



Accelerating Delivery of Quality Systems with Eclipse-based Tools

Sebastien Marineau-Mes
QNX Software Systems

May 28, 2004

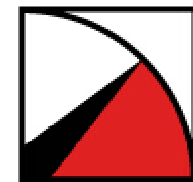
- **QNX/(KUE-nix): provider of realtime operating system (RTOS) software, development tools and services for mission-critical embedded applications**
 - > 24 years of realtime embedded experience
 - > Millions of installations worldwide
 - > Reputation for reliability, scalability, and innovation
- **QNX kernel chosen for new modular IOS and IOX**
 - > Microkernel architecture enables modularity
 - > Technology used for mission-and life-critical systems brings bullet-proof reliability to carrier-class networks
- **Commercial QNX Neutrino selected by multiple business units within Cisco**

- **Developer Environment & Challenges**
- **Eclipse Framework: An Overview**
- **The CDT Initiative**
- **Eclipse-based Tools**
 - > **Debugging**
 - > **Code Coverage**
 - > **System Analysis**
- **Case Study: Optimizing Performance in Cisco's uMG9850 Video-on-Demand (VOD) line card**
- **QNX Momentics for IOS Initiative**

- **Complex systems, large code base**
- **Aggressive delivery schedules and competitive market pressures**
- **Multiple development teams, often in remote locations**
 - > **Outsourcing**
 - > **Revision control**
 - > **Code integrity**
- **Poor tool integration or proprietary tool base**
 - > **Limited choice, reduced productivity**

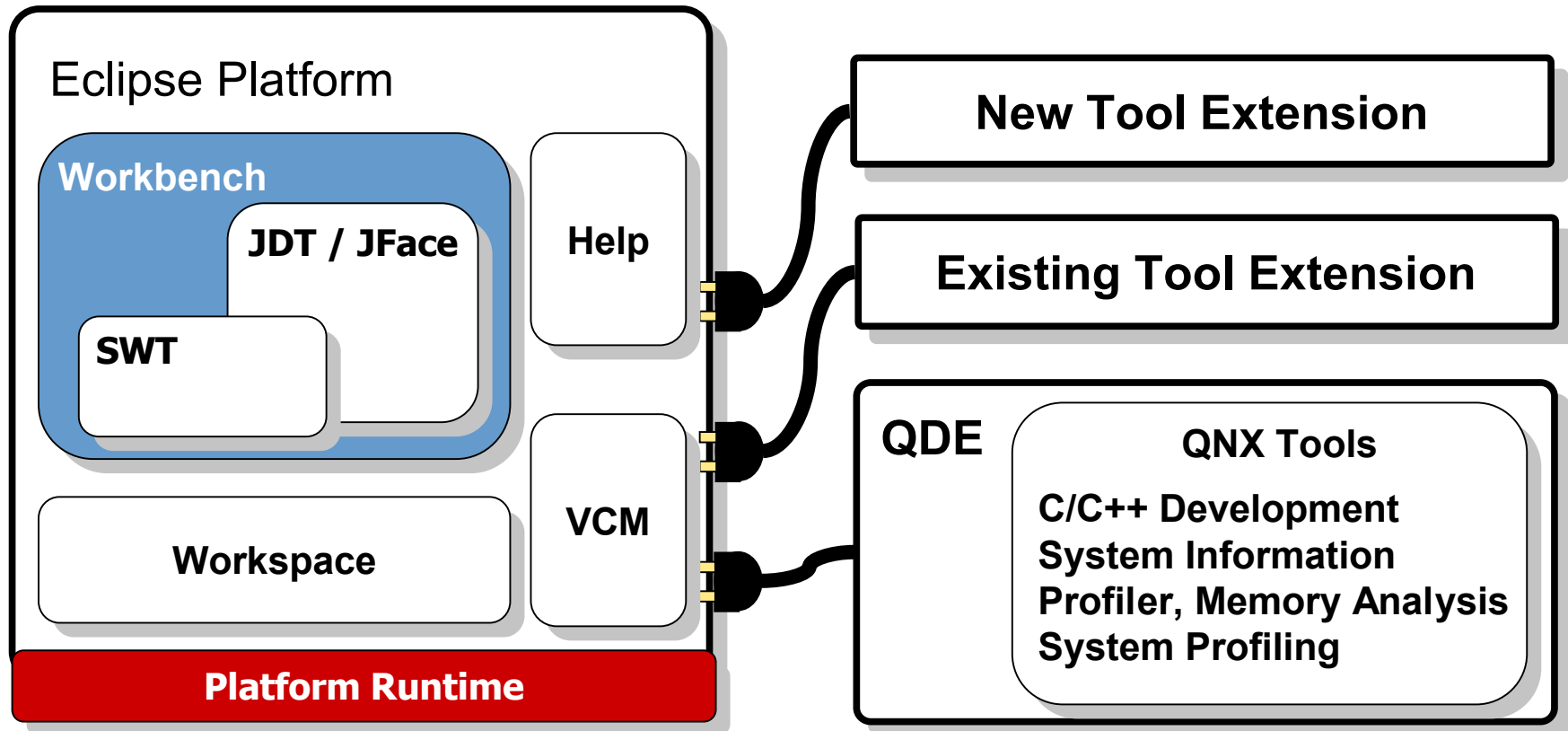
- ➔ **Open platform for tool integration**
- ➔ **Large – and rapidly growing – community of tool vendors and developers**
- ➔ **Well-defined interfaces to ensure tools work together seamlessly**
 - > all tools in the IDE share the same look-and-feel
- ➔ **Extensible plug-in architecture for virtually any type of content, such as:**
 - > navigate C/C++ source code
 - > analyze embedded flash and boot images
 - > monitor live runtime performance

Eclipse Community



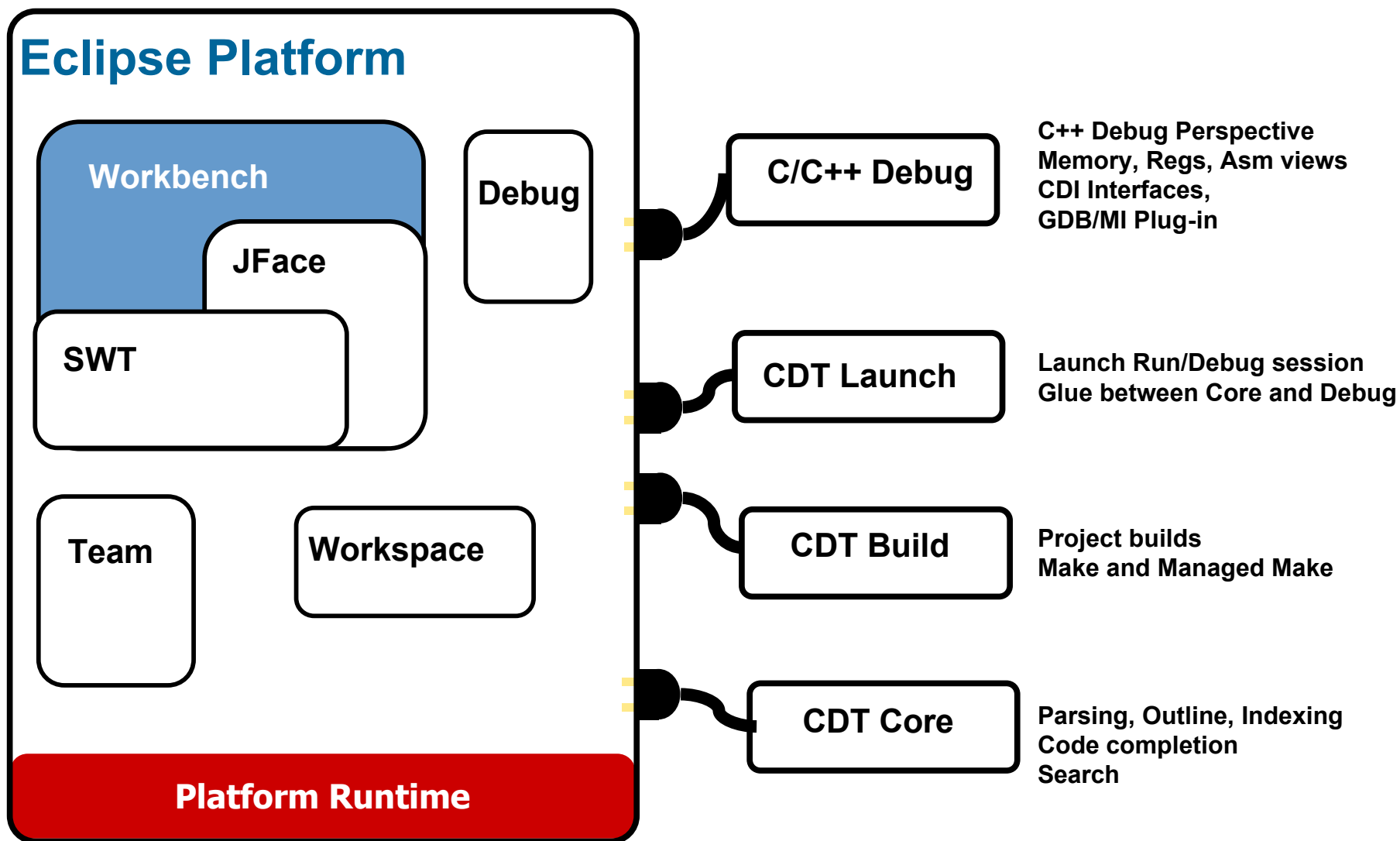
- **Core Platform**
- **Plug-In Development Tools**
- **Java Development Tools**
- **C/C++ Development Tools**
- **Cobol Development Tools**
- **Eclipse Modeling Framework**
- **Graphical Editor Framework**
- **Hyades Automated Testing Framework**
- **Visual Editor**
- **UML2**
- **Web Tools**
- **AspectJ Language Extensions**
- **AspectJ Development Tools**
- **ECESIS Educational Project**
- **Equinox Configuration and Platform Ubiquity**
- **Koi Collaboration Infrastructure**
- **GMT Generative Model Transformer**
- **Stellation Configuration Management**
- **WSVT Web Services Validation Tools**
- **XSD Infoset Model XML Schema**

Eclipse Plug-in Architecture



- **C/C++ Development Tooling (CDT) Eclipse project launched July 2002**
- **QNX CDT founding member and development lead**
 - > CDT 1.2 – Oct 2003
 - > CDT 2.0 – planned for June 2004 (sync with Eclipse 3.0)
- **Broad tool and system vendor participation**
 - > IBM/Rational, Red Hat, Intel, Tensilica, Timesys, Altera, TI...
- **Integration of components for C/C++ development**
 - > Design/Prototype and Development Tools
 - > Static Code Analysis Tools
 - > ICE and Hardware Debuggers
 - > Simulators and Optimizing Compilers

General CDT Architecture

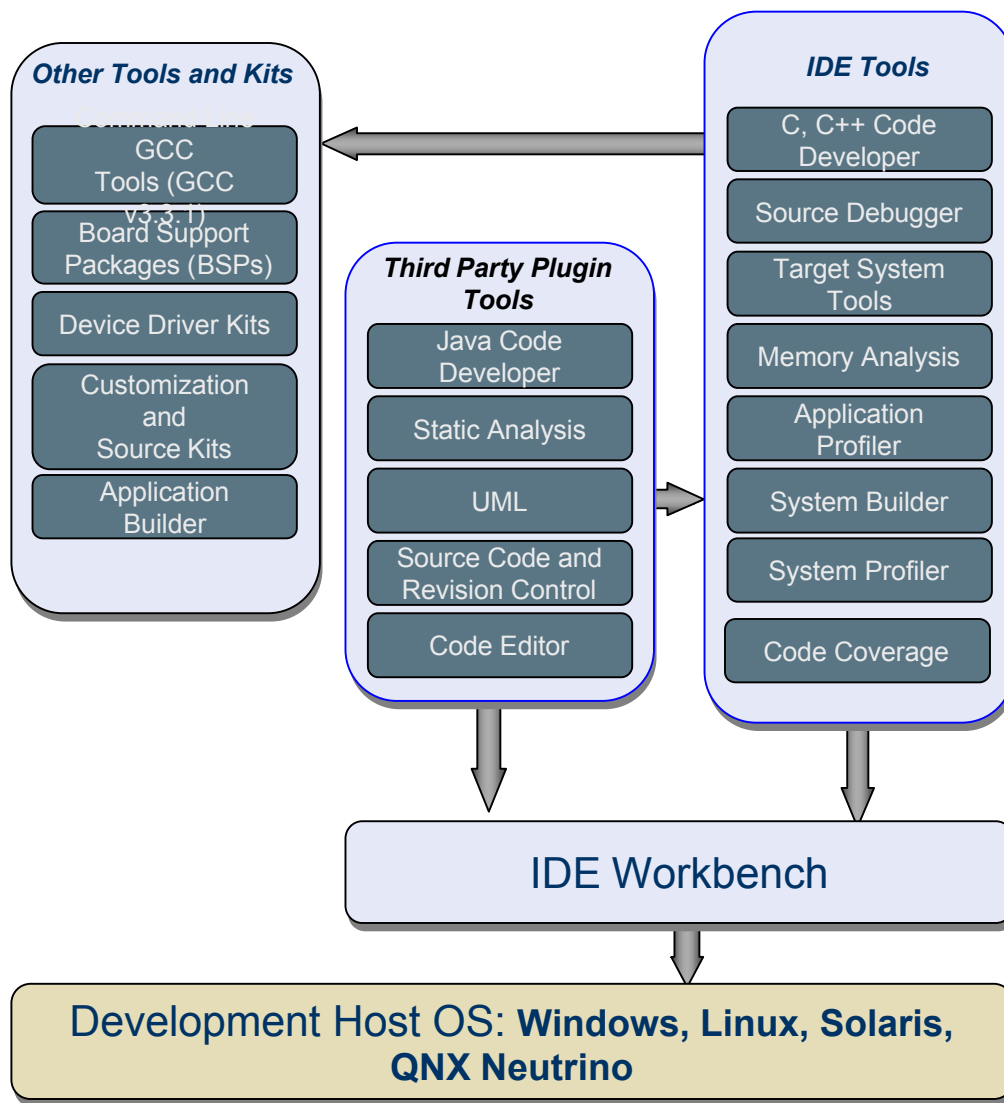




Eclipse-based Tools: QNX Momentics

Tight integration for higher productivity

QNX Momentics at a Glance



Choose your host

Windows, Linux, Solaris, QNX Neutrino

Choose your target

ARM, MIPS, PPC, SH4, XScale, x86

Choose your language

C, C++, Java

Choose your BSPs

BSPs for many popular boards and reference platforms

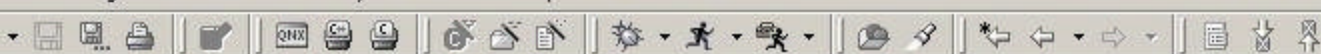
Choose command line or IDE

IDE and command line tools interoperate

Choose 3rd party tools

Growing ecosystem of tools vendors supporting Eclipse

- ➔ **Integrated revision control**
 - > **Local history of file modifications**
 - > **Built-in support for CVS**
 - > **Plug-in architecture for customized source control (e.g. Clearcase)**
- ➔ **Integrated documentation**
- ➔ **C/C++ editor features**
 - > **Syntax highlighting**
 - > **Auto code completion and code templates**
 - > **Hover help integration**
 - > **Code assist**



C/C++ Projects

- AQNXCPProject
 - Binaries
 - AQNXCPProject - [mipsle]
 - AQNXCPProject - [ppcbe]
 - AQNXCPProject - [x86le]
 - AQNXCPProject_g - [mipsle]
 - AQNXCPProject_g - [ppcbe]
 - AQNXCPProject_g - [x86le]
 - mips
 - o-le
 - o-le-g
 - AQNXCPProject_g - [mipsle]
 - Makefile
 - ppc
 - x86
 - AQNXCPProject.c
 - stdio.h
 - stdlib.h
 - directory_entry
 - name : char*
 - next : struct directory_entry*
 - get_directory_listing(char*, int) : void
 - main(int, char*[]) : int
 - directory_entry_t
 - Makefile
 - common.mk

```

Welcome AQNXCPProject.c
#include <stdlib.h>
#include <stdio.h>

typedef struct directory_entry {
    struct directory_entry *next;
    char *name;
} directory_entry_t;

void get_directory_listing(char *name, int len) {

}

int main(int argc, char *argv[]) {
    int i;
    int len;

    printf("Welcome to the QNX Momentics IDE\n");
    for (i = 0; i < argc; i++) {
        printf("Checking directory %s \n", argv[i]);
        len = strlen(argv[i]);
        get_directory_listing(argv[i], len);
    }
    return EXIT_SUCCESS;
}
    
```

Outline

- stdlib.h
- stdio.h
- directory_entry_t
- directory_entry
 - next : struct directory_entry*
 - name : char*
- get_directory_listing(char*, int) : void
- main(int, char*[]) : int

Make Targets

- AQNXCPProject
 - mips
 - ppc
 - x86
 - build
 - clean
 - rebuild

Tasks (4 items)

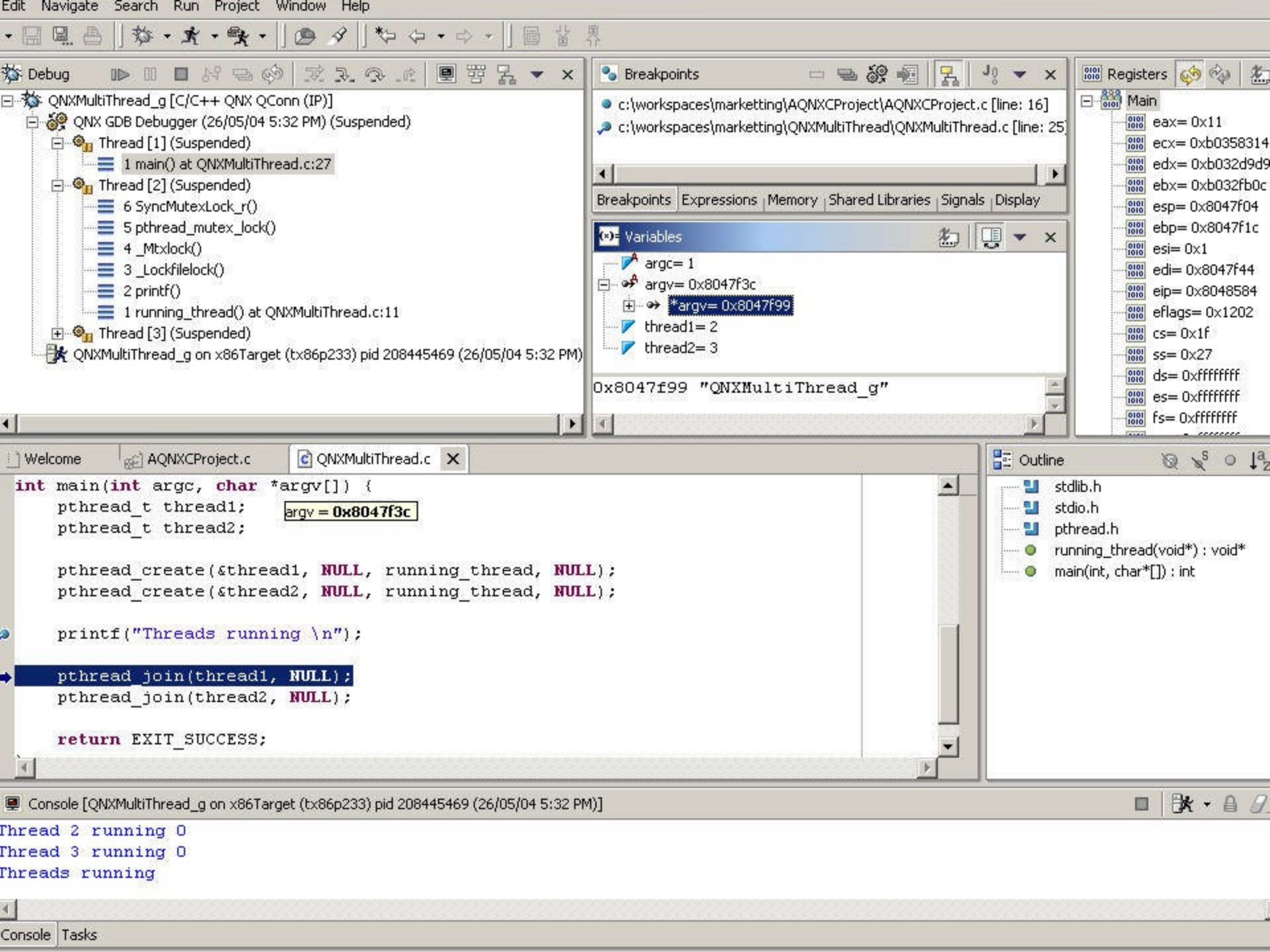
	✓	!	Description	Resource	In Folder	Location
✗			*** [AQNXCPProject.o] Error 1	AQNXCPProject		
✓			Finish implementation of get_directory_listing function	AQNXCPProject.c	AQNXCPProject	line 10
⚠			implicit declaration of function 'strlen'	AQNXCPProject.c	AQNXCPProject	line 19
✗			too few arguments to function 'get_directory_listing'	AQNXCPProject.c	AQNXCPProject	line 20

C-Build Properties Tasks

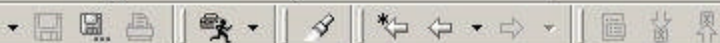
Bookmarks

Description	Resource	In Folder	Loc...
directory_entry structure definition	AQNXCPProject.c	AQNXCPProject	line 4

- ➔ **Multi-function integrated debugger**
 - > **Multi-language (C/C++/Embedded C++/Java)**
 - > **Multi-node/architecture**
 - > **Multi-process and multi-thread support**
- ➔ **Integrated with other IDE components**
- ➔ **Debugger features**
 - > **Context-sensitive editor integration**
 - > **Launch, attach or post-mortem (core file) support**
 - > **Stack-traces, variables, registers, breakpoints, ...**



- ➔ **System, Process and Thread information summary**
 - > Similar to ps/top/pidin unix utilities
- ➔ **Process resource information**
 - > Virtual memory map
 - > Memory (heap) allocator statistics and history
 - > File descriptors & CPU usage
- ➔ **Remote filesystem navigator**
- ➔ **Signals, blocking graphs, resource usage, ...**



System Summary



System Specifications

Hostname: tx86p233

Board: x86pc

OS Version: 6.3.0 (2004/04/29-21:23:19UTC)

Boot Date: Sun May 23 12:20:18 EDT 2004

CPU Details

x86 @ 199Mhz

System Memory

Used: 58M Free: 68M Total: 127M



Total Processes: 28

All Processes | Application Processes | Server Processes

Process Name	Heap Usage	CPU Usage	Start Time
pci-bios	32K	20ms	Sun May 23 12:20:18 EDT 2004
io-net	600K	15min 13sec	Sun May 23 12:20:18 EDT 2004
fs-nfs2	2M	2min 33sec	Sun May 23 12:20:20 EDT 2004
slogger	32K	43ms	Sun May 23 12:20:20 EDT 2004
devc-con	96K	20ms	Sun May 23 12:20:20 EDT 2004
devc-pty	464K	6min 38sec	Sun May 23 12:20:20 EDT 2004
pipe	48K	1sec 979ms	Sun May 23 12:20:20 EDT 2004
devc-ser8250	48K	14ms	Sun May 23 12:20:21 EDT 2004

System Summary | Thread Information | Malloc Information

Process Information



Arguments

qconn

Thread Details

Thread	Priority Name	State	Blocked on	Stack
1	10r	State Reply	procnto-instr	8K/516K
2	15r	State Receive	Channel 1	4K/132K
3	10r	State Condvar	Condvar #1345956...	4K/132K
4	10r	State Receive	Channel 4	4K/132K

Environment Variables

Identification Details

Real User ID: 0

Real Group ID: 0

Effective User ID: 0

Effective Group ID: 0

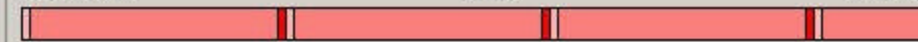
Memory of qconn



0 Process Map 0xFFFF



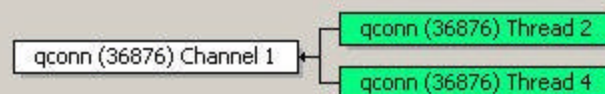
0x07F64000 Stack 0x0804...



Name	V. Addr.	Size
Stack		20K allocated 8...
Program		80K code 28K c...
qconn code/text	0x08048000	80K
qconn data/bss	0x0805c000	28K
Heap		384K allocated
heap	0x08063000	256K
heap	0x080ab000	96K
heap	0x080e3000	32K
Objects		0 mapped
Library		460K code 44K
ldgnx.so.2 code/text	0xb0300000	344K
ldgnx.so.2 data/bss	0xb0356000	16K
libsocket.so.2 code/text	0xb8200000	116K
libsocket.so.2 data/bss	0xb821d000	28K

System Blocking Graph

Servicing request Waiting for request Waiting for reply Waiting for service



Object	Object Owners	State	Blocked Folks	State
qconn (3687...	qconn (36876) Thread 2	State ...		
	qconn (36876) Thread 4	State ...		
random (368...	random (36878) Thread 2	State ...		
fs-cifs (8884...	fs-cifs (8884242) Thread 1	State ...		
	fs-cifs (8884242) Thread 2	State ...		

- **Identifies exercised source code paths**
 - > **Allows developers to target code areas for performance analysis and debugging**
- **Applications span QA, optimization, bug fixing, maintenance**
 - > **QA methodology in military, automotive and medical**
 - > **Optimizing tool**
 - > **Useful for bug fixing and maintenance groups not involved with original code development**
- **Fully integrated with other IDE components**

Code Coverage Sessions

- sever_demo [GCC Code Coverage] (26/05/04 5:44 PM)
 - sever_demo [32.56%]
 - server.c [32.56%]
 - handle_read [0%]
 - main [75.61%]

Code Coverage Sessions Navigator

Debug

Disassembly Mode On/Off

- sever_demo [C/C++ QNX QConn (IP)]
- sever_demo [GCC Code Coverage] (Running - idle)
- sever_demo_g on x86Target (tx86p233) pid 21543324

Property	Value
Coverage Info	
Lines Fully Covered	45.31% (29 lines)
Lines Not Covered	54.69% (35 lines)
Lines Partially Covered	0% (0 lines)
Total Coverage	32.56%
Info	
derived	false
editable	true
last modified	21/05/03 2:15 PM
linked	false
location	c:\workspaces\marketing\sever_de
name	server.c
path	/sever_demo/server.c
size	4586

```
server.c
return (_RESMGR_NPARTS (nparts));
}

int main(int argc, char **argv) {
    resmgr_attr_t      resmgr_attr;
    thread_pool_attr_t pool_attr;
    thread_pool_t      *tpp;
    char               *mountpoint, *filename;
    char               *defaultmount = "/dev/testdev";
    int                ret;

    filename = NULL;
    mountpoint = defaultmount;

    /* Create a dispatch context structure used by all functions below */
    if((g_dpp = dispatch_create()) == NULL) {
        fprintf(stderr, "%s: Unable to allocate dispatch context.\n", argv[0]);
        return EXIT_FAILURE;
    }
}
```

Code Coverage Report

Address: <http://127.0.0.1:1886/qnx-coverage/coverage?session=gcc1085607886954>

Report generated on: 26/05/04 5:45 PM

Session name: sever_demo [GCC Code Coverage]
Time of session creation: 26/05/04 5:44 PM
Session identifier: gcc1085607886954

Coverage					
Project	Source Line Coverage				
	Total Code Coverage	Lines Not Covered	Lines Partially Covered	Lines Fully Covered	Total Lines
sever_demo	<div><div></div></div> 32.56%	35	0	29	64

→ Memory Analysis

- > Memory allocation/deallocation tracking
- > Range checking with automatic debugger connectivity
- > Memory error detection (leak detection, overruns/underruns, double frees, etc)

→ Process Profiling

- > Statistical profiling of any active process
- > Instrumented compile provides call pair information
- > Graphical source code annotation



Target Navigator

- x86Target (tx86p233)
 - procnto-instr (1)
 - pci-bios (2)
 - io-net (3)
 - fs-nfs2 (16388)
 - slogger (32773)
 - devc-con (32774)
 - devc-pty (32775)
 - pipe (32776)
 - devc-ser8250 (32777)
 - fs-nfs2 (36874)
 - mqueue (40971)
 - qconn (36876)
 - inetd (36877)

memory_demo.c

```
int BufferOverflow()
{
    char *buffer = malloc( 64 );
    memset( buffer, 0, 66 );
    free( buffer );
    // Malloc region doesn't have a valid CRC in header
}

int BufferUnderrun()
{
    char *buffer = malloc( 64 );
    char *ptr = buffer - 1;
    ptr[0] = 0;
    free( buffer );
}
```

Memory Events

Error events tree

- tx86p233
 - memory_demo_g(216485904)
 - Data are is not in use (can't be freed or reallocated)
 - Data in free'd area has been modified
 - Pointer is not within malloc area
 - Pointer within malloc region, but outside of malloc data bounds
 - BufferOverflow [memory_demo.c:38]
 - Allocation Trace Call #19
 - Data has overrun beyond requested number of bytes

Event backtrace

```
Caught in memset
Memory allocated by malloc in c:\workspaces\marketing\memory_demo\memory_demo.c:37
64 bytes allocated on 18th call to malloc
0x08048484
main [memory_demo.c:73]
BufferOverflow [memory_demo.c:38]
errno (libmalloc.so.2) [memory_demo.c]
errno (libmalloc.so.2) [memory_demo.c]
errno (libmalloc.so.2) [memory_demo.c]
errno (libmalloc.so.2) [memory_demo.c]
```

Allocation Trace

Range	Call type	Call #	Pointer	Length	Calling address	Match call #	Match call address
	malloc	13	0x00000...	0	DoubleFree [memory_demo.c:10]	14	DoubleFree [memory_demo.c:12]
	malloc	14	0x00000...	0	DoubleFree [memory_demo.c:12]	13	DoubleFree [memory_demo.c:10]
	malloc	15	0x00000...	0	DoubleFree [memory_demo.c:13]		
	malloc	16	0x00000...	0	ReferenceAfterFree [memory_demo.c:20]	17	ReferenceAfterFree [memory_demo.c:21]
	malloc	17	0x00000...	0	ReferenceAfterFree [memory_demo.c:21]	16	ReferenceAfterFree [memory_demo.c:20]
	malloc	18	0x00000...	0	FreeOnStack [memory_demo.c:30]		
	malloc	19	0x00000...	0	BufferOverflow [memory_demo.c:37]	20	BufferOverflow [memory_demo.c:39]

Application Profiler

<terminated>profile_demo [C/C++ QNX QConn (IP)]

- <terminated>QNX Application Profiler
 - profile_demo_g
 - ldqnx.so.2
 - Unknown

<terminated>profile_demo_g on x86Target (tx86p233) pid 216649744 (26/05/04 6:00 PM)

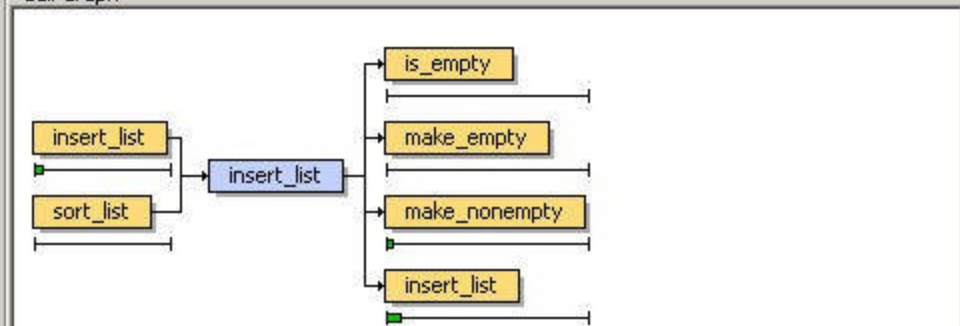
Application Profiler | Debug | Navigator | Console

Call Information

Call Pairs

Function	Called From	Call Pair Count
is_empty	insert_list	12474155
insert_list	insert_list	12463904
make_nonempty	insert_list	12002001
is_empty	append_lists	24008
make_nonempty	append_lists	24000
append_lists	append_lists	24000
is_empty	print_list	19565

Call Graph



Call Pair Details

Caller	Call Count	Called	Called Count
insert_list	12463904	is_empty	12474155
sort_list	10251	make_empty	2
		make_nonempty	12002001
		insert_list	12463904

Sampling Information

Function	Total Tim...	Total C...	Tim...	Calls si...	usec/Call	% Time Usage
insert_list	2.141	12002000	2.141	12002000	0.178	
make_nonempty	1.039	12022000	1.039	12022000	0.086	
pthread_self	0.671		0.671			
is_empty	0.405	12038007	0.405	12038007	0.034	
free	0.118		0.118			
print_list	0.043	16002	0.043	16002	2.687	
malloc	0.021		0.021			
append_lists	0.011	12004	0.011	12004	0.916	
sort_list	0.006	8001	0.006	8001	0.750	
printf	0.004		0.004			
make_list_helper	0.003	8000	0.003	8000	0.375	
sort_data	0.000	1	0.000	1		
make_empty	0.000	6	0.000	6		
get_data	0.000	1	0.000	1		

Thread Processor Usage | Sampling Information

profile_demo.c (QNX Application Profiler)

```

list->rest = rest;

return list;
}

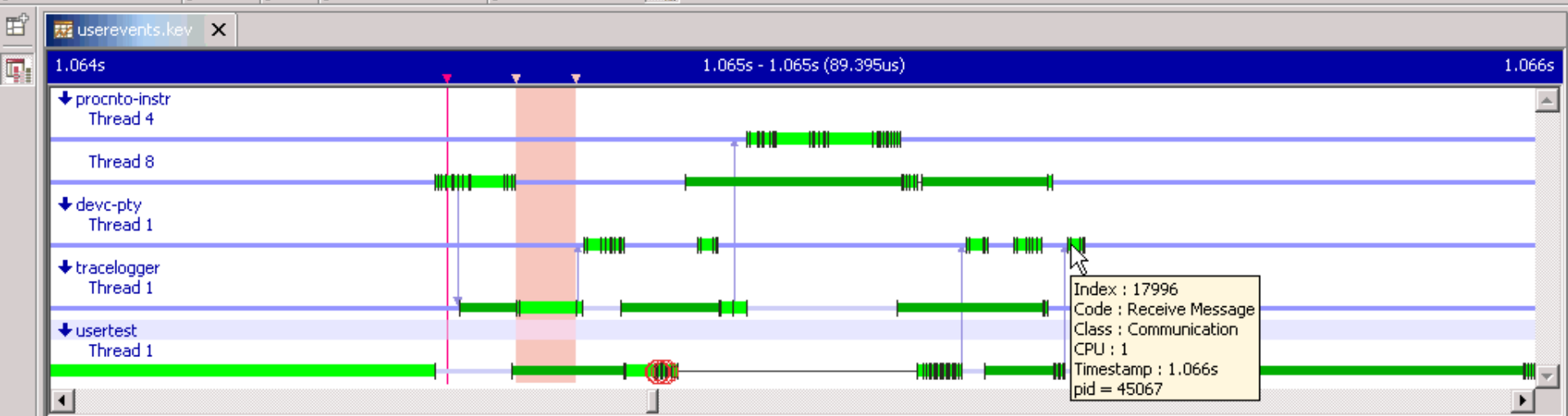
int is_empty(List list)
{
    return (list == NULL);
}

List make_list_helper(unsigned int size, List list)
{
    if (size == 1)
        return make_nonempty(size, accum);
    else
        return make_list_helper(size-1, make_nonempty(size, accum));
}

List make_list(unsigned int size)
{
    return make_list_helper(size, make_empty());
}

```

- **System level profiling and event overview**
- **IDE provides remote control of instrumented kernel**
- **Dynamic adjustable filtering**
 - > **Data depth, Event classes, User handlers**
- **Configurable instrumentation display**
 - > **Interrupt, process and thread state timelines**
 - > **Textual log view of raw event contents**
- **Full context event search, bookmarking and annotation**



Trace Event Log: userevents.kev

Event	Time	Owner	Type	Data
17785	1s 65ms 343us	procnto-instr Thr...	← _KER_RING0 Exit	ret_val 0x00000000
17786	1s 65ms 347us	procnto-instr Thr...	→ MsgReplyv Enter	status 0x00000000 rcvid 0x00000027
17787	1s 65ms 352us	usertest Thread 1	→ State Ready	pid 917541 tid 1
17788	1s 65ms 353us	procnto-instr Thr...	← MsgReplyv Exit	ret_val 0
17789	1s 65ms 356us	procnto-instr Thr...	→ MsgReceivev Enter	chid 0x00000001 rparts 2080
17790	1s 65ms 358us	procnto-instr Thr...	→ State Receive	pid 1 tid 8
17791	1s 65ms 359us	tracelogger Thre...	→ State Running	pid 913444 tid 1

Search ("UserEvents" - 4...ences in userevents.kev)

- Ev #17815 usertest Thread 1 Event Code 0x0000000C
- Ev #17818 usertest Thread 1 Event Code 0x0000000D
- Ev #17821 usertest Thread 1 Event Code 0x0000000E
- Ev #17824 usertest Thread 1 Event Code 0x0000000F

General Statistics: userevents.kev

States						Events	
Event	Calls	Avg Duration	Max Duration	Min Duration	Total Duration	Event	Calls
State Running	3702	712us	49ms 956us	3us	2s 635ms 257us	Time	1
State Ready	2559	150us	34ms 596us	0us	384ms 934us	MsgWritev Enter	1
State Receive	1717	30ms 773us	2s 151ms 130us	6us	50s 930ms 547us	SyncMutexUnlock Enter	1
State Reply	1484	5ms 459us	1s 261ms 258us	11us	8s 68ms 755us	Event Code 0x0000000C	1
State Send	258	28us	256us	20us	7ms 294us	Event Code 0x0000000D	1
State SigWaitInfo	41	76ms 821us	679ms 614us	827us	2s 381ms 460us	Event Code 0x0000000E	1
State NanoSleep	11	581ms 879us	1s 19ms 510us	121ms 899us	4s 73ms 154us	Event Code 0x0000000F	1
State WaitThread	8	6us	8us	4us	52us	← _KER_SYS_CPUPAGE_SET Enter	2
State Dead	6	280us	657us	35us	1ms 681us		

☒ Show statistics for all elements.



Case Study

Cisco's uMG9850 Video-on-Demand (VOD) line card

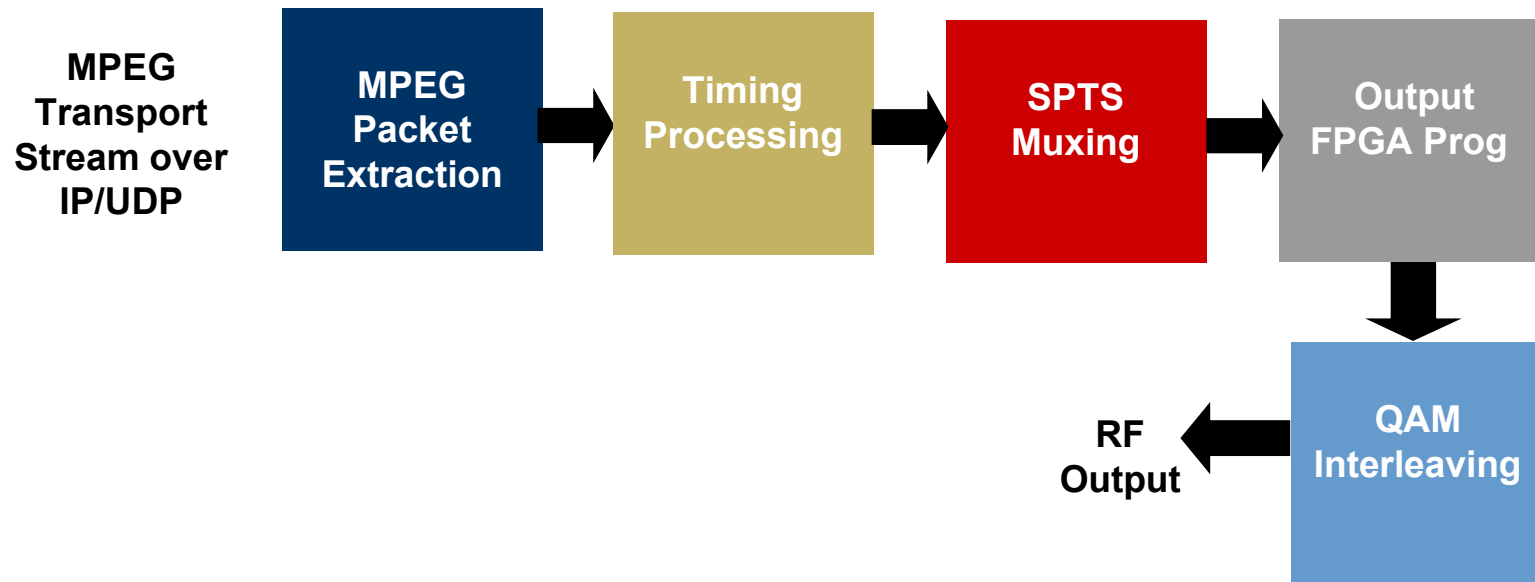
Description

- **uMG9850 QAM Module for next-generation digital video networks**
- **Can process full capacity of one Gigabit Ethernet link, which offers up to 240 standard-definition video streams – without daisy chaining**

Assumptions

- **Data-centric video processing requires precise timing and optimal usage of CPU resources**

Video Packet Flow and Processing

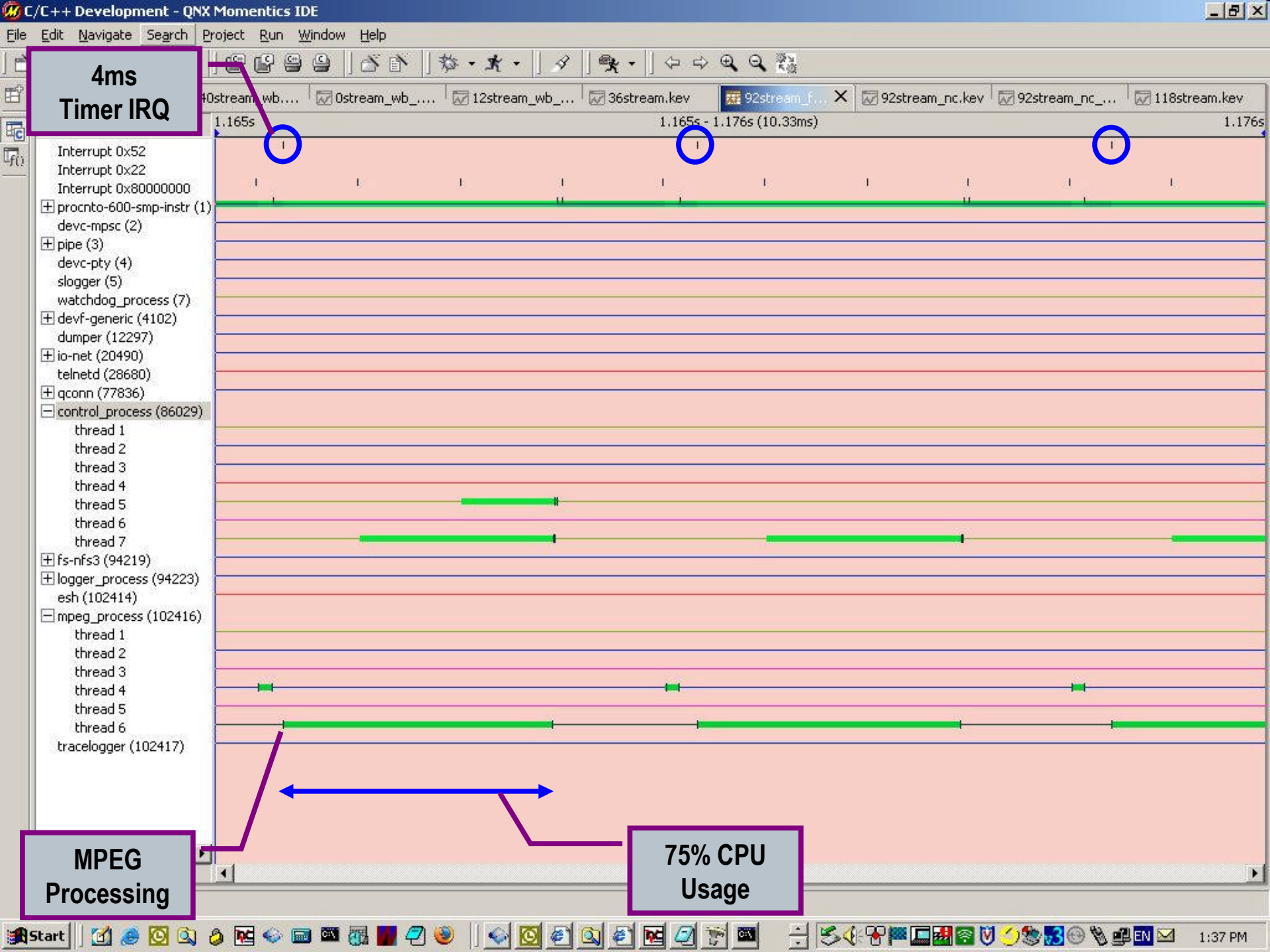


Problem:

- **Application could only process 92 MPEG video transport streams**
 - > **Hitting more than 50 percent of timing budget (4 ms window)**
- **Design required to process 240 streams**
 - > **Full 1G per second worth of bandwidth/data**
 - > **Cannot use more than 50% CPU for MPEG processing**

Approach:

- **Use system profiling to isolate which functions are being called most often and optimize where possible**
 - > **System runs on QNX Neutrino RTOS**
 - > **Began profiling eight weeks prior to testing**





Profiler

<terminated> Profile new [QNX Profiler Gmon Launcher]

<profiler-terminated> Profiler Attached to: L:/new/gmon_outs/92stream_nc_nochktp_wbout_mpeg_process<gmon file>

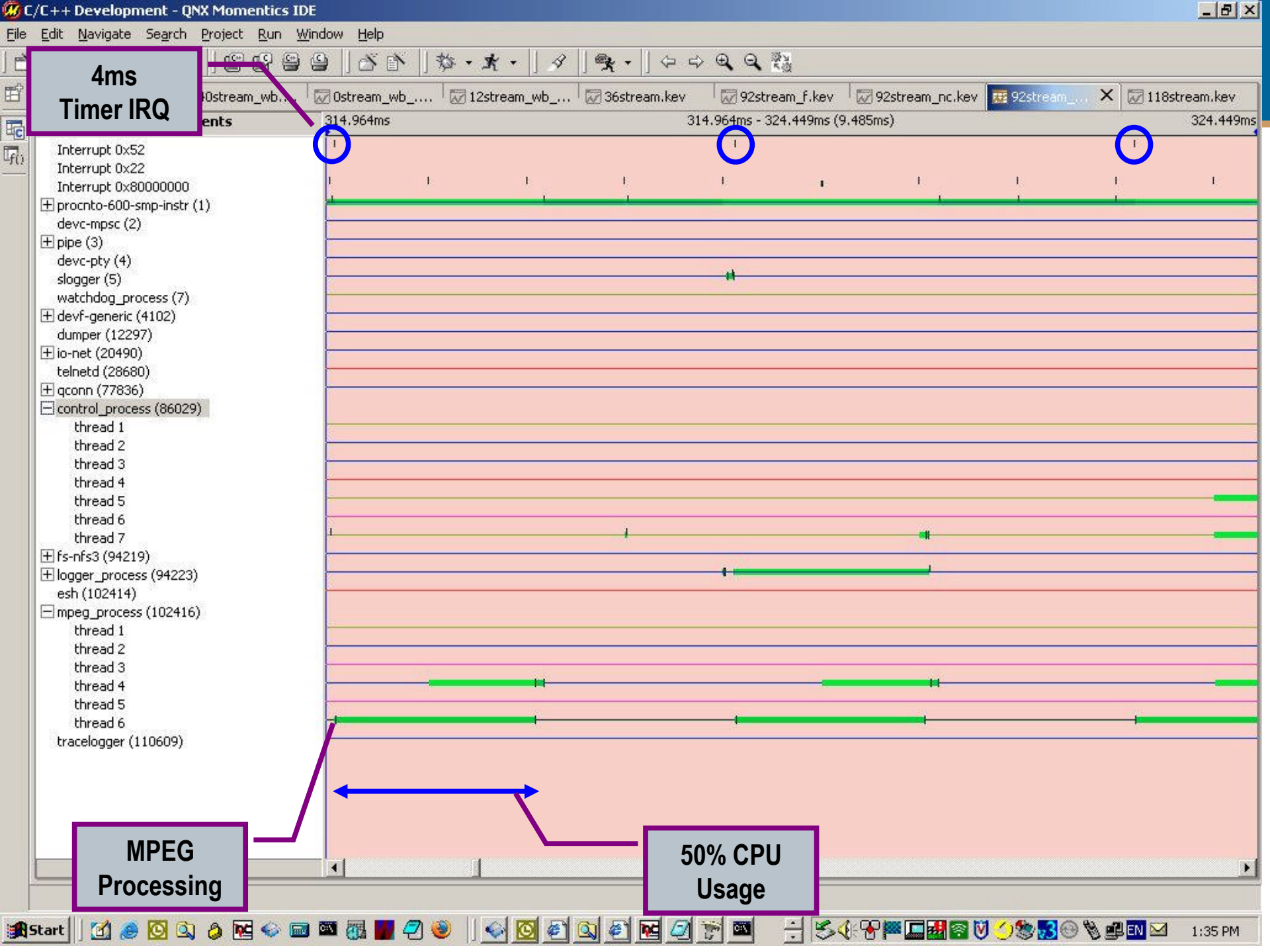
92stream_nc_nochktp_wbout_mpeg_process

Functions to optimize

Sampling Information

Function	Total Time...	Time since...	Call Count	usec/Call	% Time Usage
mux_schd_mpeg_tp	8.306	8.306	177872	46.697	
process_mpeg_tp	5.467	5.467	1971664	2.773	
clkrec_dejitter	2.633	2.633	12707216	0.207	
clkrec_check_buffer_violation	1.496	1.496	12707216	0.118	
mux_fill_null_pkt	1.426	1.426	177920	8.015	
process_ip_udp	1.299	1.299	1990180	0.653	
clkrec_non_pcr_tp	1.208	1.208	12373098	0.098	
mux_init_schd_table	0.861	0.861	177920	4.839	
input_module	0.825	0.825	15001	54.996	
psi_update_session	0.562	0.562	9187200	0.061	
mux_schd_crsi_buf	0.488	0.488	355744	1.372	
clkrec_pcr_tp	0.383	0.383	334118	1.146	
mux_cli_chck_prog_cont	0.099	0.099	177872	0.557	
mux_cli_chck_prog	0.096	0.096	177872	0.540	
mux_schd	0.037	0.037	15001	2.467	
pb_get_port	0.036	0.036			
mux_calc_schd_wndw_time	0.035	0.035	177920	0.197	
mux_scan_crsi_del_info_head	0.024	0.024	15001	1.600	
mux_cnfg_msg_handler	0.023	0.023	15001	1.533	
mux_calc_schd_slot_numb	0.022	0.022	177920	0.124	
mux_get_hw_ltch_time	0.022	0.022	22240	0.989	
mux_check_mv_idma_dscp	0.020	0.020	15001	1.333	
psi_snoop_loop	0.019	0.019	15312	1.241	
clkrec_check_monitor	0.019	0.019	334118	0.057	
mux_cmnd_mark_ltch_time	0.017	0.017	22248	0.764	
check_session_stop	0.016	0.016	14712	1.088	
get_time_diff	0.013	0.013	30002	0.433	
mpeg_thread_function	0.012	0.012			
mux_cli_chck_bw	0.011	0.011	177920	0.062	
out_hw_cnfg_handler	0.008	0.008	15001	0.533	
msgh_get_message	0.008	0.008			
psi_snoop_thread_func	0.007	0.007			

Sampling Information Call Information



4ms
Timer IRQ

MPEG
Processing

50% CPU
Usage

- Visualized where problem was occurring and used information to optimize system cache
- Reached 240 MPEG streams throughput
- Application deployed on time
 - > Were able to meet an aggressive schedule
- Now using profiling to increase processing capacity for next release of product

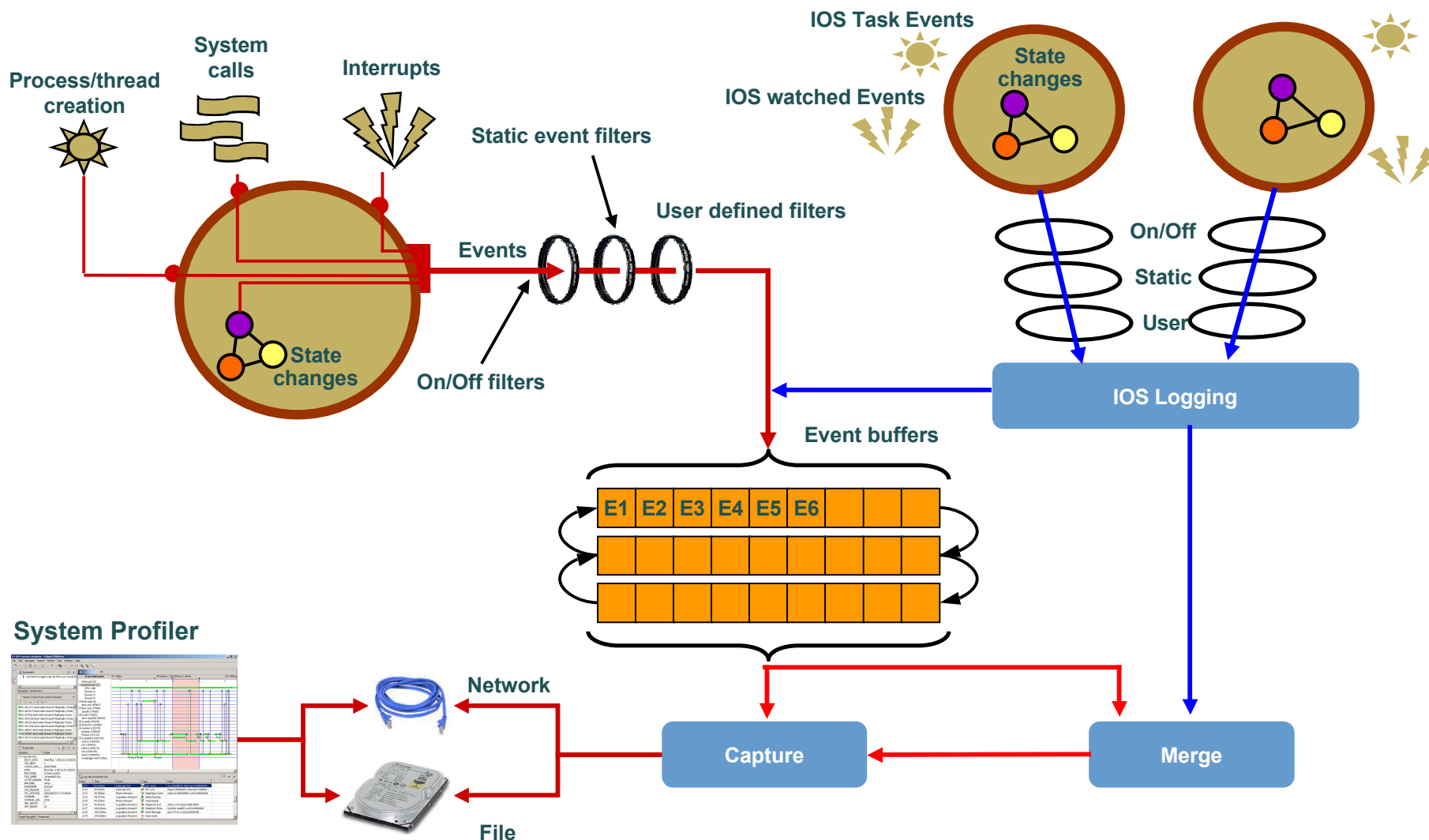


System Profiling for IOS

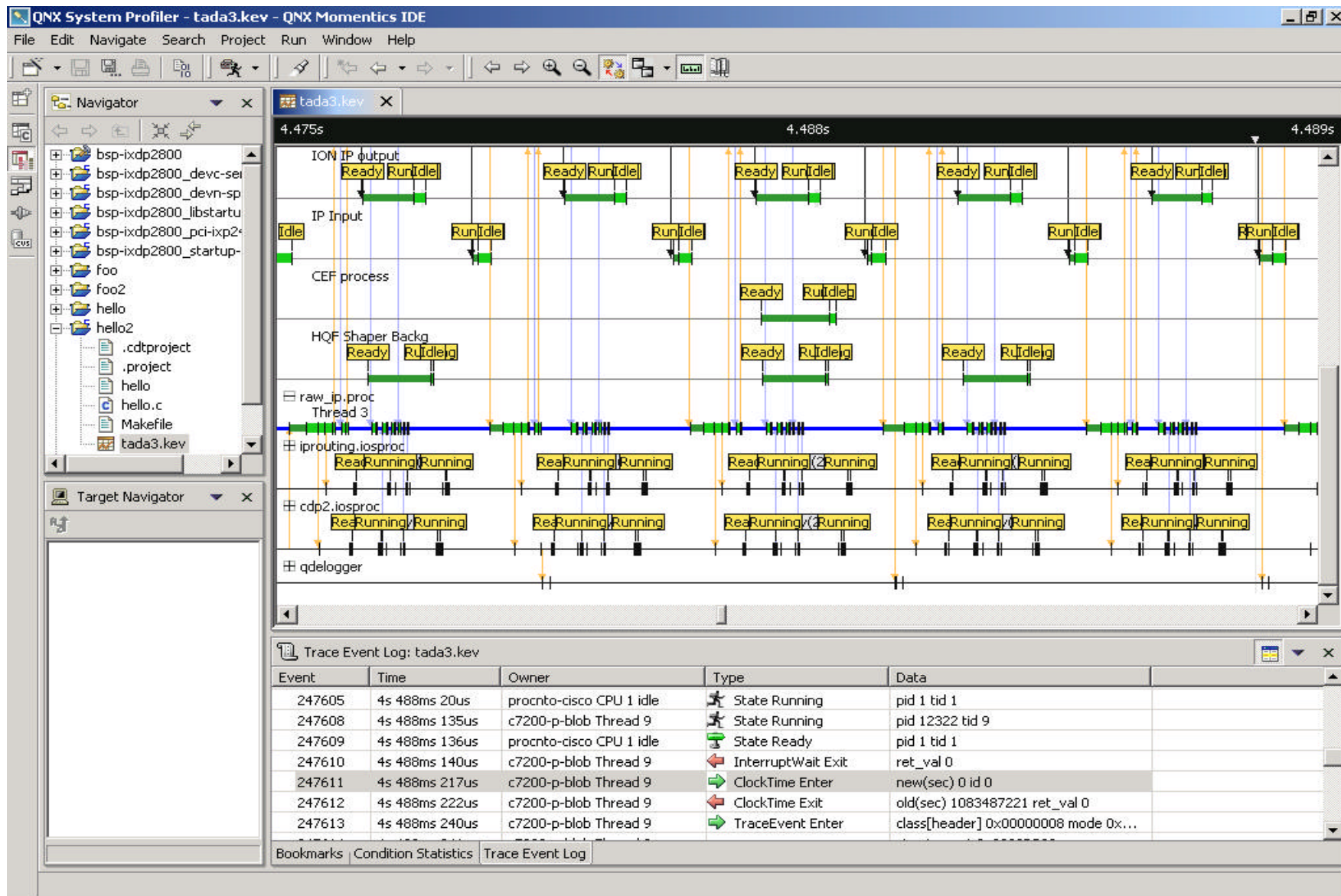
- ➔ **Use tracing extensively to find performance and scalability problems**
- ➔ **Home-grown tools**
 - > **XRay, ELog, etc.**
 - > **Most of these tools are good in tracing code paths and finding low level performance problems**
 - > **Require special image build**
 - > **Limited GUI support**

- **High-level system view to show various activity in the box**
 - > State of various processes, queues, etc., in system
 - > Process state transitions
 - > Filters to collect user-specified events
- **Graphical User Interface**
 - > Multiple views to look at the data
 - > Filters to display user specified events only
- **Cost**
 - > Low when enabled
 - > Ideally, none when not enabled (TBD)
- **Extensible**
 - > User defined events can be added

Instrumented Kernel – IOS

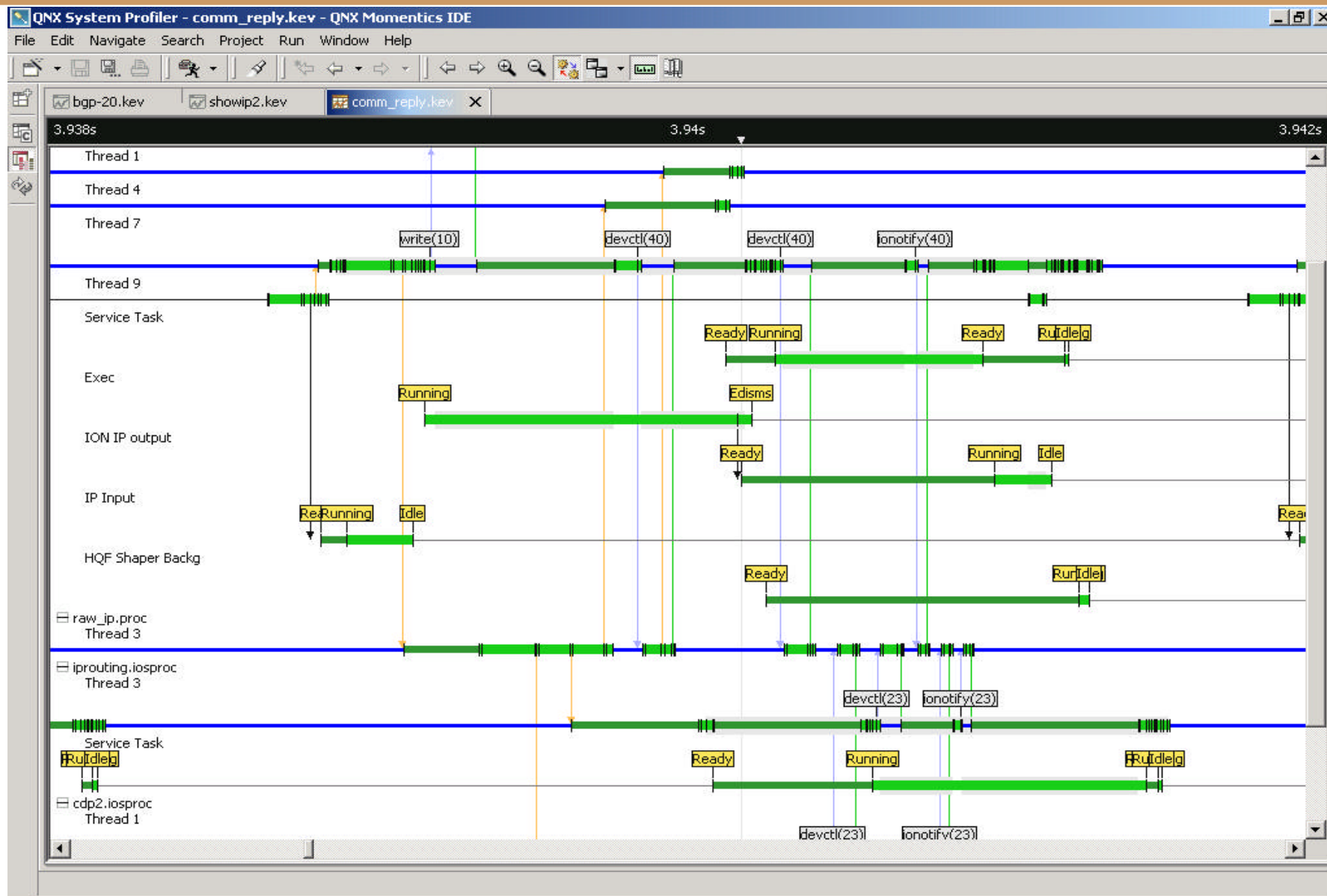


IOS Trace



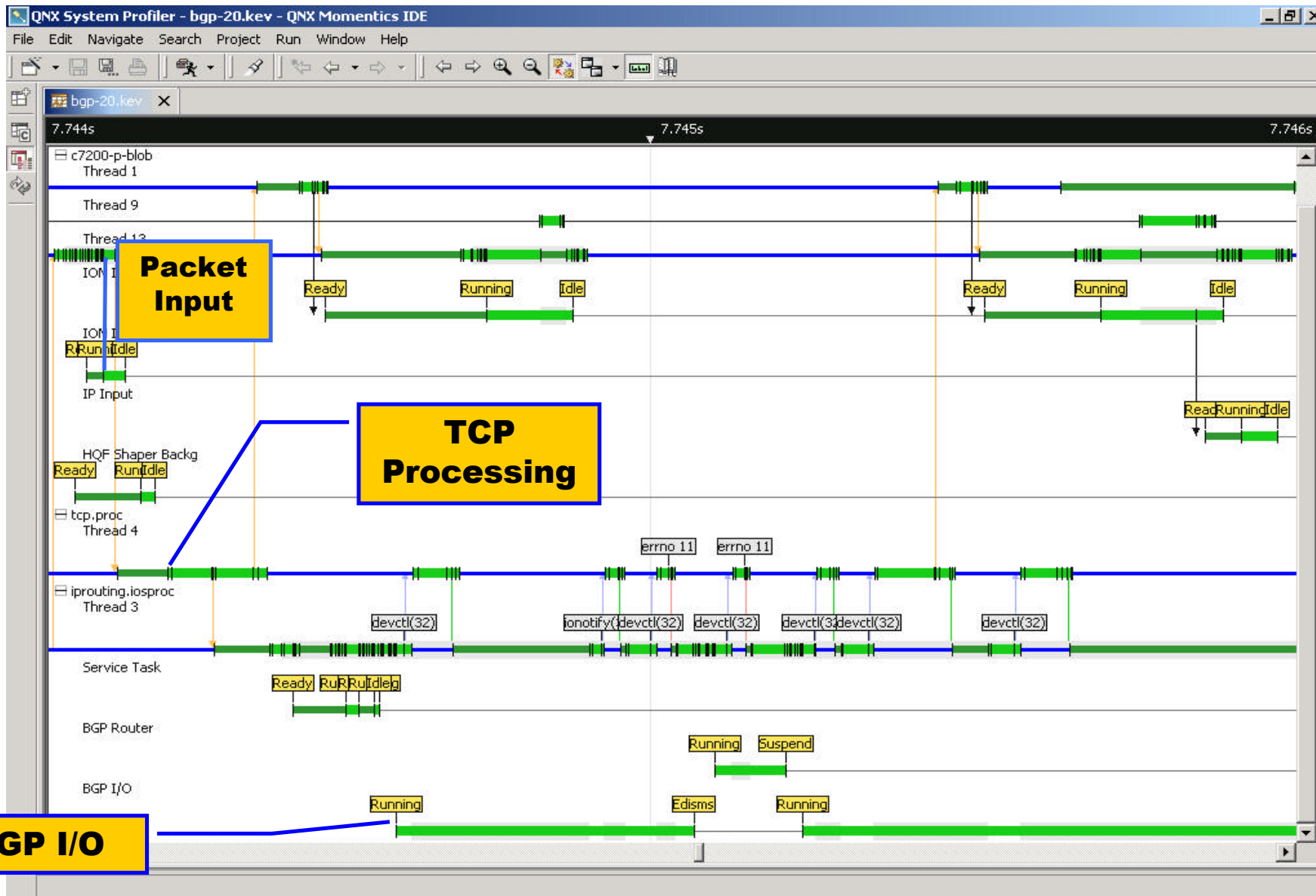
- **NPE-400 router running ION with instrumentation**
- **Trace captured during test**
- **Flood ping originated by router**

Tracing a Ping Cycle

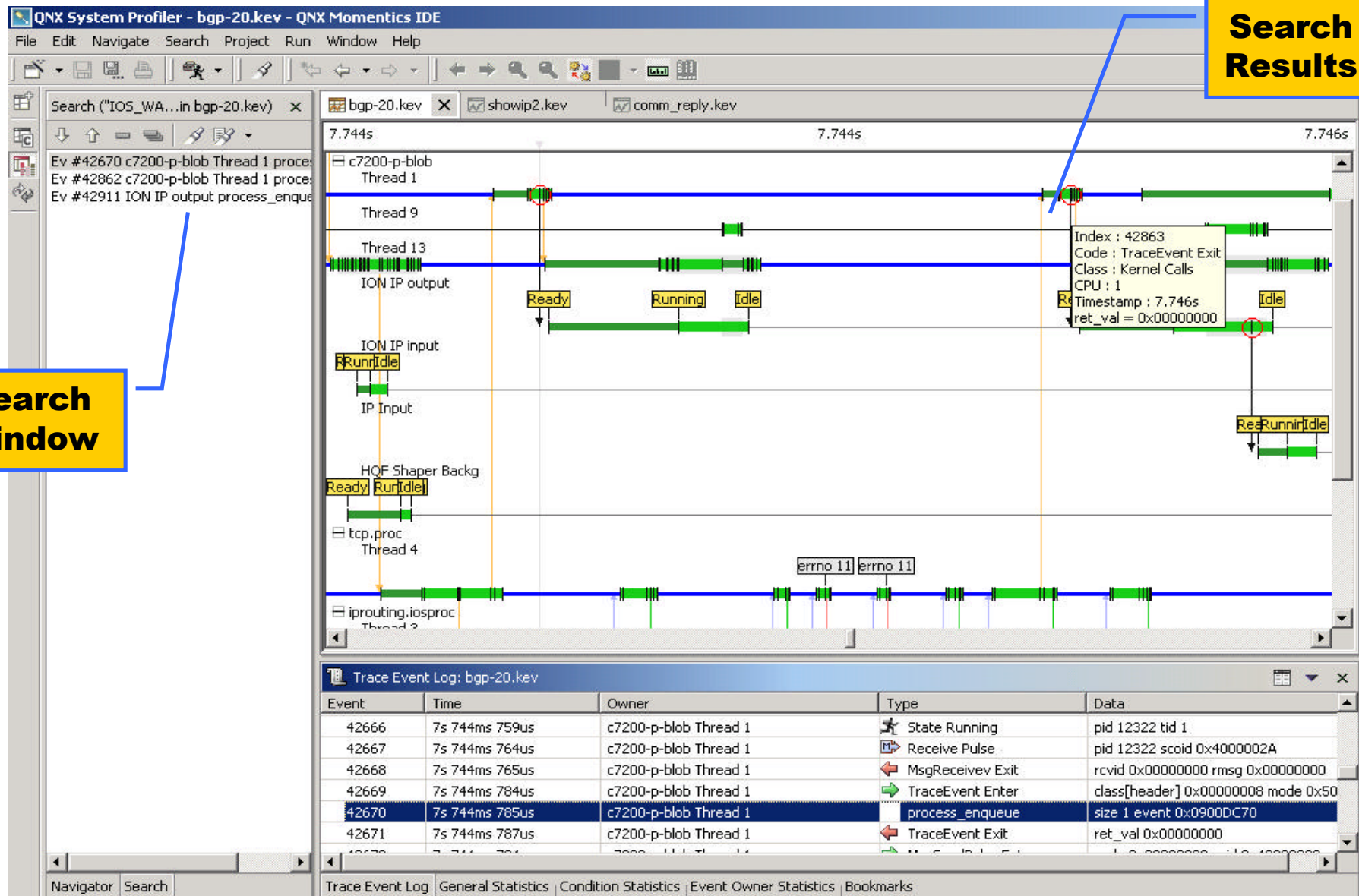


- **Standard BGP convergence test (20 routes)**
- **NPE-400 router running ION with instrumentation**
- **Trace captured during test**

BGP Convergence



Search Example



- ➔ **ION version to be delivered to Cisco beginning of June**
 - > **“Pure” IOS version to follow**
 - > **To try it out, please contact QNX or Mike Turnlund**
- ➔ **Come see a full demo at the QNX booth**
- ➔ **Discussions with Cisco on integrating other QNX tools to target IOS**

- **Eclipse-based IDE provides standard, flexible platform for tool integration**
- **Powerful suite of tools enable accelerated development, improved performance, and higher quality**
- **QNX Momentics available today for QNX-based systems**
- **Efforts underway to integrate selected Momentics tools for IOS-based systems**

- Download the case study – Using System Tracing to Improve Packet Forwarding Performance:
<http://www.qnx.com/info/systemtracing>
- Find out what's coming in QNX Momentics v6.3:
http://www.qnx.com/developers/6_3_glance.html
- Evaluate QNX Momentics today:
<http://www.qnx.com/products/eval/index.html>
- Contact Jim Gallagher (jgallagher@qnx.com) or Sebastien Marineau (sebastien@qnx.com) for more information