0014.00 :li.
0015.00 You must have <...>
0016.00 :* List object or data authorities required to run the command.
0017.00 :li.
0018.00 This command is conditionally threadsafe, <...>
0019.00 :* If conditionally threadsafe, list threadsafe conditions or restrictions.
0020.00 :li.
0021.00 <...>
0022.00 :* Describe other command-level restrictions.
0024.00 :eul.
0025.00 :ehelp.
```

```
0001.00 :pntgrp.
0002.00 :*****
0003.00 :* Help for command DSPOBJL
0004.00 :*****
0005.00 :help name='DSPOBJL'.
0006.00 Display Object List - Help
0007.00 :p.The Display Object List (DSPOBJL) command provides a list of objects in one
0008.00 or more libraries that match a specific or generic name string.
0009.00 :p.:hp2.Restrictions::ehp2.
0010.00 :ul.
0017.00 :li.
0018.00 This command is conditionally threadsafe, it is not threadsafe when the
0019.00 command output is sent to a workstation and the command is run from an
0020.00 interactive job.
0024.00 :eul.
0025.00 :ehelp.
0026.00 :*****
```



- Parameter-level help module
  - **:help** and **:ehelp** tags generated for you, including **:xh3**
  - **:p** tag generated for 1<sup>st</sup> sentence and sentence started
  - Replace **<...>** marker with one or more sentences that describe the function provided by the parameter
  - **:parml** and **:eparml** tags generated for choices list, with **:pt** and **:pd** tags generated for each possible parameter choice defined in the \*CMD, including **:pk/:epk** or **:pv/:epv** around value, with **:pk def** for default parameter value (underlines the value when formatted)
  - Headings generated for parameter elements and parameter qualifiers (up to three levels deep)



## Before

## After

```
0027.00 .* Help for parameter OBJ
0028.00 *****
0029.00 :help name='DSPOBJL/OBJ'.
0030.00 Objects (OBJ) - Help
0031.00 :wh3.Objects (OBJ)
0032.00 :p.Specifies <...>
0033.00 .* Describe the function provided by the parameter.
0034.00 :p.This is a required parameter.
0035.00 :p.:hp2.Qualifier 1: Objects:ehp2.
0036.00 :parml.
0037.00 :pt.:pk.*ALL:epk.
0038.00 :pd.
0039.00 <...>
0040.00 .* Describe the function provided by the pre-defined parameter value.
0041.00 :pt.:pw.generic-name:epv.
0042.00 :pd.
0043.00 Specify the generic name of <...>
0044.00 .* Describe the function provided by the user-defined parameter value.
0045.00 :pt.:pw.name:epv.
0046.00 :pd.
0047.00 Specify the name of <...>
0048.00 .* Describe the function provided by the user-defined parameter value.
0049.00 :eparml.
0050.00 :p.:hp2.Qualifier 2: Library:ehp2.
```

```
0027.00 .* Help for parameter OBJ
0028.00 *****
0029.00 :help name='DSPOBJL/OBJ'.
0030.00 Objects (OBJ) - Help
0031.00 :wh3.Objects (OBJ)
0032.00 :p.Specifies the objects to be listed.
0033.00 .* Describe the function provided by the parameter.
0034.00 :p.This is a required parameter.
0035.00 :p.:hp2.Qualifier 1: Objects:ehp2.
0036.00 :parml.
0037.00 :pt.:pk.*ALL:epk.
0038.00 :pd.
0039.00 List all objects, regardless of the object name.
0041.00 :pt.:pw.generic-name:epv.
0042.00 :pd.
0043.00 Specify the generic name of the objects to be listed. A generic
0044.00 name is specified as a character string that contains one or
0044.01 more characters followed by an asterisk (*).
0045.00 :pt.:pw.name:epv.
0046.00 :pd.
0047.00 Specify the name of the object to be listed.
0049.00 :eparml.
0050.00 :p.:hp2.Qualifier 2: Library:ehp2.
```

- Examples help module

- **:help** and **:ehelp** tags generated for you, including **:xh3**
- **:p** tag generated for each example title, with **:hp2/:ehp2** tags for emphasis; can delete title if only one example
- **:xmp** and **:exmp** tags generated around example CL; replace KWD1, KWD2 with appropriate keywords/values
- One example OK if \*CMD has zero or one parameter
- Good examples are very effective way for you to demonstrate how to leverage the features of your command
- Replace <...> marker with one or more sentences that describe the behavior of the example command string



## Before

## After

```
0168.00 :help name='DSPOBJL/COMMAND/EXAMPLES'.
0169.00 Examples for DSPOBJL - Help
0170.00 :xh3.Examples for DSPOBJL
0171.00 :p.:hp2.Example 1: Simple Command Example:ehp2.
0172.00 :xmp.
0173.00 DSPOBJL  KWD1(PARMLVAL1)
0174.00 :xmp.
0175.00 :p.This command <...>
0176.00 .* Describe a simple invocation of the command.
0177.00 .*
0178.00 :p.:hp2.Example 2: More Complex Command Example:ehp2.
0179.00 :xmp.
0180.00 DSPOBJL  KWD1(PARMLVAL1)  KWD2(PARMLVAL2)
0181.00           KWD3(PARMLVAL3)
0182.00 :xmp.
0183.00 :p.This command <...>
0184.00 .* Describe a more complex invocation of the command.
0185.00 :ehelp.
```

```
0168.00 :help name='DSPOBJL/COMMAND/EXAMPLES'.
0169.00 Examples for DSPOBJL - Help
0170.00 :xh3.Examples for DSPOBJL
0171.00 :p.:hp2.Example 1: Display Generic List of Objects in Single Library:ehp2.
0172.00 :xmp.
0173.00 DSPOBJL  OBJ(VIG/Q*)
0174.00 :xmp.
0175.00 :p.This command will list all of the objects with names that begin with the
0176.00 letter 'Q' in library VIG. The list will be shown on the requesting
0177.00 workstation if the command is run from an interactive job, or generated to
0177.01 a spooled file if the command is run from a batch job.
0177.02 .*
0178.00 :p.:hp2.Example 2: List Objects to an Output File:ehp2.
0179.00 :xmp.
0180.00 DSPOBJL  OBJ(VIG/*ALL)  OUTPUT(*OUTFILE)
0181.00           OUTFILE(QTEMP/MYOBJECTS)  OUTMBR(ONOC19)
0182.00 :xmp.
0183.00 :p.This command will list all objects in library VIG to member ONOC19 of file
0184.00 MYOBJECTS in library QTEMP. The output file and member will be created if they
0184.01 do not exist when the command is run.
0185.00 :ehelp.
```

- Error messages help module
  - **:help** and **:ehelp** tags generated for you, including **:xh3**
  - **:p** tag generated for sample heading for list of monitorable \*ESCAPE messages sent by the command
    - Suggest separate heading and message list for monitorable \*STATUS or \*NOTIFY messages; not applicable for most \*CMD
    - Replace with **:p.&msg(CPX0007,QCPFMSG)**. if no messages
  - **:dl**, **:dt**, **:dd**, and **:edl** tags generated for list of error message IDs and the associated message 1<sup>st</sup>-level text
    - Demonstrates how to use **&msg** UIM built-in function to pull in message text when panel group is created
    - Suggest **not** listing CPF0001 (syntax error) or CPF9999 (\*FC)
  - Edit message list to match those signaled by your \*CMD



## Before

```
0188.00 * Error messages for DSPOBJL
0189.00 *
0190.00 *****
0191.00 'help name='DSPOBJL/ERROR/MESSAGES'.
0192.00 &msg(CPF0005,QCPFMSG). DSPOBJL - Help
0193.00 :wh3.&msg(CPF0005,QCPFMSG). DSPOBJL
0194.00 :p.:hp3.*ESCAPE &msg(CPF0006,QCPFMSG).:ehp3.
0195.00 *****
0196.00 * List the *ESCAPE, *STATUS, and *NOTIFY messages signalled from the command.
0197.00 * The following are generic messages defined in message file QCPFMSG.
0198.00 * Modify this list to match the list of error messages for the command.
0199.00 *****
0200.00 :DL COMPACT.
0201.00 :DT.CPF9801
0202.00 :DD.&MSG(CPF9801,QCPFMSG,*LIBL,nosub).
0203.00 :DT.CPF9802
0204.00 :DD.&MSG(CPF9802,QCPFMSG,*LIBL,nosub).
0205.00 :DT.CPF9803
0206.00 :DD.&MSG(CPF9803,QCPFMSG,*LIBL,nosub).
0207.00 :DT.CPF9807
0208.00 :DD.&MSG(CPF9807,QCPFMSG,*LIBL,nosub).
0209.00 :DT.CPF9808
```

## After

```
0190.00 *****
0191.00 'help name='DSPOBJL/ERROR/MESSAGES'.
0192.00 &msg(CPF0005,QCPFMSG). DSPOBJL - Help
0193.00 :wh3.&msg(CPF0005,QCPFMSG). DSPOBJL
0194.00 :p.:hp3.*ESCAPE &msg(CPF0006,QCPFMSG).:ehp3.
0195.00 *****
0196.00 * List the *ESCAPE, *STATUS, and *NOTIFY messages signalled from the command.
0199.00 *****
0200.00 :DL COMPACT.
0201.00 :DT.CPF2110
0202.00 :DD.&MSG(CPF2110,QCPFMSG,*LIBL,nosub).
0203.00 :DT.CPF2113
0204.00 :DD.&MSG(CPF2113,QCPFMSG,*LIBL,nosub).
0205.00 :DT.CPF2124
0206.00 :DD.&MSG(CPF2124,QCPFMSG,*LIBL,nosub).
0207.00 :DT.CPF2176
0208.00 :DD.&MSG(CPF2176,QCPFMSG,*LIBL,nosub).
0209.00 :DT.CPF2182
0210.00 :DD.&MSG(CPF2182,QCPFMSG,*LIBL,nosub).
0221.00 :DT.CPF9899
0222.00 :DD.&MSG(CPF9899,QCPFMSG,*LIBL,nosub).
0223.00 :EDL.
0224.00 :ehelp.
0225.00 :epnlgrp.
```

#### 4. Create the \*PNLGRP from edited UIM source

```
CRTPNLGRP    PNLGRP(VIG/DSPOBJLHLP)  
             SRCFILE(VIG/QPNLSRC)  SRCMBR(DSPOBJL)
```

#### 5. Change the \*CMD object to link to \*PNLGRP

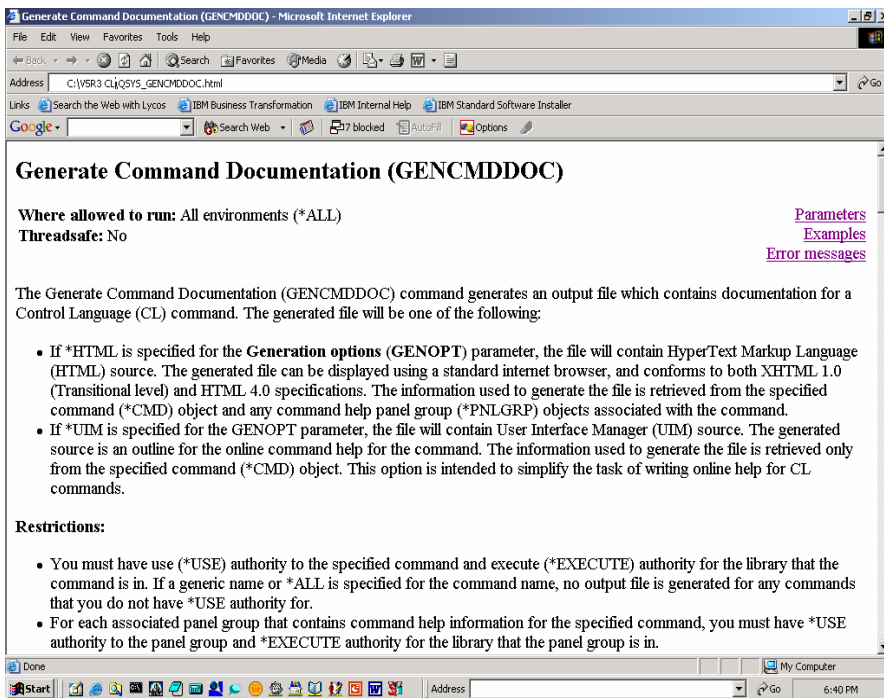
```
CHGCMD      CMD(VIG/DSPOBJL)  
            HLPID(*CMD)  HLPPNLGRP(VIG/DSPOBJLHLP)
```

#### 6. Test help from Prompter and command line

#### 7. Update UIM source and re-create panel group (if needed), until you're satisfied with the online help



# HTML Command Documentation



- Use GENCMDDOC command with GENOPT(\*HTML)
- Command uses information from the \*CMD and the online help from \*PNLGRP to build the HTML
- HTML can be put on web page or imbedded in other documentation



## Doing CL Help “Like IBM”

- Command definitions pull prompt text from message file to allow for translation
  - Use **&msg** in UIM source to ensure consistent text
- Group online help for component/product \*CMDs
- Use **:import** tag to share common help between panel groups and **:imhelp** tag to imbed common help within a help module
- Generated HTML documentation ships in the iSeries Information Center

## Summary

- In the past, adding online help for your own CL commands was a real “brute force” exercise
- With V5R3, combination of GENCMDDOC tool and new Chapter 10 in “CL Programming” book make this task MUCH easier
- Side benefits of being able to generate HTML documentation to put on a web page or import into a document (e.g. using Word or WordPro)



@server®

Thank  
YOU



# Trademarks and Disclaimers

© IBM Corporation 1994-2004. All rights reserved.

References in this document to IBM products or services do not imply that IBM intends to make them available in every country.

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

AS/400

iSeries

eServer

IBM



IBM (logo)

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both. Other company, product or service names may be trademarks or service marks of others.

Information is provided "AS IS" without warranty of any kind.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.

Information concerning non-IBM products was obtained from a supplier of these products, published announcement material, or other publicly available sources and does not constitute an endorsement of such products by IBM. Sources for non-IBM list prices and performance numbers are taken from publicly available information, including vendor announcements and vendor worldwide homepages. IBM has not tested these products and cannot confirm the accuracy of performance, capability, or any other claims related to non-IBM products. Questions on the capability of non-IBM products should be addressed to the supplier of those products.

All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. Contact your local IBM office or IBM authorized reseller for the full text of the specific Statement of Direction.

Some information addresses anticipated future capabilities. Such information is not intended as a definitive statement of a commitment to specific levels of performance, function or delivery schedules with respect to any future products. Such commitments are only made in IBM product announcements. The information is presented here to communicate IBM's current investment and development activities as a good faith effort to help with our customers' future planning.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.