z/TPF API5 Driver

User's Guide

© Copyright IBM Corp. 2005, 2011

This page intentionally left blank.

© Copyright IBM Corp. 2005, 2011

ZTEST API5

Use this command to execute the API5 driver. The API5 driver executes test cases for Application Program Interfaces (APIs), for example, macros, C functions, and grouped functional interfaces such as heap or resource control. The online help function displays the APIs supported by the API5 driver.

Requirements and restrictions

None.

Format

```
                    .- -HELP-----------------------------------.
>>--ZTEST-- --API5--+------------------------------------------+---------><
                    |                                          |
                    +-+- -API-name-+--+---------------------+--+
                    | '- -API-ALL--'  '-| API5 Parameters |-'  |
                    |                                          |
                    '- -?--------------------------------------'

API5 Parameters:

  .- -NOERR-.  .- -Verbose---.  .- -CNT-1--------.  .- -BSO---.
|-+---------+--+-------------+--+----------------+--+---------+---------->
  '- -ERR---'  '- -NOVerbose-'  +- -CNT-count----|  '- -NOBSO-'
                                '- -TIme-seconds-'

  .- -CSO---.                      .- -PName-CVZZ-.
>-+---------+--+----------------+--+--------------+--+--------------+---->
  '- -NOCSO-'  '- -LSET-loadset-'  '- -PName-prog-'  '- -SNAP-bytes-'

>-+-----------+----------------------------------------------------------|
  '- -CASE-xx-'
```

HELP | ?

HELP, ?, or entering an incorrect format will cause a display describing the format of the ZTEST API5 command. A list of supported APIs is included in the help display.

API-*name*

specifies the application program interface to be executed, where *name* must be a 3 to 6 character supported API name. See the online help for the list of supported APIs.

API-ALL

specifies that all APIs supported by this driver will be executed.

ERR

executes the error test cases.

NOERR

execute the normal test cases. NOERR is the default.

© Copyright IBM Corp. 2005, 2011

Verbose
displays all the informational messages. Verbose is the default.

NOVerbose
displays only error messages.

CNT-*count*
specifies the number of times to execute the request, where *count* can be 1 to 5 digits. The default value for *count* is 1. CNT and TIME are mutually exclusive.

TIme-*seconds*
specifies the amount of time in seconds to execute the request, where *seconds* can be 1 to 4 digits. CNT and TIME are mutually exclusive.

BSO
exercises only BSO macros.

NOBSO
does not exercise BSO macros.

CSO
exercises only CSO function calls.

NOCSO
does not exercise CSO function calls.

LSET-*loadset*
specifies the loadset name to be used, where *loadset* is 1 to 8 alphanumeric characters. This parameter is only valid for those APIs supporting a specified loadset. The default value for *loadset* is all blanks.

PName-*prog*
specifies the program name to be used, where *prog* must be a 4 character program name. This parameter is only valid for those APIs supporting a specified program name. The default value for *prog* is CVZZ.

SNAP-*bytes*
specifies the number of bytes to display, where *bytes* is a numerical value from 0 to 999. This parameter is only valid for those API exercise cases that support it.

CASE-*xx*
specifies the test case number to execute, where *xx* is a valid test case number. This parameter is only valid for those API exercise cases that support it.

© Copyright IBM Corp. 2005, 2011

Source code information

The API5 driver consists of the following program segments:

Header Files

Header File	Description
c_api5.h	Defines function prototypes, structures, and constants for this driver.
qmac23.h	Used by API-movcr segments. It contains prototypes and various #defines.

Macros

Macro	Description
drv51.mac	Used to print a message if MSGON indicator is set.
prerr.mac	Used to print an error message if MSGON indicator is set.
prmsg.mac	Used to print a message if MSGON indicator is set.

BSOs

Module	Makefile	Segment	Description
QMAJ	qmaj.mak	qmaj.asm	Called by qmac50.asm to tests the link utilities tools.
		qman.asm	Called by qmaj.asm to test the link utilities tools.
QMAK	N/A	qmak.asm	Tests the 31-bit GETPC macro.
QMAL	qmal.mak	qmal.asm	Called by qmac50.asm to tests the link utilities tools.
		qman.asm	Called by qmal.asm to test the link utilities tools.
QMAM	qmam.mak	qmam.asm	Called by qmac50.asm to tests the link utilities tools.
QMAX	qmax.mak	qmax.asm	Test cases for PNAMC or extended register save capability.
		qm11.asm	Used as a step to build the function nesting area in QM22 to test the PNAMC macro.
		qm22.asm	Tests the PNAMC macro.
QMAY	N/A	qmay.asm	Handles created entries from QMAC functions, and is used to create a queue of IOBs so the ecblc_macro function can purge them.
QMAZ	N/A	qmaz.asm	Contains BSO linkage tests.

CSOs

Module	Makefile	Segment	Description
QMAC	qmac.mak	qmac00.cpp	Main entry point for the API driver. It will perform parsing and invoke the appropriate routine.
		qmac10.cpp	Tests the attac, caloc, crosc, fiwhc, flipc, freec, gsysc, maloc, raloc, rehka, rsysc, and unhka functions.
		qmac20.cpp	Tests the cratc, crexc, corhc, lodic, movec, tmslc, uatbc, and wgtac functions.
		qmac21.cpp	Tests trace groups.

© Copyright IBM Corp. 2005, 2011

Module	Makefile	Segment	Description
		qmac23.cpp	Tests the tpf_movec_return() and tpf_movec_EVM_return() functions.
		qmac40.cpp	Test the getpc() and progc() functions.
		qmac60.cpp	Tests the cifrc, numbc, serrc, snapc, and sonic functions.
		qmac70.cpp	Test the tpf_stpoc() function with the TPF_STPOC_RESTORE parameter.
		qmac80.cpp	drv_zdcrt_func() issues cretcs to QMAP in order to create entries in the CRET table. Used to test the new ZDCRT command
		qmac85.cpp	pausc_func() to test the new CTL-574 system error
		qmac90.cpp	Tests the swisc_enter and tpf_genlc functions.
		qmac22.asm	Test cases for extended register save capability.
		qmac30.asm	Tests the SONIC macro.
		qmac41.asm	Tests the PROGC macro.
		qmac42.asm	Tests the GETPC macro.
		qmac50.asm	Tests various aspects of the CALOC, CFISVC, CORHC, CORUC, CREXC, DETAC, ECBLC, ERRNOC, FREEC, FSYSC, GSVAC, GSWBC, GSYSC, MALOC, MAXBC, NUMBC, PHYBC, RALOC, RECVC, RITID, RSYSC, SWISC, TAMCC, VALBC macros.
		qmac55.asm	Tests the CLINKC macro.
QMAD	qmad.mak	qmad.cpp	This segment is used for error dump test cases of the attac (case 0), crosc (case 1), freec (case 2), crexc (case 3), cremc (case 4), and unhka (case 5) functions.
QMAE	qmae.mak	qmae.cpp	Returns the number 12 and then verifies that this is the number returned after having done a crosc() to this program.
QMAF	qmaf.mak	qmaf.cpp	Tests crosc()'s ability to pass a parameter list. It modifies the data, and then returns double the initial value. The calling program verifies the returned value.
QMAG	qmag.mak	qmag.cpp	Returns two times the number passed in the first register if the second register has been passed to it correctly and contains the word "test". Otherwise, it returns the number 1.
QMAH	qmah.mak	qmah.cpp	Test cases 1-3: Called by corhc_func in

© Copyright IBM Corp. 2005, 2011

Module	Makefile	Segment	Description
			qmac20.cpp to test the exclusive control over a resource.
		qmac22.asm	Test cases for extended register save capability.
QMAI	qmai.mak	qmai.cpp	Tests the tpf_movec_EVM_return() function.
QMAO	qmao.mak	qmao.cpp	QMAO() is swisc_create-d by the pausc_func() in qmac85.cpp. It will loop uninterrupted for the specified amount of time. CTL-10 processing was modified for this segment in the API5 control file .
QMAP	qmap.mak	qmap.cpp	This program is cretc-ed to from drv_zdcrt_func() in qmac80.cpp. It displays the time to activate the ECB after the cretc was called

© Copyright IBM Corp. 2005, 2011

Additional information

Most API names specify a single macro or C function. Some APIs are grouped together by a function. See Appendices for functionally grouped APIs. The following explains how to invoke the APIs which cannot be called that intuitively:

- Use API-UNHKA ERR to invoke the testcase for unhka. The API5 driver only has an error testcase for this API.
- Use API-ENTRC to invoke the testcase for tpf_swisc_enter.
- Use API-GENLC to invoke the testcase for tpf_genlc.
- Use API-CLINKC to test BSO internal linkage macros (CLINKC, SLINKC, RLINKC, and ELINKC).

Examples

The following example invokes ENTRC three times:

```
ZTEST API5 API-ENTRC CNT-3
```

The following example invokes the BSO error cases for SONIC:

```
ZTEST API5 API-SONIC ERR BSO
```

The following example invokes all of the BSO and CSO non-error cases for all supported APIs and only displays error messages:

```
ZTEST API5 API-All NOV
```

Messages

Below is a sampling of some API5 driver messages:

ZTEST API5 API-ATTAC

```
QMAC0001I 15.32.02 attac:  Assembler driver not provided+
CSMP0097I 15.32.02 CPU-B SS-BSS  SSU-HPN  IS-01
attac: entering attac_func+
CSMP0097I 15.32.02 CPU-B SS-BSS  SSU-HPN  IS-01
attac: 1-Attach D1 to previously Detached D1..+
CSMP0097I 15.32.02 CPU-B SS-BSS  SSU-HPN  IS-01
attac: 1-Successfully Completed.+
CSMP0097I 15.32.02 CPU-B SS-BSS  SSU-HPN  IS-01
attac: 2-Attach and Detach D2 multiple times.+
CSMP0097I 15.32.02 CPU-B SS-BSS  SSU-HPN  IS-01
attac: 2-Successfully Completed.+
CSMP0097I 15.32.02 CPU-B SS-BSS  SSU-HPN  IS-01
attac: completed attac_func.+
CSMP0097I 15.32.02 CPU-B SS-BSS  SSU-HPN  IS-01
API5 driver exited +
```

ZTEST API5 API-CORUC

```
QMAC0001I 15.34.20 coruc:  1-holding resource w. 64 bit size option+
CSMP0097I 15.34.20 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC0001I 15.34.20 coruc:  1-Resource held+
CSMP0097I 15.34.20 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC0001I 15.34.20 coruc:  1-Releasing resource w. 32bit size option+
CSMP0097I 15.34.20 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC0001I 15.34.20 coruc:  1-test executed successfully+
```

© Copyright IBM Corp. 2005, 2011

```
CSMP0097I 15.34.20 CPU-B SS-BSS  SSU-HPN  IS-01
API5 driver exited +
```

ZTEST API5 API-DETAC

```
QMAC0001I 15.32.27 DETACed 260 blocks+
CSMP0097I 15.32.27 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC0001I 15.32.27 ATTACed 260 Blocks - Test 1 Complete+
CSMP0097I 15.32.27 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC0001I 15.32.27 DETACed 260 blocks from DECB+
CSMP0097I 15.32.27 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC0001I 15.32.27 ATTACed 260 Blocks - Test 2 Complete+
CSMP0097I 15.32.27 CPU-B SS-BSS  SSU-HPN  IS-01
API5 driver exited +
```

ZTEST API5 API-ENTRC

```
entrc: Entering entrc_func test.+
CSMP0097I 15.34.32 CPU-B SS-BSS  SSU-HPN  IS-01
entrc: Processing entrc_func normal cases.+
CSMP0097I 15.34.32 CPU-B SS-BSS  SSU-HPN  IS-01
entrc: 1-Create ECB for swisc_enter test.+
CSMP0097I 15.34.32 CPU-B SS-BSS  SSU-HPN  IS-01
entrc: Completed entrc_func test.+
CSMP0097I 15.34.32 CPU-B SS-BSS  SSU-HPN  IS-01
API5 driver exited +
```

ZTEST API5 API-FREEC

```
CSMP0097I 15.32.38 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC0001I 15.32.38 freec:  1-free address of zero in 64 bit mode+
CSMP0097I 15.32.38 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC0001I 15.32.38 freec:  1-test executed successfully+
CSMP0097I 15.32.38 CPU-B SS-BSS  SSU-HPN  IS-01
freec: entering freec_func.+
CSMP0097I 15.32.38 CPU-B SS-BSS  SSU-HPN  IS-01
freec: no freec_func regular cases to process.+
CSMP0097I 15.32.38 CPU-B SS-BSS  SSU-HPN  IS-01
freec: completed freec_func.+
CSMP0097I 15.32.38 CPU-B SS-BSS  SSU-HPN  IS-01
API5 driver exited +
```

ZTEST API5 API-GSYSC

```
QMAC0001I 15.34.39 gsysc:  1-Executing 64 bit heap option+
CSMP0097I 15.34.39 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC0001I 15.34.39 gsysc:  1-buf above bar 0000000880500000+
CSMP0097I 15.34.39 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC0001I 15.34.39 gsysc:  1-test executed successfully+
CSMP0097I 15.34.39 CPU-B SS-BSS  SSU-HPN  IS-01
gsysc: entering gsysc_func.+
CSMP0097I 15.34.39 CPU-B SS-BSS  SSU-HPN  IS-01
gsysc: 1-Attempting to get 31-bit storage from the system heap.+
CSMP0097I 15.34.39 CPU-B SS-BSS  SSU-HPN  IS-01
gsysc: 1-Storage obtained from the system heap.+
CSMP0097I 15.34.39 CPU-B SS-BSS  SSU-HPN  IS-01
gsysc: 1-Address below 2-Gig Bar.  Address returned: 0xf6af000.+
CSMP0097I 15.34.39 CPU-B SS-BSS  SSU-HPN  IS-01
gsysc: 1-Storage released from the system heap.+
CSMP0097I 15.34.39 CPU-B SS-BSS  SSU-HPN  IS-01
gsysc: completed gsysc_func.+
CSMP0097I 15.34.39 CPU-B SS-BSS  SSU-HPN  IS-01
API5 driver exited +
```

ZTEST API5 API-MALOC

```
QMAC0001I 15.32.47 maloc:  1-Executing 64 bit heap option+
CSMP0097I 15.32.47 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC0001I 15.32.47 maloc:  1-buf above bar 0000000480000F30+
CSMP0097I 15.32.47 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC0001I 15.32.47 maloc:  1-test executed successfully+
CSMP0097I 15.32.47 CPU-B SS-BSS  SSU-HPN  IS-01
maloc: entering maloc_func.+
CSMP0097I 15.32.47 CPU-B SS-BSS  SSU-HPN  IS-01
maloc: 1-Attempting to reserve heap storage using tpf_malloc64()+
CSMP0097I 15.32.47 CPU-B SS-BSS  SSU-HPN  IS-01
maloc: 1-Address below 2-Gig Bar.  Address returned: 0xcf0fb38.+
CSMP0097I 15.32.47 CPU-B SS-BSS  SSU-HPN  IS-01
maloc: completed maloc_func.+
CSMP0097I 15.32.47 CPU-B SS-BSS  SSU-HPN  IS-01
API5 driver exited +
```

ZTEST API5 API-RECVC

```
QMAC0001I 15.34.53 recvc:  1-if a dump follows this msg, test failed+
CSMP0097I 15.34.53 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC0001I 15.34.53 recvc:  1-test executed successfully+
CSMP0097I 15.34.53 CPU-B SS-BSS  SSU-HPN  IS-01
API5 driver exited +
```

ZTEST API5 API-SNAPC

```
snapc: entering snapc_func test+
CSMP0097I 15.32.57 CPU-B SS-BSS  SSU-HPN  IS-01
snapc: no valid case for snapc_func test+
CSMP0097I 15.32.57 CPU-B SS-BSS  SSU-HPN  IS-01
snapc: exiting snapc_func test+
CSMP0097I 15.32.57 CPU-B SS-BSS  SSU-HPN  IS-01
API5 driver exited +
```

ZTEST API5 API-UNHKA

```
unhka: entering unhka_func.+
CSMP0097I 15.33.09 CPU-B SS-BSS  SSU-HPN  IS-01
unhka: no unhka_func regular cases to process.+
CSMP0097I 15.33.09 CPU-B SS-BSS  SSU-HPN  IS-01
unhka: completed unhka_func.+
CSMP0097I 15.33.09 CPU-B SS-BSS  SSU-HPN  IS-01
API5 driver exited +
```

ZTEST API5 API-VALBC

```
QMAC0001I 15.41.50 valbc:  1-Executing valbc for inuse 1 meg block+
CSMP0097I 15.41.50 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC0001I 15.41.50 valbc:  1-buf above bar 0000000880500000+
CSMP0097I 15.41.50 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC0001I 15.41.50 valbc:  2-Executing valbc for released meg block+
CSMP0097I 15.41.50 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC0001I 15.41.50 valbc:  2-test executed successfully+
CSMP0097I 15.41.50 CPU-B SS-BSS  SSU-HPN  IS-01
API5 driver exited +
```

References

For more information about reading syntax diagrams, also referred to as railroad diagrams, see *Accessibility information* in the TPF Product Information Center.

© Copyright IBM Corp. 2005, 2011

Appendix A: GETPC API Group

The GETPC API group exercises the various parameters supported by the GETPC macro and the getpc() C function call. Each of these parameters can be exercised separately by specifying the appropriate case. Some cases require the user to specify a program name or loadset name. This API group also supports the BSO/CSO option which allows the user to limit the exercises to GETPC or getpc() test. The SNAP option is also supported, allowing the user to specify the number of bytes to display of the retrieved data.

The following test cases are supported by this API group.

BSO non-error cases

Case Number	Description
1	Default parms, NAME=(Rx)
2	NAME=(Rx),DBI
3	NAME=(Rx),PBI
4	NAME=CVZZ
5	FETCHONLY=NO, NAME=(Rx)
6	FETCHONLY=NO,NAME=(Rx),DBI
7	NAME=(Rx),FETCHONLY=NO,PBI
8	FETCHONLY=YES, NAME=(Rx)
9	FETCHONLY=YES,NAME=(Rx),DBI
10	NAME=(Rx),FETCHONLY=YES,PBI
11	NAME=CVZZ,FETCHONLY=YES
12	NAME=(Rx),LOADSET=(Ry)

BSO error cases (specify ERR option)

Case Number	Description
13	ERROR CASE-NAME=ZZZZ
14	ERROR CASE-NAME=(Rx),LOADSET=(Ry)
15	ERROR CASE-NAME=(Rx),LOADSET=(Ry) DUMP

31-bit error case

Case Number	Description
16	ERROR CASE-NAME=(Rx)

CSO non-error cases

Case Number	Description
1	GETPC_DBI.
2	GETPC_PBI
3	GETPC_DUMP
4	LOADSET NULL
5	GETPC_NOLOCK
6	GETPC_SPECIAL

CSO error cases

None.

© Copyright IBM Corp. 2005, 2011

Examples

The following example invokes case 1 for GETPC and getpc() using the default program name CVZZ and displays 100 bytes:

```
ZTEST API5 API-GETPC CASE-1 SNAP-100
```

The following example invokes GETPC error case 13 and displays 200 bytes:

```
ZTEST API5 API-GETPC CASE-13 ERR BSO SNAP-200
```

Messages

Below is a sampling of some API5 GETPC driver messages:

ZTEST API5 API-GETPC PN-CVZZ

```
QMAC0001I 13.57.13 STARTING THE GETPC MACRO TEST+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC:  PROCESSING VALID CASES+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: CASE#1-Default parms, NAME=(Rx)+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: EXPECTED RESULT- CASE#1 SUCCESSFUL+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: CASE#2-NAME=(Rx),DBI+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: EXPECTED RESULT- CASE#2 SUCCESSFUL+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: CASE#3-NAME=(Rx),PBI+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: EXPECTED RESULT- CASE#3 SUCCESSFUL+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: CASE#4-NAME=CVZZ+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: EXPECTED RESULT- CASE#4 SUCCESSFUL+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: CASE#5-FETCHONLY=NO, NAME=(Rx)+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: EXPECTED RESULT- CASE#5 SUCCESSFUL+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: CASE#6-FETCHONLY=NO,NAME=(Rx),DBI+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: EXPECTED RESULT- CASE#6 SUCCESSFUL+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: CASE#7-NAME=(Rx),FETCHONLY=NO,PBI+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: EXPECTED RESULT- CASE#7 SUCCESSFUL+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: CASE#8-FETCHONLY=YES, NAME=(Rx)+
7CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: EXPECTED RESULT- CASE#8 SUCCESSFUL+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: CASE#9-FETCHONLY=YES,NAME=(Rx),DBI+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: EXPECTED RESULT- CASE#9 SUCCESSFUL+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: CASE#10-NAME=(Rx),FETCHONLY=YES,PBI+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: EXPECTED RESULT- CASE#10 SUCCESSFUL+
```

© Copyright IBM Corp. 2005, 2011

```
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: CASE#11-NAME=CVZZ,FETCHONLY=YES+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: EXPECTED RESULT- CASE#11 SUCCESSFUL+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: CASE#12-NAME=(Rx),LOADSET=(Ry)+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 GETPC: EXPECTED RESULT- CASE#12 SUCCESSFUL+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.57.13 COMPLETING THE GETPC MACRO TEST+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
getpc: entering getpc_func.+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
getpc: Case-1 GETPC_DBI.+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
getpc: case-1 completed successfully.+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
getpc: Case-2 GETPC_PBI.+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
getpc: case-2 completed successfully.+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
getpc: Case-3 GETPC_DUMP.+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
getpc: case-3 completed successfully.+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
getpc: Case-4 LOADSET NULL+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
getpc: case-4 completed successfully.+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
getpc: completed getpc_func.+
CSMP0097I 13.57.13 CPU-A SS-BSS  SSU-BSS  IS-01
API5 driver exited +
```

Appendix B: PROGC API Group

The PROGC API group exercises the various parameters supported by the PROGC macro and the progc() C function call. Each of these parameters can be exercised separately by specifying the appropriate case. Some cases require the user to specify a program name or loadset name. This API group also supports the BSO/CSO option which allows the user to limit the exercises to PROGC or progc() test. The SNAP option is also supported, allowing the user to specify the number of bytes to display of the retrieved data.

The following test cases are supported by this API group.

BSO non-error cases

Case Number	Description
1	Default parms, NAME=(Rx)
2	NAME=(Rx),DBI
3	NAME=(Rx),PBI
4	NAME=CVZZ

BSO error cases (specify ERR parameter)

Case Number	Description
5	ERROR CASE-NAME=ZZZZ

CSO non-error cases

Case Number	Description
1	progc(name,PROGC_PBI)
2	progc(name,PROGC_DBI)

CSO error cases (specify ERR parameter)

Case Number	Description
1	ERROR CASE-progc(NULL,PROGC_PBI)

Examples

The following example invokes PROGC and progc() case 1 using the default program name CVZZ and displays 100 bytes:

```
ZTEST API5 API-PROGC CASE-1 PName-CVZZ SNAP-100
```

The following example invokes progc() error case 1:

```
ZTEST API5 API-PROGC CASE-1 CSO ERR
```

The following example invokes PROGC error case 5:

```
ZTEST API5 API-PROGC CASE-5 BSO ERR
```

Messages

Below is a sampling of some API5 PROGC driver messages:

ZTEST API5 API-PROGC PN-CVZZ

```
QMAC0001I 13.59.12 STARTING THE PROGC MACRO TEST+
CSMP0097I 13.59.12 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.59.12 PROGC:  PROCESSING VALID CASES+
CSMP0097I 13.59.12 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.59.12 PROGC: CASE#1-Default parms, NAME=(Rx)+
CSMP0097I 13.59.12 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.59.12 PROGC: EXPECTED RESULT- CASE#1 SUCCESSFUL+
CSMP0097I 13.59.12 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.59.12 PROGC: CASE#2-NAME=(Rx),DBI+
CSMP0097I 13.59.12 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.59.12 PROGC: EXPECTED RESULT- CASE#2 SUCCESSFUL+
CSMP0097I 13.59.12 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.59.12 PROGC: CASE#3-NAME=(Rx),PBI+
CSMP0097I 13.59.12 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.59.12 PROGC: EXPECTED RESULT- CASE#3 SUCCESSFUL+
CSMP0097I 13.59.12 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.59.12 PROGC: CASE#4-NAME=CVZZ+
CSMP0097I 13.59.12 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.59.12 PROGC: EXPECTED RESULT- CASE#4 SUCCESSFUL+
CSMP0097I 13.59.12 CPU-A SS-BSS  SSU-BSS  IS-01
QMAC0001I 13.59.12 COMPLETING THE PROGC MACRO TEST+
CSMP0097I 13.59.12 CPU-A SS-BSS  SSU-BSS  IS-01
progc: entering progc_func+
CSMP0097I 13.59.12 CPU-A SS-BSS  SSU-BSS  IS-01
progc: case-1 progc(name,PROGC_PBI)+
CSMP0097I 13.59.12 CPU-A SS-BSS  SSU-BSS  IS-01
progc: case-1 completed successfully
CSMP0097I 13.59.12 CPU-A SS-BSS  SSU-BSS  IS-01
progc: 2-progc(name,PROGC_DBI)+
CSMP0097I 13.59.12 CPU-A SS-BSS  SSU-BSS  IS-01
progc: case-2 completed successfully
CSMP0097I 13.59.12 CPU-A SS-BSS  SSU-BSS  IS-01
progc: completed progc_func.+
CSMP0097I 13.59.12 CPU-A SS-BSS  SSU-BSS  IS-01
API5 driver exited +
```

© Copyright IBM Corp. 2005, 2011

Appendix C: UXRIT API

The UXRIT test cases test the RITID user exit UCCRIT. There are two record IDs Y1 and YC defined for this test. These two records are unique in the respect of having the attribute of UEXIT set to YES (default) in RIAT table. This attribute along with other attributes in the table can be altered by "ZRTDM MODIFY RECID-recid,UEXIT-NO,...", By altering the attributes of the records, the user can use these test cases to explore a wide variety of test cases. Since the attributes can be altered on line, the test cases only issue the "User Exit (NOT) entered" message. There are four test cases in a row, testing record ID Y1, Y7, YC and non-existent ?Z. With the default setting, we expect to see Y1 entered, Y7 NOT entered, YC entered and ?Z NOT found.

Messages

Below is a sampling of some API5 UXRIT driver messages:

ZTEST API5 API-UXRIT

```
QMAC/uxrit: Test record ID Y1+
CSMP0097I 09.14.38 CPU-B SS-BSS  SSU-HPN  IS-01
CUSR0001I 09.14.38 Message from User Exit: RECORD ID Y1 detected+
CSMP0097I 09.14.38 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC/uxrit: User Exit entered (record ID-Y1)+
CSMP0097I 09.14.38 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC/uxrit: Test record ID Y7+
CSMP0097I 09.14.38 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC/uxrit: User Exit NOT entered (record ID-Y7)+
CSMP0097I 09.14.38 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC/uxrit: Test record ID YC+
CSMP0097I 09.14.38 CPU-B SS-BSS  SSU-HPN  IS-01
CUSR0001I 09.14.38 Message from User Exit: RECORD ID YC detected+
CSMP0097I 09.14.38 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC/uxrit: User Exit entered (record ID-YC)+
CSMP0097I 09.14.38 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC/uxrit: Test record ID ?Z+
CSMP0097I 09.14.38 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC/uxrit: getfc can NOT find record ?Z+
CSMP0097I 09.14.38 CPU-B SS-BSS  SSU-HPN  IS-01
QMAC/uxrit: Completes all test cases, now return+
CSMP0097I 09.14.38 CPU-B SS-BSS  SSU-HPN  IS-01
API5 driver exited +
```

ZTEST API5 API-UXRIT NOV

```
CUSR0001I 09.14.07 Message from User Exit: RECORD ID Y1 detected+
CSMP0097I 09.14.08 CPU-B SS-BSS  SSU-HPN  IS-01
CUSR0001I 09.14.08 Message from User Exit: RECORD ID YC detected+
CSMP0097I 09.14.08 CPU-B SS-BSS  SSU-HPN  IS-01
API5 driver exited +
```