TPF Toolkit 4.6 enhancements

Matt Gritter

Agenda

What’s new in Fix Pack 4

What’s new in Fix Pack 5

What’s new in Fix Pack 6

What’s new in Fix Pack 7

What’s new in Fix Pack 8

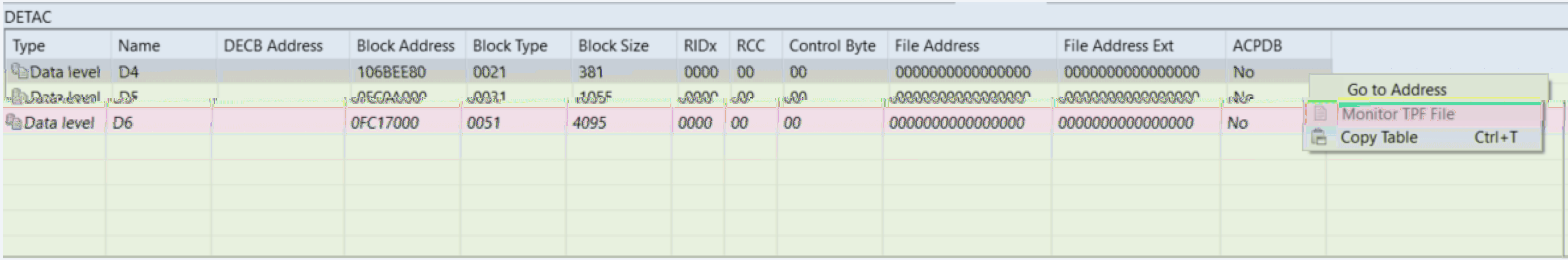
What’s new in Fix Pack 4

Available 2019-06-28

• DETAC view

The DETAC view displays information about all working storage blocks that are detached from the ECB data level or data event control block (DECB).

You can use this view to monitor those working storage blocks.



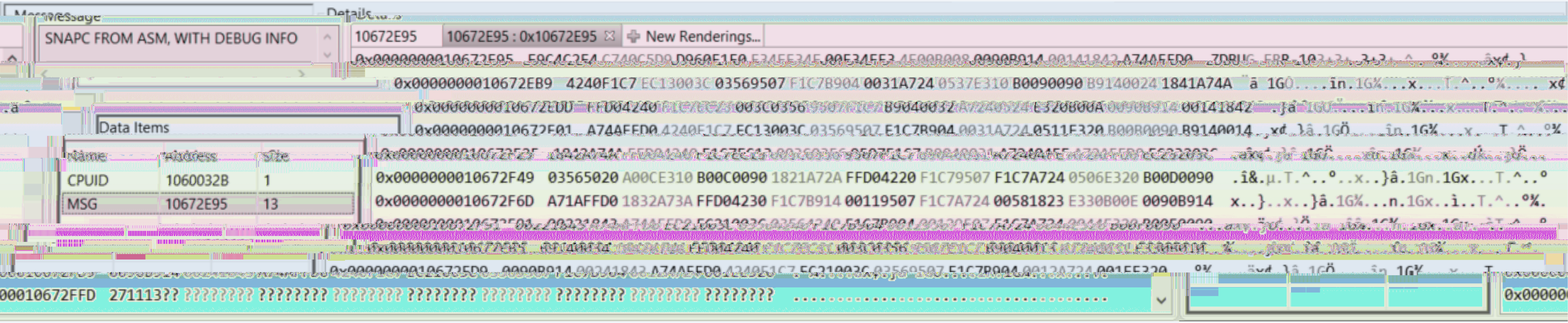
DETAC

Type	Name	DECB Address	Block Address	Block Type	Block Size	RIDx	RCC	Control Byte	File Address	File Address Ext	ACPDB
Data level	D4		106BEE80	0021	381	0000	00	00	0000000000000000	0000000000000000	No
Data level	D5		0FC0A000	0031	1055	0000	00	00	0000000000000000	0000000000000000	No
Data level	D6		0FC17000	0051	4095	0000	00	00	0000000000000000	0000000000000000	No

• Dump Data Items view

The Dump Data Items view displays the data items that are captured by the LISTC macro or the dump viewer user exit for snap dump processing.

This view is populated only when you are debugging a SNAP dump that is selected from the Dump File view.



SNAPC FROM ASM, WITH DEBUG INFO

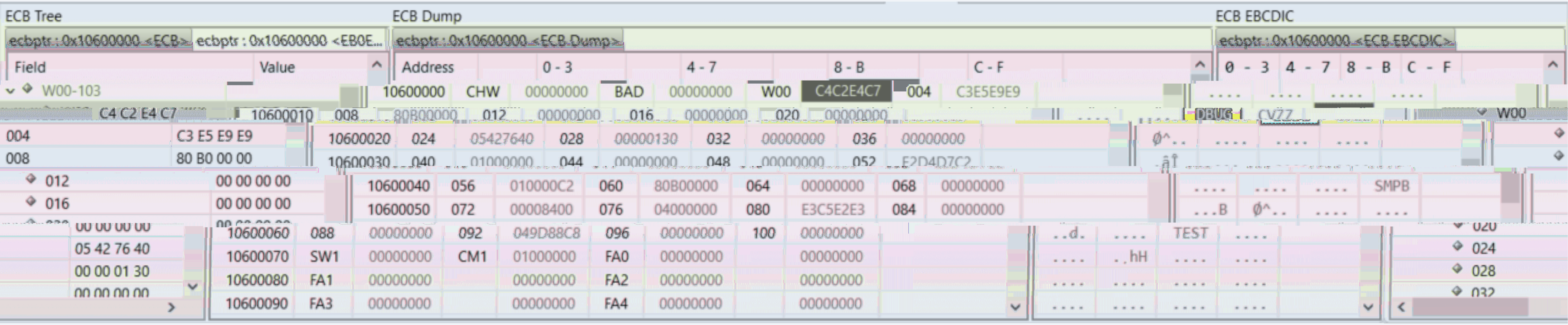
Name	Address	Size
CPUID	1060032B	1
MSG	10672E95	13

```
0x0000000010672E95  E9C4C2E4 C740C5D9 D960F1F0 F34FF34F 00F34FF3 4F00B008 0000B914 00141842 A74AFFD0  .ZDBUG ERR-10...
0x0000000010672EB9  4240F1C7 EC13003C 03569507 F1C7B904 0031A724 0537E310 B0090090 B9140024 1841A74A  ā 1GÖ....in.1G%...x...T.^..°%.... x¢
0x0000000010672EDD  FFD04240 F1C7EC13 003C0356 9507F1C7 B9040032 A7240524 E320B00A 00908914 00141842  .}ā 1GÜ....in.1G%...x...T.^...%...
0x0000000010672F01  A74AFFD0 4240F1C7 EC13003C 03569507 F1C7B904 0031A724 0511E320 B00B0090 B9140014  .x¢.}ā.1GÖ....in.1G%...x...T.^..°%
0x0000000010672F49  03565020 A00CE310 B00C0090 1821A72A FFD04220 F1C79507 F1C7A724 0506E320 B00D0090  .ì&.µ.T.^..°..x..}â.1Gn.1Gx...T.^..°
0x0000000010672F6D  A71AFFD0 1832A73A FFD04230 F1C7B914 00119507 F1C7A724 00581823 E330B00E 0090B914  x..}..x..}â.1G%...n.1Gx..ì..T.^..°%.
0x0000000010672FD9  0090B914 00241843 A74AFFD0 4240F1C7 EC21003C 03569507 F1C7B904 0012A724 001EE320  .°%....x¢.}â.1GÖ....in.1G%...x...T
0x0000000010672FFD  271113?? ???????? ???????? ???????? ???????? ???????? ???????? ???????? ????????  ....................................
```

• ECB view

The ECB view is a memory view that displays page one of the ECB.

The displayed memory data is similar to the Entry Blk section of a z/TPF dump.



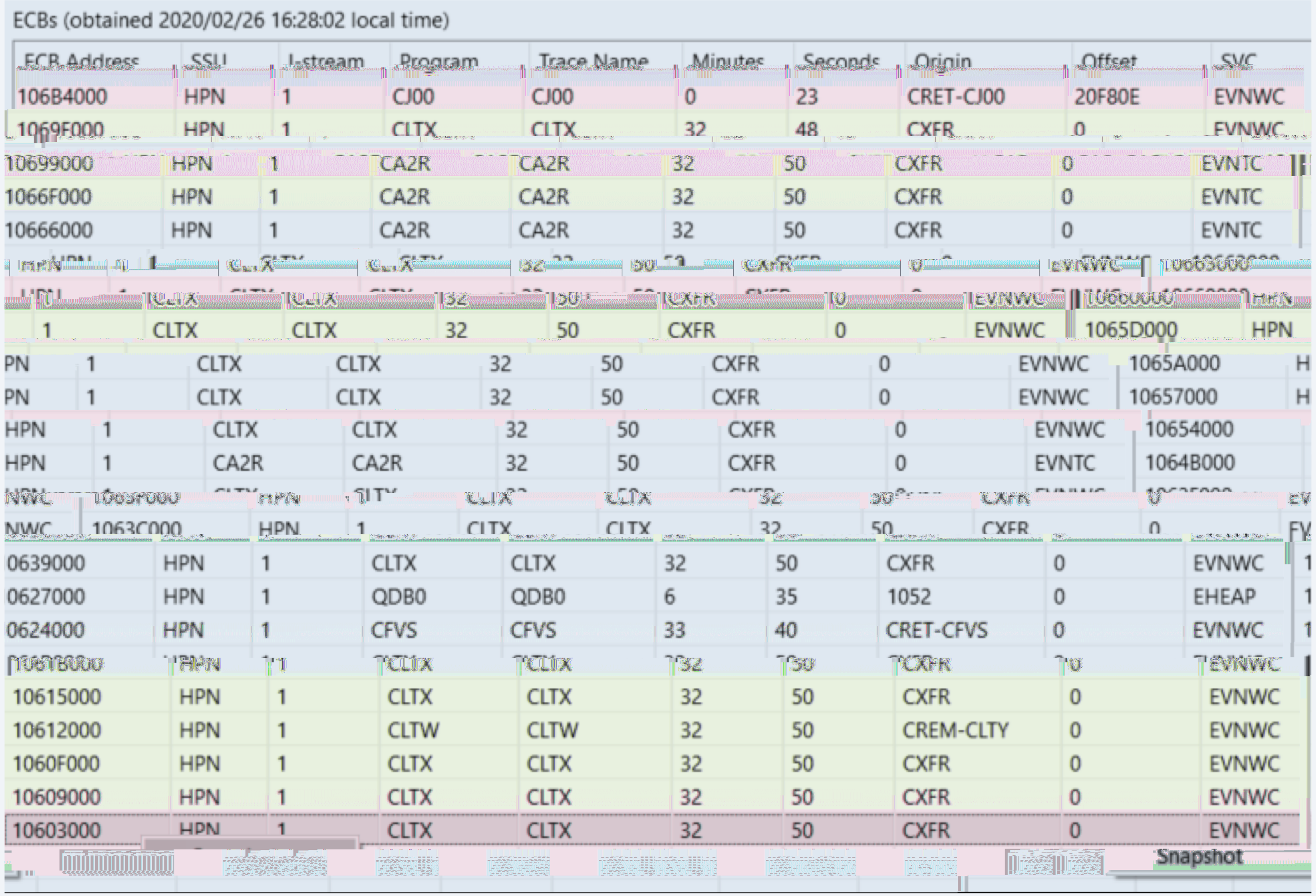
ECB Tree
ecbptr : 0x10600000 <ECB>
ecbptr : 0x10600000 <EB0E...
ECB Dump
ecbptr : 0x10600000 <ECB Dump>
ECB EBCDIC
ecbptr : 0x10600000 <ECB EBCDIC>

```
Field          Value
W00-103
               C4 C2 E4 C7
004            C3 E5 E9 E9
008            80 B0 00 00
   012         00 00 00 00
   016         00 00 00 00
   020         00 00 00 00
   024         05 42 76 40
   028         00 00 01 30
   032         00 00 00 00
```

```
Address     0 - 3          4 - 7          8 - B          C - F
10600000    CHW 00000000   BAD 00000000   W00 C4C2E4C7   004 C3E5E9E9
10600010    008 80B00000   012 00000000   016 00000000   020 00000000
10600020    024 05427640   028 00000130   032 00000000   036 00000000
10600030    040 01000000   044 00000000   048 00000000   052 E2D4D7C2
10600040    056 010000C2   060 80B00000   064 00000000   068 00000000
10600050    072 00008400   076 04000000   080 E3C5E2E3   084 00000000
10600060    088 00000000   092 049D88C8   096 00000000   100 00000000
10600070    SW1 00000000   CM1 01000000   FA0 00000000       00000000
10600080    FA1 00000000       00000000   FA2 00000000       00000000
10600090    FA3 00000000       00000000   FA4 00000000       00000000
```

```
0 - 3   4 - 7   8 - B   C - F
....    ....    ....    ....
DBUG    CVZZ
Ø^..    ....    ....    ....
.ãÎ
....    ....    ....    SMPB
...B    Ø^..    ....    ....
..d.    ....    TEST    ....
....    ..hH    ....    ....
....    ....    ....    ....
....    ....    ....    ....
```

What’s new in Fix Pack 4



ECBs (obtained 2020/02/26 16:28:02 local time)

ECB Address	SSU	I-stream	Program	Trace Name	Minutes	Seconds	Origin	Offset	SVC
106B4000	HPN	1	CJ00	CJ00	0	23	CRET-CJ00	20F80E	EVNWC
1069F000	HPN	1	CLTX	CLTX	32	48	CXFR	0	EVNWC
10699000	HPN	1	CA2R	CA2R	32	50	CXFR	0	EVNTC
1066F000	HPN	1	CA2R	CA2R	32	50	CXFR	0	EVNTC
10666000	HPN	1	CA2R	CA2R	32	50	CXFR	0	EVNTC
[illegible]	HPN	1	CLTX	CLTX	32	50	CXFR	0	EVNWC
10663000	HPN	1	CLTX	CLTX	32	50	CXFR	0	EVNWC
10660000	HPN	1	CLTX	CLTX	32	50	CXFR	0	EVNWC
1065D000	HPN	1	CLTX	CLTX	32	50	CXFR	0	EVNWC
1065A000	HPN	1	CLTX	CLTX	32	50	CXFR	0	EVNWC
10657000	HPN	1	CLTX	CLTX	32	50	CXFR	0	EVNWC
10654000	HPN	1	CLTX	CLTX	32	50	CXFR	0	EVNWC
1064B000	HPN	1	CA2R	CA2R	32	50	CXFR	0	EVNTC
1063F000	HPN	1	CLTX	CLTX	32	50	CXFR	0	EVNWC
1063C000	HPN	1	CLTX	CLTX	32	50	CXFR	0	EV
0639000	HPN	1	CLTX	CLTX	32	50	CXFR	0	EVNWC
0627000	HPN	1	QDB0	QDB0	6	35	1052	0	EHEAP
0624000	HPN	1	CFVS	CFVS	33	40	CRET-CFVS	0	EVNWC
1061B000	HPN	1	CLTX	CLTX	32	50	CXFR	0	EVNWC
10615000	HPN	1	CLTX	CLTX	32	50	CXFR	0	EVNWC
10612000	HPN	1	CLTW	CLTW	32	50	CREM-CLTY	0	EVNWC
1060F000	HPN	1	CLTX	CLTX	32	50	CXFR	0	EVNWC
10609000	HPN	1	CLTX	CLTX	32	50	CXFR	0	EVNWC
10603000	HPN	1	CLTX	CLTX	32	50	CXFR	0	EVNWC

- **ECB Snapshot view**

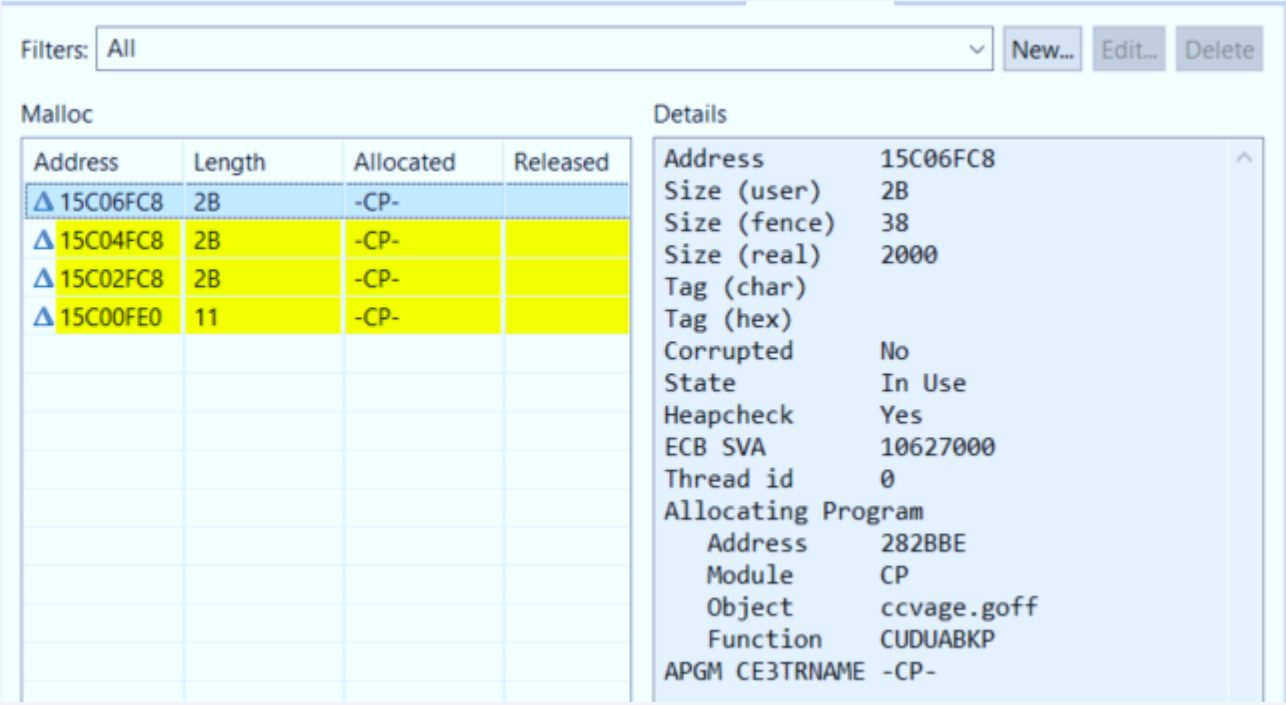
The ECB Snapshot view provides a way for you to learn about ECBs on a z/TPF system that are older than a specified number of seconds.

ZDECB command with a graphical user interface

What’s new in Fix Pack 4

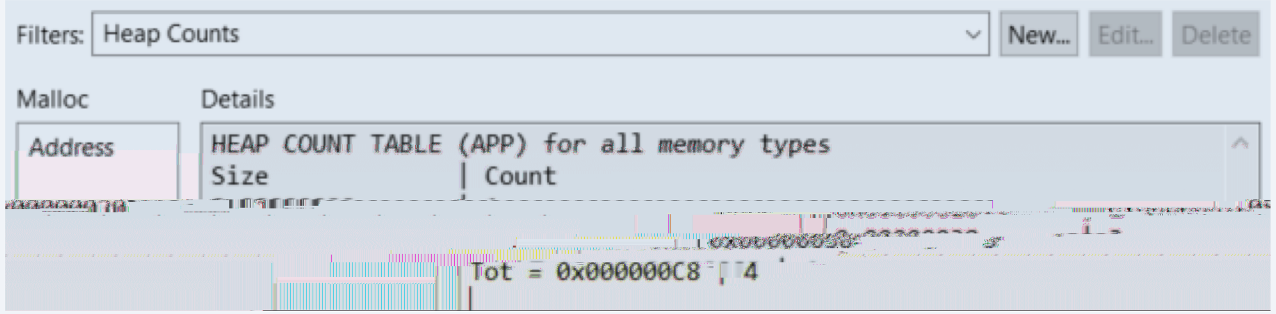
• Malloc view

The Malloc view displays the status of allocated memory blocks for the active debug session in the Debug view.

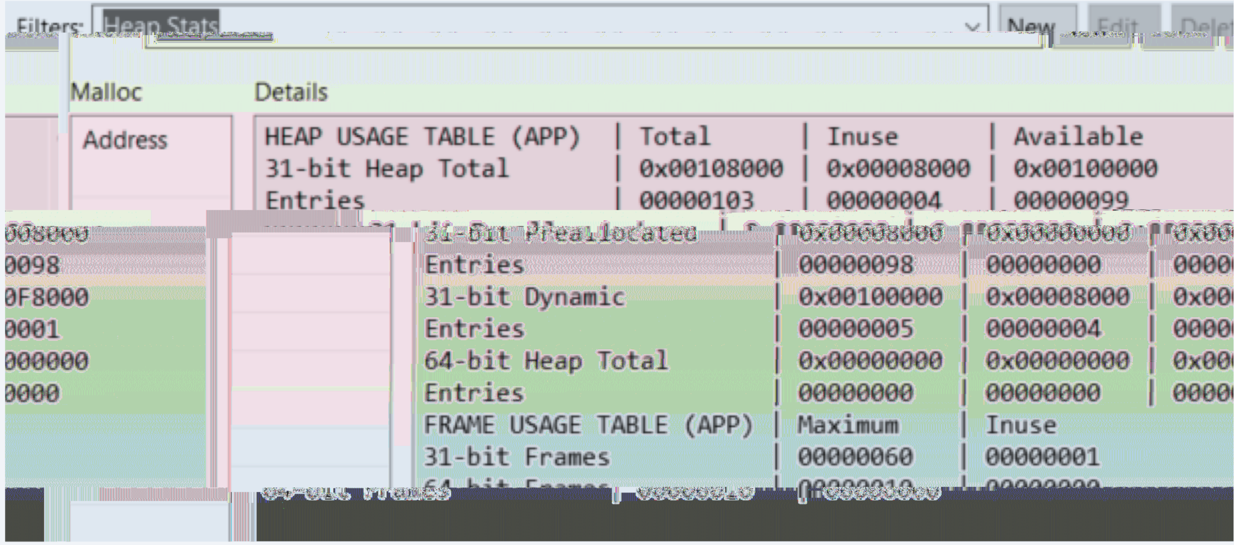


Address	Length	Allocated	Released
Δ 15C06FC8	2B	-CP-	
Δ 15C04FC8	2B	-CP-	
Δ 15C02FC8	2B	-CP-	
Δ 15C00FE0	11	-CP-	

```
Address          15C06FC8
Size (user)      2B
Size (fence)     38
Size (real)      2000
Tag (char)
Tag (hex)
Corrupted        No
State            In Use
Heapcheck        Yes
ECB SVA          10627000
Thread id        0
Allocating Program
   Address       282BBE
   Module        CP
   Object        ccvage.goff
   Function      CUDUABKP
APGM CE3TRNAME -CP-
```



```
HEAP COUNT TABLE (APP) for all memory types
Size                   | Count
                       Tot = 0x000000C8 | 4
```

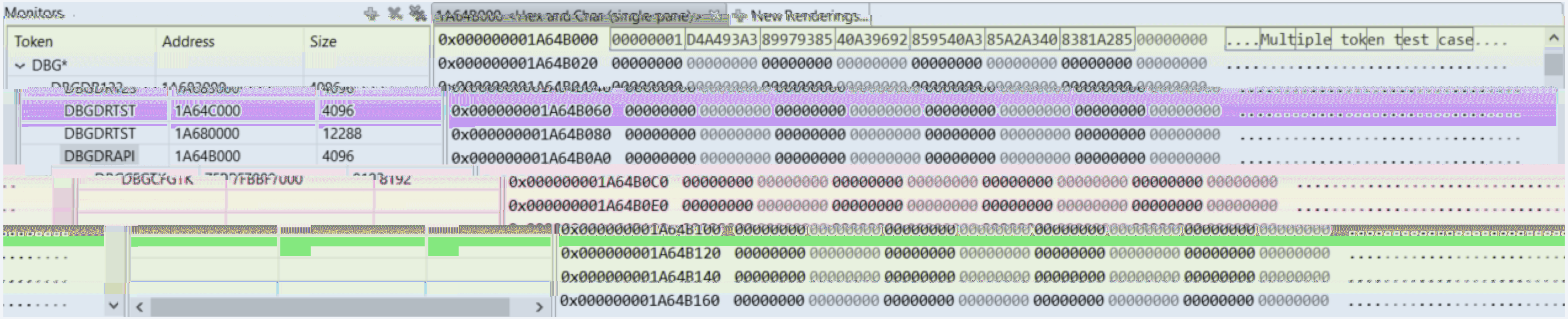


```
HEAP USAGE TABLE (APP)   | Total      | Inuse      | Available
31-bit Heap Total        | 0x00108000 | 0x00008000 | 0x00100000
Entries                  | 00000103   | 00000004   | 00000099
31-bit Preallocated      | 0x00008000 | 0x00000000 | 0x00
Entries                  | 00000098   | 00000000   | 0000
31-bit Dynamic           | 0x00100000 | 0x00008000 | 0x00
Entries                  | 00000005   | 00000004   | 0000
64-bit Heap Total        | 0x00000000 | 0x00000000 | 0x00
Entries                  | 00000000   | 00000000   | 0000
FRAME USAGE TABLE (APP)  | Maximum    | Inuse
31-bit Frames            | 00000060   | 00000001
```

• System Heap view

The System Heap view accepts the token that identifies a system heap area and displays the storage for that token.

You can monitor and modify the system heap contents in this view.



Token	Address	Size
⌄ DBG*		
DBGDR123	1A683000	4096
DBGDRTST	1A64C000	4096
DBGDRTST	1A680000	12288
DBGDRAPI	1A64B000	4096
DBGCFGTK	7FFBBF7000	8192

```
0x000000001A64B000  00000001 D4A493A3 89979385 40A39692 859540A3 85A2A340 8381A285 00000000  ....Multiple token test case....
0x000000001A64B020  00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000  ................................
0x000000001A64B040  00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000  ................................
0x000000001A64B060  00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000  ................................
0x000000001A64B080  00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000  ................................
0x000000001A64B0A0  00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000  ................................
0x000000001A64B0C0  00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000  ................................
0x000000001A64B0E0  00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000  ................................
0x000000001A64B100  00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000  ................................
0x000000001A64B120  00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000  ................................
0x000000001A64B140  00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000  ................................
0x000000001A64B160  00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000  ................................
```

What’s new in Fix Pack 5

Available 2019-09-26

- **Custom debug registration**

A custom debug registration type contains the basic information about a set of conditions that must be satisfied to start debugging an ECB.

Allows stopping at a specific point under specific conditions in the application by implementing a function prototype.

- In order to implement a custom debug registration someone must:
  - Implement the user exit that will run on the z/TPF system
  - Create or update a plug-in project that is distributed to TPF Toolkit users

Custom debug registration

• Implement user exit

Update segment cdbxud.c switch statement with new ID number (Valid IDs are 101-255)

Call user exit by function in application code, see c_udrt.h for function prototype

Detailed steps available for Custom debug registration support in IBM Knowledge Center

• Plug-in project

Add com.ibm.tpf.toolkit.register as a dependency.

Contribute the extension point

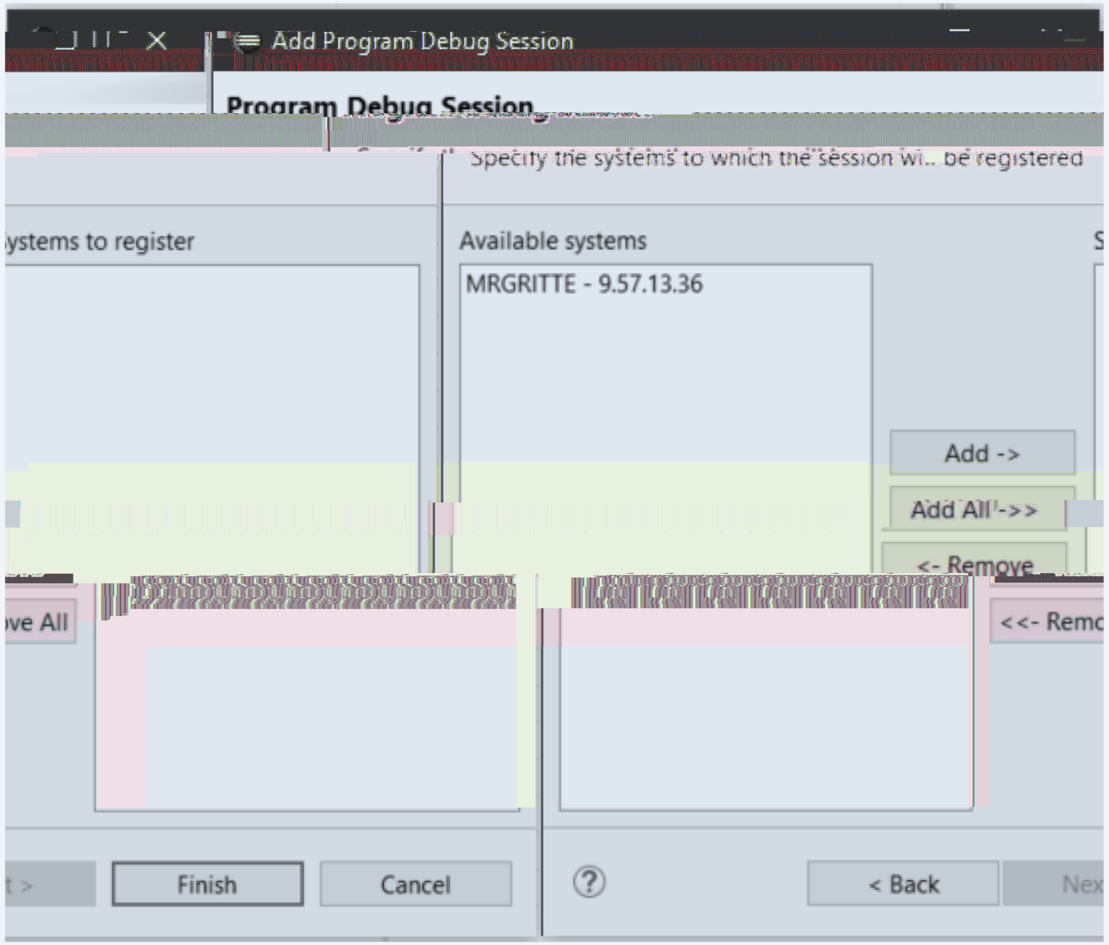
com.ibm.tpf.toolkit.register.customRegistration

specifying the to display to the user and the that was added to segment cdbxud.c (101-255)

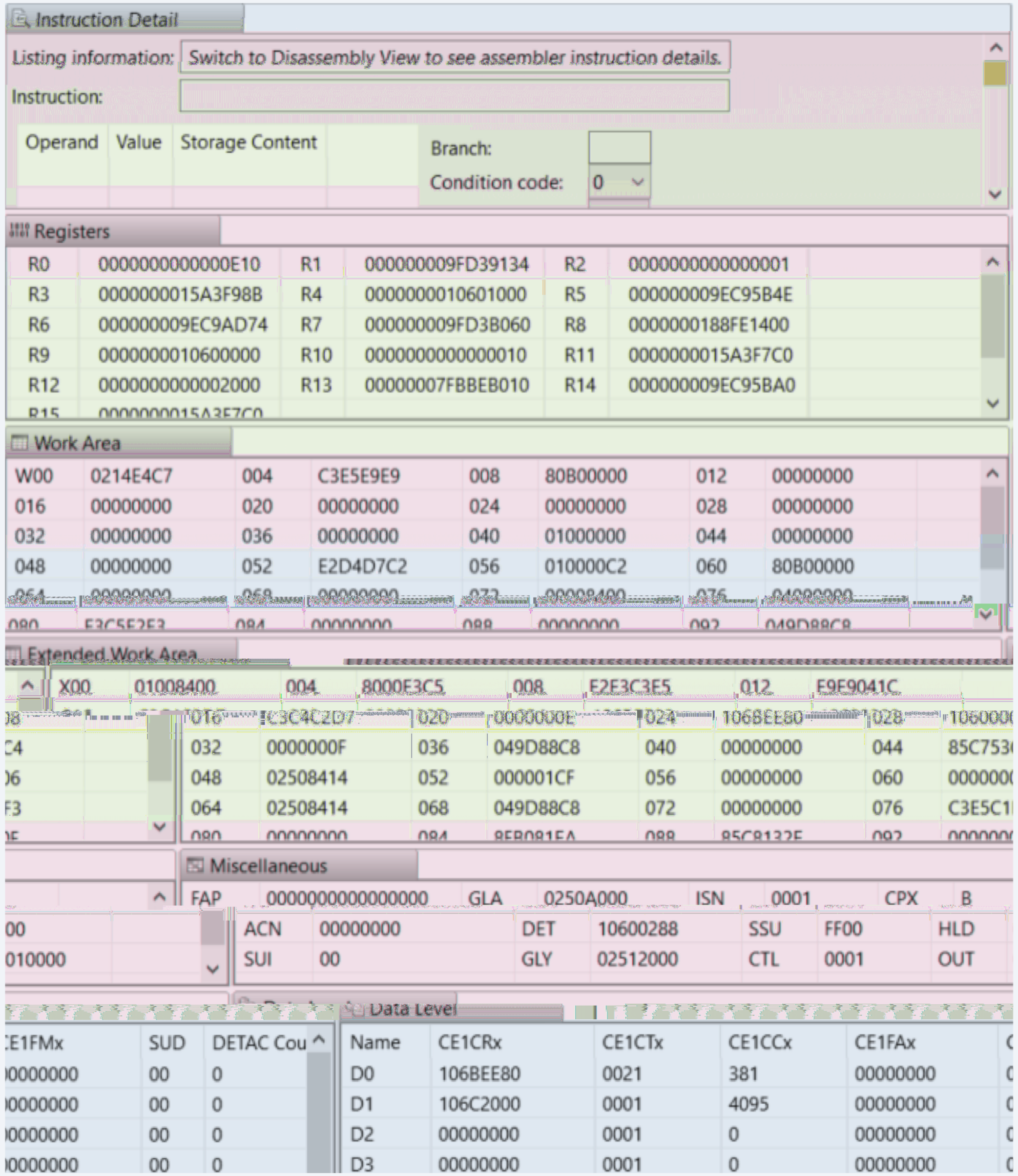
Detailed steps available in the administrator guide.

- **Register on session create**

Additional page included in the debug session and code coverage sessions wizards where a user specifies which systems the registration entry should be added to immediately after the registration entry object is created in the TPF Toolkit client.



What’s new in Fix Pack 5



Instruction Detail

Listing information: Switch to Disassembly View to see assembler instruction details.

Instruction:

Operand	Value	Storage Content

Branch:

Condition code: 0

Registers

R0	0000000000000E10	R1	000000009FD39134	R2	0000000000000001
R3	0000000015A3F98B	R4	0000000010601000	R5	000000009EC95B4E
R6	000000009EC9AD74	R7	000000009FD3B060	R8	0000000188FE1400
R9	0000000010600000	R10	0000000000000010	R11	0000000015A3F7C0
R12	0000000000002000	R13	00000007FBBEB010	R14	000000009EC95BA0
R15	0000000015A3F7C0				

Work Area

W00	0214E4C7	004	C3E5E9E9	008	80B00000	012	00000000
016	00000000	020	00000000	024	00000000	028	00000000
032	00000000	036	00000000	040	01000000	044	00000000
048	00000000	052	E2D4D7C2	056	010000C2	060	80B00000
080	E3C5E2E3	084	00000000	088	00000000	092	049D88C8

Extended Work Area

X00	01008400	004	8000E3C5	008	E2E3C3E5	012	E9E9041C
016	C3C4C2D7	020	0000000E	024	1068EE80	028	106000
032	0000000F	036	049D88C8	040	00000000	044	85C753
048	02508414	052	000001CF	056	00000000	060	000000
064	02508414	068	049D88C8	072	00000000	076	C3E5C1
080	00000000	084	8EB081EA	088	85C8132F	092	000000

Miscellaneous

FAP	0000000000000000	GLA	0250A000	ISN	0001	CPX	B
ACN	00000000	DET	10600288	SSU	FF00	HLD	
SUI	00	GLY	02512000	CTL	0001	OUT	

E1FMx	SUD	DETAC Cou
0000000	00	0
0000000	00	0
0000000	00	0
0000000	00	0

Data Level

Name	CE1CRx	CE1CTx	CE1CCx	CE1FAx
D0	106BEE80	0021	381	00000000
D1	106C2000	0001	4095	00000000
D2	00000000	0001	0	00000000
D3	00000000	0001	0	00000000

- **ECB Summary view**

The ECB Summary view highlights the commonly used information in registers, instructions, and the ECB, such as the work areas, data levels, and other selected fields.

In this view, you can monitor and modify the registers, currently running instructions, and important fields in the ECB memory of the application that you are debugging to see whether the application behaves as expected.

• ECB Trace view

The ECB Trace view displays the call history of the ECB that is being debugged.

To use the ECB Trace view, APAR PJ45879 (Sept 2019) must be applied to the z/TPF system.

tpf_ecb_trace.swagger.json documents the REST APIs
tpf_ecb_trace.srvc.json must be deployed to discover service

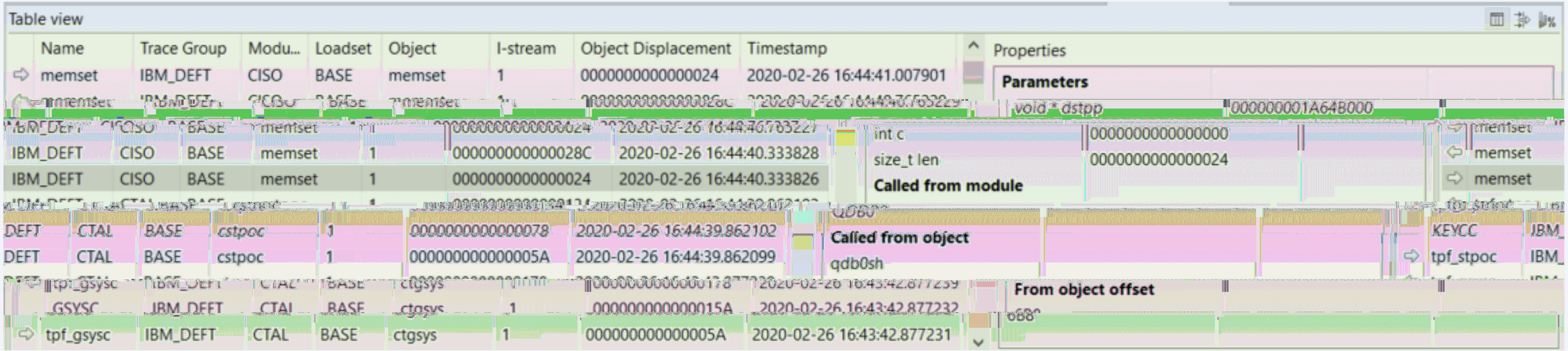


Table view

Name	Trace Group	Modu...	Loadset	Object	I-stream	Object Displacement	Timestamp
⇨ memset	IBM_DEFT	CISO	BASE	memset	1	0000000000000024	2020-02-26 16:44:41.007901
memset	IBM_DEFT	CISO	BASE	memset	1	000000000000028C	2020-02-26 16:44:40.765229
[illegible]	IBM_DEFT	CISO	BASE	memset	1	0000000000000024	2020-02-26 16:44:40.765227
memset	IBM_DEFT	CISO	BASE	memset	1	000000000000028C	2020-02-26 16:44:40.333828
memset	IBM_DEFT	CISO	BASE	memset	1	0000000000000024	2020-02-26 16:44:40.333826
[illegible]	IBM_DEFT	CTAL	BASE	cstpoc	1	[illegible]	[illegible]
KEYCC	IBM_DEFT	CTAL	BASE	cstpoc	1	0000000000000078	2020-02-26 16:44:39.862102
⇨ tpf_stpoc	IBM_DEFT	CTAL	BASE	cstpoc	1	000000000000005A	2020-02-26 16:44:39.862099
tpf_gsysc	IBM_DEFT	CTAL	BASE	ctgsys	1	0000000000000178	2020-02-26 16:43:42.877239
tpf_gsysc	IBM_DEFT	CTAL	BASE	ctgsys	1	000000000000015A	2020-02-26 16:43:42.877232
⇨ tpf_gsysc	IBM_DEFT	CTAL	BASE	ctgsys	1	000000000000005A	2020-02-26 16:43:42.877231

Properties

Parameters

void * dstpp	000000001A64B000
int c	0000000000000000
size_t len	0000000000000024

Called from module

QDB0

Called from object

qdb0sh

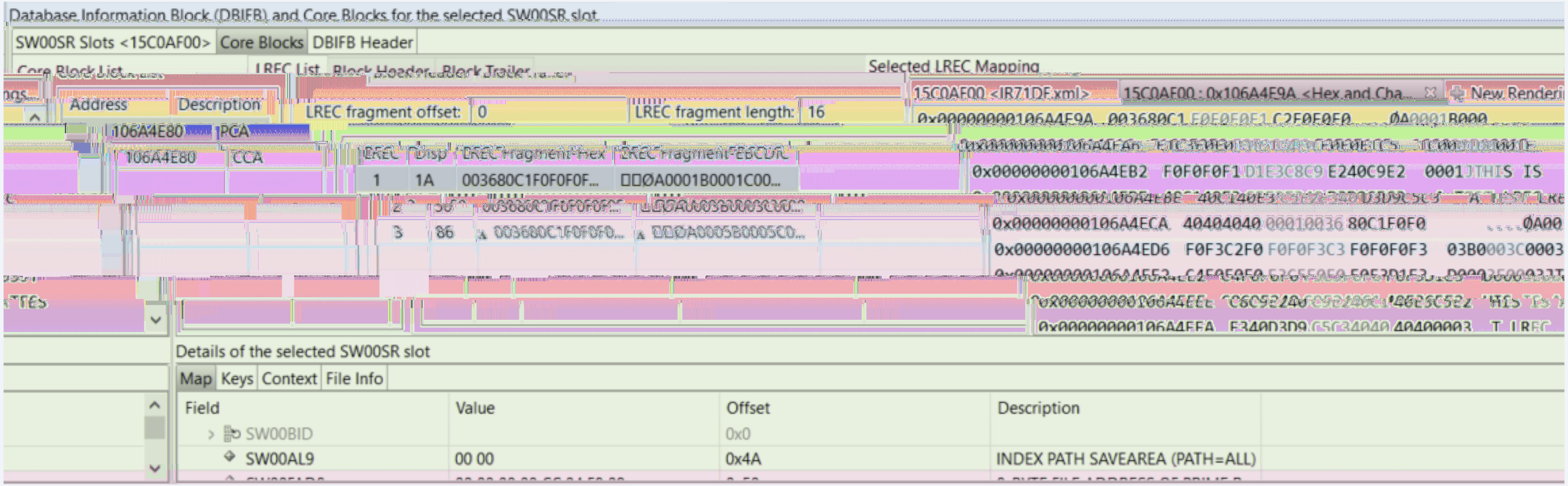
From object offset

6B8

• SW00SR view

The SW00SR view displays information about the SW00SR slots that are opened by a z/TPFDF application.

You can view and edit information about a SW00SR slot in the various tabs in this view.



Database Information Block (DBIFB) and Core Blocks for the selected SW00SR slot

Address	Description
106A4E80	PCA
106A4E80	CCA

LREC fragment offset: 0 LREC fragment length: 16

LREC	Disp	LREC fragment-Hex	LREC fragment-EBCDIC
1	1A	003680C1F0F0F0F...	□□ØA0001B0001C00...
2	50	003680C1F0F0F0F...	□□ØA0003B0003C00...
3	86	003680C1F0F0F0...	□□ØA0005B0005C0...

```
0x00000000106A4E9A  003680C1 F0F0F0F1 C2F0F0F0   ..ØA0001B000
0x00000000106A4EA6  [illegible]
0x00000000106A4EB2  F0F0F0F1 D1E3C8C9 E240C9E2   0001JTHIS IS
0x00000000106A4EBE  40C140E3 [illegible]          A TEST LRE
0x00000000106A4ECA  40404040 00010036 80C1F0F0       ....ØA00
0x00000000106A4ED6  F0F3C2F0 F0F0F3C3 F0F0F0F3   03B0003C0003
0x00000000106A4EE2  C4F0F0F0 F3C5F0F0 F0F3D1E3   D0003E0003JT
0x00000000106A4EEE  C8C9E240 C9E240C1 40E3C5E2   HIS IS A TES
0x00000000106A4EFA  E340D3D9 C5C34040 40400003   T LREC
```

Details of the selected SW00SR slot

Field	Value	Offset	Description
SW00BID		0x0	
SW00AL9	00 00	0x4A	INDEX PATH SAVEAREA (PATH=ALL)

What’s new in Fix Pack 6

Available 2019-12-12

What’s new in Fix Pack 6

- **ECB Trace Analysis Report**

Produce and export a report from the contents of the ECB Trace view.

Metrics include:
Call Hierarchy
File / Record ID activity *
Macro call counts *
Memory allocations counts *
Modules entered counts *

* Includes caller information

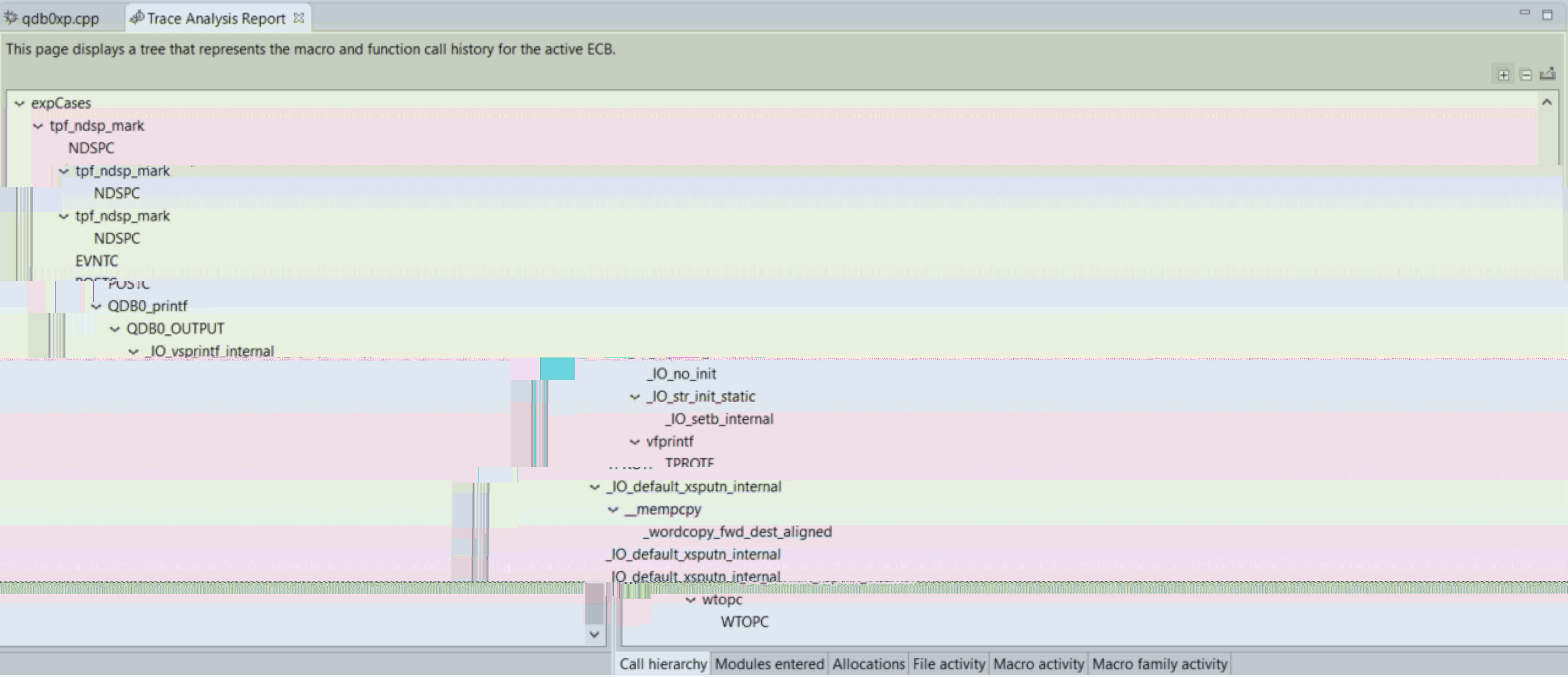
- **SonarQube format for code coverage results**

Prior to Fix Pack 6, only code coverage results in IBM format were produced and the results could only be viewed using TPF Toolkit.

With this enhancement, you can generate code coverage results in the generic SonarQube format from the TPF Toolkit web application and view the results by using a SonarQube instance.

Analysis Report Pictures

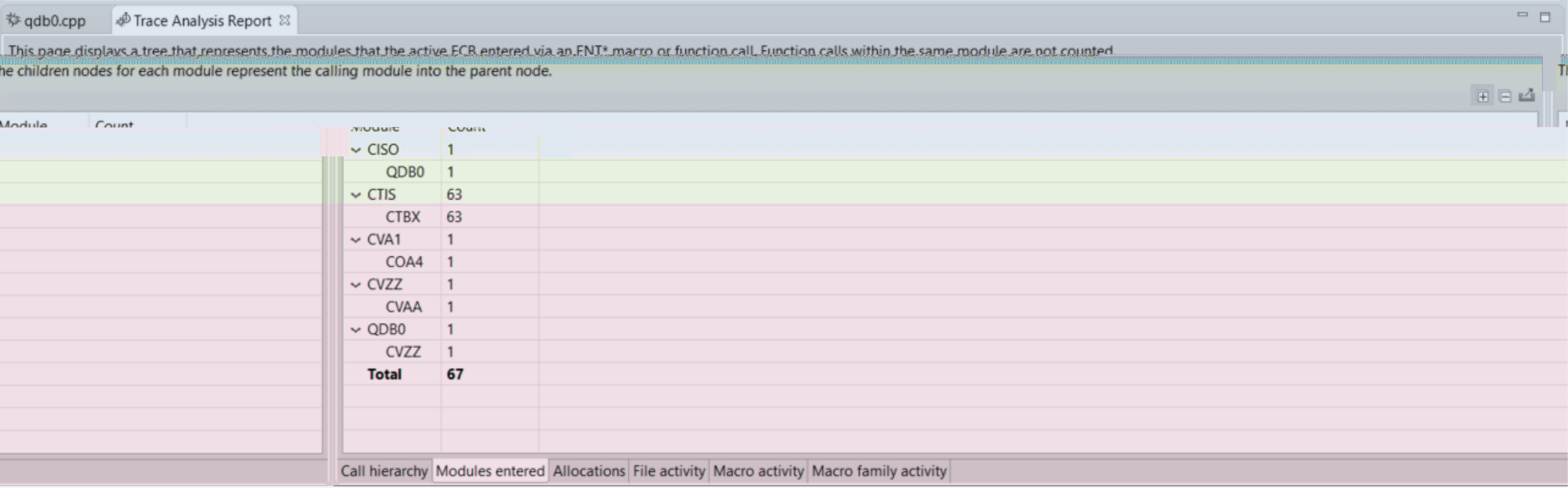
Call hierarchy page



This page displays a tree that represents the macro and function call history for the active ECB.

```
expCases
  tpf_ndsp_mark
    NDSPC
     tpf_ndsp_mark
       NDSPC
     tpf_ndsp_mark
       NDSPC
     EVNTC
       POSTC
       QDB0_printf
         QDB0_OUTPUT
           _IO_vsprintf_internal
                                   _IO_no_init
                                   _IO_str_init_static
                                     _IO_setb_internal
                                   vfprintf
                                     TPROTE
                                 _IO_default_xsputn_internal
                                   __mempcpy
                                     _wordcopy_fwd_dest_aligned
                                 _IO_default_xsputn_internal
                                 _IO_default_xsputn_internal
                                         wtopc
                                           WTOPC
```

Modules entered page

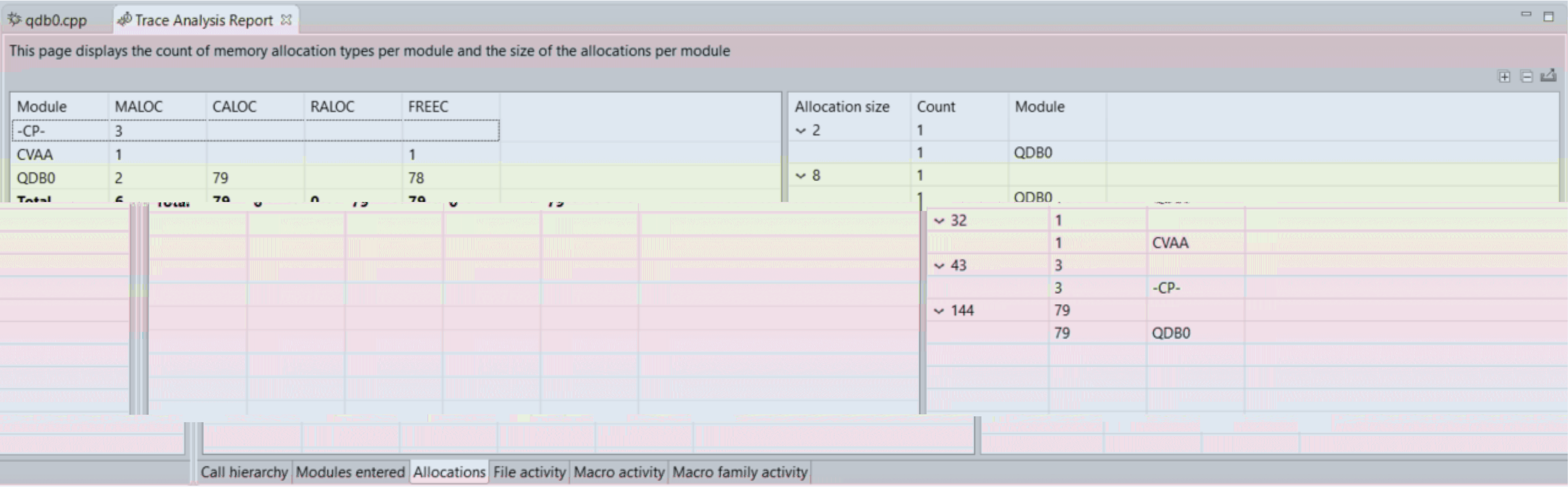


This page displays a tree that represents the modules that the active ECB entered via an ENT* macro or function call. Function calls within the same module are not counted.

he children nodes for each module represent the calling module into the parent node.

Module	Count
CISO	1
QDB0	1
CTIS	63
CTBX	63
CVA1	1
COA4	1
CVZZ	1
CVAA	1
QDB0	1
CVZZ	1
Total	**67**

Allocations page

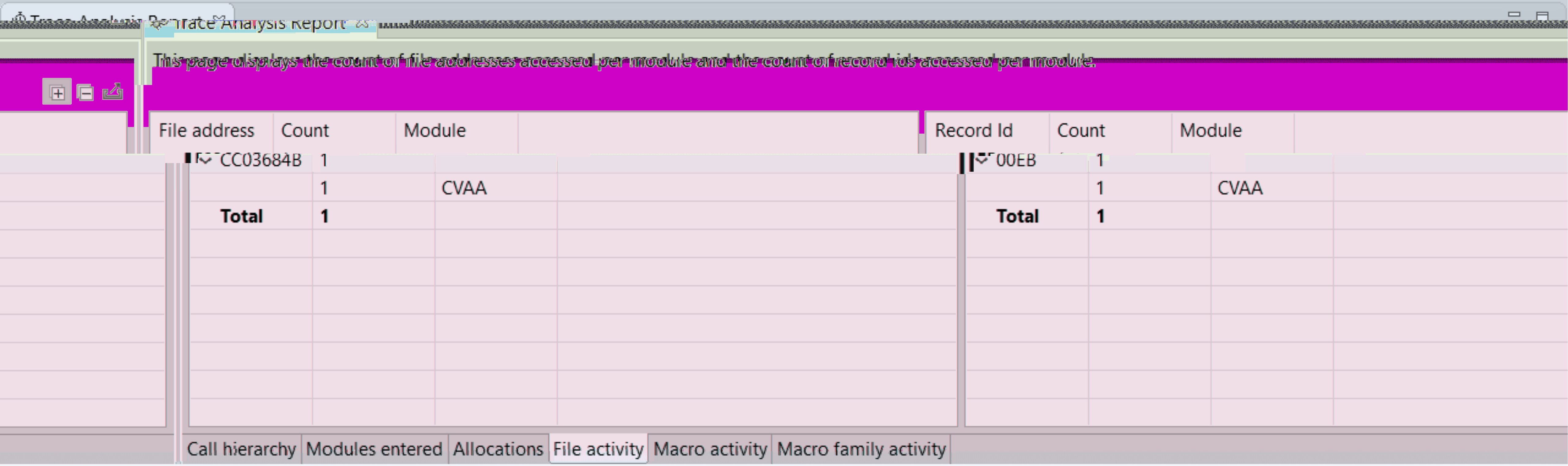


This page displays the count of memory allocation types per module and the size of the allocations per module

Module	MALOC	CALOC	RALOC	FREEC
-CP-	3			
CVAA	1			1
QDB0	2	79		78
Total	**6**	**79**	**0**	**79**

Allocation size	Count	Module
2	1	
	1	QDB0
8	1	
	1	QDB0
32	1	
	1	CVAA
43	3	
	3	-CP-
144	79	
	79	QDB0

File activity page

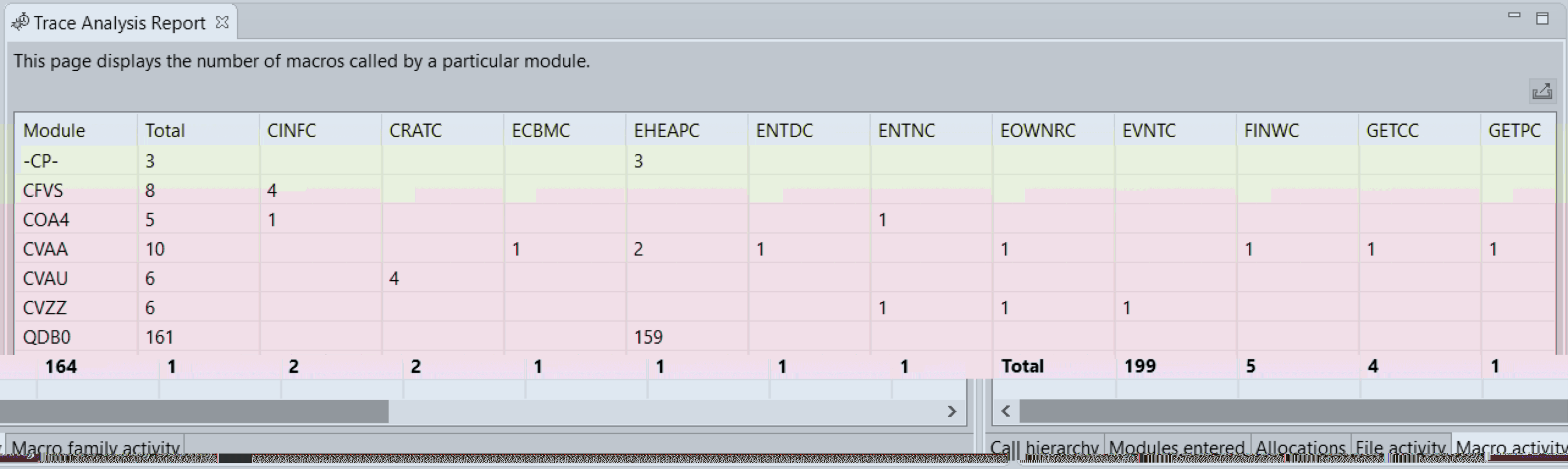


This page displays the count of file addresses accessed per module and the count of record ids accessed per module.

File address	Count	Module
CC03684B	1	
	1	CVAA
Total	**1**	

Record Id	Count	Module
00EB	1	
	1	CVAA
Total	**1**	

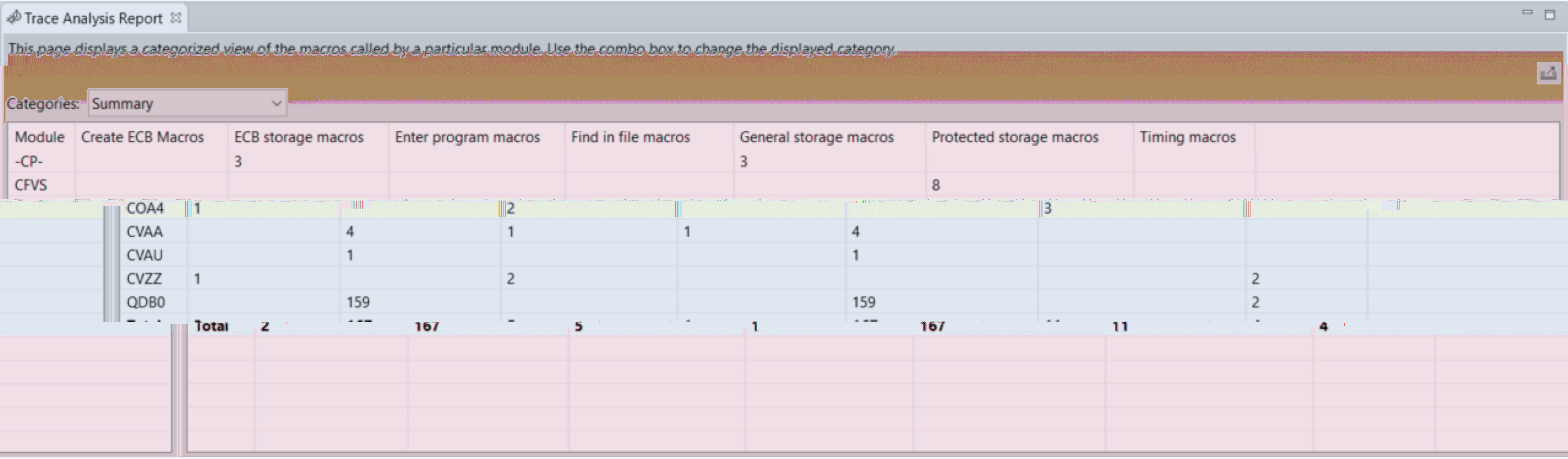
Macro activity page



This page displays the number of macros called by a particular module.

Module	Total	CINFC	CRATC	ECBMC	EHEAPC	ENTDC	ENTNC	EOWNRC	EVNTC	FINWC	GETCC	GETPC
-CP-	3				3							
CFVS	8	4										
COA4	5	1					1					
CVAA	10			1	2	1		1		1	1	1
CVAU	6		4									
CVZZ	6						1	1	1			
QDB0	161				159							
164	**1**	**2**	**2**	**1**	**1**	**1**	**1**	**Total**	**199**	**5**	**4**	**1**

Macro family page

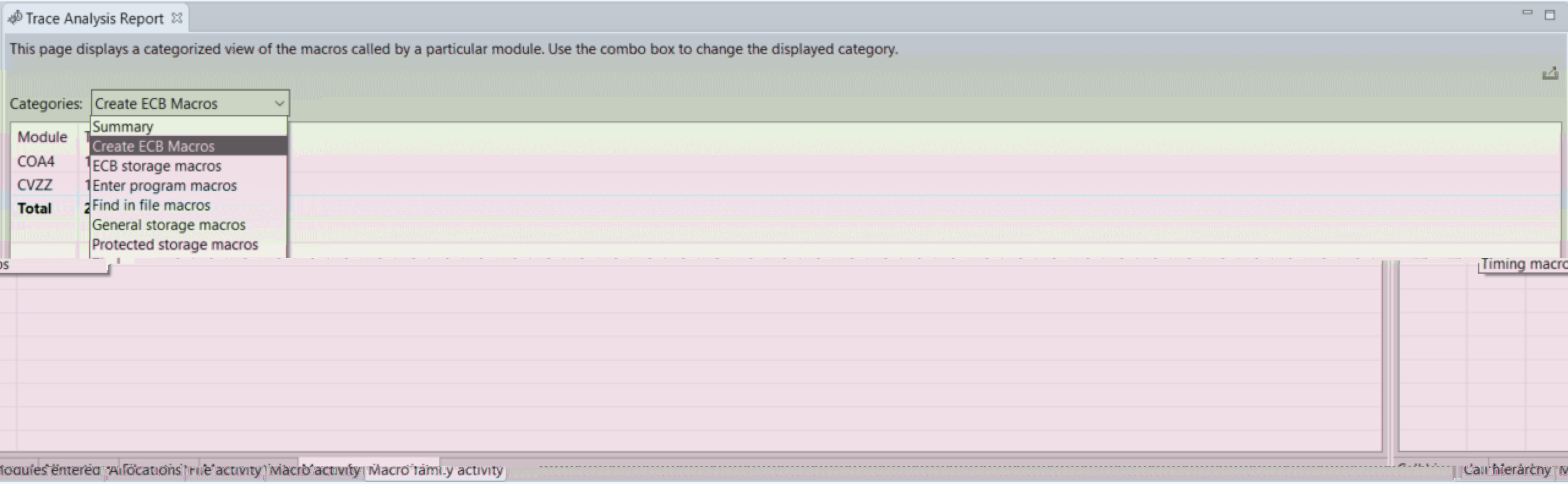


This page displays a categorized view of the macros called by a particular module. Use the combo box to change the displayed category.

Categories: Summary

Module	Create ECB Macros	ECB storage macros	Enter program macros	Find in file macros	General storage macros	Protected storage macros	Timing macros
-CP-		3			3		
CFVS						8	
COA4	1		2			3	
CVAA		4	1	1	4		
CVAU		1			1		
CVZZ	1		2				2
QDB0		159			159		2
Total	**2**	**167**	**5**	**1**	**167**	**11**	**4**

Macro family options



This page displays a categorized view of the macros called by a particular module. Use the combo box to change the displayed category.

Categories: Create ECB Macros

Module
COA4
CVZZ
Total

What’s New in Fix Pack 7

Available 2020-03-30

- **Trace log collection REST services and view**

APAR PJ46051 (March 2020) must be applied to the z/TPF system

tpf_ecb_trace.swagger.json documents the services interfaces.
tpf_ecb_trace.srvc.json must be deployed to discover services.
tpf_debugger.cfg.json must be updated with file system path for storage of collection files.

Collections can be created/stopped/deleted via the Trace Log Collection view in TPF Toolkit client.

Analysis report created and exported similar to the ECB Trace view.

What’s New in Fix Pack 7

- **AbstractCommandHandler class and samples**

Provides some orchestration of the existing Eclipse framework to better enable implement commands using the remote or local copies of files within a project from the Project Navigator view user interface.

Helpful for customizing the TPF Toolkit client for your workflow.

Sample code archive file available from FixCentral with the rest of the product files.

What’s New in Fix Pack 8

Available 2020-06-30

- **Loadset view**

PJ46043 must be applied to the z/TPF system to provide REST services for program management

The view displays a maximum of 45 loadsets that are loaded to the selected system.

User can activate, deactivate, delete and display additional details for each loadset in the view.

- **Loader input file editor**

Provides a new workflow to define the contents of a loadset as well as load and activate the loadset to a z/TPF system.

Browse and select programs to add to the @LOADSET control statement.

Browse and select files to add to the @FILE control statement.

Load and activate the loadset defined in the loader input file to a z/TPF system from the editor or the file’s context menu. (PJ46043 required).

What’s New in Fix Pack 8

- **Simplify OpenAPI editor**

Reorganizes the user experience for creating OpenAPI descriptors

Validates the file against the OpenAPI specification using the swagger.io Java libraries

Create Definition JSON objects from DFDL schemas

UI interface to add Path, Operation, Parameter, Response, Request and Definition objects to the OpenAPI descriptor.

- **Updates to AbstractCommandHandler API**

The AbstractCommandHandler class provided in Fix Pack 7 was updated with the following features:

- An option to write the command response line-by-line to the Console view.
- An option to specify a command response timeout value.
- Improved progress indication.

Trademarks

IBM, the IBM logo, ibm.com and Rational are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Notes

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput 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 improvements equivalent to the performance ratios stated here.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

This presentation and the claims outlined in it were reviewed for compliance with US law. Adaptations of these claims for use in other geographies must be reviewed by the local country counsel for compliance with local laws.

Disclaimer

Any reference to future plans are for planning purposes only. IBM reserves the right to change those plans at its discretion. Any reliance on such a disclosure is solely at your own risk. IBM makes no commitment to provide additional information in the future.