

Tracing for Performance Monitoring on Parallel and Distributed Systems

Dr. Robert W. Wisniewski Manager Blue Gene Software IBM T. J. Watson Research http://www.research.ibm.com/people/b/bob/





	 The second s
-313	
. E	 376.

Outline

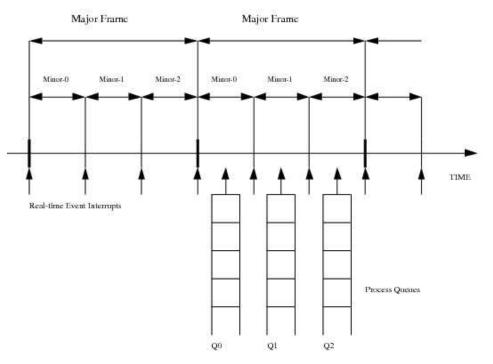
- > SGI (RTAS Real-Time Technology and Applications Symposium 95)
 - rtmon
 - Kernel and Cray Unification
 - Lessons
- K42 (Supercomputing 03)
 - Approach, scalability, and use
 - Lessons
- CPO (Continuous Program Optimization) (PAC2 2004)
 - **PEM (Performance Environment Monitoring)**
 - Lessons
- CSO (Commercial Scale-Out) (europar07 slides thanks Jose Moreira)
 - Goals
 - Lessons
- Blue Gene / P (internal slides thanks Valentina Salapura)
- Observations on Linux and LTT
- The "next system" concluding remarks



Frame Scheduler

Major Frames: Determines period - a complete cycle of processes

Minor Frames: Independent units within major frame - used for setting up sepcific application behavior

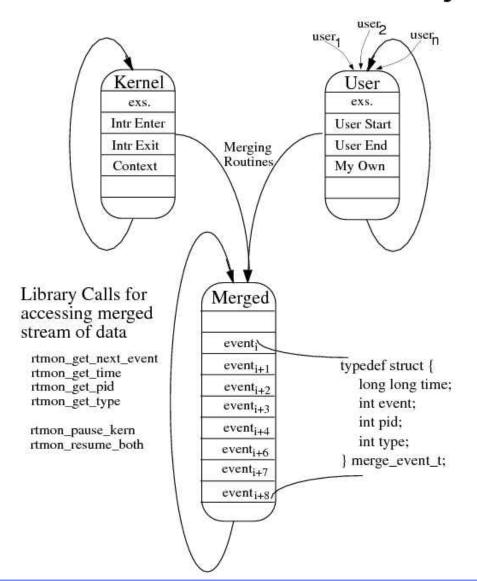


Processes can be enqueued in more than one process queue

	æ,	= =
- T		
3		278

FrameView - Bottom Layer

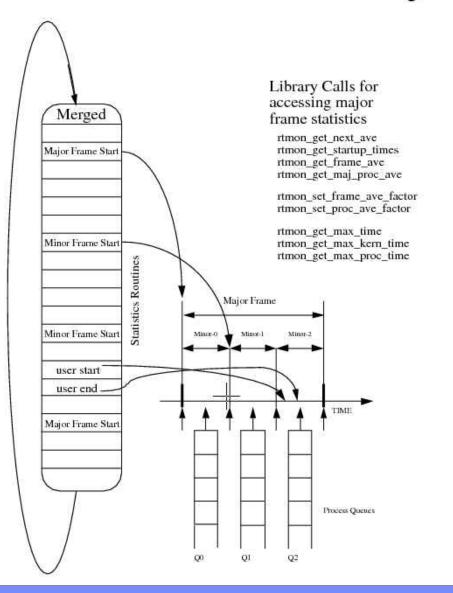
- rtmon bottom layer
 - Per-processor
 - Multiple writes user q
 - •atomicIncWrap gets index
 - Set valid bit when done
 - •Reader clears valid bit





FrameView - Middle Layer

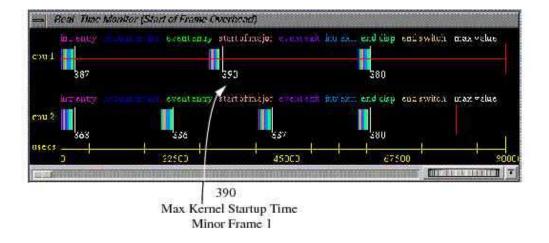
- rtmon middle layer
 - Assign meaning to events, recreate frames
 - Report discrepancies
 - Calculate extreme value





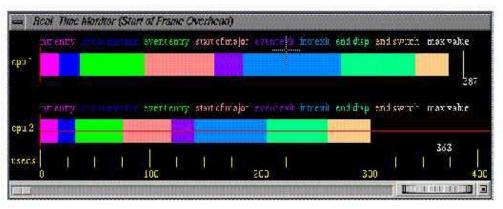
Kernel Startup Graph Real-Time Monitoring Tools

- rtmon top layer
 - Multiple view



Above represents full view matching the view as seen in the main graph Below represents a blown up image of the kernel startup time for minor frame 0

Colors of event labels match color bars



		_	
12			
Ξ.			2.0
	-	and a	

SGI Kernel and Cray Unification

- rtmon extended to kernel
 - 3 separate tracing schemes depending on what you were doing
 - •Confusing
 - •Error prone
 - •Hurts performance
- SGI purchases Cray
 - 5 separate tracing schemes...
 - Cray introduces another aspect

•Need data from machines in field that are not possible to build in house – requires extensive events and black-box capability

E.	and and and

SGI Lessons

rtmon

- + Collect cheaply on line more expensive off line processing
- Roughly ¹/₄ of machine needed to get events off
- Tradeoff between application-specific design and generality
- + Single system of trace events useful
- + Possible to do non-locking tracing
- Fixed events are cumbersome
- Visualization is key
 - •It's the killer app for tracing

-	
- 21	
<u> </u>	
	Contract design of stands

Outline

- SGI
 - rtmon
 - Kernel and Cray Unification
 - Lessons
- ≻ K42
 - Approach, scalability, and use
 - Lessons
- CPO (Continuous Program Optimization)
 - **PEM (Performance Environment Monitoring)**
 - Lessons
- CSO (Commercial Scale-Out)
 - Goals
 - Lessons
- Blue Gene / P
- Observations on Linux and LTT
- The "next system" concluding remarks



K42's Goals (started 1997)

- Scalability
 - Up to large MP and large applications
 - down for small-scale MP and small apps on large-scale MP
- Flexibility/Customizability:
 - policies/implementations of every physical/virtual resource instance can be customized to application needs
 - system can adapt to security and performance faults without penalizing common case performance
- Portability:
 - can be easily ported to new 64-bit platforms
 - can exploit features of HW
- Availability:
 - fault containment: should be able to survive HW failures on large MP
 - can be dynamically upgraded without bringing system or apps down
- Maintainability/Extensibility:
 - highly module structure
 - re-enable the OS research community
- Full Functionality and Linux compatibility:
 - support huge numbers of Linux apps and drivers without modification
 - transfer technology back and forth to vanilla Linux

Goals: Performance Monitoring

- Provide unified events for correctness and performance
- Allow events to be gathered efficiently on a multiprocessor
- Allow efficient logging of events from applications, servers, and the kernel into a unified buffer with monotonically increasing timestamps
- Have the infrastructure always compiled into the system allowing data gathering to be dynamically enabled
- Separate the collection of events from their analysis
- Have minimal impact on the system when tracing is not enabled; allow for zero impact by providing the ability to "compile out" events
- Provide cheap and flexible collection of data for either small or large amounts of data per event



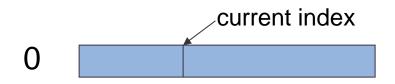


Key Ideas

- * lockless logging
- * random access variable length events
- \rightarrow unified events
- ^ user-mapped per-processor buffers
- ^ major and minor ids



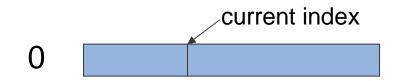


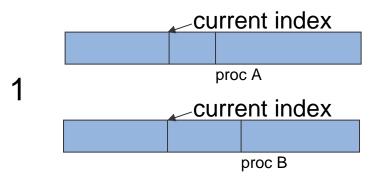


Process A - to log 2 words Process B - to log 3 words



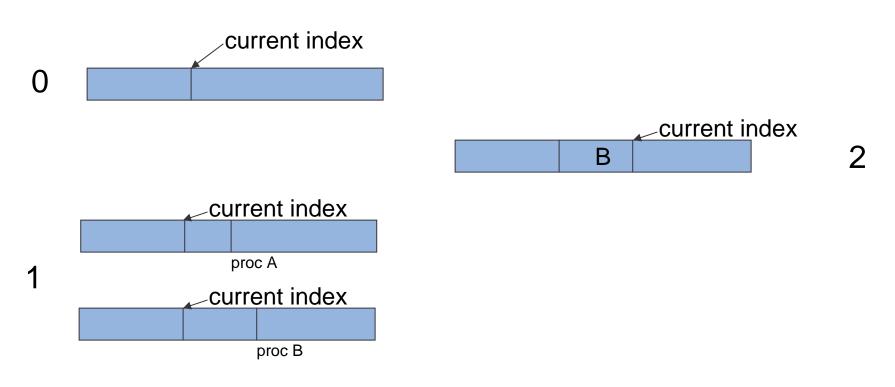






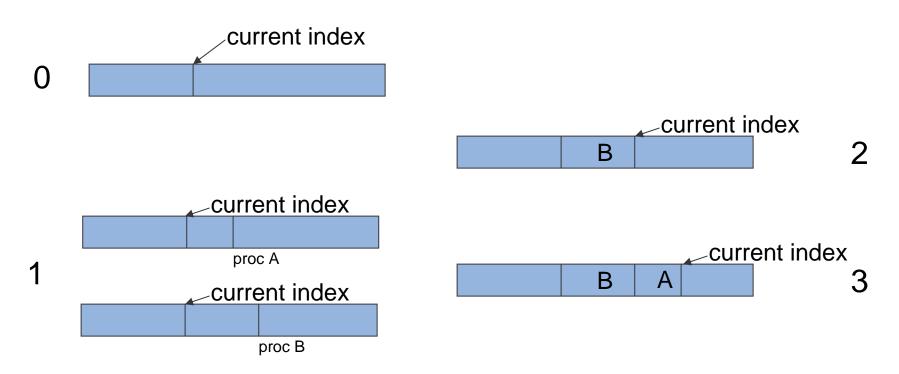












works between user, servers, kernel potential problems – event loss etc.





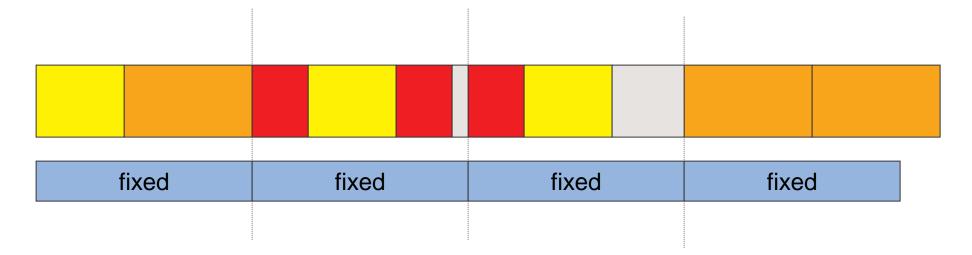
Random Access Variable Length Events

- Variable length events (vs fixed length)
 - more flexible
 - cheaper
 - space
 - time
 - easier for longer events









works for RAM and disk





Use Event Listing

21.4747350 TRC_USER_RUN_UL_LOADER 21.4747422 TRC_EXCEPTION_PGFLT 21.4747882 TRC_EXCEPTION_PGFLT_DONE 21.4748091 TRC_EXCEPTION_PGFLT_DONE 21.4748530 TRC_MEM_FCMCOM_ATCH_REG 21.4748709 TRC_MEM_FCMCRW_CREATE 21.4749142 TRC_EXCEPTION_PPC_RETURN 21.4749247 TRC_EXCEPTION_PPC_CALL 21.4749573 TRC_MEM_REG_CREATE_FIX 21.4749773 TRC_MEM_REG_DEF_INITFIXED 21.4749873 TRC_MEM_ALLOC_REG_HOLD 21.4749962 TRC_MEM_ALLOC_REG_HOLD 21.4750293 TRC_MEM_FCMCOM_ATCH_REG process 6 created new process with id 7 name /shellServe PGFLT, kernel thread 8000000c12b0f90, faultAddr 405e628, PGFLT DONE, kernel thread 8000000c12b0f90, faultAddr 405 PPC CALL, commID 0 Region 80000001022cc98 attached to FCM e10000000003f30 TRC_MEM_FCMCRW_CREATE ref e10000000003f90 PPC RETURN, commID 60000000 PPC CALL, commID 0 Region default 1000000 created fixlen addr 113000 region default 1000000 created fixlen addr 113000 alloc region holder addr 10000000 size 113000 alloc region holder addr 1000000 size 113000 Region e1000000003fa0 attached to FCM e1000000003f90





Use Fine-Grained Behavior

Exec:./runte SCbrk SCchild SCexecve SCexit SCmmap SCrmdir dispatcher user	est.sh /b : : : : : :	8.39/4/8	f: f: f: f: f:	1041.17/80 273.20/15 87.53/7 1304.87/76		107.45/18 691.53/34 24.19/5
fault ppc Ex-process wall 10800	: : : total: .11/0	929.41/1/5 2804.31/184/186 1274.52/93/210 5008.23/401			p: p: p:	
CRT::Fork COSMgrO	rDefault Worker bject::C	ase2 ::AsyncMsgHandler :leanupDaemon ed::ProcessMsgList		255.3 4.05/ 246. 185.6 3.56	/3 10/4 51/2	





Use

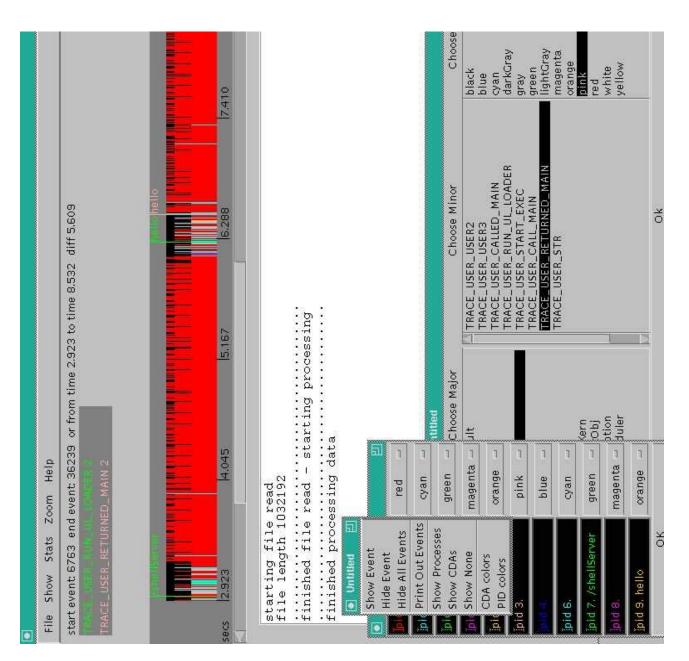
Lock Contention Analysis

top 10 contended locks by time - for full list see traceLockStatsTime

time (secs)	count	spin	max time	pid
		call chair	-	
3.466320753	1209	188795433	0.012220087	0x1
		AllocReg	ionManager::all	oc(unsigned
		PMalloc[Default::pMalloc(unsigned
		GMalloc:	::gMalloc()	
0.684612632	573	37233770	0.007647854	0x0
		AllocReg	jionManager::all	oc(unsigned
		PMalloc[Default::pMalloc(unsigned
		GMalloc:	::gMalloc()	
0.104643241	11885	4910595	0.000322320	0x1
		PageAllc	catorDefault::de	allocPages(unsigned
		PageAllc	catorUser::deall	ocPages(unsigned
		AllocPoo	l::largeFree(void	*,











and a local local series and a	
Strength (Management Strength out) - Annual	

K42 Lessons

• K42

23

+ Static trace points valuable

•More efficient (94 cycles on K42)

•Modified when code is modified

- + Separate definition files useful
- + Breakdown into major and minor classes useful
- + Variable length events
- + Single unified system for events
- + Dynamic enabling and disabling useful
- No dynamic events
- No flexibility at event time
- Visualization is key

	1.00	Statement of the local division of the	
1.0.3	A. 1994		

Outline

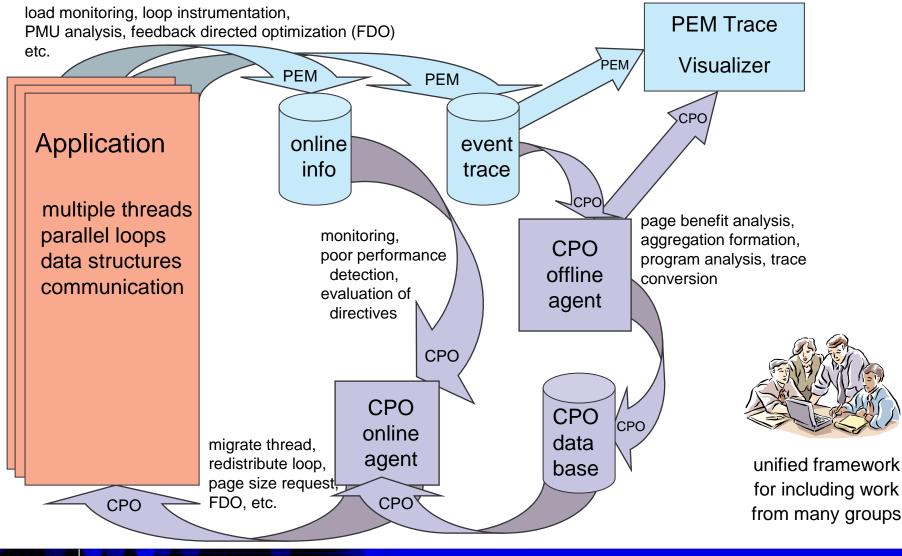
- SGI
 - rtmon
 - Kernel and Cray Unification
 - Lessons
- K42
 - Approach, scalability, and use
 - Lessons
- CPO (Continuous Program Optimization)
 - **PEM (Performance Environment Monitoring)**
 - Lessons
- CSO (Commercial Scale-Out)
 - Goals
 - Lessons
- Blue Gene / P

24

- Observations on Linux and LTT
- The "next system" concluding remarks

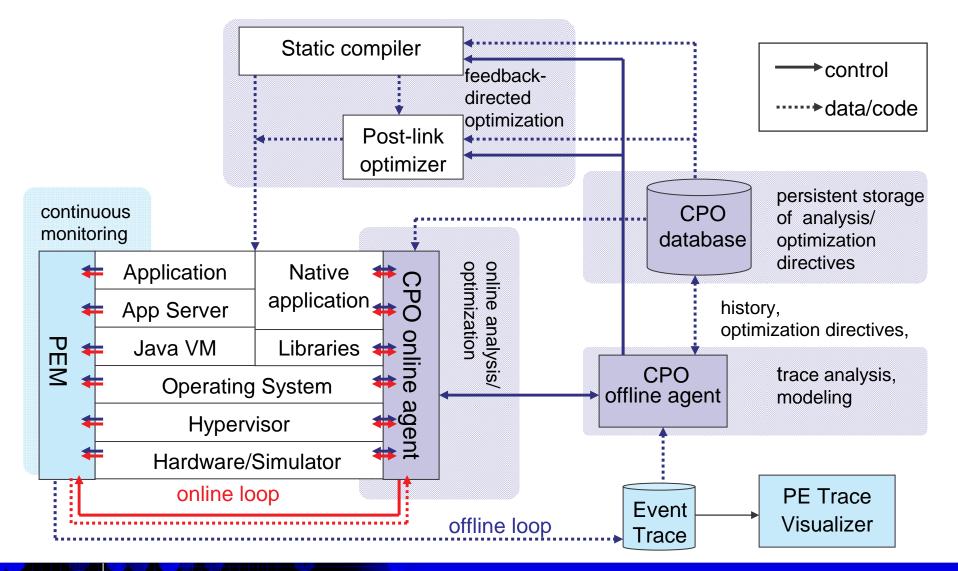


CPO Vision and Potential





CPO Architecture



Overview of cth performance using PE

ile	xplorer (PE)																						
🗖 Strip Array																							ŕď
	coom all Zoom out Z			400 400	******		919199999999								iaiaiaiaia	aaaaaaaa							1 121
+ - ^ V ? Z	.00m ali 2.00m out 2	oomin	< > Cer																	1			
		0G	T/		16			20			30			4G				5G		Т.	.6G		
				1 10 10	Docto do d			35573			2689846	4	239 (6894					xeelo lo lo			d and a loss		
		OG	CH H H H	19.6	1G	- x - x - x	2.2.7	¹ 2G	<u>i i i i</u>	4.9.9	'3G	¥.		4G	1.1	i ii ii	2.2. 19.	5G	R R R 7		¹ 6G	- 4 4 A A	33
pplication Markers	1 records 1 records		t ¹																	1			
mDynamic	3 records	1																					
tatic	6 records 6 records																						
gDynamic	19 records		ttt																				
itack	3 records		19961																				
lalloc/Free	55 records 77 records		mt t																	t			
	6043.0		4																	1			
Page Faults	pgfits 1 recor1 recor		t ¹																	+ ¹			
	0.0(Min		·																				
	1.7893		1																	Ŷ			
CPI	cpi 1 recor1 recor	1	t ¹						~ •											+ ¹			
	0.2013	-		. فيتريم	المه يعفر	1 miles	ريبين	14 ju	_ أنظى ا	له الله ال	الرفع ل	-	لعن 12			المبلى الم		1.40	ويعرز و	-	_		2.4
	72686.				1. 1 ×	200	Sa:			1	ke :	11:	62	1	-		100	101 -	àsi :	Ť			
ILB misses	TIbMis:1 recor1 recor		+ ¹				語語	and the second				観察		和教育	and the second				题	+ 1			
	22.0(Mi					1	.H.		•		1			1	1. An								
	711350						-			 			in the second se	ija.						Ť			
Data ERAT misses	DeratM1 recor1 recor		† ¹	71	1997) 	2.00	.e1	4 <u>7</u>	1	and .	- 94	1/27	- stri	940		49.4	171	147		+1			
	2348.0	-									Control of)	ti	-		1	1 × 1	1				
	707672			2	*	*	12	and the second s			ale .		it	ala	1		*	1	器			and States and	
.oad L1 misses	LdMiss1 recor1 recor		• ¹	M.	纲	¥\$;	1	200		2	1	1	W.	1			•	×.	教	+ ¹			
	794.0()			-	-		- ÷	a	-				-		i.			-		-			3



Comparison of large page mapping categories shown in PE

🛅 Strip Array																							ೆ ರೆ ಡ
Tel Tel Tel Tel	oom al	Zoom out Z	oom in	< > Cen	ter																		
				- the	Land Land	موسد		-	ساريس	- Mark		i maga	h.s				رمسهريس	- magazie	-	-	· † ¹		
			1 - 11 - 14 - 14 - 14 - 14 - 14 - 14 -	,0G		0.0.0	16	0.30.30	91 (A. 94	12G	20.20.20		,3G	10.10.10	15 15 15	40)		,5G		4	1 ⁶⁰	
				en Ar an ar					40.00	1 12 2	-114	48814	5594152	607 (651)		i Harrisona			in in th			1 12.01	
	1.8		0G ¥			16				¹ 2G				3(3			40	÷			5G	
o large pages	срі	1 recor1 recor	1																				
lo la go pagoo	0.0			in the second	بعفرش فم	34 L.	والمستحجين	inter -	مد م	میلوده	وسنسية			معن المراج	س فندم م	_and-	Jane_	بير فساق	لسيصي فللجا	الم مرضوا	s-s_	المراجع عليه	ی العصول
	0.0 1.8		1																				
mDynamic	срі	1 recor1 recor	1																				
	0.0			Les . to	_		- المديمون			فيتقحص	<u>م</u> رس	····	-		<u>نے صبح</u>	ما ر المعن ور		منر م سر	فيتشريه	المع المريك	وليرقيهم	رف برم	
	0.0 1.8		1				• (etc)																
static	срі	1 recor1 recor	1						-				14	20				-		14			
	0.0		·	ines the	_	ر ک		in the second	<u>_่มม</u>	يستعس		~~	en l'and	بعد ال		للسجيعقر			ي ^و يتريكور	لععريث	14	والمستحدقة	م العمرز
	1.8		1																				
smDynamic + static	cpi	1 recor1 recor	1						-+														
	0.0				يعفرس	74. July	and the state		نەر 🖌	يستشر				میشد. ا	<u> </u>	لدجيعر			ی ^{عیری} تمدیکا		in the second	Martinet .	
	1.8		ş																				
gDynamic	cpi	1 recor1 recor	1 -											3						2	47		+ ¹
	0.0			L-of-	وسنسرة	شريكر فس	بالرقيق		مرحلرا	See .	للمؤسور	ini-	لمشرهب	رقبار_	المست	يسترقه	ليرفعني	فبخر_	- مناس	ونسام	برمينين. مراجعين	in the second	
	1.8		1														0.5						
gDynamic + static	cpi	1 recor1 recor	1			**:										a							<mark>↑</mark> 1
	0.0		~~ <u>~</u>	نې <u>مەر</u> ك		فلنالجرهم	ند منب ر	-	متريحين	the second	فبجسير		منشوهب		إسنيه	<u>سنين الم</u>	لىر ^{ىيىن} ەر	فيتجني			مرقبتهم	in the second	
	1.8		ŧ																				
gDynamic + smDy	срі	1 recor1 recor	12			ā., .,												. 4.		a		Ť	Ę.
	0.0		ليرجده	المتحريقين		and the second		بر النبي من السر السرام	جر <mark>مر</mark> ب	برلده		1. -	مر المرا لي الي		- ^ت سنرن	بمسيلوه	م لير من يج	ومستموه	in the	×	بشرقه بم	-	
	1.8		1																				
ill large pages	cpi	1 recor1 recor	1					40		4							-	+				+1	
	0.0		-	والسبخر		-		ب من م	جريا	إستسرع		i-i-	- المستند الم	1	ليجتنب	رم المر		برجيني	الم الم	Li-	منسخت		



Ξ.:	
	ł

CPO Implementation

- Extended K42's infrastructure
 - Events from a wider range of layers
 - •Extended notion of majors and minors to layers
 - Integrated HW performance counters
 - Self describing event definitions in XML
 - Extended to more than tracing, at each "event":
 - •Trace event
 - •Gather statistics on event, with tracing at threshold
 - •Call a handler for event
 - •All of the above

CPO Lessons

CPO

30

- + Vertical integration with HPCs powerful
- + Addition of statistics option good for online monitoring
- + Multiplexing hardware counters (ICS 05)
- No dynamic events
- No automatic packaging of trace and description files
- Visualization was valuable

	Survey of Street, or other	
 and party	-	
-		÷.

Outline

- SGI
 - rtmon
 - Kernel and Cray Unification
 - Lessons
- K42
 - Approach, scalability, and use
 - Lessons
- CPO (Continuous Program Optimization)
 - **PEM (Performance Environment Monitoring)**
 - Lessons
- CSO (Commercial Scale-Out)
 - Goals
 - Lessons
- Blue Gene / P

31

- Observations on Linux and LTT
- The "next system" concluding remarks

Introduction

- In scientific/technical computing, parallel processing became mainstream in the 80's
- Since the early 90's there has been a strong move of commercial computing away from single-processor machines to multi-processor systems, as the latter became more cost efficient
- Two different approaches to multiprocessors:
 - Scale-up: large shared-memory machines
 - Scale-out: clusters of interconnected smaller machines

Introduction

- In scientific/technical computing, parallel processing became mainstream in the 80's
- Since the early 90's there has been a strong move of commercial computing away from single-processor machines to multi-processor systems, as the latter became more cost efficient
- Two different approaches to multiprocessors:
 - Scale-up: large shared-memory machines
 - Scale-out: clusters of interconnected smaller machines



Scale-up

Scale-out

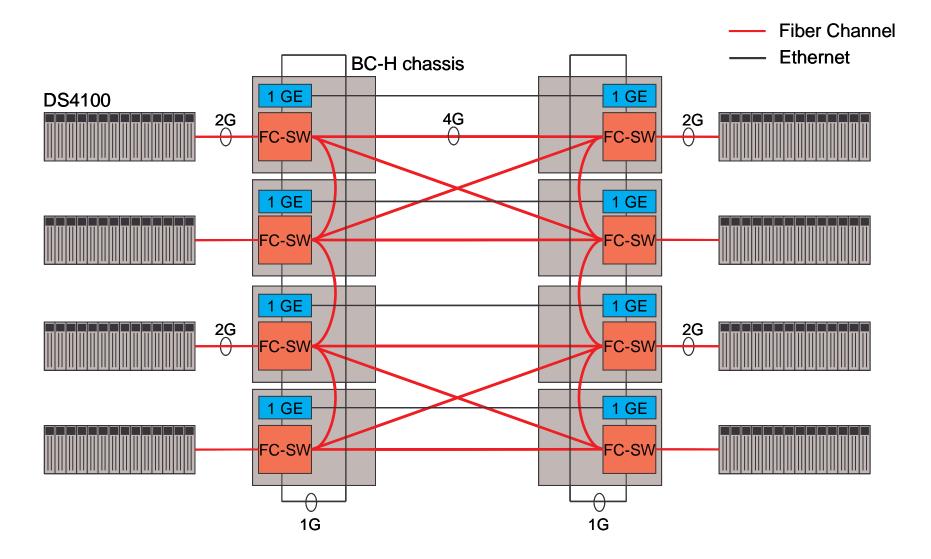




34



Commercial Scale Out experimental system



_			 -
		<u> </u>	
	-		

Understanding performance in commercial scale-out

Two challenges similar to scientific computing:

- Lots of processing elements \rightarrow lots of trace data: need techniques to limit data and identify important parts
- Correlate events from different machines → need synchronized time

Two challenges unique to commercial:

- Complexity of the software stack → hypervisor, operating system, Java, middleware, application
- Many threads of execution per processing element → multiple threads per process and multiple processes per processor – it is not unusual to see hundreds to thousands of threads per machine!

	-		$ \rightarrow =$
	-	-	
_	-	_	
=	_	_	

Starting point

Linux Trace Toolkit Next Generation (LTTng):

- Extracts information from hypervisor to application
- Requires instrumentation but it is uniform across layers
- Low overhead

Linux Trace Toolkit Viewer (LTTV):

- Merges data collected by each software layer
- Identifies the producer of each event (node, process, thread)
- Classifies the execution context (process, trap, interrupt, system call)

Enhancements to LTTng:

- PowerPC-specific instrumentation
- Tracing support for Java addition of *thread branding* (also LTTV)

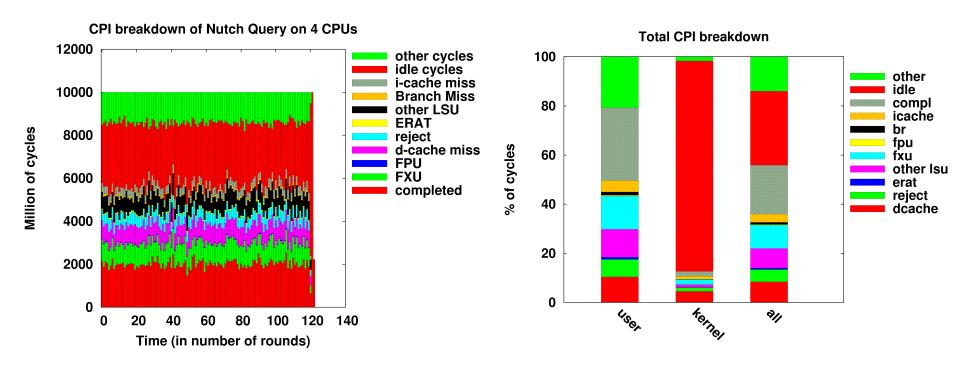
Performance monitoring facility

- Uses hardware performance counters in PowerPC
- Identified bottlenecks through statistical sampling

No. York	Linux Trace Toolkit Viewer	E
Image: Provide Control Image: Provide Contro <tth contro<="" image:="" provide="" th=""> Image: Provide</tth>		Help
Note Note Order Part Order Part		
Control Difference Difference <thdifference< th=""> Difference</thdifference<>		
Image:	Traceset	
Control No. <	8469s 743409507ns 8499s 243409507ns	8469s 743409507
	€ vertical	
Image: Internet		
Image: Internet		
Image: Internet		
Image: Internet		
races Brand Dir Droid Dir Dir <thdir< th=""> Dir Dir <thdir< td=""><td></td><td></td></thdir<></thdir<>		
races Brand Dia Dia <thdia<< td=""><td>i sebula in th<mark>a</mark> anay <mark>M</mark>abra, Inc. 1996 - Ana and an an</td><td></td></thdia<<>	i sebula in th <mark>a</mark> anay <mark>M</mark> abra, Inc. 1996 - Ana and an	
races Brand Dir Droid Dir Dir <thdir< th=""> Dir Dir <thdir< td=""><td></td><td></td></thdir<></thdir<>		
races Brand Dir Droid Dir Dir <thdir< th=""> Dir Dir <thdir< td=""><td></td><td>and the second</td></thdir<></thdir<>		and the second
unspander UNSPANDED 0		the states of the state
UNBRANDED 0		
add UNBRANDED 1 0 0 8468 78575247 0 signation/U UNRANDED 2 1 0 9468 78575394 0 stdhdg/U UNBRANDED 1 0 9468 78575497 0 stdhdg/U UNBRANDED 1 0 9468 78575987 0 stdhdg/U UNBRANDED 1 0 9468 78575867 0 stdhdg/U UNBRANDED 1 0 9468 78575867 0 stdthdg/U UNBRANDED 12 0 9468 78575867 0 stdtt UNBRANDED 12 0 9468 78575867 0 stdtt UNBRANDED 12 0 9468 78575843 0 stdttt UNBRANDED 12 0 8468 785772870 0 stdtttt 0 9776 8468 785778424 0 kerred prrikt<846875777876	Process Brand PID PPID CPU Birth nsec TRACE \$468s 743409507ms \$469s 243409507ms	8469s 743409507ns
igration 0 UNBRANDED 2 1 1 0 8468 785754822 0 stoching 0 UNBRANDED 3 1 0 8468 785754822 0 stoching 0 UNBRANDED 5 1 0 8468 78575698 0 invead UNBRANDED 5 1 0 8468 78575698 0 invead UNBRANDED 5 1 0 8468 78575698 0 invead UNBRANDED 5 1 1 27 0 8468 78576247 0 isockin 0 UNBRANDED 5 1 1 27 0 8468 78576247 0 isockin 0 UNBRANDED 5 1 1 27 0 8468 78576247 0 isockin 0 UNBRANDED 5 1 1 27 0 8468 78576247 0 isockin 0 UNBRANDED 5 1 1 27 0 8468 78576247 0 isockin 0 UNBRANDED 5 1 1 27 0 8468 78576247 0 isockin 0 UNBRANDED 5 1 1 27 0 8468 78576247 0 isockin 0 UNBRANDED 5 1 1 27 0 8468 78576247 0 isockin 0 UNBRANDED 5 1 1 27 0 8468 78576247 0 isockin 0 UNBRANDED 5 1 1 27 0 8468 78576247 0 isockin 0 UNBRANDED 5 1 1 1 27 0 8468 78576247 0 isockin 0 UNBRANDED 5 1 1 27 0 8468 78576247 0 isockin 0 UNBRANDED 5 1 1 1 27 0 8468 78576247 0 isockin 0 UNBRANDED 5 1 1 27 0 8468 78576247 0 isockin 0 UNBRANDED 5 1 1 27 0 8468 78576247 0 isockin 0 UNBRANDED 5 0 1 1 1 1 27 0 8468 78571476 0 core state dump field 8487.8571491 0 UNBRANDED 0, 0,0,0,5YSCALL { iso 0 core state dump field 8487.8571491 0 UNBRANDED, 0,0,0,0,5YSCALL { iso 0 core state dump field 848.78571396 (isochif 468.78571396 (isochif 468	swapper UNBRANDED 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	
WIBRANDED 3 1 0 846 78575482 0 satchdogi UNBRANDED 4 1 0 846 78575697 0 vertadio UNBRANDED 5 1 0 8468 78575697 0 vertadio UNBRANDED 1 0 8468 78575697 0 vertadio UNBRANDED 127 1 0 8468 78575697 0 vertadio UNBRANDED 127 0 8468 785776978 0 vertadio UNBRANDED 127 0 8468 785776927 0 vertadio UNBRANDED 127 0 8468 785776927 0 vertadio UNBRANDED 127 0 8468 785776927 0 vertadio UNBRANDED 128 127 0 8468 785776927 0 vertadio UNBRANDED 128 127 0 8468 785776927 0 vertadio vertadio vertadio vertadio vertadio vertadio	bash UNBRANDED 1 0 0 8468 785752447 0 🗙	
atthodogi UNBRANDED 4 1 0 8468 78375939 0 verta/0 UNBRANDED 5 1 0 8468 78375959 0 verta/0 UNBRANDED 27 1 0 8468 78375959 0 verta/0 UNBRANDED 137 0 8468 78376249 0 envade UNBRANDED 137 0 8468 78376220 0 envade UNBRANDED 137 0 8468 78376220 0 envade UNBRANDED 137 0 8468 78376220 0 envade UNBRANDED 137 0 8468 78371624 0 kernel prink: 8468/783716200 (rpu.0), 0, UNINAMED, UNBRANDED, 0, 0x0, 5VSCALL (p.e. 0xc00000000000000000000000000000000000		
versite UNBRANDED 5 1 0 8468 785756907 0 versite UNBRANDED 127 0 8468 785756907 0 versite UNBRANDED 127 1 0 8468 78575090 0 versite UNBRANDED 127 0 8468 78575090 0 versite UNBRANDED 120 127 0 8468 78575090 0 versite UNBRANDED 120 127 0 8468 78576225 0 versite		
heiper UNBRANDED 6 1 0 8468 78575950 0 htread UNBRANDED 127 1 0 8468 78575050 0 enwatch UNBRANDED 130 127 0 8468 78576020 0 shock UNBRANDED 130 127 0 8468 785762275 0 dfuld UNBRANDED 130 127 0 8468 785762275 0 dfuld UNBRANDED 200 127 0 8468 785762275 0 dfuld UNBRANDED 200 127 0 8468 78576247 0 sace Tracefile 0 Vrint k Facility Time (nz) PID Event Description mg/exp3test /cput 0 vrint k Karnel 8468 78571737 0 coreattag_dump_facility.load 8468.78571737767 0 coreattag_dump_facility.load 8468.7857174767 0 coreattag_dump_facility.load 8468.7857174767 0 coreattag_dump_facility.load 8468.7857174767 0 co		
hread UNBRANDED 12 1 0 8468 785758943 0 emwach UNBRANDED 130 127 0 8468 785760260 0 enbus UNBRANDED 135 127 0 8468 785761247 0 enbus UNBRANDED 135 127 0 8468 785763412 0 ace Tracefile CPUD Event Facefile 790 8468 785763412 0 ace Tracefile CPUD Event Facefile 790 8468 785763412 0 Ventel printic 8468/78571494 (rpu.0), 0, UNNAMED, UNBRANDED, 0, 00, SYSCALL { pe do::000000002028589 } mplexp3test /cpu vprintk kernel 8468 78571620 0 ventel printic 8468/785712976 (rocm/s/facilites, 0), 0, UNNAMED, UNBRANDED, 0, 00, SYSCALL { pe do::0000000022889 } mplexp3test /cpu vprintk kernel 8468 78571943 0 kernel printic 8468/78571291620 (ropu.0), 0, UNNAMED, UNBRANDED, 0, 00, SYSCALL { pe do::0000000021889 } mplexp3test /cpu vprintk kernel 8468 <t< td=""><td></td><td></td></t<>		
unstach UNBRANDED 130 127 0 8468 78376000 0 enbus UNBRANDED 131 127 0 8468 783761247 0 olockd/0 UNBRANDED 350 127 0 8468 783762225 0 ace Tracefile 0 8468 783763120 0 8468 783763120 0 ace Tracefile 0 VPUD Event Facility Time (s) Time (s) PD Event Description mplexp3test /control/facilities 0 vprintk Kernel 8468 785717876 0 kernel.printk: 8468.7857127876 (control/facilities, 0), 0, UNNAMED, UNBRANDED, 0, ox0, SYSCALL { loglevel = 7, text = { Dumping facility core }, ip = 0 mplexp3test /control/facilities 0 vprintk kernel 8468 78571976 0 kernel.printk: 8468.785712876 (control/facilities, 0), UNNAMED, UNBRANDED, 0, ox0, SYSCALL { loglevel = 7, text = { Dumping facility core }, ip = 0 mplexp3test /cpu 0 printk kernel 8468 785719902 0 kernel.printk: 8468.785722094 (control/facility, 0), UNNAMED, UNBRANDED, 0, ox0, SYSCALL { lo		
unbranue UNBRANDED 131 127 0 8468 785761247 0 alockd/0 UNBRANDED 135 127 0 8468 785762225 0 addud UNBRANDED 200 127 0 8468 78576225 0 addud UNBRANDED 200 127 0 8468 785761247 0 addud UNBRANDED 200 127 0 8468 785761247 0 addud UNBRANDED 200 127 0 8468 785714164 0 kernel printl: 8468.78571454 (cpu.0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218983 } mplexp3/test /cpu 0 vprintl: kernel 8468 78571476 0 kernel printl: 8468.785712620 (cpu.0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218983 } mplexp3/test /cpu 0 vprintl: kernel 8468 78571970 0 core.state_dump_facility_load 8468.78571260 (cpu.0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218983 } mplexp3/test /cpu vprintl: kernel printl: 8468.785719902 (/pu.0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218983 } mplexp3/t		
JUNBRANDED 135 127 0 8468 785762225 0 dflush UNBRANDED 200 127 0 8468 785763412 0 ace Tracefile CPUD Event Facility Time (n) PD Event Description mp/sxp3test (rpu 0 vrintk kernel 8468 785712454 0 kernel printk: 8468.785712400 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc000000000218988 } mp/sxp3test (rpu 0 vrintk kernel 8468 785712460 c kernel printk: 8468.785712400 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218988 } mp/sxp3test (rpu 0 vrintk kernel 8468 785712460 c cer.state_dump_facility_load: 8468.7857129707 (control/facilities, 0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/sxp3test /rpu 0 vrintk kernel 8468.785712902 kernel-printk: 8468.785712902 (/pu_0, 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip excc00000000218898 } mp/sxp3test /rpu 0 vrintk kernel 846		
Hubb UNBRANDED 200 127 0 8468 785763412 0 race Tracefile CPUID Event Facility Time (n) Time (n) PID Event Description mg/exp3/test /Cpuin 0 printk kernel 8468 78571495 0 kernel printk 8468.78571454 /(cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898} mg/exp3/test /Cpuin 0 state_dump_facility_load core 8468 785713910 0 kernel printk 8468.785713910 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility core }, ip = 0 mg/exp3/test /Cpuin 0 printk kernel 8468 78571990 0 kernel printk: 8468.785719413 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility core }, ip = 0 mg/exp3/test /Cpuin 0 printk kernel 8468 78571990 0 kernel printk: 8468.78571990 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility core }, ip = 0 mg/exp3/test /Cpuin 0 printk kernel 8468 78571990 0 kernel printk: 8468.785719900 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { l		
Tracefile CPUD Event Facility Time (n) Time (n) PiD Event Description mp/sxp3/test /cpu 0 printk kernel 8468 785712630 0 kernel printk: 8468.7857126200 (/cpu_0), 0.UNNAMED, UNBRANDED, 0.0x0, SYSCALL { log evel = 7, text = { Dumping facility core }, ip = 0 mp/sxp3/test /cpu 0 yrintk kernel 8468 78571276 0 kernel uprintk: 8468.7857126200 (/cpu_0), 0.UNNAMED, UNBRANDED, 0.0x0, SYSCALL { log evel = 7, text = { Dumping facility core }, ip = 0 mp/sxp3/test /cpu 0 printk kernel 8468 785712902 0 kernel uprintk: 8468.785719502 (/cpu_0), 0.UNNAMED, UNBRANDED, 0.0x0, SYSCALL { log evel = 7, text = { Dumping facility time + }, ip = mp/sxp3/test mp/sxp3/test /cpu 0 yrintk kernel 8468 785712902 0 kernel uprintk: 8468.785719902 (/cpu_0), 0.UNNAMED, UNBRANDED, 0.0x0, SYSCALL { log evel = 7, text = { Dumping facility time + }, ip = mp/sxp3/test /control/facilities 0 yrintk kernel 8468 78572049 0 core.state_dump_facility_load: 8468.785720949 (/control/facilities_0), 0.UNNAMED, UNBRANDED, 0.0x0, SYSCALL { log evel = 7, text = { Dumping facility socket }, ip = mp/sxp3/test /control/facilities 0 yrintk kernel vprintk:		
mp/exp3/test /cpu 0 printk kernel 8468 78571450 kernel printk: 8468.78571450 (cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /cpu 0 vprintk kernel 8468 785716200 (cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /cpu 0 printk kernel 8468 78571976 0 core 8468 78571976 0 core state_dump_facility_load: 8468.785717876 (/control/facilities_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /cpu 0 printk kernel 8468 785719902 0 kernel vprintk: 8468.785719902 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /cpu 0 vprintk kernel 8468 785719902 0 kernel vprintk: 8468.785719902 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /cpu 0 vprintk kernel 8468 785721980 0 core state_dump_facility_load: 8468.785729949 0 core.state_dump_facility_load: 8468.785729949 (/control/facilities_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /cpu 0 printk kernel 8468 785721368 0 kernel vprintk: 8468.785721368 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /cpu 0 vprintk kernel 8468 785721857 0 kernel vprintk: 8468.785721857 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /cpu 0 vprintk kernel 8468 785721857 0 kernel vprintk: 8468.785721857 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /cpu 0 vprintk kernel 8468 785721557 0 kernel vprintk: 8468.785721265 (/control/facilities_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /cpu 0 vprintk kernel 8468 78572520 (kernel vprintk: 8468.78572265 (/control/facilities_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /cpu 0 vprintk kernel 8468 785725698 (kernel vprintk: 8468.785725698 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /cp		
mp/exp3/test /cpu 0 vprintk kernel 8468 785712600 0 kernel.vprintk: 8468.785712600 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility core }, ip = 0 mp/exp3/test /cpu 0 printk kernel 8468 785712767 0 core.state_dump_facility_load: 8468.7857197676 (/control/facilities_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility core }, ip = 0 mp/exp3/test /cpu 0 printk kernel 8468 78571970 0 kernel.printk: 8468.785719413 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility timer }, ip = 0 mp/exp3/test /cpu 0 vprintk kernel 8468 785721943 0 kernel.printk: 8468.785719302 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility timer }, ip = 0 mp/exp3/test /cpu 0 printk kernel 8468 785721949 0 core.state_dump_facility_load: 8468.785721949 (/control/facilities_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility timer }, ip = 0 mp/exp3/test /cpu 0 printk kernel 8468 785721368 kernel.printk: 8468.785721368 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility socket }, ip = 0 mp/exp3/test /cpu 0 printk kernel 8468 785721368 kernel.printk: 8468.785721367 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility socket }, ip = 0 mp/exp3/test /cpu 0 printk kernel 8468 78		
mp/exp3/test /control/facilities 0 state_dump_facility_load core 8468 785717876 0 core.state_dump_facility_load: 8468.785717876 (/control/facilities_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { name = "core", checksump/exp3/test /cpu 0 printk kernel 8468 78571941 0 kernel printk: 8468.785719413 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /control/facilities 0 state_dump_facility_load core 8468 78571940 0 kernel.vprintk: 8468.785719402 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility timer }, ip = mp/exp3/test /control/facilities 0 state_dump_facility_load core 8468 78571368 0 kernel.vprintk: 8468.785720949 (/control/facilities_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility timer }, ip = mp/exp3/test /cpu 0 printk kernel 8468 785721368 0 kernel.vprintk: 8468.785721368 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility socket }, ip = mp/exp3/test /cpu 0 vprintk kernel 8468 785721857 0 kernel.vprintk: 8468.785721367 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility socket }, ip = mp/exp3/test /cpu 0 printk kernel 8468 785721857 0 kernel.vprintk: 8468.785722765 (/control/facilities_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility socket }, ip = mp/exp3/test /cpu 0 printk kernel 8468 78572550 0 kernel.vprintk: 8468.7857252100 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility socket }, ip = mp/exp3/test /cpu 0 printk kernel 8468 78572550 0 kernel.vprintk: 8468.78572550 (/control/facilities_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility merory }, ip mp/exp3/test /cpu 0 printk kernel 8468 78572550 0 kernel.vprintk: 8468.785725508 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility merory }, ip mp/exp3/test /cpu 0 vprintk kernel 8468 78572550 0 kernel.vprintk: 8468.7		facility core }, ip = 0
mp/exp3/test /cpu 0 printk kernel 8468 785719413 0 kernel.printk: 8468.785719403 (/cpu_0), 0. UNNAMED, UNBRANDED, 0. 0x0, SYSCALL { ip = 0xc0000000218898 } mp/exp3/test /cpu 0 vprintk kernel 8468 785719902 0 kernel.printk: 8468.785719902 (/cpu_0), 0. UNNAMED, UNBRANDED, 0. 0x0, SYSCALL { loglevel = 7, text = { Dumping facility imer }, ip = mo/exp3/test /cpu 0 printk kernel 8468 785721957 0 kernel.printk: 8468.785719902 (/cpu_0), 0. UNNAMED, UNBRANDED, 0. 0x0, SYSCALL { loglevel = 7, text = { Dumping facility imer }, ip = mo/exp3/test /cpu 0 printk kernel 8468 785721957 0 kernel.printk: 8468.7857219694 (/control/facilities_0), 0. UNNAMED, UNBRANDED, 0. 0x0, SYSCALL { ip = 0xc0000000218898 } mp/exp3/test /cpu 0 vprintk kernel 8468 785721857 0 kernel.printk: 8468.785721857 (/cpu_0), 0. UNNAMED, UNBRANDED, 0. 0x0, SYSCALL { loglevel = 7, text = { Dumping facility socket }, ip = mo/exc00000000218898 } mp/exp3/test /cpu 0 vprintk kernel 8468 785721857 0 kernel.vprintk: 8468.785721857 (/cpu_0), 0. UNNAMED, UNBRANDED, 0. 0x0, SYSCALL { loglevel = 7, text = { Dumping facility socket }, ip = mo/exc00000000218898 } mp/exp3/test /cpu 0 vprintk kernel 8468 78572157 0 kernel.vprintk: 8468.785721857 (/cpu_0), 0. UNNAMED, UNBRANDED, 0. 0x0, SYSCALL { loglevel = 7, text = { Dumping facility socket }, ip = mo/exc000000000218898 } mp/exp3/test /cpu 0 printk kernel 8468 78572510 kernel.vprintk: 8468.78572569 (/cpu_		
mp/exp3/test /control/facilities 0 state_dump_facility_load core 8468 785720949 0 core.state_dump_facility_load: 8468.785720949 (/control/facilities_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { name = "timer", checks mp/exp3/test /cpu 0 printk kernel 8468 785721368 0 kernel.printk: 8468.785721368 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /cpu 0 vprintk kernel 8468 785721857 0 kernel.printk: 8468.785721367 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility socket }, ip = mp/exp3/test /cpu 0 printk kernel 8468 785722765 0 core.state_dump_facility_load: 8468.785722765 (/control/facilities_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility socket }, check mp/exp3/test /cpu 0 printk kernel 8468 785725210 0 kernel.printk: 8468.785722765 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { in ame = "socket", check mp/exp3/test /cpu 0 printk kernel 8468 785725210 0 kernel.printk: 8468.785722765 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { in ame = "socket", check mp/exp3/test /cpu 0 vprintk kernel 8468 785725698 0 kernel.printk: 8468.785725098 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { in ame = "socket", check mp/exp3/test /cpu 0 vprintk kernel 8468 785725698 0 kernel.printk: 8468.785725698 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { in ame = "socket", check mp/exp3/test /cpu 0 vprintk kernel 8468 785725698 0 kernel.vprintk: 8468.785725698 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { inglevel = 7, text = { Dumping facility memory }, ip mp/exp3/test /cpu 0 vprintk kernel 8468 785725698 0 kernel.vprintk: 8468.785725698 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility memory }, ip memory }, ip memory }, ip memory kernel start: 8468 * s743409507 * ns end: 8469 * s743409507 * ns Time Interval: 1 * s 0 * Time Interval: 1 * s 0 * Time Interval: 1 * s 0 * Time		
mp/exp3/test /cpu 0 printk kernel 8468 785721368 0 kernel.printk: 8468.785721368 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /cpu 0 vpintk kernel 8468 785721357 0 kernel.vpintk: 8468.785721367 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility socket }, ip = mp/exp3/test /cpu 0 printk kernel 8468 785722765 0 core.state_dump_facility_load: 8468.785722765 (/control/facilities_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility socket }, check mp/exp3/test /cpu 0 printk kernel 8468 78572520 0 kernel.printk: 8468.7857252165 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { in = = "socket", check mp/exp3/test /cpu 0 vpintk kernel 8468 785725698 0 kernel.printk: 8468.785725698 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { in = 0xc00000000218898 } mp/exp3/test /cpu 0 vpintk kernel 8468 785725698 0 kernel.vprintk: 8468.785725698 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility memory }, ip mp/exp3/test /cpu 0 vpintk kernel 8469 * \$743409507 * ns Time Interval: 1 * s 0 * ns Current Time: 8468 * 5743409507 * ns Current Time: 8468 * 5743409507 * ns Time Interval: 1 * s 0 * ns Current Time: 8468 * 5743409507 * ns Current Time: 8468 * ns Current Time: 8468 * 5743409507 *	/tmp/exp3/test /cpu 0 vprintk kernel 8468 785719902 0 kernel.vprintk: 8468.785719902 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping t	facility timer }, ip =
mp/exp3/test /cpu 0 vprintk kernel 8468 785721857 0 kernel.vprintk: 8468.785721857 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility socket }, ip = mp/exp3/test /cpu 0 printk kernel 8468 785722765 0 core.state_dump_facility_load: 8468.785722765 (/control/facilities_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility socket }, ip = mp/exp3/test /cpu 0 printk kernel 8468 785722765 0 kernel.printk: 8468.785722765 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { in = "socket", check mp/exp3/test /cpu 0 vprintk kernel 8468 78572569 0 kernel.printk: 8468.785725010 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { in = 0 xc000000000218898 } mp/exp3/test /cpu 0 vprintk kernel 8468 78572569 0 kernel.vprintk: 8468.785725698 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility memory }, ip = mp/exp3/test /cpu 0 vprintk kernel 8468 785725698 0 kernel.vprintk: 8468.785725698 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility memory }, ip = mp/exp3/test /cpu 0 vprintk kernel 8469 * \$743409507 * ns end: 8469 * \$743409507 * ns Time Interval: 1 * \$0 * ns Current Time: 8468 * \$743409507 * ns Time Interval: 1 * \$0 * ns Current Time: 8468 * \$743409507 *	/tmp/exp3/test /control/facilities 0 state_dump_facility_load core 8468 785720949 0 core.state_dump_facility_load: 8468.785720949 (/control/facilities_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { name	ne = "timer", checks
mp/exp3/test /control/facilities 0 state_dump_facility_load core 8468 785722765 0 core.state_dump_facility_load: 8468.785722765 (/control/facilities_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { name = "socket", check mp/exp3/test /cpu 0 printk kernel 8468 78572510 kernel.printk: 8468.785725210 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /cpu 0 vprintk kernel 8468 785725698 kernel.vprintk: 8468.785725698 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /cpu 0 vprintk kernel 8468 785725698 kernel.vprintk: 8468.785725698 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility memory }, ip memory }, ip memory kernel start: 8468 * \$743409507 * ns end: 8469 * \$743409507 * ns Time Interval: 1 * s 0 * ns Current Time: 8468 * \$743409507 * ns Time Interval: 1 * s 0 * ns Current Time: 8468 * \$743409507 * ns Current Time: 8468 * \$743409507 * ns Time Interval: 1 * s 0 * ns Current Time: 8468 * \$743409507 * ns Current Time: 8468 * \$743409507 * ns Time Interval: 1 * s 0 * ns Current Time: 8468 * \$743409507 * ns Time Interval: 1 * s 0 * ns Current Time: 8468 * \$743409507 * n	/tmp/exp3/test /cpu 0 printk kernel 8468 785721368 0 kernel.printk: 8468.785721368 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 }	
mp/exp3/test /cpu 0 printk kernel 8468 785725210 kernel.printk: 8468.785725210 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /cpu 0 vprintk kernel 8468 785725698 kernel.printk: 8468.785725200 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { ip = 0xc00000000218898 } mp/exp3/test /cpu 0 vprintk 8468 785725698 kernel.printk: 8468.785725698 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility memory }, ip mp/exp3/test /cpu 0 vprintk 8468 785725698 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility memory }, ip me Frame start: 8468 \$ 743409507 ns ns me Interval: 1 s 0 ns Current Time: 8468 743409507 743409507	/tmp/exp3/test /cpu 0 vprintk kernel 8468 785721857 0 kernel.vprintk: 8468.785721857 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping the second	facility socket }, ip =
mp/exp3/test /cpu 0 vprintk kernel 8468 785725698 0 kernel.vprintk 8468.785725698 (/cpu_0), 0, UNNAMED, UNBRANDED, 0, 0x0, SYSCALL { loglevel = 7, text = { Dumping facility memory }, ip		ne = "socket", check
ne Frame start: 8468 + \$ 743409507 + ns end: 8469 + \$ 743409507 + ns Time Interval: 1 + s 0 + ns Current Time: 8468 + \$ 743409507 +		2 22 2002
ne Frame start: 8468 + s 743409507 + ns end: 8469 + s 743409507 + ns Time Interval: 1 + s 0 + ns Current Time: 8468 + s 743409507 + ns		
	Time Frame start: 8468 🛊 s 743409507 🗼 ns end: 8469 🛊 s 743409507 🎲 ns Time Interval: 1 🛊 s 0 📫 ns Current Time: 8468 🛊	

_			_
	_		
	-	_	
		_	
-			

Stall breakdown



- ~2 billion completing cycles/sec (20% of total 10 billion)
- 6 billion instructions/sec

Non-stall CPI (CPI_c): 0.34 Average for SPECcpu 2000: 0.35

Average bundle size: 3

39

_		
-		
	1.1.1	
		-

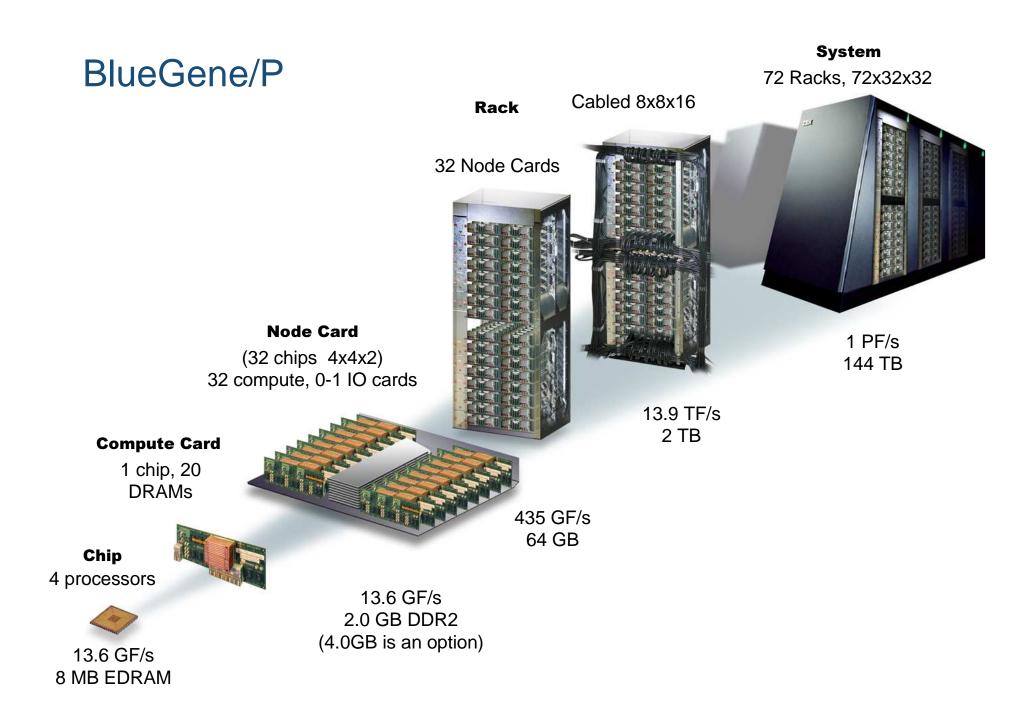
CSO Lessons

- CSO
 - + Tracing useful
 - + HPCs useful
 - Performance monitoring for distributed commercial workloads needs more work
 - •Handling many small, in terms of CPU usage, tasks
 - Automatic process branding
 - Inter-machine timer synchronization
 - Automatic idle determination
 - •Cross machine logical causality
 - Tree-based causality
 - •Selective aggregation of performance data
 - Virtualization

- E. STA 87	

Outline

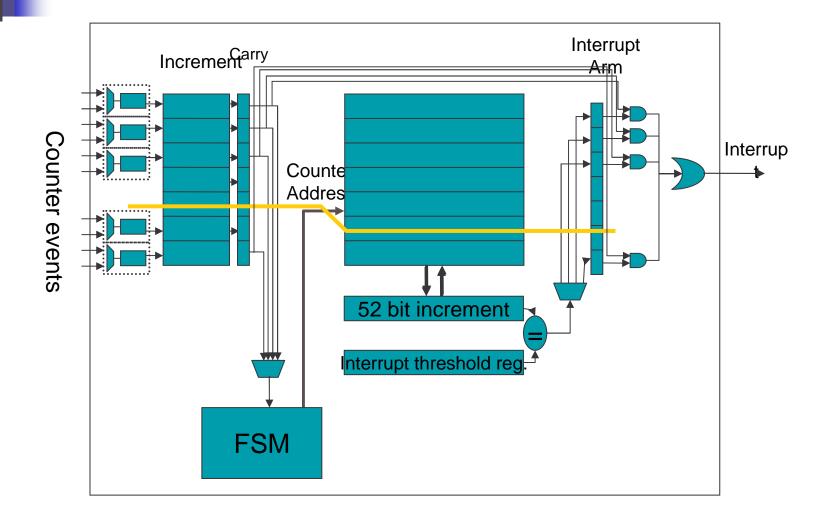
- SGI
 - rtmon
 - Kernel and Cray Unification
 - Lessons
- K42
 - Approach, scalability, and use
 - Lessons
- CPO (Continuous Program Optimization)
 - **PEM (Performance Environment Monitoring)**
 - Lessons
- CSO (Commercial Scale-Out)
 - Goals
 - Lessons
- Blue Gene / P
- Observations on Linux and LTT
- The "next system" concluding remarks



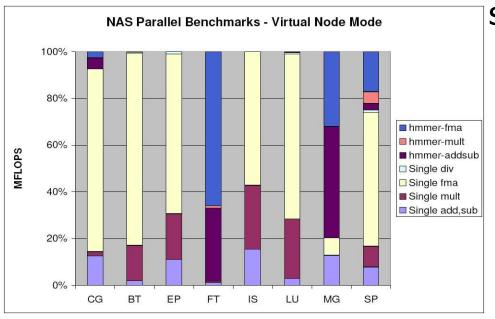
Performance monitoring unit in the Blue Gene/P system

- Implements 256 counters, 64bits wide
 - 1024 possible counter events
 - Monitors 4 processor cores and FPU, L3, L2, snoop filters, torus and collective network
- Novel architecture
 - Hybrid implementation using SRAM arrays
 - High density, high capacity on-chip performance monitor unit
- Hybrid architecture
 - 12 low order bits of a counter implemented using discrete logic
 - 52 high order bits stored in an SRAM array
 - SRAM state updated at a regular basis under state machine control
 - Configurable input selection and interrupt
 - Interrupt indication when the threshold value is reached

Hybrid PMU architecture



Usage of PMU in BGP



Breakdown of FP operations

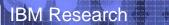
- Opens countless possibilities some usage examples
 - Analyze the execution profile for different compiler optimizations and infer their effectiveness
 - Conclude on the effectiveness of the various hardware & software settings to determine the optimal configuration
 - Profile and characterize workloads for various modes of operation to achieve maximum performance on multiple cores



Coreprocessor showing program counter on 4 racks

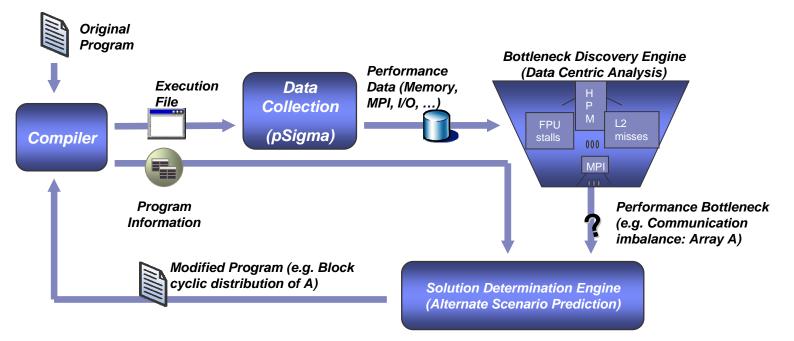
iroup Mode:	Stack Traceback (condensed)	Session 1 (CO	RE)	Common nodes:
:Compute Nod : Oxfffff	iffe (4096) iart_blrts (4096) main (4096) MPL_pdtest (4096) our_pdgev (4096) Parallel_LU_Factor (1) MPIDI_BGLTS_RectBcast (BGLML_Messager_CM_i Parallel_LU_Factor (127) MPIDI_BGLTS_RectBcast (BGLML_Messager_advi BGLML_Messager_advi Gover clip depth tl BGLML_Messager_advi Gover clip depth	(1) advance (1) (127) mce (1) Evessage (1) ast (4) preshold> (8) mce (113) th threshold> (53) _CM_advance (60) depth threshold> (60)		rts/bk/./core.1158
<u>الــــــــــــــــــــــــــــــــــــ</u>		P	A D	
orefile: /bglh oftware sig enerated by hile execut 00=0x300091 04=0x000000 08=0x007a32 12=0x400004 16=0x000000	home/shok/DRV340_2004-040817/ppc/src/bg ome/tgooding/HPLforBGL-xlrts/bk/./core.1153 mal0x00000009 (SIGK. 7 interrupt0x00000010 (soft: sing instruction at0x0015b608 bose registers: 118 r01=0x0feadf80 r02=0x1eeeeeeee r03= 118 r05=0x0000008 r06=0x007a3648 r07= 1280 r09=0x30000140 r10=0x007820bc r11= 142 r13=0x1eeeeee r14=0xffffffff r15= 1000 r17=0x00000000 r18=0x000000000 r19= 101 r21=0x00088de4 r22=0x00000000 r23=	B ILL - kill) ware interrupt) 0x00000000 0xb1155000 0x00270000 0xfffffff 0x00000000	icast.	.c:/94

46





Data and Control Flow of HPCS Toolkit



HPCS Toolkit provides Autonomic Application Performance Capability.

- Intelligent automation of performance evaluation and decision system
- Interactive capability with graphical/visual interface always available, but always optional

	-	2.2
音道		
<u>–</u>		

Outline

- SGI
 - rtmon
 - Kernel and Cray Unification
 - Lessons
- K42
 - Approach, scalability, and use
 - Lessons
- CPO (Continuous Program Optimization)
 - PEM (Performance Environment Monitoring)
 - Lessons
- CSO (Commercial Scale-Out)
 - Goals
 - Lessons
- Blue Gene / P
- Observations on Linux and LTT

-		-	
- EF			
	57	. S	
		10.0	5.4

What is right for Linux and How

- Patches versus dynamic points versus markers versus static
- One infrastructure versus many
- Get performance monitoring community active on lkml
- Get nose in tent

and a second	·3*	
minutes principling from the desire	<u>-</u>	<u> 38376</u>

Outline

- SGI
 - rtmon
 - Kernel and Cray Unification
 - Lessons
- K42
 - Approach, scalability, and use
 - Lessons
- CPO (Continuous Program Optimization)
 - PEM (Performance Environment Monitoring)
 - Lessons
- CSO (Commercial Scale-Out)
 - Goals
 - Lessons
- Blue Gene / P
- Observations on Linux and LTT



The Final Next System

- Efficient, Flexible tracing
 - Single unified space over all layers including HW counters
 - Use static events or event markers
 - Enable system to trace, gather stats, or callback at event
 - Allow additional dynamic events
 - Break into categories and allow dynamic enabling
 - Provide automatic tool for packaging up data and description
 - Timer synchronization built into infrastructure
 - Variable sized events
 - Non-locking and scalable gathering
 - Efficient online gathering for more extensive offline analysis
 - Negligible impact when disabled

The Final Next System

- Configurable visualization
 - Ability to add new graphs and have system save view
 - Pluggable modules to interpret application-specific events
 - Ability to handle massive (100G +) trace data
 - •Quick start up
 - •Summary and stats information on selectable portion
 - Handle multicore, multiprocessor, and distributed data
 - Handle real-time, scientific, and commercial data
 - Lots of interesting work left to understand commercial systems
 - Nice default views

•Time-centric time by process, thread-centric view, statistics, histogram, event list