Micro-architectural Anatomy of a Commercial TCP/IP Stack

7th Annual IEEE Workshop on Workload Characterization (WWC-7)

October 25, 2004

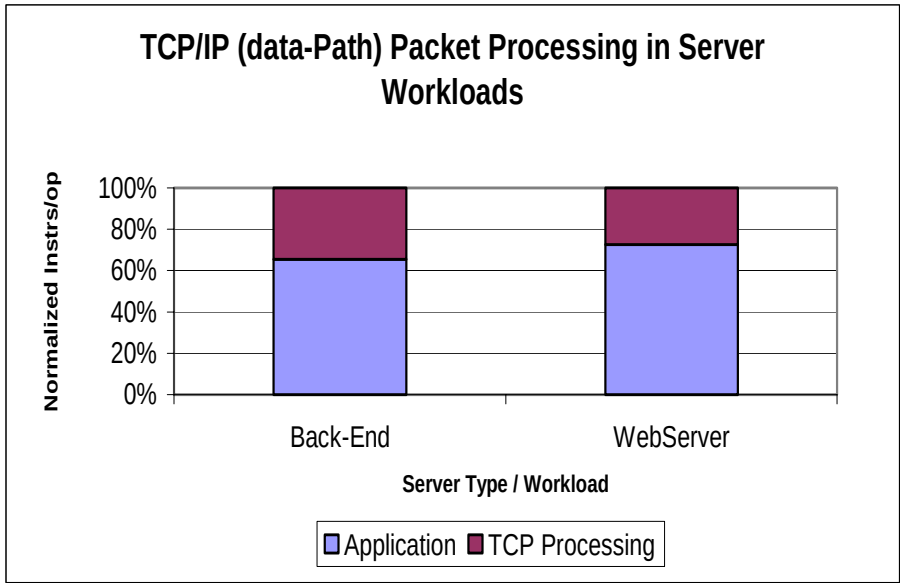
Ramesh Illikkal, Ravi Iyer and Don Newell
Communications Technology Lab
Intel Corporation

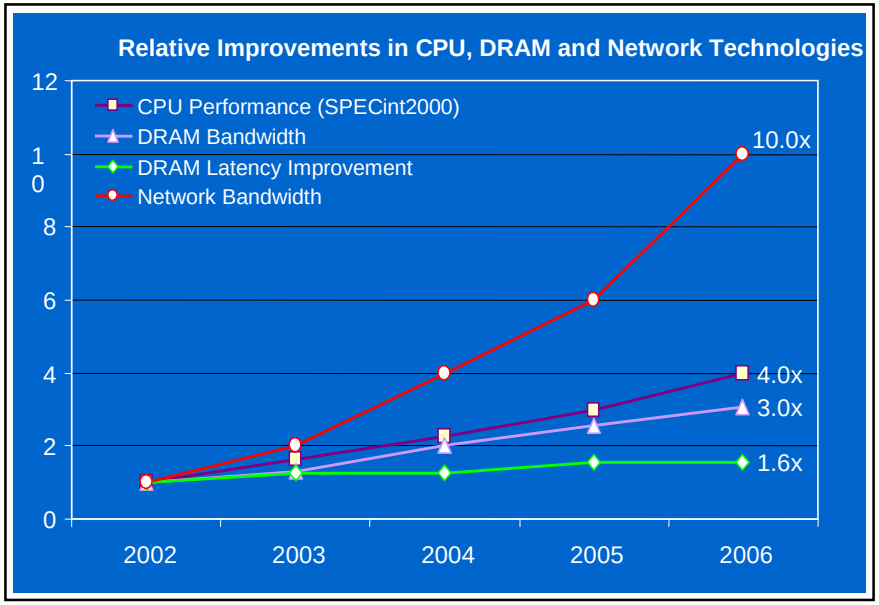
{Ramesh.g.illikkal, Ravishankar.iyer, Don.newell}@intel.com

Outline

- Problem Statement: Why TCP/IP
  - Importance in Server Workloads
  - Performance issue with TCP/IP
- How do we attack the problem
  - Metrics of interest & Current Tools
  - Trace based Analysis
- Trace based workload analysis
  - Instruction tracing
  - Performance simulation
  - Symbol Annotation
- TCP/IP performance breakdown
  - Bottom half
  - Upper Half
- Summary/Conclusions

Why TCP/IP?





(SPECint2000 data for IA32 CPU P4 based family)

- TCP/IP Processing contributes to ~30% in a typical web server and backend servers with network storage
- Very high CPU utilization for TCP/IP processing on servers - 100 % CPU for small packets
- Problem gets worse with 10 Gigabits per second

Performance Tools and Metrics

System Level Metrics (Perfmon)

NTTTCP Receive Workload (4KB)	
Perfmon Events	Counts
Bytes Received/sec	88224948
Bytes Sent/sec	961899
Packets Received/sec	62159
Packets Sent/sec	17813
Processor(_Total)\% Idle Time	15
Processor(_Total)\% Interrupt Time	20
Processor(_Total)\% Processor Time	85
Processor(_Total)\DPCs Queued/sec	8753
Processor(_Total)\Interrupts/sec	8862
System\Context Switches/sec	22460
System\System Calls/sec	50588

Architectural Metrics (eMon)

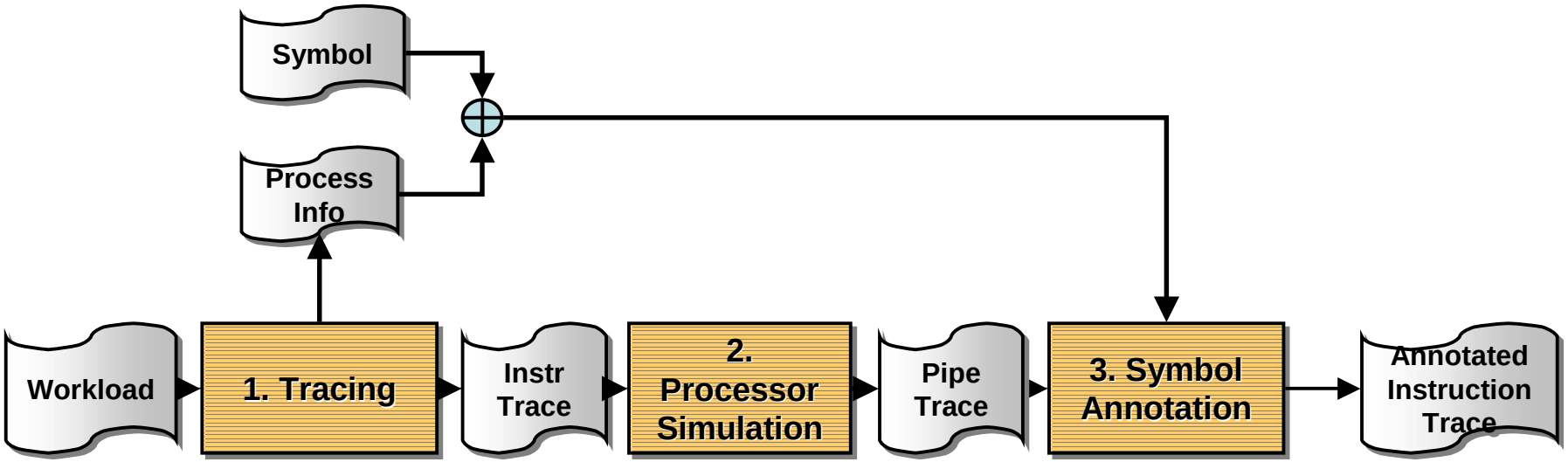
	#of Instructions per buffer	# of branches per buffer	Memory accesses per buffer	TLB Misses Per buffer	Instruction Cache Misses per buffer	Data Cache misses per buffer
64	3,759	88.1	0.4	0.71	1.0	2.1
128	3,892	90.6	0.8	0.73	1.1	3.2
256	4,134	97.5	1.5	0.78	1.3	5.3
512	4,607	110.3	3.0	0.90	1.7	9.6
1024	5,608	134.5	6.0	1.18	2.5	18.0
1460	6,473	157.6	8.7	1.31	3.1	24.0
2048	9,779	244.4	12.3	1.63	4.8	36.8
4096	21,547	546.0	26.5	5.31	13.2	95.7
8192	31,955	788.0	58.1	6.93	16.5	196.0
16384	56,225	1451.4	133.3	11.97	28.1	411.2
32768	130,440	2805.8	282.9	21.78	50.6	890.3
65536	245,148	6327.0	588.7	41.01	92.9	1783.3

- Workload Metrics of interest
  - Network Throughput (Bits per second)
  - CPU Utilization (%of CPU)
- System Level Metrics give next level of details
  - System Software Metrics (Perfmon)
  - Platform HW Architecture Metrics (eMon)
- Profiling for Software Hotspots
  - Sampling based (Vtune)
  - Misses the process flow details

Hotspot Profilers (Vtune)

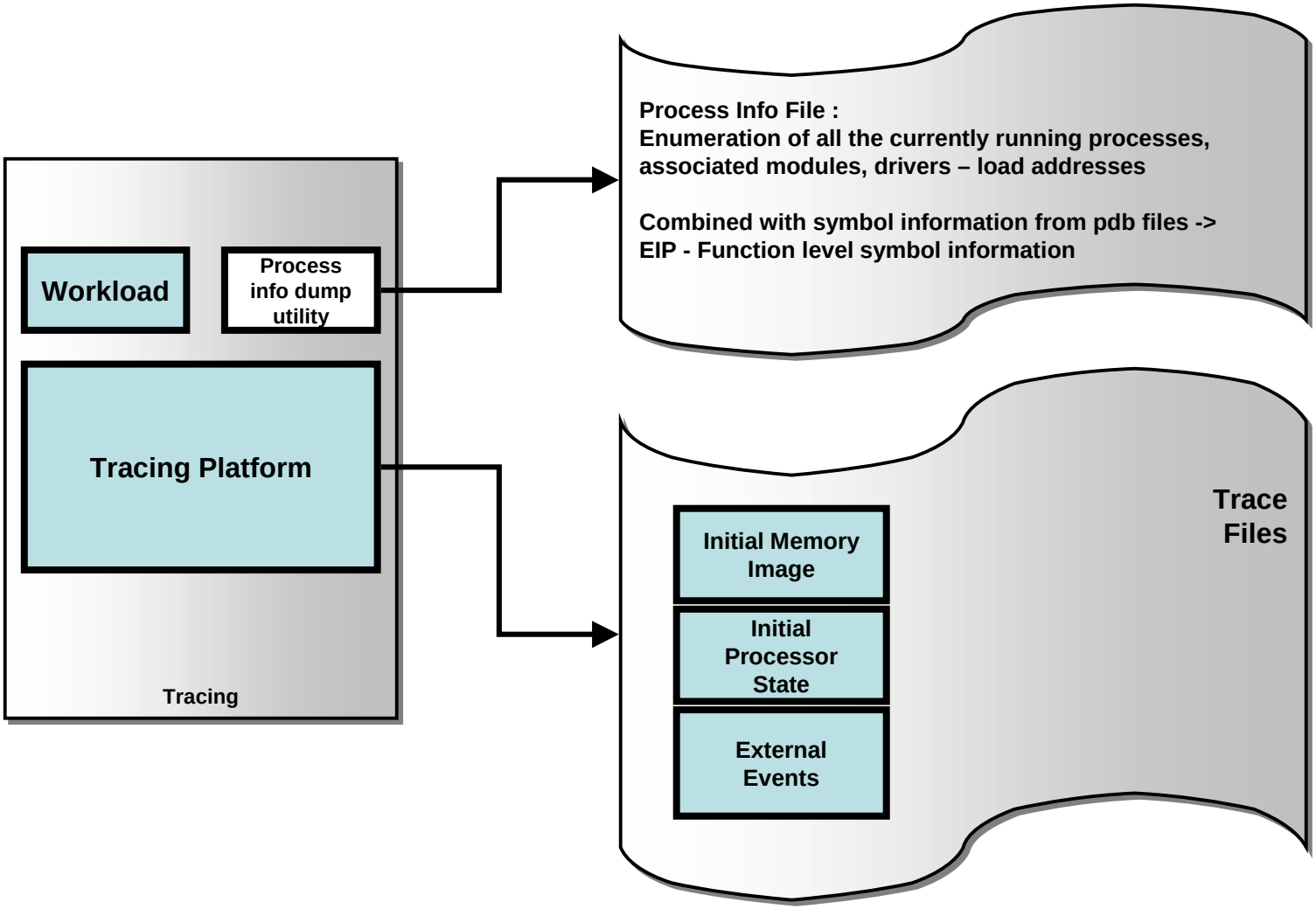
Function	Cycles %	Instruction %	L1 Miss	TLB Miss	Module
KiDispatchInterrupt	26.23	1.20	0.11	0.32	ntoskrnl.exe
ModuleStart	11.35	15.95	0.06	9.16	VTRun.dll
MmIsThisAnNtAsSystem	2.14	1.33	0.03	0.02	ntoskrnl.exe
LsaDeregisterLogonProcess	1.66	2.24	0.02	0.00	ntoskrnl.exe
KeReleaseInStackQueuedSpinL	1.53	1.87	0.08	0.14	ntoskrnl.exe
DllGetClassObject	1.51	2.91	0.02	0.13	pdm.dll
KfLowerIrql	1.34	1.53	0.07	0.00	hal.dll
LCMapStringW	1.16	4.32	0.00	0.06	KERNEL32.dll
NtDuplicateObject	1.04	1.58	0.04	0.02	ntoskrnl.exe
STROBJ_vEnumStart	0.94	1.39	0.15	1.23	win32k.sys
READ_REGISTER_ULONG	0.89	0.03	1.13	0.00	ntoskrnl.exe
HalBeginSystemInterrupt	0.85	0.08	0.25	0.02	hal.dll

Process flow

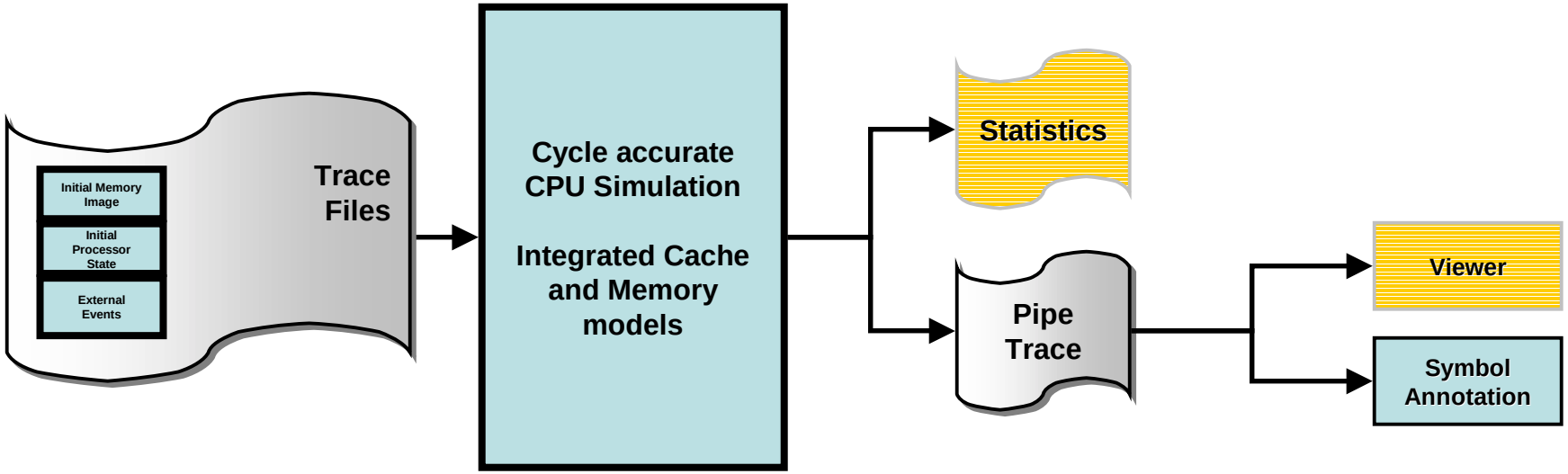


- Instruction traces collected from HW/SW tracing platforms
- Micro-architectural level timing, event and latency information added to traces by running it through CPU simulators of interest
- Symbol information added to the traces for function level understanding of the software

Tracing…
Recording a snapshot of execution



Simulation…
Replay of the snapshot



The CPU simulated could be different than the CPU we collected the traces on.

Detailed clock by clock representation of all the Instructions through the processor pipeline

Add symbol information by matching EIP to symbol load address

Symbol Annotation Example…

```
DRIVERS
-------

   #  ADDRESS      SIZE NAME
-----------------------------------------------------------------
   1: 804DE000    266000 \WINDOWS\system32\ntoskrnl.exe
   2: 80744000     28000 \WINDOWS\system32\hal.dll
   3: F7707000      8000 \WINDOWS\system32\KDCOM.DLL
   4: F770F000      8000 \WINDOWS\system32\BOOTVID.dll
   5: F7435000     31000 ACPI.sys
   6: F7487000      9000 \WINDOWS\system32\DRIVERS\WMILIB.SYS
   7: F7420000     15000 pci.sys
   8: F7497[illegible]
   9: F7717[illegible]
  10: F74A7[illegible]DEX.SYS
  11: F74B7[illegible]
  12: F73FB[illegible]
  13: F771F[illegible]
  14: F7301[illegible]
  15: F73B0000    21000 volsnap.sys
  16: F74C7000     9000 ACPIEC.sys
  17: F7727000     8000 \WINDOWS\system32\DRIVERS\OPRGHDLR.SYS
  18: F74D7000     E000 PartMgr.sys
  19: F7394000    1C000 atapi.sys
  20: F737B000    18E80 adpu160m.sys
  21: F7355000    26000 \WINDOWS\system32\DRIVERS\SCSIPORT.SYS
  22: F7338000    19680 adpu320.sys
  23: F74E7000     F000 disk.sys
  24: F7325000    16000 \WINDOWS\system32\DRIVERS\CLASSPNP.SYS
  25: F74F7000     C000 Dfs.sys
```

Process, Module Enumeration

⊕

```
Preferred load address is 00400000

Start         Length     Name                    Class
0001:00000000 000Ddc4cH .text                   CODE
0002:00000000 000000e4H .idata$5                DATA
0002:000000f0 0000id18H .rdata                  DATA
0002:00001e08 0000003fH .rdata$debug            DATA
0002:00001e50 00000008H .rdata$xdata            DATA
0002:00001[illegible]                           DATA
0002:00001[illegible]                           DATA
0002:00001[illegible]                           DATA
0002:00001[illegible]                           DATA
0002:00001[illegible]                           DATA
0002:00001[illegible]                           DATA
0002:00001[illegible]                           DATA
0002:00001f74 000003e4H .idata$6                DATA
0002:00002358 00000000H .edata                  DATA
0003:00000000 00000004H .CRT$XCA                DATA
0003:00000004 00000004H .CRT$XCAA               DATA
0003:00000008 00000004H .CRT$XCZ                DATA
0003:0000000c 00000004H .CRT$XIA                DATA
0003:00000010 0000000cH .CRT$XIC                DATA
0003:0000001c 00000004H .CRT$XIZ                DATA
0003:00000020 00000004H .CRT$XPA                DATA
```

Symbol Information (PDB)

*** This is a map file, but similar info is available in PDB**

```
8074645C        HalpAcpiTimerQueryPerfCount
80746DB4        HalpmmTimerClockInterruptStub
80767CD4        rgzSessionManager
807656F8        HalpGetResourceSortValue
80752F80        HalpPICINTToVector
807578A0        HalpFixedAcpiDescTable
8074C2D8        HalpInitializeApicAddressing
8074AF90        HalpSendNodeIpi64
807671B8        DetectAcpiMP
8075B7A8        HalpSaveContextTargetProcessor
8074E21C        HalpVectorToINTI
8074CFCA        HalaAcpiTimerInit
807611BE        HaliAcpiTimerInit
80761AC0        HalpGetMceInformation
80752C1C        HalpAcpiMultiNode
8075BA9A        Hal[illegible]
8074D134        Ke[illegible]
8075FE40        Ha[illegible]
8074A6AE        HalpPCIWriteUlongType2
8074727C        HalpBroadcastCallService
80746D8C        V86_Hmmt_a
8074C180        HalpEnablePicInti
8075D10E        HalpSaveInterruptControllerState
80752260        HalpMoveMemory
80762DFA        HalpSearchForPciDebuggingDevice
80757CE0        HiberFreeCR3
80745110        _imp__InbvSolidColorFill
8074AC4C        HalpPassIrpFromFdoToPdo
80752C90        HalpTimerWatchdogEnabled
80761856        HalTranslatorReference
80752C10        HalpVirtAddrForFlush
807480A0        KeReleaseQueuedSpinLock
807527B0        rgzNoApic
80758E00        HalpNewAdapter
```

EIP-Symbol Map

Three phases of Trace based analysis

EIP	OpCode
80748538	xorl %eax, %eax
8074853a	movb 0x4(%esp), %al
8074853e	movb -0x7f8b8660(%eax), %al
80748544	movl -0x1ff80, %ecx
8074854a	movl %eax, -0x1ff80
8074854f	movl 0xc(%esp), %eax
80748553	shrl $0x4, %ecx
80748556	movb -0x7f8adde4(%ecx), %cl
8074855c	movb %cl, (%eax)
8074855e	movl $0x1, %eax
8074856	
8074856	
8074856	
8074856	

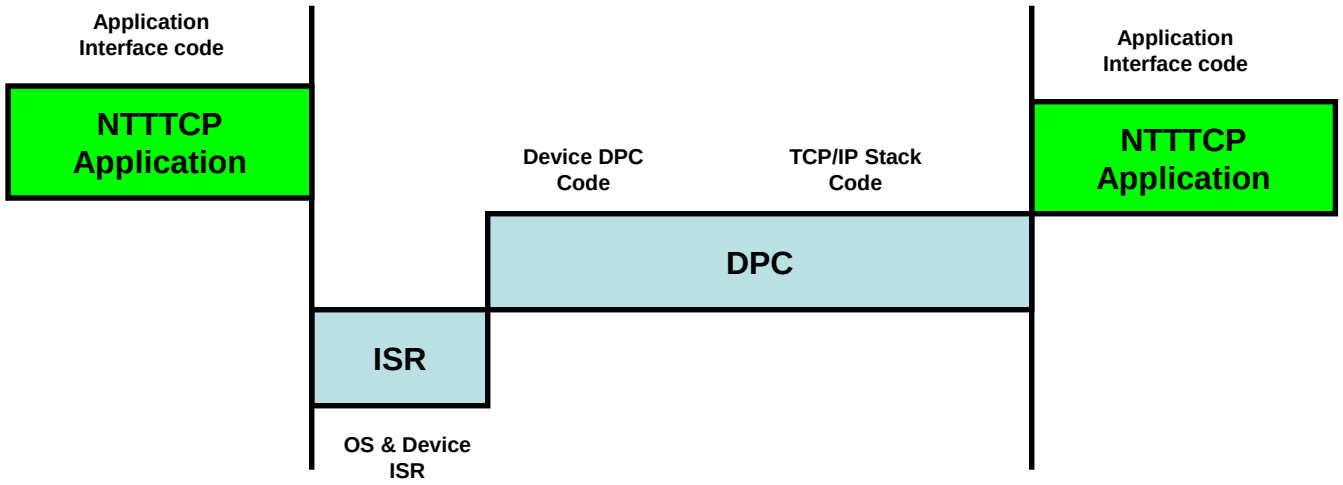
Eip	Clock	L1dStoreHit	L2StoreHit	L2StoreMiss	L1dLoadHit	L2LoadHit	L2LoadMiss	dtlbMiss	dtlbFlush	itlbMiss	itlbFlush	RobFull	PrefetchDropped	Opcode
80748538	998670	0	0	0	0	0	0	0	0	0	0	0	0	xorl %eax, %eax
8074853a	998672	0	0	0	1	0	0	0	0	0	0	0	0	movb 0x4(%esp), %al
8074853e	998675	0	0	0	1	0	0	0	0	0	0	0	0	movb -0x7f8b8660(%eax),
80748544	999396	0	0	0	0	0	0	0	0	0	0	0	0	movl -0x1ff80, %ecx

Call depth	Eip	Clock	Instructions	L1dStoreHit	L2StoreHit	L2StoreMiss	L1dLoadHit	L2LoadHit	L2LoadMiss	dtlbMiss	dtlbFlush	itlbMiss	itlbFlush	RobFull	PrefetchDropped	Symbol/Opcode
-11	80748538	998670								**HAL:HalBeginSystemInterrupt**						
	80748538	998670		0	0	0	0	0	0	0	0	0	0	0	0	xorl %eax, %eax
	8074853a	998672		0	0	0	1	0	0	0	0	0	0	0	0	movb 0x4(%esp), %al
	8074853e	998675		0	0	0	1	0	0	0	0	0	0	0	0	movb -0x7f8b8660(%eax),
	80748544	999396		0	0	0	0	0	0	0	0	0	0	0	0	movl -0x1ff80, %ecx
	8074854a	999396		1	0	0	0	0	0	0	0	0	0	0	0	movl %eax, -0x1ff80
	8074854f	999396		0	0	0	1	0	0	0	0	0	0	0	0	movl 0xc(%esp), %eax
	80748553	999400		0	0	0	0	0	0	0	0	0	0	0	0	shrl $0x4, %ecx
	80748556	999402		0	0	0	1	0	0	0	0	0	0	0	0	movb -0x7f8adde4(%ecx),
	8074855c	999402		1	0	0	0	0	0	0	0	0	0	0	0	movb %cl, (%eax)
	8074855e	999402		0	0	0	0	0	0	0	0	0	0	0	0	movl $0x1, %eax
	80748563	999415		0	0	0	0	0	0	0	0	0	0	0	0	sti
	80748564	999415		0	0	0	0	0	0	0	0	0	0	0	0	cmpb $0x2, %cl
	80748567	999415		0	0	0	0	0	0	0	0	0	0	0	0	jae .+0xa
	80748569	999426		0	0	1	0	0	0	0	0	0	0	0	0	movb $0x2, %fs:0x95
-12	80748571	999430	14	2	0	1	4	0	0	0	0	0	0	0	0	**RETURN**

%eax, -0x1ff80
0xc(%esp), %eax
0x4, %ecx
-0x7f8adde4(%ecx),
%cl, (%eax)
$0x1, %eax

$0x2, %cl
-0xa
$0x2, %fs:0x95
0xc

What we are about to look at…



- NTTTCP Application running, reading data from the TCP/IP stack
- ISR gets invoked when an interrupt is received
- Device ISR acknowledges the interrupt, queues a DPC
- Device DPC code reads 64 descriptors and indicates to TCP/IP
- TCP stack processes 64 packets
- NTTTCP Application resumes, reading data from the TCP/IP stack

Annotated Trace: Bottom Half

Event	Clock	Function Name
Start of interrupt	0	HAL:HalBeginSystemInterrupt
	2225	NDI:ndisMIsr
	3137	ETH:E1000ISR
	5586	ETH:E1000DisableInterrupt
	6511	NTO:KeInsertQueueDpc
	6531	HAL:KfRaiseIrql
	7726	NTO:KiAcquireSpinLock
	7909	HAL:HalRequestSoftwareInterrupt
	7726	NTO:KiAcquireSpinLock
	8206	HAL:KfLowerIrql
End of interrupt	9041	HAL:HalEndSystemInterrupt
Scheduler code	9131	NTO:KiDispatchInterrupt
	10225	NTO:KiRetireDpcList
	11287	NTO:KefAcquireSpinLockAtDpcLevel
	11343	NTO:KefReleaseSpinLockFromDpcLevel
DPC Start	11681	NDI:ndisMDpcX
	12210	ETH:E1000HandleInterrupt

Event	Clock	Function Name
DPC Begin	0	ETH:E1000HandleInterrupt
	1087	ETH:RxProcessReceiveArray
Packet 1	1983	ETH:RxPacketAssemble (1)
	2114	NTO:KefAcquireSpinLockAtDpcLevel
	6575	ETH:RxVlanProcess
	8496	ETH:RxProcessPacketActions
	9095	ETH:RxTcpIpChecksumVerify
	10217	NTO:KefReleaseSpinLockFromDpcLevel
Packet 2	10312	ETH:RxPacketAssemble (2)
	…	…
Packet 3	11658	ETH:RxPacketAssemble (3)
	…	…
Packet 4	12874	ETH:RxPacketAssemble (4)
	…	…
	…	…
Packet 64	90138	ETH:RxPacketAssemble (64)
RX Handler End	91558	ETH:ReceivePacketArrayIndicate

- Interrupt Service Routine
- Software interrupt queuing
- Scheduling of Software Interrupt

- Software Interrupt Handler
- Receives all the packets received
- Indicates the arrival of packets to protocol stack

Annotated Trace: Contd.

Event	Clock	Function Name
Start TCP	0	TCP:ARPRcvPacket (1)
Packet 1	848	TCP:ARPRcvIndicationNew
	2368	TCP:IPRcvPacket
	3968	TCP:GetLocalNTE
	4737	TCP:ProcessFirewallQ
	6768	TCP:DeliverToUser
	7226	TCP:ProcessFirewallQ
	9561	TCP:UpdateIPSecRcvBuf
	11077	TCP:FindUserRcv
	12124	TCP:TCPRcv
	13552	TCP:PrefetchRcvBuf
	14319	NTO:RtlPrefetchMemoryNonTemporal
	16096	TCP:FindTCB
	17613	NTO:KefAcquireSpinLockAtDpcLevel
	17964	NTO:KefAcquireSpinLockAtDpcLevel
	17990	NTO:KefReleaseSpinLockFromDpcLevel
	20430	TCP:IndicateData
	20611	NTO:KefReleaseSpinLockFromDpcLevel
	20751	AFD:AfdBChainedReceiveEventHandler
	20760	AFD:AfdCheckAndReferenceConnection
	22149	HAL:KeAcquireInStackQueuedSpinLock
	23246	AFD:AfdBFillPendingIrps
	23515	AFD:AfdGetPendedReceiveIrp
	23709	AFD:AfdMapMdlChain
	23765	NTO:MmMapLockedPagesSpecifyCach
	23812	NTO:ExRemoveHeadNBQueue
	24221	NTO:ExfInterlockedCompareExchang
	24249	NTO:RtlpInterlockedPushEntrySList
	24602	HAL:KeReleaseInStackQueuedSpinLock
	25100	AFD:AfdCopyMdlChainToMdlChain
	26311	NTO:IofCompleteRequest
	26373	NTO:MmUnlockPages
	26385	NTO:MmUnmapLockedPages
	26514	NTO:ExInsertTailNBQueue
	26519	NTO:InterlockedPopEntrySList
	26609	NTO:ExfInterlockedCompareExchang
	26638	NTO:ExfInterlockedCompareExchang
	26769	NTO:InterlockedPopEntrySList
	26940	NTO:KeInitializeApc
	27242	NTO:KeInsertQueueApc
	27733	HAL:KeAcquireInStackQueuedSpinLockR
	28136	NTO:KiInsertQueueApc
	28247	NTO:KeAcquireQueuedSpinLockAtDpcL
	28519	NTO:KiUnwaitThread
	28617	NTO:KiReadyThread
	28727	NTO:KeReleaseInStackQueuedSpinLockI
	28772	NTO:KiExitDispatcher
	28793	NTO:KiProcessDeferredReadyList
	29360	NTO:KiDeferredReadyThread
	30974	HAL:KfLowerIrql
	32288	NTO:KefAcquireSpinLockAtDpcLevel
	33265	TCP:START_TCB_TIMER_R
	34277	TCP:StartTCBTimerR
	34378	TCP:RemoveAndInsertIntoTimerWheel
	34850	NTO:KefAcquireSpinLockAtDpcLevel
	35015	NTO:KefReleaseSpinLockFromDpcLevel
	35061	TCP:DerefTCB
	35090	HAL:KfReleaseSpinLock
	36112	TCP:CompleteSends
	36670	ETH:E1000ReturnPacket
	36719	NTO:KefAcquireSpinLockAtDpcLevel
	36755	NDI:NdisUnchainBufferAtFront
	36870	NTO:KefReleaseSpinLockFromDpcLevel
Packet 2	37060	TCP:ARPRcvPacket (2)
…	…	…
Packet 3	46947	TCP:ARPRcvPacket (3)
…	…	…
Packet 4	53038	TCP:ARPRcvPacket (4)
…	…	…
…	…	…
Packet 64	291410	TCP:ARPRcvPacket (64)
IP Complete	296736	TCP:IPRcvComplete
TCP Complete	297289	TCP:TCPRcvComplete

Event	Clock	Function Name
	0	TCP:ProcessTCBDelayQ
ACK 1	383	TCP:SendACK (1)
	2156	TCP:ClassifyPacket
	2222	TCP:IPTransmit
	2520	TCP:GetIPPacket
	4375	TCP:ARPTransmit
	5009	TCP:ARPSendData
	5140	NDI:ndisMSendX
	6669	ETH:E1000SendPackets
	7250	ETH:PACKET_SEND
	8616	TCP:DerefTCB
	8901	TCP:CompleteRcvs
	9529	TCP:TCPDataRequestComplete
	13501	NTO:KeInitializeApc
	13533	NTO:KeInsertQueueApc
	15021	TCP:IndicatePendingData
	15095	AFD:AfdBReceiveEventHandler
	23689	TCP:FreeTcpIpr [Rep Mov]
ACK 2	42525	TCP:SendACK (2)
…	…	…
ACK 3	94050	TCP:SendACK (3)
…	…	…
ACK 4	134506	TCP:SendACK (4)
…	…	…

Event	Clock	Function Name
	0	GetOverlappedResult
	39	ReadFile
	299	ObReferenceObjectByHandle
	458	IoGetRelatedDeviceObject
	705	IoAllocateIrp
	1068	IoAllocateMdl
	1162	MmProbeAndLockPages
	2001	IopSynchronousServiceTail
	2077	AfdDispatch
	2089	AfdReceive
	2125	AfdBReceive
	2710	AfdGetReceiveBuffer
	2747	AfdMapMdlChain
	2756	MmMapLockedPagesSpecifyCache
	3208	AfdCopyMdlChainToMdlChain
	4121	AfdFreeBuffer
	8164	AfdGetReceiveBuffer
	8903	KfRaiseIrql
	9378	IopCompleteRequest
	9478	IoFreeMdl
	9536	KeSetEvent
	10292	IoFreeIrp
	10418	KfLowerIrql
	10662	WaitForMultipleObjects
	12352	GetOverlappedResult

Analysis Summary

Metric	Top Half	Bottom Half
CPI	5.9	3.4
MPI	0.0135	0.0199
% Cycles	57%	43%

- Application interface constitutes more than half of the processing (57%)
  - New application interfaces are in the horizon

Metric	ISR	Device DPC	Stack
Cycles	11343	92086	445322
Instructions	330	22591	137035
MPI	0.0009	0.0051	0.0061
% Cycles	2%	17%	81%

- TCP processing constitutes most of the bottom half processing
- Optimization in SW and HW can reduce this part considerably
  - ETA and MARS stacks are our steps towards this
  - More information can be found in IEEE Computer Nov 04 Issue

Summary/Conclusions

- Symbol Annotated Traces enable in-depth workload analysis
  - Provides Performance Characterization at Software and Hardware levels
- We provide a u-Arch level Analysis of TCP stack using this methodology
  - We follow a packet from the interrupt to application delivery (life of a packet)
  - Top half processing consumes more than 50 % of processing
  - Most of the Bottom half processing is in the protocol processing itself