

Intel® Xeon 5500 Platforms, Integrated Memory Controllers and NUMA

David Levinthal, Julia Fedorova, Dmitry Ryabtsev SSG/DPD/PAT

* Intel, the Intel logo, Intel Core and Core Inside are trademarks of Intel Corporation in the U.S. and other countries.



NUMA and Enabling: Overview Topology Overview BIOS Options OS dependent NUMA concerns Identifying memory locality (and lack thereof) on Intel® Xeon 5500 processors

Summary



Intel and core are a trademark or registered trademark of Intel Corporation or its subsidiaries in the United States or other countries

2 11/10/2009



DP Platform dominant validation vehicle

Intel® Xeon™ 5500 Platforms





3



NUMA, Quickpath and Intel® Xeon[™] 5500 Platforms

Quickpath Interfaces greatly increase memory bandwidth of our platforms

- **Integrated memory controllers on each socket** access dimms
- Quickpath interconnctions provide cache coherency
- Bandwidth improves by ~4X
- **Bandwidth improvement comes at a price**
- Non uniform memory access
- Latency to dimms on remote sockets is ~2X larger

Pealing away the Bandwidth layer reveals the NUMA Latency layer





NUMA Modes on DP Systems Controlled in BIOS Non Numa

- Even/Odd lines assigned to sockets 0/1 - Line interleaving
- NUMA mode
- First Half of memory space on socket 0
- Second half on socket 1
- Default on Intel® Xeon[™] 5500 Processors





5



NON-NUMA/NUMA Timings for Specomp* and NAS* Parallel Benchmarks



* Other names and brands may be claimed as the property of others.



11/10/2009

6



Non Uniform Memory Access and Parallel Execution

Process parallel is intrinsically NUMA friendly

- Affinity pinning maximizes local memory access
- MPI
- Parallel submission to batch queues
- Standard for HPC
- Shared memory threading is more problematic
- Explicit threading, TBB, openMP*
- NUMA friendly data decomposition (page based) has not been required
- OS scheduled thread migration can aggravate situation



* Other names and brands may be claimed as the property of others.



7



HPC Applications will see Large Performance Gains due to Bandwidth Improvements

A remaining performance bottleneck may be due to non uniform memory access latency

Intel® PTU data access profiling feature was designed to address NUMA

 Intel® Xeon[™] 5500 processors events were designed to provide the required data



8



Data Access Events on Intel® Xeon[™] 5500 processors Reveal NUMA Access Pattern

"miss" events are inclusive

 Sum over all data sources and their individual latencies

Intel® Xeon[™] 5500 processor Precise events are exclusive

Per data source



9



Data Access Events Reveal NUMA Access Pattern

💕 Help					
Hide Back Print Options					
Contents Index Search	MEM_UNCORE_RETIRED Event Code: 0x0F Mask: See in table below. Available counters: 0,1,2,3 Category: Precise Event Based Sampl Definition: Precise events further det	ling Performanc ailing load acce	e Tuning Events	s; uncore	
	Event Name Extension	Mask	Definition	Description	
INST_RETIRED.AN THREAD_ACTIVE Fort Execution Perf Fort End Performance Front End Performance TLB Performance Tun TLB Performance Tun L1 Data Cache Perform L2 Unified Cache Perfc	OTHER_CORE_L2_HITM	0x02	Load instructions retired that HIT modified data in sibling core (Precise Event)	Counts number of memory load instructions retired where the memory reference hit modified data in a sibling core residing on the same socket.	3
Offcore Performance T Offcore Performance T Precise Event Based \$ - ? About Precise Ever - ? BR_INST_RETIREL - ? BR_MISP_RETIRE - ? FP_ASSIST - ? INST_RETIRED - ? MEM_INST_RETIR	REMOTE_CACHE_LOCAL_HOME_HIT	0x08	Load instructions retired remote cache HIT data source (Precise Event)	Counts number of memory load instructions retired where the memory reference missed the L1, L2 and LLC caches and HIT in a remote socket's cache. Only counts locally homed lines.	
MEM_LUAD_KEII MEM_STORE_RET MEM_UNCORE_RET SSEX_UOPS_RET UOPS_RETIRED Snoop Performance Tu Uop Issue Performance Perfore for Nobelem baced	REMOTE_DRAM	0x10	Load instructions retired remote DRAM and remote home- remote cache HITM	Counts number of memory load instructions retired where the memory reference missed the L1, L2 and LLC caches and was remotely homed. This includes both DRAM access and HITM in a remote socket's cache for remotely homed lines.	





Controlling NUMA Data Locality on Linux* and Windows*

Linux* assigns physical pages on "first touch"

- ie buffer initialization not malloc
- If each thread initializes its data, things are good
- Can also use numactl or numalib

Windows assigns physical pages with "allocation"

- VirtualAlloc works like malloc on Linux*
 - Physical pages assigned at first use
- malloc & VirtualAllocExNuma allocation must be parallelized
 - Buffers are no longer contiguous linear address ranges
 - Much MUCH harder

11/10/2009



* Other names and brands may be claimed as the property of others.





Data Locality, Threaded Applications and Bandwidth

```
Consider a threaded triad
int triad(int len, double *a, double *b,
```

```
double *c, double *x);
```

```
int i,bytes = 24;
#pragma omp parallel
```

```
*pragina omp paraner {
```

```
#pragma omp for private (i)
#pragma vector nontemporal
for(i=0;i<len;i++)a[i]=b[i]+x*c[i];</pre>
```

```
}
return bytes
```

```
Parallelizes the work
function called 1000 times, len=8192000
~ 1B cachelines written NT, 2B read
```





Data Locality, Threaded Applications and Bandwidth

Run an OpenMP* triad under my usual mini_app driver, the resulting BW is only ~ 5bytes/cycle for 8 threads

Running in Non Numa Mode results in ~8.5 Bytes/cycle

Why?

Default Version Allocates Buffers on Thread 0 Using only one Memory Controller



* Other names and brands may be claimed as the property of others.





Performance Events and NUMA Sources

• Offcore_Response_0 8 flavors of Request Type X 8 flavors of \$line Source - + all combinations..

(~65K possible programmings)

 One "gotcha"... NT stores to local Dram appear to go to another core's cache (data source = 2 instead of 0x40)





PTU Display Shows Local and Remote Access for OpenMP Triad

V2 levinth-nhma:1 (lev	rinth)						
			Intel(R) Performance Tuning [[til	itry - NUI	MA triac	l adr. omn - Eclinse Platform	^ •
e Edit Navigate Pro	oject Run Wind	ow Help		ity - noi	ana_entere	Tau Toub - Feiber Lanoun	
3- 🗋 🗁 │ 🚱 - │	Ç. – –						E ,
NUMA_triad_omp_Inx	NUMA_tria	id_adr_omp	23				- <i>s</i>
unction	Module	CPU_C	OFFCORE_RESPONSE_0.ANY_REQUEST.ANY	OF 🔻	OFF	OFFCORE_RESPONSE_0.ANY_DATA.REMOTE_DRAM	OFFCORE_RESPONSE_0.ANY_DATA.LOCAL_DRAM
TRIAD	triad_adr_o	111,561	31,269	15,539	15,628	10,254	10,23
<unknown(s)></unknown(s)>	vmlinux-2.6	4,260	555	69	458	27	5
kmp_wait_sleep	libguide.so	21,169	7	1	2	1	
kmp_x86_pause	libguide.so	13,311	5	0	1	0	
Limit 95% 🔻 Grar	Function	▼ Pro	cess All Thread All		-	Module: All Cpu Total	▼
□◆							
😻 🔲 levinth@levi	nth-nhma:~		📔 levinth@levinth-nhma:~,	/mini_app	/triad_orr	p 📄 Intel(R) Performance Tuning Util	ity - NUMA triad adr omp



15 11/10/2009



Need to Distribute "Allocation" "Allocate" on First Touch

Other names and brands may be claimed as the property of others.

Original allocation

Initialization must also be done in Parallel

(intel) Software

16 11/10/2009

}



Parallel "Allocation" for Linux* Requires Parallel Initialization

```
Parallel allocation
      buf1 = (char *) malloc(DIM*(sizeof (double))+1024);
      buf2 = (char *) malloc(DIM*(sizeof (double))+1024);
      buf3 = (char *) malloc(DIM*(sizeof (double))+1024);
      a = (double *) buf1;
      b = (double *) buf2;
      c = (double *) buf3;
#pragma omp parallel
#pragma omp for private(num)
      for(num=0;num<len;num++)</pre>
      {
             a[num]=10.;
             b[num]=10.;
             c[num]=10.;
      }
}
```

* Other names and brands may be claimed as the property of others.





inte

Event	Triad_omp	
CPU_CLK_UNHALTED.THREAD	2.23E+11	
CPU_CLK_UNHALTED.THREAD;Socket 0	7.51E+10	
CPU_CLK_UNHALTED.THREAD;Socket 1	1.48E+11	
OFFCORE_RESPONSE_0.ANY_REQUEST.ANY_LOCATION	3.13E+09	
OFFCORE_RESPONSE_0.ANY_REQUEST.ANY_LOCATION;Socket 0	1.56E+09	
OFFCORE_RESPONSE_0.ANY_REQUEST.ANY_LOCATION;Socket 1	1.56E+09	
OFFCORE_RESPONSE_0.ANY_REQUEST.LOCAL_CACHE_DRAM	1.56E+09	
OFFCORE_RESPONSE_0.ANY_REQUEST.LOCAL_CACHE_DRAM; Socket 0	1.55E+09	
OFFCORE_RESPONSE_0.ANY_REQUEST.LOCAL_CACHE_DRAM; Socket 1	8000000	
OFFCORE_RESPONSE_0.ANY_REQUEST.REMOTE_DRAM	1.55E+09	
OFFCORE_RESPONSE_0.ANY_REQUEST.REMOTE_DRAM;Socket 0	1.55E+09	
OFFCORE_RESPONSE_0.ANY_REQUEST.REMOTE_DRAM;Socket 1	100000	



18 11/10/2009

Note socket 0/1 switch between PTU runs Software and Solutions Group 2008 Software Technology Open Forum



Event	Triad_omp	Triad_NUMA
CPU_CLK_UNHALTED.THREAD	2.23E+11	1.17E+11
CPU_CLK_UNHALTED.THREAD;Socket 0	7.51E+10	5.84E+10
CPU_CLK_UNHALTED.THREAD;Socket 1	1.48E+11	5.83E+10
OFFCORE_RESPONSE_0.ANY_REQUEST.ANY_LOCATION	3.13E+09	3.11E+09
OFFCORE_RESPONSE_0.ANY_REQUEST.ANY_LOCATION;Socket 0	1.56E+09	1.56E+09
OFFCORE RESPONSE 0.ANY REQUEST.ANY LOCATION;Socket 1	1.56E+09	1.55E+09
OFFCORE RESPONSE 0.ANY REQUEST.LOCAL CACHE DRAM	1.56E+09	3.11E+09
OFFCORE_RESPONSE_0.ANY_REQUEST.LOCAL_CACHE_DRAM;	1 55E±09	1 55E±09
OFFCORE_RESPONSE_0.ANY_REQUEST.LOCAL_CACHE_DRAM; Socket 1	800000	1.55E+09
OFFCORE_RESPONSE_0.ANY_REQUEST.REMOTE_DRAM	1.55E+09	400000
OFFCORE_RESPONSE_0.ANY_REQUEST.REMOTE_DRAM;Socket 0	1.55E+09	300000
OFFCORE_RESPONSE_0.ANY_REQUEST.REMOTE_DRAM;Socket 1	100000	100000



19

5.1 B/cyc vs 8.5 B/cyc vs 12.5 B/cyc on a poorly tuned machine 2008 Software Technology Open Forum



OpenMP and Core Affinity Pinning

Export KMP_AFFINITY=compact,0,verbose will pin affinity of threads

Just not reproducibly (per socket) on Red Hat 5.1 from run to run

Causing problems in multi run PTU collections

Problem is that an app does not use OMP runtime libs to pin affinity until there is a #pragma parallel {}

You must add this around first instruction to pin affinity of Main thread







Multi-thread Scaling and NUMA

When measuring scaling between 4 and 8 threads (assuming no SMT) the affinity of the 4 threads matters

4 threads all on one socket has the same LLC cache size/core as 8 threads

BUT

2 threads/socket has closer to the same memory BW as the 8 thread run

Thus 4->8 scaling will always have a non scaling contribution due to one of these 2 effects







Per Socket Display + Data Source events Show NUMA / Cross Socket Traffic

VI levinth-nhma:1 (levinth)										
▾▯◬▯◙₌▾▯⇔▾⇔▾								E		**
Tuning Navigator 🕱 🛛 🗖 🗖	2008-04	-28-11-08-58 🛚 📑	2008-04-28-11-17-04	2008-04-28-11	-09 [2008-04-2	28-11-30-20	4 [»] 2		
□ 🔄 🏹	Function	Module	MEM_UNCORE_RETIRE	D.LOCAL_DRAM	MEM_U	MEM_U	MEM_UNCORE_RETIRED.REMOTE_DRA	M MEM_UN	MEM_U	
🗞 CG	TRIAD	gather fmal6 omp		18,866	9,44	9,426	11	3,986 9,472	2 9,514	
😪 gather_numa		· ·		······						
109-36 2008-04-28-11-09-36										
2008-04-28-11-30-20										
1008-04-28-11-32-54										
😪 gather_omp										
108-58 2008-04-28-11-08										
2008-04-28-11-17-04	Limit 95	% 🔻 Granularity Fu	nction Process	All	Threa		Module: All	Cpu	Per Socket	-
🗞 OMP_triad										
🚓 sp.B	Experim	ient Summary 🛿 🔾 Co	nsole Cachelines View	Memory Access an	d Latency H	listograms				
	Applicatio Work Dire Processo Frequenc L2 Cache L3 Cache	on: /home/levinth ectory: /home/levinth r: Genuine Intel(:y: 2.67 GHz e Size: 256 KBytes e Size: 8192 KBytes	/mini_app/gather_omp/m /mini_app/gather_omp R) CPU @ 0000 @	un_omp.sh Event MEM 2.67GHz MEM	Based Sa	mpling RETIRED.REM RETIRED.LOO	MOTE_DRAM_AND_REM, HOME_CACHE_	38.87 HITM 19.24 219 s HE_HITM 19.19 214 s	4 samples 7 samples x amples x 10 4 samples x amples x 10	100(000 = 100(000 =
]\$										
levinth@levinth-nhma:~	Ievinth@I	evinth-nhma: ~/	levinth@levinth-nhma:~,	/) 🞯 [levinth]		🔳 levi	inth@levinth-nhma:~/) 🥃 Intel(R) Pe	rformance Tun		



22 11/10/2009





Default Initialization Breaks Array into 8 Contiguous Pieces→ 50% Non Local Access

levinth-nhma:1 (levinth)	Intel(R)) Performance Tu	ning Utility - 2008-	04-27-16-08-17 (2) - Eclipse Pla	atform		
igate <u>P</u> roject <u>R</u> un <u>W</u>	indow <u>H</u> elp)	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,					
9 ∎•							E\$	
gator 🛿 🗖 🗖	2008-04	-27-16-07-53	2008-04-27-16-08-17	2008-04-27	12-33	2) 23		
	Function	Module					M AND REM HOME CAC	HE HITM
	TRIAD	gather_fma16_om	ip	387,180,000	193,330,000		19	93,850,000
am								
04-27-15-54-44								
04-27-15-55-56								
04-27-15-56-55								
04-27-16-08-17								_
04-27-17-12-33	Limit 95	% 💌 Granularity F	Function Pro	all All	Thread All	Module: All	▼ Cpu Total ▼	·
d	Experim	ent Summary X	Console Cachelines	View Memory Access	and Latency Histograms			- 6
04-26-09-42-00		•						
04-26-09-49-31	Applicatio	on: /home/levintl	h/mini_app/gather_d	mp/run_omp.sh Ev	ent Based Sampling		78,884 samples	
04-26-09-59-51	Work Dire	ectory: /home/levintl	h/mini_app/gather_c	mp i	MEM_UNCORE_RETIRED.REMOTE_DR.	AM_AND_REM_HOME_CACHE_HITM	19,450 samples x 1000	0 = 194,500,0
04-26-10-04-21	Processo	or: Genuine Inte	I(R) CPU @ 00	00 @ 2.67GHz			212 samples x 10000 =	2,120,000 e
04-26-12-11-01	Frequenc	y: 2.67 GHz		'	MEM_UNCORE_RETIRED.LOCAL_DRA	M_AND_LOC_HOME_REM_CACHE_HITM	218 samples x 1000	0 = 194,620,0
04-26-16-01-18	L2 Cache	e Size: 256 KBytes		,	IEM LOAD RETIRED.LLC MISS		38.924 samples x 1000	0 = 389.240.0
04-27-16-07-53	L3 Cache	e Size: 8192 KBytes					618 samples x 10000 =	6,180,000 e
_triad_adr_omp								
_triad_omp_lnx								
	4							•
						1		
	la ma a c	🔳 lewipth@lewir	nth-nhma ~/mini_a	Ievinth@levinth	nhma:/mini_a	ath phma: -/mini_a	The second se	



24 11/10/2009



Address Histogram for all Dram Accesses

Valevinth-nhma:1 (levinth)												
┍ 🛛 👌 💁 🔶 - ↔-										E	• • •	» 1
Tuning Navigator 🕱 🗖 🗖	2008-04-28-11	-08-58	08-04-28-11-09-36	5 🚺 20	008-04-28	-11-08-58	(2) 🕱					e e
	Function	Module	Ue »	0e »	LC >>	RC >>	Local DRAM	RemRAM	MEM_UHITM	MEM_UNHITM		1
a cg	TRIAD	gather_fma16	_omp 0 (N/A)	0 (N/A)	0 (N/A)	0 (N/A)	18,866 (97.5%) 18,986 (97.8%) 18,866 (97.5%)) 18,986 (97.8%)		
😪 gather numa	main	gather_fma16	_omp 0 (N/A)	0 (N/A)	0 (N/A)	0 (N/A)	307 (1.6%) 253 (1.3%) 307 (1.6%)) 253 (1.3%)		
2008-04-28-11-09-36	<unknown(s)></unknown(s)>	vmlinux-2.6.1	8-53.el5 0 (N/A)	0 (N/A)	0 (N/A)	0 (N/A)	186 (1.0%	b) 184 (0.9%) 186 (1.0%)) 184 (0.9%)		
😪 gather_omp												
2008-04-28-11-08-58	Total Select		0 (N/A)	0 (N/A)	0 (N/A)	0 (N/A)	18,866 (97.5%) 18,986 (97.8%) 18,866 (97.5%)) 18,986 (97.8%)		
🗞 OMP_triad	Granularity Fund	tion 🔻 Proce	ss gather_fma16	omp 🔻	Thread	All	▼ Mo	dule All	▼ Filt	er by selection 🛛 🗨	🔿 🗙	
🚓 sp.B												5
🚓 triad	Experiment Summ	ary Console Ca	achelines View	Memory A		d Latency	Histograms 🕱		Memory Access Bir	n Size: 4 MByte	<u> </u>	1
	2008-04-28-11-08	-58 (2)										π
	Description	1							1			
	Low VA displa		hl d	1	1.		1					
	High VA displa	لاال الارتار		يا ال	ւ հետ Մ				ան շերեր	n 1 n		
	Number of hin	11. I.I.I.I.I.I.I.I.I.I.I.I.I	hiller the state	lik Ithia	la Malilaina		lah, belah belah belah b		adhair i dhinin. Na	h a dhatar an dhatar		
	Rin Size					فالظرافان			Malaki Kuri J. Di Laiki . Jak	ALL		
	Filter	udindi i di ud		" • • •	101		ana Miri I. A	t de el 19. falletelle		n di si Manana ana		
	riter					W ' '		i an	and a state of the second s			
			- ' - I									
J ◆									▶ _			
😵 🔲 levinth@levinth-nhma:~	Ievinth@levinth-	nhma:~/ 🔲 🗉 I	evinth@levinth-nhi	ma:~/	🗋 [levint	h]		evinth@levinth-nhr	na:~/ 😂 Intel(R) Performance Tun.	.)	
<					1111							> .:



25 11/10/2009



Filtering to a Single Thread Displays the Data Decomposition

🚾 levinth-nhma:1 (levinth)										
┍╸┇┢╽┇╸│ҫ╸╺⇒									Ē	»
Tuning Navigator 🕱 🛛 🗖 🗖	2008-04-28-11	-08-58	-28-11-09-36	2008-04-28-11-08-58	3 (2) 🔀					
	Function	Module	Ue >> Oe >	→ LC → RC →	Local DRAM 🔉	RemRAM >>	MEM_UHITM	MEM_UHITM		
🚓 cg	TRIAD	gather_fma16_omp	0 (N/A) 0 (N/A) O (N/A) O (N/A)	2,359 (99.8%)	2,368 (98.2%)	2,359 (99.8%)	2,368 (98.2%)		
😪 gather_numa	<unknown(s)></unknown(s)>	vmlinux-2.6.18-53.0	e15 O (N/A) O (N/A	(N/A) 0 (N/A)	5 (0.2%)	44 (1.8%)	5 (0.2%)	44 (1.8%)		
2008-04-28-11-09-36										
😪 gather_omp										
2008-04-28-11-08-58	Total Select									
🗞 OMP_triad	Granularity Fund	tion 🔻 Process ga	ther_fma16_omp	Thread 000039	b(0003) 🔻 Mo	dule All		Filter by selection	X 1	
😪 sp.B										
🐟 triad	Experiment Summ	nary Console Cachelin	es View 🔝 Memory	Access and Latency	Histograms 🔀		Memory Access	Bin Size: 4 MByte		
	2008-04-28-11-08	-58 (2)								
	Description	1				1	1			
	Low VA displa					O Camples E 2		1		
	High VA displa			1. htm:// 10. h		Jo Samples:55				
	Max Referenc	hi da i cui di la fica	la a <mark>n I</mark> . I	Louisball, Libre,	i. Itilla balada	. 	ին հերաներին հերաներ	- Illah, - Itt hills and	dad . I tite Lind	
	Number of bin		all all control on the second				a li <mark>ka s</mark> ili Juni li tuk ali tu		Jaco J <mark>e</mark> nista al Culture -	1
	Bin Size	1919 - 1	, <mark>1</mark>		N Hand M H h H , i t a h i	illi di di di di di di di di	₽₽ <mark>↓</mark> [™] ₩1₩₽₽, ₽11↓	1. 11. <mark>1.</mark> 1	PU <mark>P</mark> U PUP	
	Filter	i de la conde	ן ווייז יון זן יו	ןיןן אן יין			ווין יועוי א	יון די וי		
	Low VA displa	. <mark>.</mark>							- ' <u>'</u> ' '	
	High VA displa									
	Number of bin									
*										
🕢 🔲 levinth@levinth-nhma:~	levinth@levinth-	nhma:~/ 🔲 Jevinth	@levinth-nhma:~/	[] [levinth]	le le	evinth@levinth-nh	maː~/	el(B) Performance	Tup	
								en, en on hance		



26 11/10/2009



A Different Thread

Ve levinth-nhma:1 (levinth)															
▾▯◬▯◙₌▾▯⇔▾⇔▾													E		»
Tuning Navigator 🛿 🗖 🗖	2008-04-28-11	L-08-58	2008-04-28	-11-09-36	20	08-04-28-	11-08-58	(2) 🔀							
	Function	Module		Ue »	0e »	LC »	RC >>	Local DRAM	» Rem	RAM »	MEM_UHITM	MEM_UHITM			
🚓 cg	TRIAD	gather_f	ma16_omp	0 (N/A)	0 (N/A)	0 (N/A)	0 (N/A)	2,363 (99.9	9%) 2,36	59 (98.2%)	2,363 (99.9%) 2,369 (98.2%)			
😪 gather_numa	<unknown(s)></unknown(s)>	vmlinux-:	2.6.18-53.el5	0 (N/A)	0 (N/A)	0 (N/A)	0 (N/A)	2 (0.1	1%)	43 (1.8%)) 2 (0.1%	6) 43 (1.8%)			_
109-36 2008-04-28-11-09-36															
🗞 gather_omp															_
2008-04-28-11-08-58	Total Select														
😪 OMP_triad	Granularity Fund	ction 💌	Process gathe	er_fma16_o	mp 💌	Thread	000039b	9(0008) 💌	Module	All	-	Filter by selection	י 🕄 者	×	
🗞 sp.B				vr(te ··	k .				~			- Din Gine 4 MDrd			
🗞 triad	Experiment Summ	nary Conso	Cachelines	view III. M	embry A	ccess and	Latency	Histograms 2	25		Memory Acce	ss Bin Size: 4 MBy		ब द –	
	2008-04-28-11-08	3-58 (2)													
	Leve VA display	.				1									
	Low VA displa		di lil												
	Max Referenc		d			dina 1				1011		l kunt li dal a			
	Number of hin	di da bada da b	n hi i i li <mark>n</mark> adi.	llite i dadite	h dhad	, de Califatione		, , , , , , , , , , , , , , , , , , ,		, e n lin l	, 1997 , 1997, 1997, 1997, 1997, 1997, 1997, 1997, 1997, 1997, 1997, 1997, 1997, 1997, 1997, 1997, 1997, 1997, 1	lis baller all diad de		Juliu	
	Bin Size		del de l'hel. <mark>L'</mark> hen.		17.61	11''''''''	J. d. <mark>1.</mark> del del			fillia a d		1 <mark>.</mark> 4194.000	l <mark>e</mark> se en el		
	Filter	T I I I I I I I I I I I I I I I I I I I	'' '''' 	וןין אף יו		יווי וי	ha fill.	an a biri i	.	" ' 	1 r i 1 r i 1	b de de la Constante de la C	1 h (11.1 d 1		
	Low VA displa				t <mark>i</mark> l		III	I	1	· · ·	I.			U 1	
	High VA displa														
	Number of bir														
]*															
😻 🔲 levinth@levinth-nhma:~	Ievinth@levinth-	nhma:~/	evinth@le	evinth-nhm	a:~/	🔞 [levinth	1]		Ievinth	@levinth-n	hma:~/ 🗦 li	ntel(R) Performance	Tun		
	and the second														>



27 11/10/2009



Using Only Precise Remote Dram Event Only Half the entries shown Gaps due to lack of events are suppressed

V2 levinth-nhma:1 (levinth)										
▾ ▯ ◬ ◙;▾ ⇔▾ ⇔▾								E	**	
Tuning Navigator 🛿 🗖 🗖 🔀 2008-0	04-28-11-08-58	2008-04-28-11-09-36	2008-04-28-	11-08-58 (2)	2008-04-28-11-	17-04 🗙			- 8	L
E 🔄 🗸 Function	Module	Ue »	0e » LC »	RC » LM »	RemRAM >>	MEM_UHITM				L
CG TRIAD	gather_fm	a16_omp 0 (N/A)	D (N/A) O (N/A)	0 (N/A) 0 (N/A)	2,480 (99.0%)	2,480 (99.0%)				L
😪 gather_numa <unkno< td=""><td>wn(s)> vmlinux-2.</td><td>5.18-53.el5 0 (N/A)</td><td>0 (N/A) 0 (N/A)</td><td>0 (N/A) 0 (N/A)</td><td>26 (1.0%)</td><td>26 (1.0%)</td><td></td><td></td><td></td><td></td></unkno<>	wn(s)> vmlinux-2.	5.18-53.el5 0 (N/A)	0 (N/A) 0 (N/A)	0 (N/A) 0 (N/A)	26 (1.0%)	26 (1.0%)				
2008-04-28-11-09-36										
😪 gather_omp										
2008-04-28-11-08-58 Total Se	lect									
🔓 2008-04-28-11-17-04 Granular	ity Function 🔻 Pr	ocess gather_fma16_o	mp 🔻 Thread	00003bcd(0003)	▼ Module All		▼ Filter by selectio	n 🔩 🔒	×	
🗞 OMP_triad										
sp.B Experime	nt Summary Console	Cachelines View 🛄 M	emory Access and	Latency Histogram	ns 🖾	Memory A	ccess Bin Size: 4 MBy	te 🗸 🤇	Q Q - D	
🗞 triad	28-11-17-04			N N						
Descripti	on									
Low VA o	lispla		alter th							
High VA	displa	d								
Max Refe	erenc	n	dia M. U.A Librar							
Number	ofbin	' 1 1 1 1 1 1 1 1 1 1 1		JAN IN <mark>N</mark> IN						
Bin Size				l la						
Filter										
Low VA o	lispla									
High VA	displa									
Number	of bin									
*						•	1][
L										
levinth@levinth_nhma:~	Nevinth-nhma:~/		a:~/			winth-phma:~/	Intol(R) Portomana	Tun		
										>



28 11/10/2009



Using Only Precise Remote Dram Event Only Half the entries shown Gaps due to lack of events are suppressed

VE levinth-nhma:1 (levinth)	
▼ □ △] Q ▼] Φ ▼ ⇔ ▼ E	
Tuning Navigator 🛿 🗧 🗖 🔀 2008-04-28-11-08-58 📓 2008-04-28-11-09-36 📓 2008-04-28-11-08-58 (2)	
E 🔄 🗸 Fun Module Ue » Oe » LC » RC » RDRAM » MEM_UNHITM	١.
CG TRIAD gather_fma16_omp 0 (N/A) 0 (N/A) 0 (N/A) 2,486 (100.0%) 2,486 (100.0%)	
🗞 gather_numa	
2 2008-04-28-11-09-36	
🗞 gather_omp	
2008-04-29-11-08-58 Tot	
🖺 2008-04-28-11-17-04 Granularity Function 💌 Process gather_fma16_omp 🔍 Thread (00003bd2(0013) 🔍 Module All 💽 Filter by selection 🕄 😭 🗶	
🗞 OMP_triad	
Agesp.B. 2008-04-29-11-17-04	
A triad	
Low VA displa	
Max Reference and a second built and the second secon	
Bin Size Dx2AAB9C400000 Samples:1	
Filter	
Low VA displa	
High VA displa	
Number of bir	
	>



29 11/10/2009



Change Initialization to Follow Work Access Pattern

Thread initialization with same access sequence as work

Expect ~33% improvement

- 1/2 of accesses get lower latency by 2
- Simple OMP ran in 14.3 cycles/cell
- NUMA initialized version ran in 11.2 cycles/cell

Every access has serious DTLB issues, which don't change with the improved NUMA layout







Sampling View for Correctly Initialized Array has no Remote Access

V2 levinth-nhma:1 (levinth)					
▾▯◬▯◙▾▯▻▾⇔▾					🖻 ×
Tuning Navigator 😫 📃 🔲	2008-04-28-11-08-58	2008-04-28-11-09	-36 🕱		
_ € ▼ F	Function	Module	MEM_UNCORE_RETIRED.LOCAL_DRAM		_DRAM
🚓 cg	TRIAD	gather_fma16_numa	39,3	77	17
🚓 gather_numa	<unknown(s)></unknown(s)>	vmlinux-2.6.18-53.el5		51	258
2008-04-28-11-09-36	main	gather_fma16_numa		29	119
😪 gather_omp					
2008-04-28-11-08-58					
😪 OMP_triad	Limit 95% 🔻 Gran	ularity Function 🔻 P	rocess All Thread	All Module: All	▼ Cpu Total ▼
😪 sp.B		·			
🚓 triad Ex	xperiment Summary 토	Console 🕄 Cachelin	es View Memory Access and Latency Hist	ograms	
<	terminated> dram [Intel	(R) PTU] vtsarun /gather_i	numa/2008-04-28-11-09-36 -s -dl -ec MEM	UNCORE_RETIRED.LOCAL_DRAM_AN	ID_LOC_HOME_REM_CACHE_HITM:sa=10000,M
	MP_AFFINITY: Interna MP_AFFINITY: Interna	al thread 3 bound to al thread 4 bound to	US proc set {b} OS proc set {1}		
KI	MP_AFFINITY: Interna	al thread 5 bound to	OS proc set {3}		
KI	MP_AFFINITY: Interna MP_AFFINITY: Interna	al thread 6 bound to	OS proc set {5}		
	ounter = 4000000	at thread 7 bound to	os proc set {/}		
1	in a loop of 400000	D pointer chases the	re were 4000000 unique accesses		
NU	UM: 1024, DIM: 400000	9			
E	lapsed time = 48132. lapsed time summed i	35/84.000000 cycles per iteration = 48131	76524 000000 cvcles		
	100 good timings		() () () () () () () () () () () () () (
c)	ycles per iteration	= 12.032941			
by	ytes per cycle = 2.0	559366			
	orofiling finished :	= 2.705387 => 04/28/2008 11:09:4	12 AM		
·	······	workload			
v .	workload stopped	=> 04/28/2008 11:09:4	12 AM		
(
]¢					
levinth@levinth-nhma:~	levinth@levinth-nhma:-	-/ 🔳 levinth@levinth-	nhma:~/) 🙀 [levinth]	evinth@levinth-nhma:~/	Intel(R) Performance Tun





Page Allocation Order Matters







Conclusions

NUMA will add complexity to software performance analysis and optimization

We have the infrastructure to manage this



33 11/10/2009



Backup



34 11/10/2009

