Beyond Multi-core: Achieving killer performance with storage, network and compute in a NUMA world.

Lessons learned developing AMPS

Jeffrey M. Birnbaum jmb@crankuptheamps.com
Website: http://www.crankuptheamps.com

60East TECHNOLOGIES

- Fast Publish/Subscribe Solution
- High Performance Content Filtering
- Filters resemble SQL-92 + Xpath
- Sub-microsecond processing latencies
- Capacity to do >1M messages/sec/core

Example subscription filters:

XML:

```
•   /FIXML/Order@Sym = “IBM” and
•       /FIXML/Order/OrdQty@Qty >= 5000
```

FIX:

```
•   /55 = “IBM” and /35 in (‘D’, ‘C’)
```

AMPS

- State of the World (Database)
- Content filtered queries
- Atomic query + subscribe
- Message deltas (both in and out)
- Focus Tracking
- Analytics Engine (Real-time Aggregation)
- Parallel and lock-free design

Analytics Engine (Real-time Aggregation)

- Projects one topic into another
  - Think: Real-time SQL-92 “View”

Example:

- Project:
  - /11 as /customer
  - /55 as /symbol
  - sum(/14 * /99)/sum(/14) AS /vwap
- GroupBy: /11, 55
- New Topic Name: VWAP

This:

- 11=c01;55=INTC;14=1000;99=34.50;
- 11=c01;55=INTC;14=5000;99=34.75;
- 11=c01;55=INFA;14=100;99=18.75;

Becomes:

- customer=c01;symbol=INTC;vwap=34.70833;
- customer=c01;symbol=INFA;vwap=18.75;

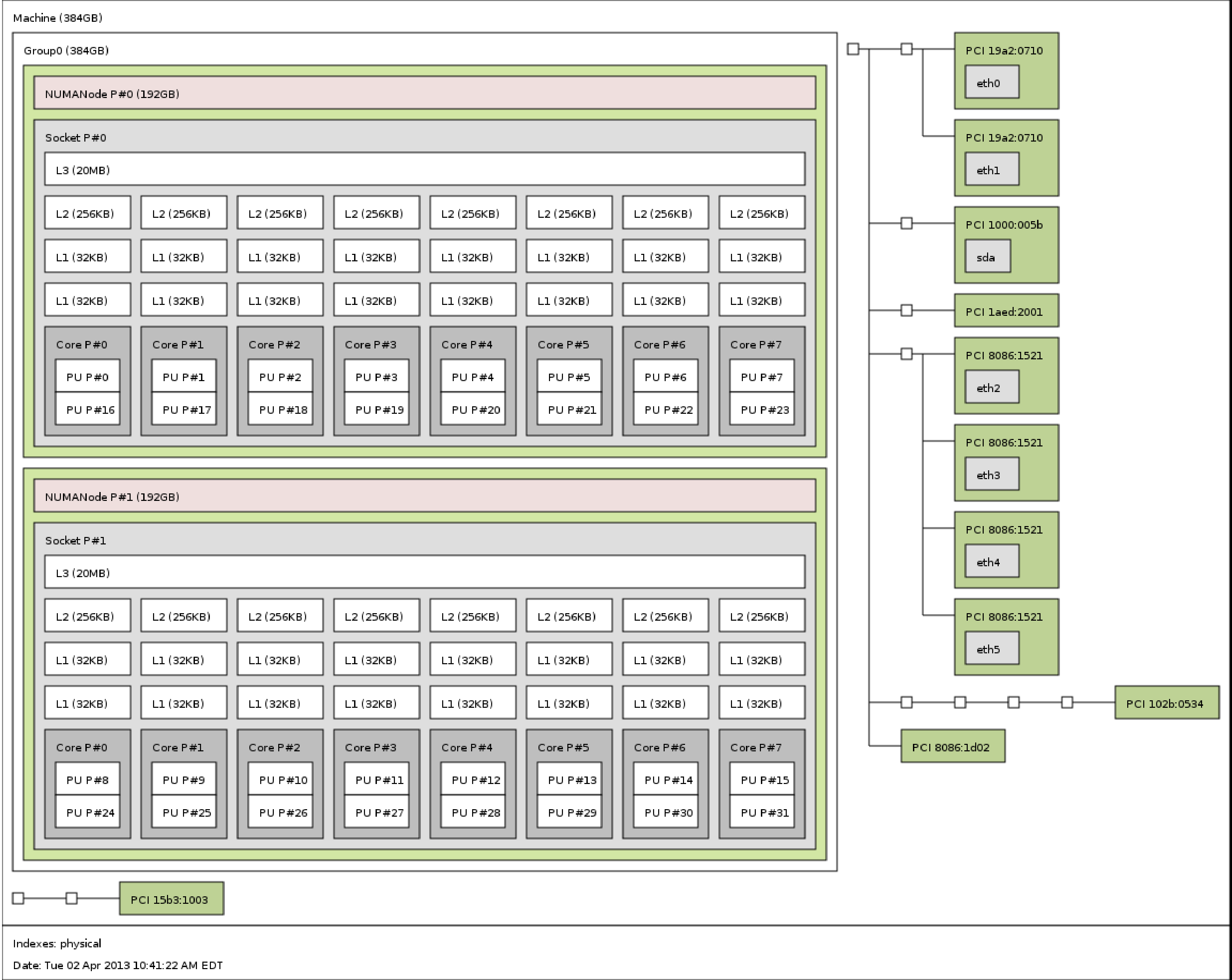
Achieving Killer Performance

KILLER PERFORMANCE

- Cache aware data structures
- NUMA awareness
  - Threads
  - Memory
  - PCIe IO Devices (network and storage)
  - Intra and Inter package communication latency
- Lock-free concurrency
- Generic is almost always a loser
- Only share static data

60East TECHNOLOGIES

SANDYBRIDGE NUMA

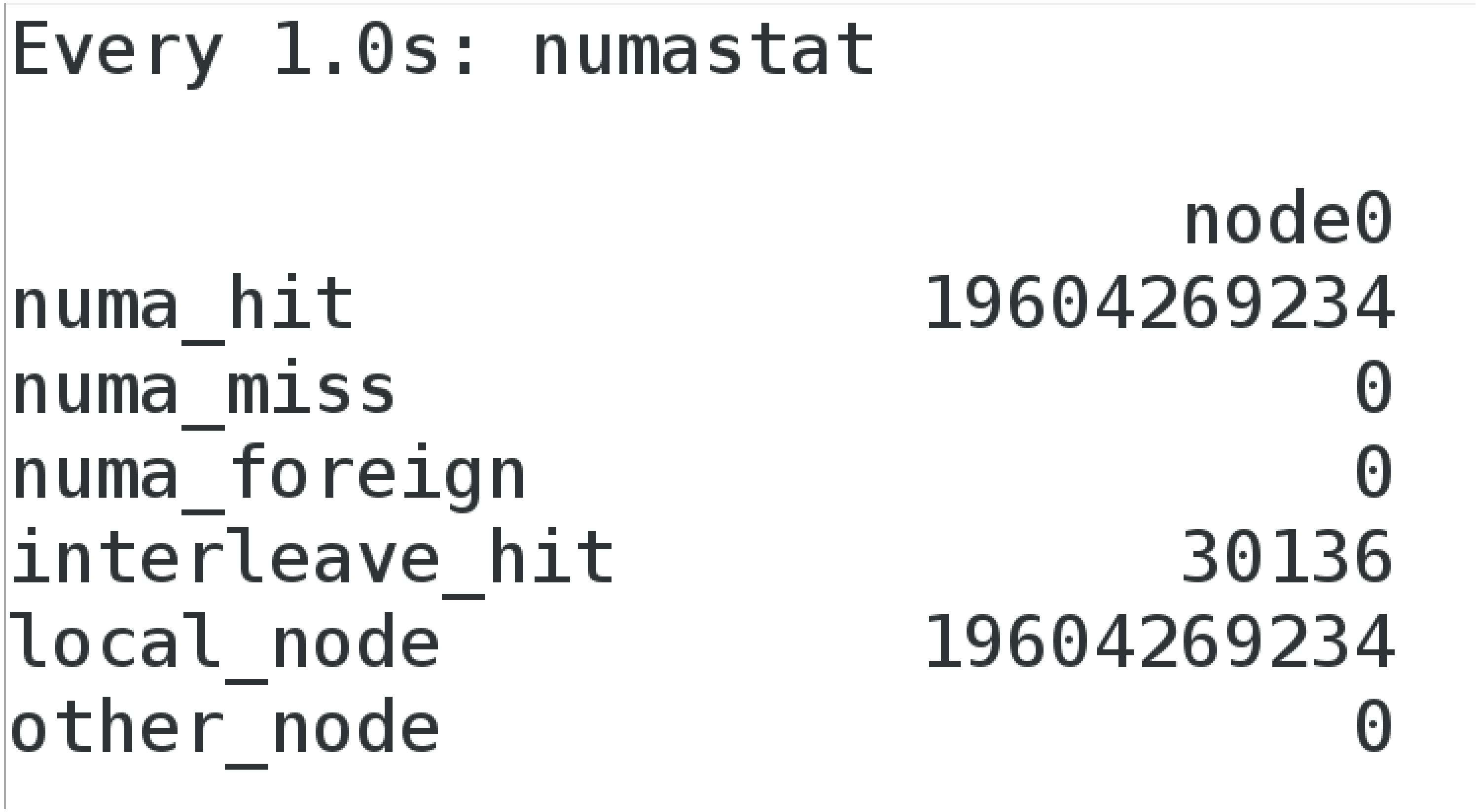


Indexes: physical

Date: Tue 02 Apr 2013 10:41:22 AM EDT

Understanding NUMA Memory Performance

$watch -n1 numastat



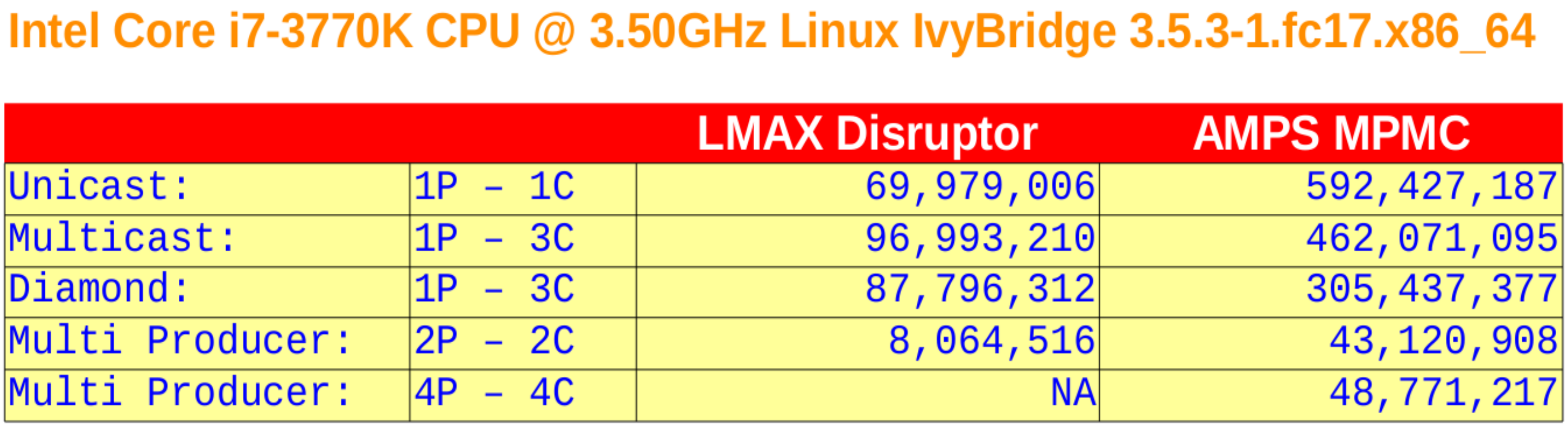
```
Every 1.0s: numastat

                           node0
numa_hit             19604269234
numa_miss                      0
numa_foreign                   0
interleave_hit             30136
local_node           19604269234
other_node                     0
```

numastat

Cache Aware Data Structures

MPMC Ring Buffer Performance



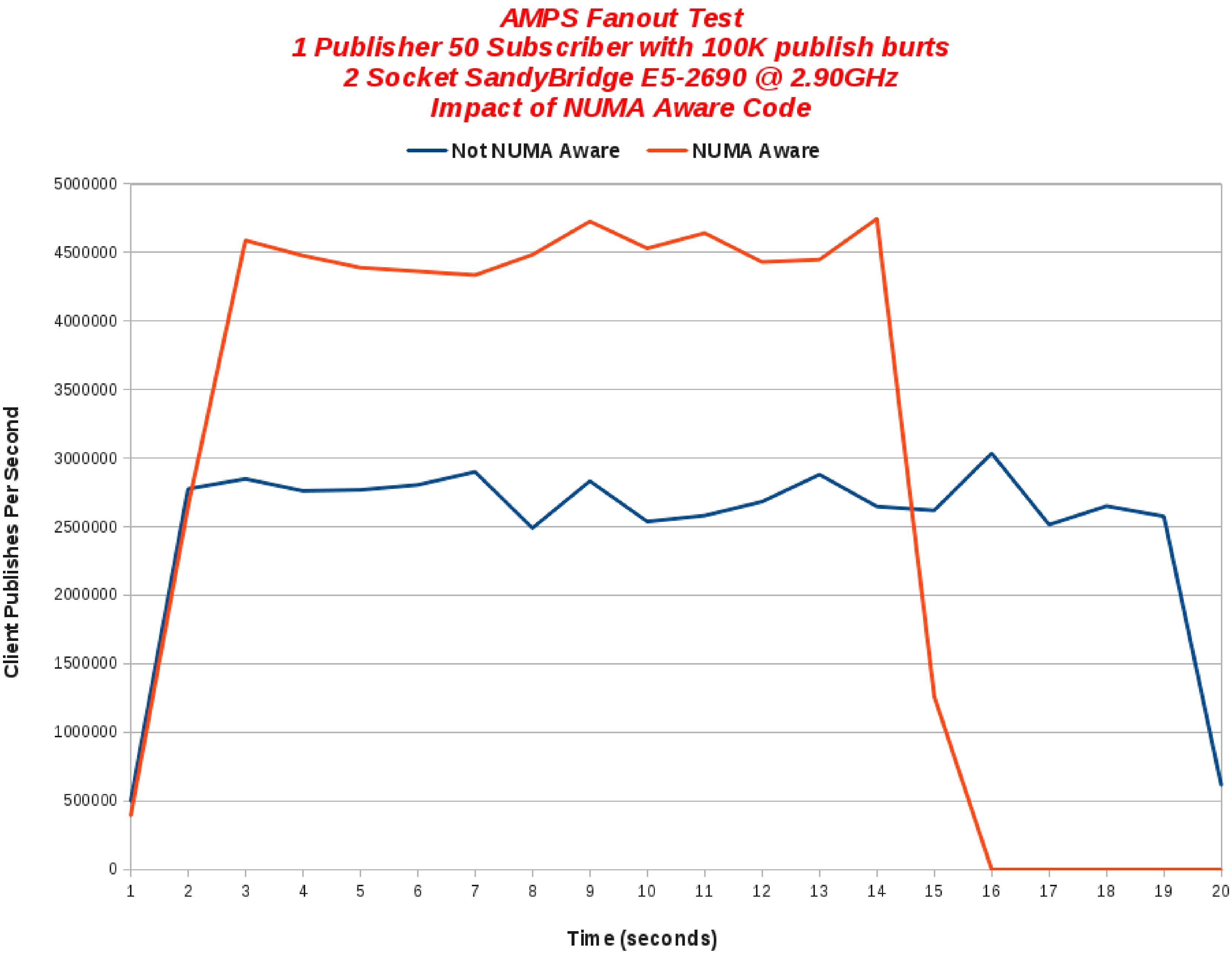
Intel Core i7-3770K CPU @ 3.50GHz Linux IvyBridge 3.5.3-1.fc17.x86_64

		LMAX Disruptor	AMPS MPMC
Unicast:	1P – 1C	69,979,006	592,427,187
Multicast:	1P – 3C	96,993,210	462,071,095
Diamond:	1P – 3C	87,796,312	305,437,377
Multi Producer:	2P – 2C	8,064,516	43,120,908
Multi Producer:	4P – 4C	NA	48,771,217

cas vs fetch_and_add
less shared state

KILLER PERFORMANCE

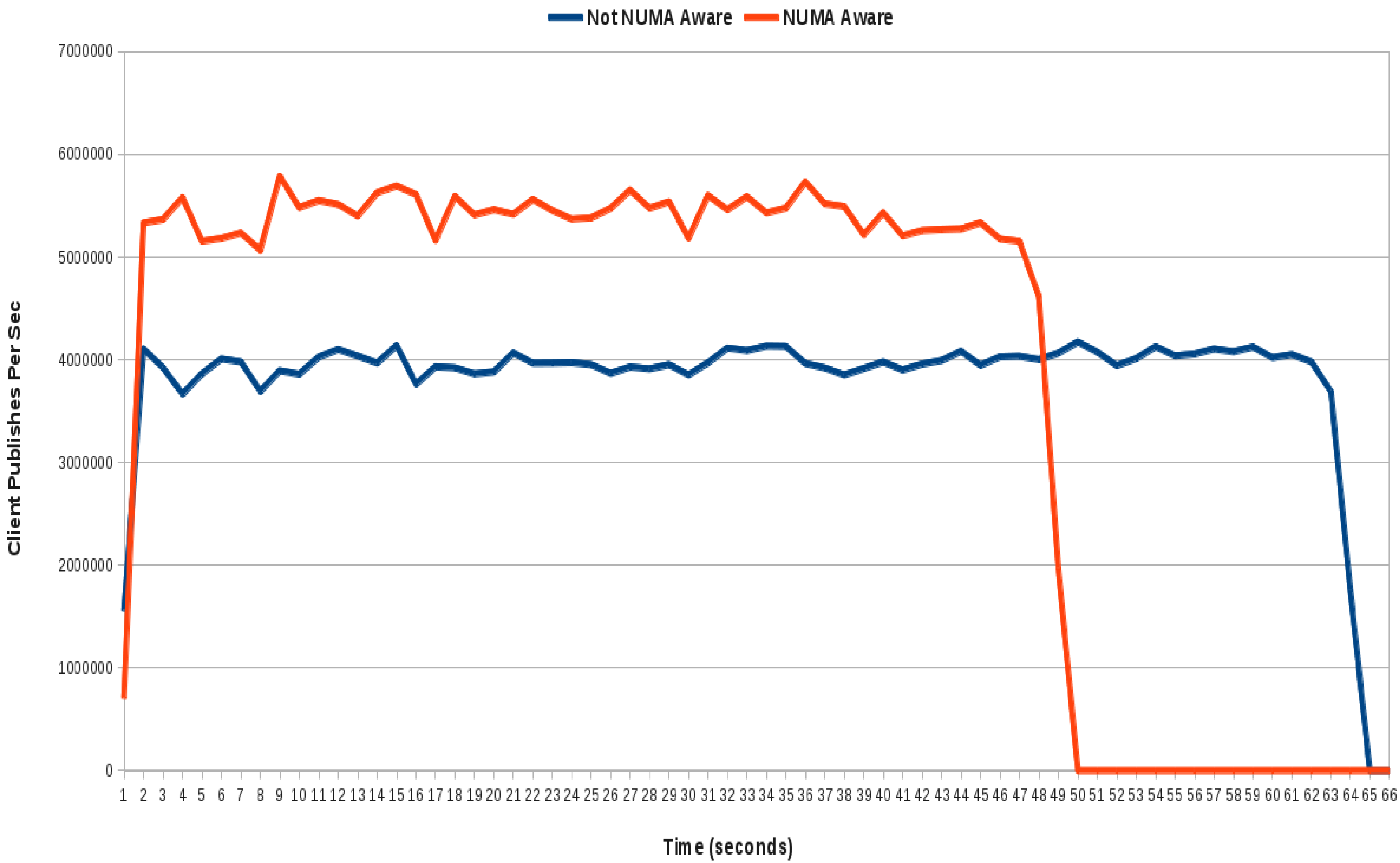
NUMA AWARE CODE



60East TECHNOLOGIES

NUMA IMPACT

AMPS Fanout Test
5 Publishers 50 Subscribers with 100K publish bursts
2 Socket SandyBridge E5-2690 0 @ 2.90GHz
impact of 'numatl -N 0 -m 0'



Advice

- Experiment
- Read and Learn
  - Dave Dice Blog
  - https://blogs.oracle.com/dave/entry/numa_aware_reader_writer_locks
- Portable Hardware Locality (hwloc)
  - lstopo – display system topology
  - numactl – control NUMA policy
  - numstat – observe cross-node memory requests
  - libnuma – control affinity of threads and memory
- Design with non-uniform access in mind
  - Locality of threads and memory is critical so design processing paths accordingly
  - Try to reduce inter-package communication especially wrt memory access patterns

THANKS

Slides will be available on our blog at http://crankuptheamps.com

Come see us at future conferences where we will share techniques and things we think about when delivering top tier performance.

Thanks!