BIWA SIG
Better Information Better Results
www.oraclebiwa.org

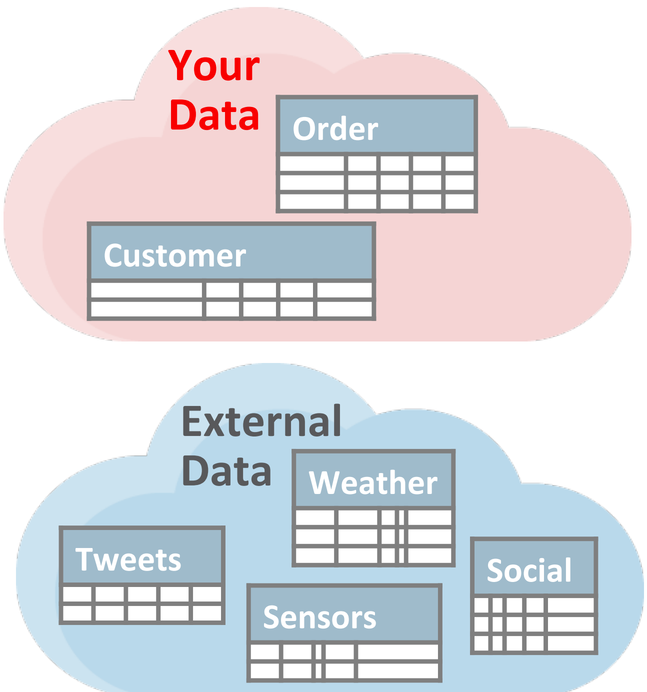
Advanced Analytics & Graph: Transparently taking advantage of Hardware innovations in the Cloud

Brad Carlile, Sr. Direction Solutions Architecture Engineering
Oracle

Additional info on Proof points:
http://blogs.oracle.com/bestperf

Modern Analytics is about Your Data & External Data

Better information when you can analyze all relevant data



- Analytics is now your data ***PLUS*** external data
  - Oracle Database In-Memory Fastest on SPARC
  - Continuously analyzing same data in different ways – Fastest if store data in-memory
- Many ways to access external data
  - Social networks, public data: sentiments, weather, traffic, events, ratings, trends, IoT sensors, behaviors,...
  - Lots of ETL/filtering needed to find useful data
  - Big Data SQL

Oracle Database Optimized for OLTP Transactions

SQL handles data manipulation and management of information

Order	F1	F2	…	F50

Customer	F1	F2	…	F100

- OLTP Database were optimized for transactions and/or events
  - Each transaction/event has many features
  - Demographics, social, geography, behaviors,…
- **A single transaction's features are stored together in row for memory**
  - Why? Better computational and memory efficiency when working a transaction
- *…but Analysis wants to look at specific features across many transactions*

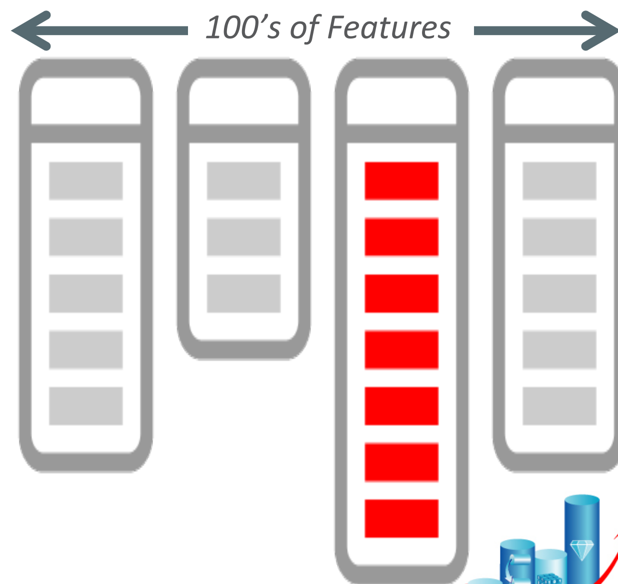
Oracle Database also Optimized for Analytic Queries

SQL optimized for analytic queries

- Column Store is optimal for Analytics
  - Analytics Looks across transactions at specific columns
  - Database columns are Machine Learning features
- **Columnar faster computational and better memory efficiency when analyzing features**
- Can also exploit data characteristics >20x benefit
  - Storage optimizations – data repeatedly analyzed
    - Dictionary + layered compression means 10's TB of data fit
    - into TB's of server storage
  - Processing optimizations
    - Vector Processing of Dictionary-encoded Columns, Bloom Filter join processing, operator pushdown, metadata optimizations

Big Feature Revolution

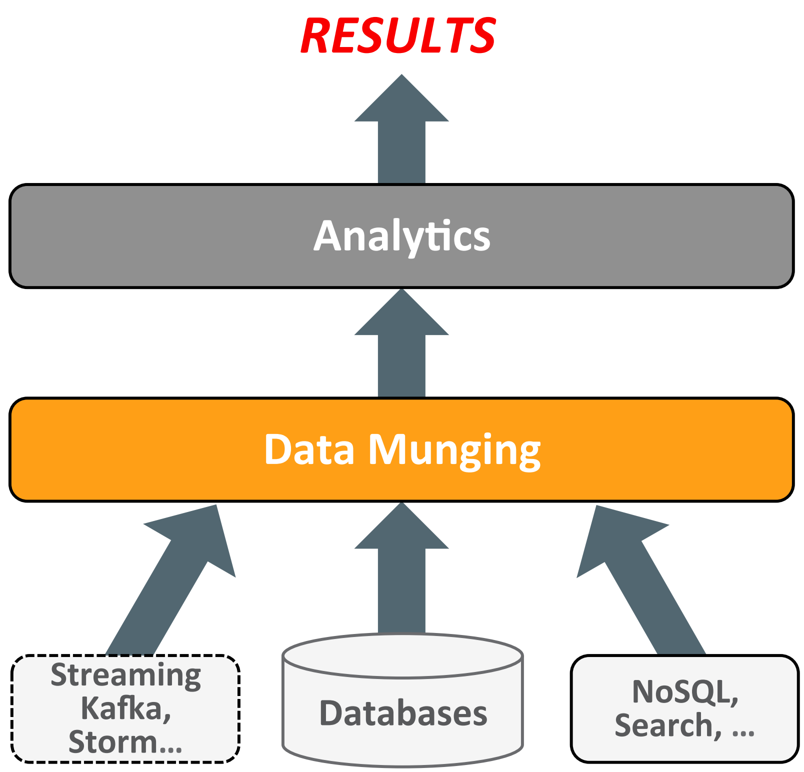
Number of features collected for each Transaction/event are exploding: 10's to 100's



The Basic Analytics Flow

A lot of time spent in Data Munging

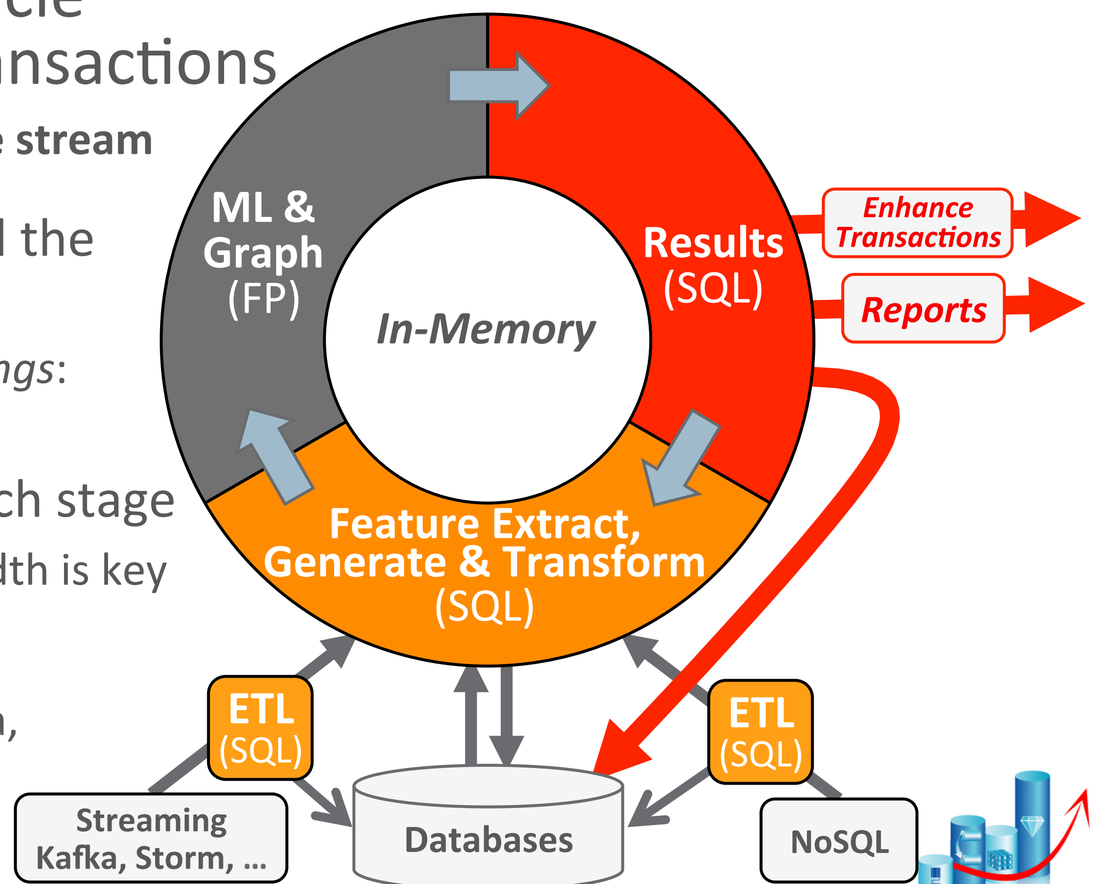
- Data can come from many sources
  – Databases, NoSQL, csv, feeds…
- We need to prepare it
  – Data Munging of all sorts!
- Analyze the data
  – Find the "right way" to analyze it
    - ML, Graph, SQL…



Continuous Analytics Cycle
Results Often Enrich Transactions

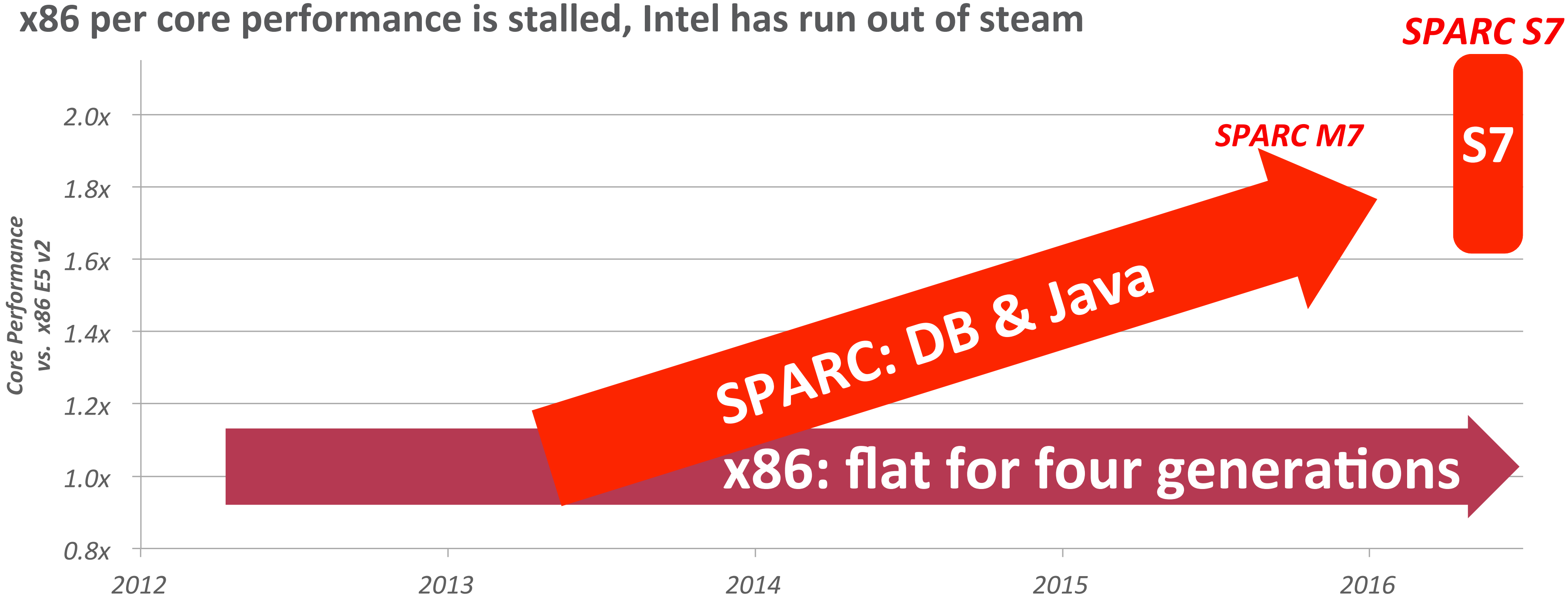
Analytics is more than one pipeline stream

- Continuous iterations around the data analytics wheel
  - Save, catalog, and re-use *all things*: data, SQL, code, and analytics
- In-memory advantages at each stage
  - SPARC's DAX & leading bandwidth is key
- Many sources of data
  - Internal proprietary, public data, external streaming, archives



SPARC's Innovations are Continually Increasing Core Performance

x86 per core performance is stalled, Intel has run out of steam

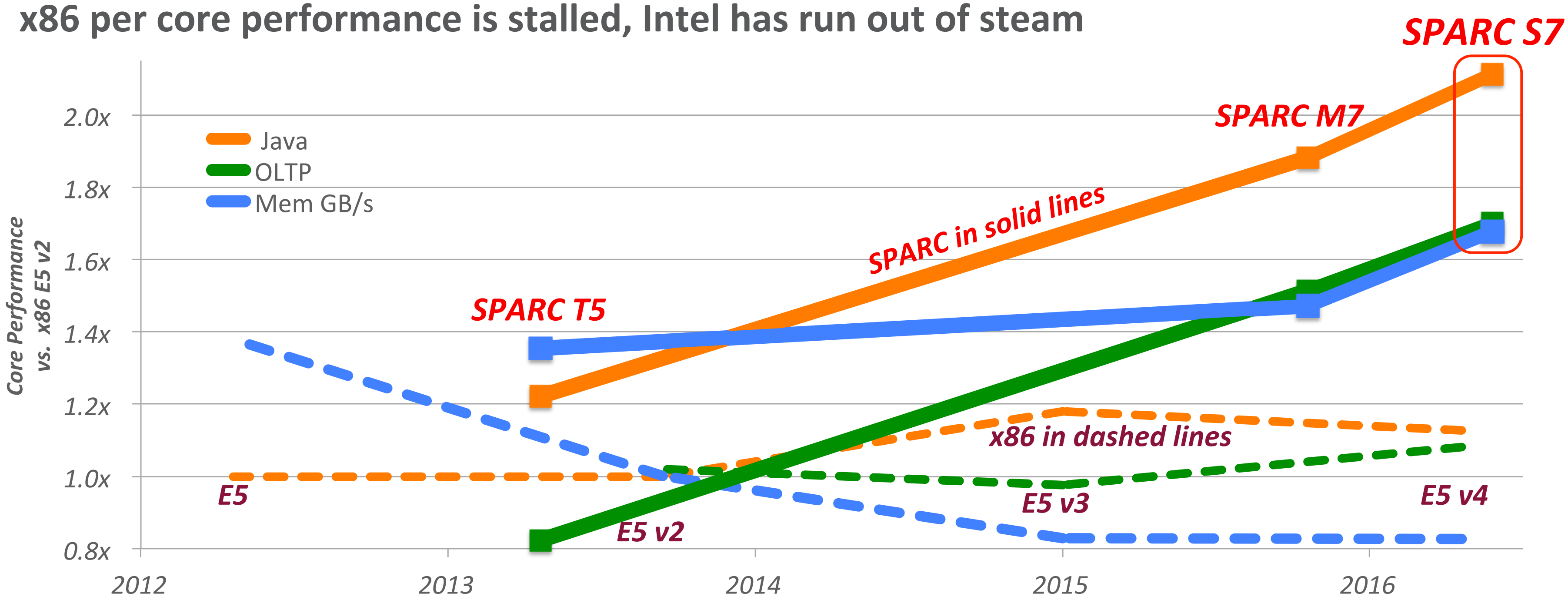


- SPARC has faster cores for Database, Java, Apps, ...
- SPARC has more cores per chip which also drives efficiency

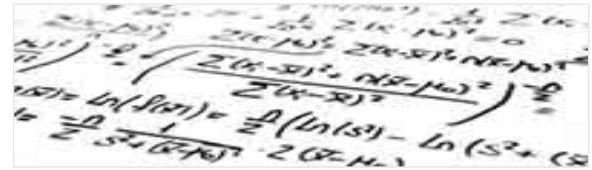
"per core = (server performance)/(server core count)"

SPARC S7 is 1.6x to 2.1x faster than x86

x86 per core performance is stalled, Intel has run out of steam

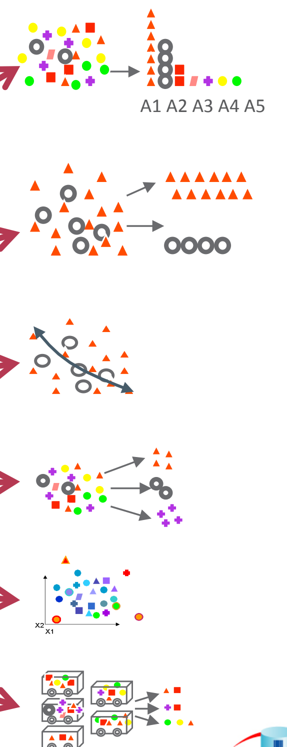


- It's more important how you use transistors, than the number transistors you make (Moore's Law)



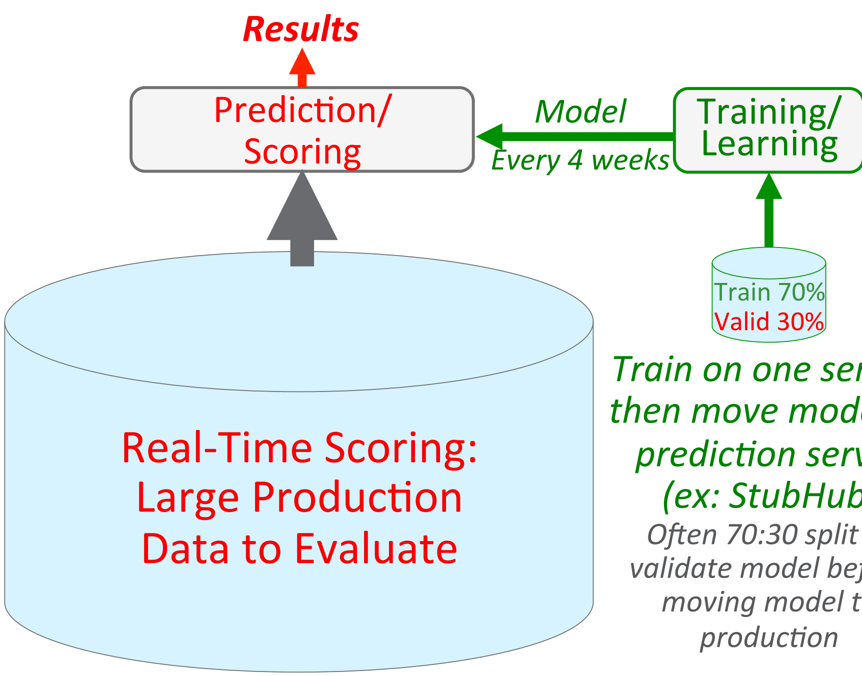
ML - Machine Learning

- ***Automatically*** sifting through many features & data
  - to find previously hidden patterns,
  - to discover valuable new insights and make predictions
- Examples:
  - Id most important factors *(Attribute Importance)*
  - Predict customer behaviors *(Classification)*
  - Predict or estimate a value *(Regression)*
  - Segment a population *(Clustering)*
  - Find fraudulent or "rare events" *(Anomaly Detection)*
  - Determine co-occurring items in a "baskets" *(Associations)*
  - Find profiles of targeted people or items *(Decision Trees)*



Machine Learning (ML): Prediction vs. Training
Prediction critical for Real-time scoring engines

SPARC 3x-5x faster at Prediction & 2x faster at Training



Train on one server, then move model to prediction server (ex: StubHub)

Often 70:30 split to validate model before moving model to production

Prediction requires highly-efficient server design

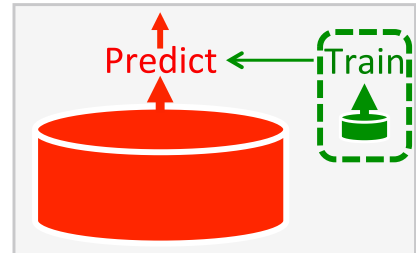
	ML Score/ Prediction	ML Learn/ Train
% of activity	Most Data	*periodic
SPARC core advantage vs x86	3x to 5x per core	Up to 2x per core

**periodically updates to models: quarterly, monthly, weekly, nightly*

BIWA SIG
Better Information Better Results
www.oraclebiwa.org

Oracle Database Advanced Analytics Option Machine Learning on SPARC

ML training SPARC M7 up to 2.0x faster per core than x86

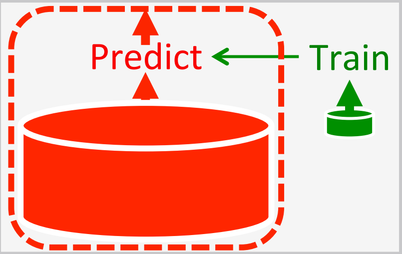


	Training: ***Creating Model from data***	**Attri-butes**	**X5-4 4-chip**	**T7-4 4-chip**	**SPARC per chip Advantage**	**SPARC per core Advantage**
Supervised	SVM IPM Solver	900	1442s	404s	3.6x	**2.0x**
	GLM Classification	900	331s	154s	2.1x	**1.2x**
	SVM SGD Solver	9000	157s	84s	1.9x	**1.1x**
	GLM Regression	900	78s	55s	1.4x	**0.8x**
Cluster Model	Expectation Maximization	9000	763s	455s	1.7x	**0.9x**
	K-Means	9000	232s	161s	1.4x	**0.8x**

- Oracle Advanced Analytics in Oracle Database 12.2
  - SPARC M7 faster per core on training 64-bit floating point intensive
  - In-memory 640 million records, Airline On-time dataset

SGD (Stochastic Gradient Descent), IPM (Interior Point Method)

Oracle Database Advanced Analytics Option Machine Learning on SPARC



ML prediction SPARC 3.0x - 4.8x faster per core than x86

	Prediction *Using model on 1B records*	**Attri-butes**	**X5-4 4-chip**	**T7-4 4-chip**	**SPARC per chip Advantage**	**SPARC per core Advantage**
Supervised	SVM IPM Solver	900	206s	24s	8.6x	**4.8x**
	GLM Regression	900	166s	25s	6.6x	**3.7x**
	GLM Classification	900	156s	25s	6.2x	**3.5x**
	SVM SGD Solver	9000	132s	24s	5.5x	**3.1x**
Cluster Model	K-Means	9000	222s	35s	6.3x	**3.6x**
	Expectation Maximization	9000	243s	40s	6.1x	**3.4x**

- Oracle Advanced Analytics in Oracle Database 12.2
  - SPARC M7 much faster per core on scoring bandwidth-intensive
  - In-memory 1 Billion records, Airline On-time dataset

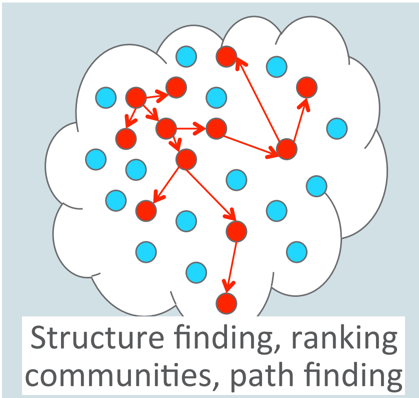
SGD (Stochastic Gradient Descent), IPM (Interior Point Method)

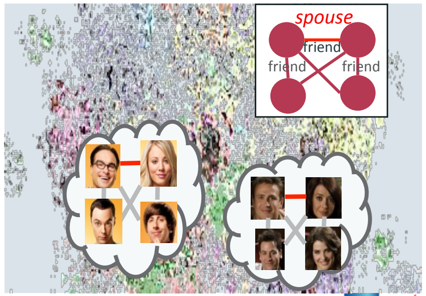
"per core = (server performance)/(server core count)"

Graph Analysis

Inter-relationships between data and networks are growing in importance

- Graphs everywhere:
  - Facebook (friends of friends), Twitter, LinkedIn, etc…
    - Most data has inter-relationships that contain insights
- Two major types of graph algorithms
  - Computational Graph Analytics: Analysis of entire Graph
    - Influencer ID, community detect, patter machine, recommendations
  - Graph Pattern Matching
    - Queries that find sub-graphs fitting relationship patterns





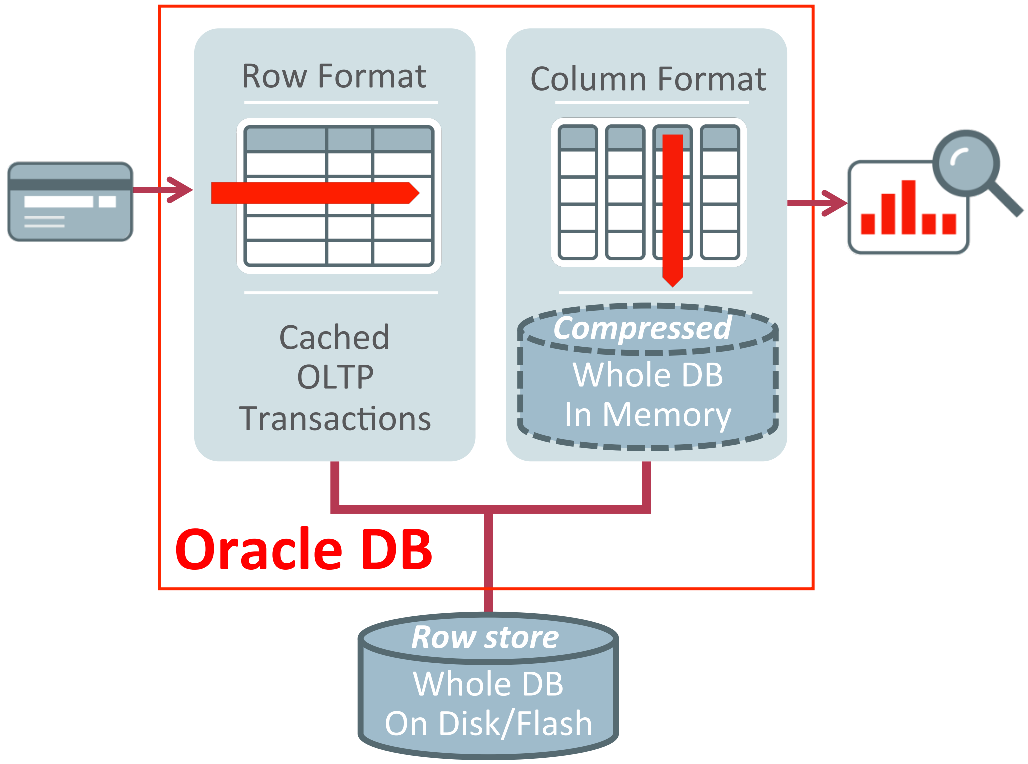
Computational Graph Algorithms: PageRank & Single-Source Shortest Path (SSSP)

Graph: SPARC M7 up to 1.5x faster per core than x86

Graph Algorithm	*Workload Size*	4-chip X86 E5 v3	4-chip SPARC T7-4	SPARC per chip Advantage	SPARC per core Advantage
SSSP Bellman-Ford	448M vertices, 17.2B edges	39.2s	14.7s	2.7x	**1.5x**
	233M vertices, 8.6B edges	21.3s	8.5s	2.5x	**1.4x**
PageRank	448M vertices, 17.2B edges	136.7s	62.6s	2.2x	**1.2x**
	233M vertices, 8.6B edges	72.1s	27.6s	2.6x	**1.5x**

- Graph computations accelerated by SPARC's memory bandwidth
  - Bellman-Ford/SSSP (single-source shortest path) – optimal route or connection
  - PageRank - measuring website importance

Oracle Database In-Memory & SPARC DAX



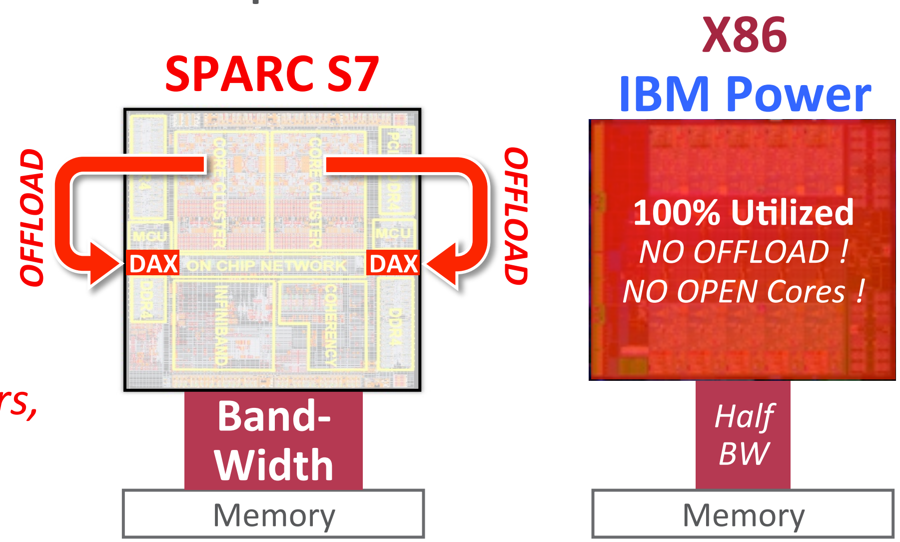
- OLTP uses proven row format
- Analytics use in-memory Column
  - 10x faster analytics due to software
  - Columnar compression means huge databases can now fit in-memory
- Oracle database **stores BOTH** row and column formats
  - Simultaneously active and transactionally consistent
- Even faster with SPARC DAX (Data Accelerator) HW innovation
  - Additional 10x to 20x faster for Analytics

SPARC's Multi-generational Leapfrog in Performance

Radical Innovation: Integrated Offload offers 10x faster performance!

- Integrated Offload
  - Data Analytics Acceleration (DAX)
  - Encryption & Security

- *It's more important how you use transistors, than Moore's Law (the number transistors you make)*

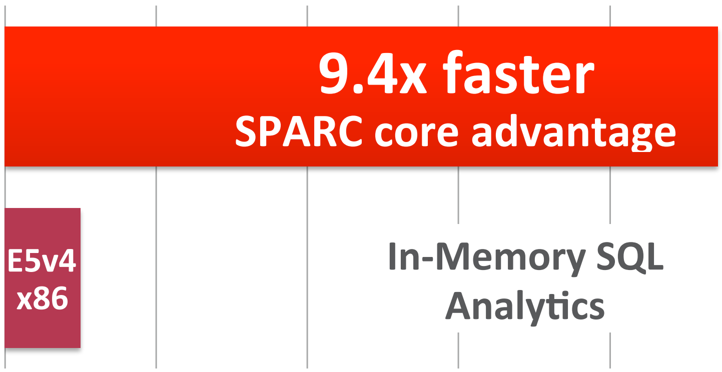


Other vendors focused on Detached GPUs

- *Detached GPUs are poorly designed for SQL*
- *Slow GPU interconnection robs performance*

SPARC Dramatically Faster In-Memory SQL Analytics

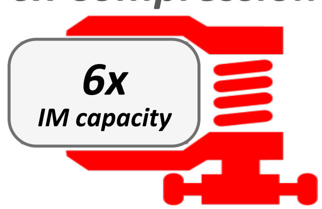
SPARC DAX (Software on Silicon) is multigenerational lead in performance



SPARC Offload

DAX frees
Cores for other
processing

6x Compression



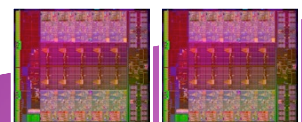
- **SPARC S7 9.4x faster per core than x86 E5 v4**
  - SPARC S7-2 422 query/m vs 2-chip x86 56 query/m
    - S7-2 (18-cores total), 2-chip x86 E5 v4 (20-cores total)
  - x86 per core performance flat for 4 generations
- DAX offloads In-Memory Scans
  - DAX offload allows cores to increase throughput
- 160GB in memory represents 1TB DB on disk
  - 6.2x compression with Oracle 12.2 In-memory

Real-Time Enterprise: Simultaneous OLTP & In-memory

SPARC S7: Faster analytics, faster OLTP, and better response time (per core)

Analytics: SPARC S7 5.12x faster per core
OLTP: SPARC S7 2.04x faster per core

***Sixteen SPARC S7** cores*
same load as
***Fifty-six x86** cores*

x86 E5 v3 36 cores		**SPARC S7-2 16 cores**
		
OLTP Performance Throughput: **216k Transactions per sec** Response Time: **~8ms** **Analytics** **47 Queries per min**	VS	**Analytics** **107 Queries per min** **OLTP Performance** Throughput: **196k Transactions per sec** Response Time: **~9ms**

SF 1050 = 1.05TB Data Warehouse
Oracle 12*c* 12.1.0.2.2

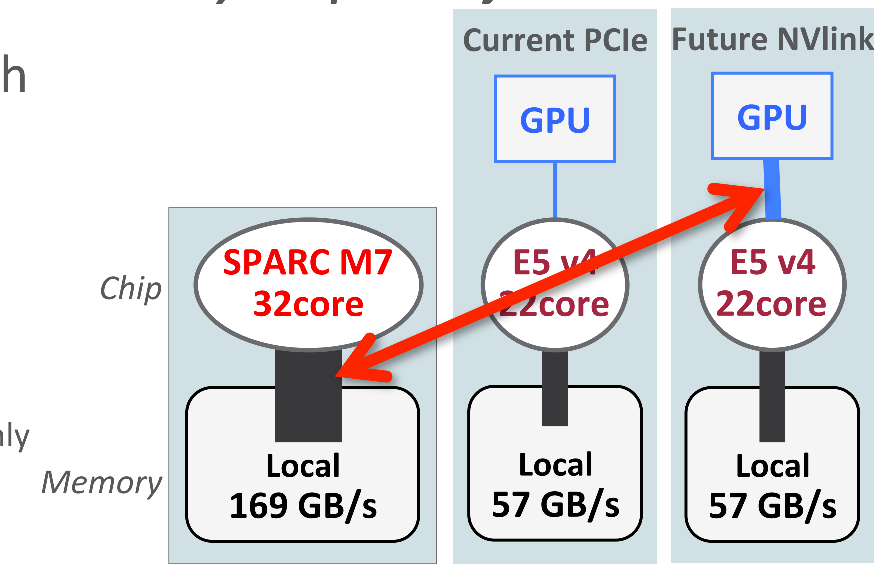
"per core = (server performance)/(server core count)"

ORACLE

Graphic Processing Units (GPUs)
GPUs Only Help Very Compute Intensive Algorithms

Complex ML/Statistics often not fast on GPUs… "Can only compute as fast as move data"

- Detached GPUs have limited bandwidth
  - GPUs fit for lots of simple graphics
    - Graphics have tiny data movement: Coords & textures in, video frame out
  - Big Data ML much more demanding: GPUs have limited applications
    - ML Learning & Scoring
      - Some ML Learning can be compute intensive, but only done initially to create Model
    - Nvidia Math Libraries only have BLAS3 routines
      - BLAS2 and BLAS1 do NOT have compute intensity therefore are not in library
      - Complicated ML has mix of BLAS1, BLAS2, BLAS3



Bandwidth lines to scale
GPU compute often stalled by lack of bandwidth

SQL is the Powerful Language for Data Scientists

SQL provide ETL and Query power for Analytics, Machine Learning(ML), and Graph

- SQL a powerful, concise, expressive, that enables rapid development
  - Decades of advanced algorithms optimizing SQL
  - Can also write SQL in DSL (Domain Specific Languages) for Spark, Python, R, Scala, etc…
- Examples of uses
  - Apache Spark SQL (67% growth in SQL users)
  - Java Streams - DSL SQL for Java
    - Used by many customers as well as Oracle Coherence
  - Apache Eclipse – Goldman Sachs Collections
  - Apache Hadoop Hive
  - Oracle Database Analytics & OLTP

Because SQL is so important
Oracle put SQL in Silicon
in the SPARC processor (DAX)

SPARC is Over 10x Faster than 86
Because of Unique Design

SQL & the Many Flavors of DSL – All potentials for DAX

SQL can be written in Domain Specific Languages (aka Language Integrated Queries)

- *SQL:*
  - *SELECT count(*) from person WHERE citizen.age > 18*
- *Apache Spark SQL (DSL)*
  - *val voters : Int = citizen.filter($"citizen.age" > 18).count()*

Apache Spark feeds both formats into SQL optimizer
Joins can be written & optimized

- Java Streams (DSL)
  - int voters = arrayList.parallelStream().filter(citizen-> citizen.olderThan(18)).count();
- Goldman Sachs Collections – Apache Eclipse (DSL)
  - int voters = fastList.count(citizen-> citizen.olderThan(18));

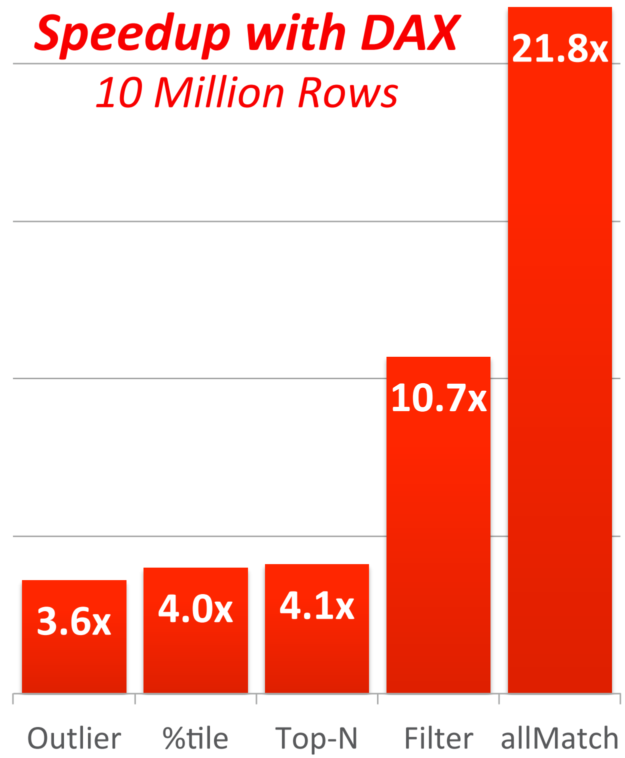
SPARC DAX Integrated with Java JDK8 Streams

DAX accelerates Java Streams: 3.6x to 21.8x faster

- Java Streams provide SQL-style code in Java
  – Java Streams is a perfect fit for DAX acceleration
- DAX & Java Streams
  – Integer Stream filter, allMatch, anyMatch, noneMatch, map(ternary operator), toArray & count functions
  – Simple code changes to use DAX
    - import com.Oracle.Stream *instead of* import java.util.Stream
    - DaxIntStream *instead of* IntStream
- Public access: http://SWiSdev.oracle.com/DAX

Java Stream API:

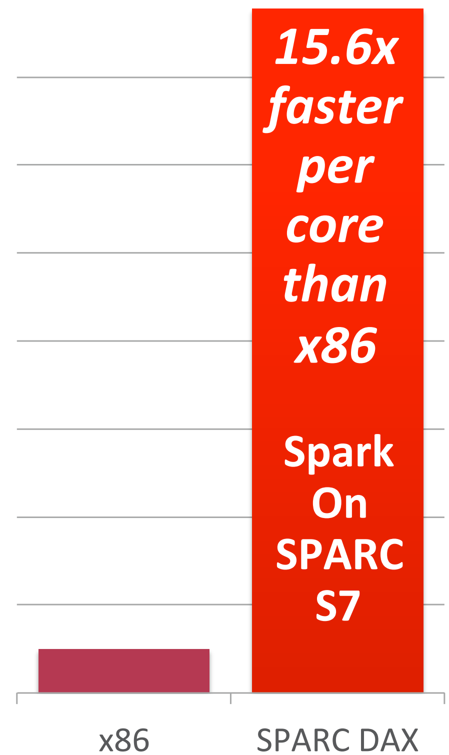
filter2_data.parallelStream().filter(w->w.temp<100).count()



SPARC DAX Accelerating Apache Spark 2.1.0

Spark SQL on SPARC S7-2 DAX over 15.6x faster per core than x86

- *Proof-of-Concept Prototype on SPARC S7:*
  Apache Spark 2.1 POC: DAX & in-memory columnar
  - 2-chip SPARC S7 DAX: 9.8 Billion rows/sec on 2-predicate scan
    - SPARC S7-2(16 total cores) 0.061 sec
    - 2-chip x86 (20 total cores) 0.760 sec *latest generation E5 v4 Broadwell*
  - *These advantages are over and above Tungsten's improvements*
- SPARC DAX offloads critical functions
  - SQL: filtering, dictionary encoding, 1 & 2 predicate, join processing, additional compression, etc.
- Libdax open API: http://SWiSdev.oracle.com/DAX



SPARC Advantages for Every Form of Analytics

SPARC is 1.3x to over 10x faster per core on Analytics than x86

Analytics	*In-memory Columnar SW & Software in Silicon (SPARC DAX)*	*SPARC's efficient CPU & memory design*
Database Analytics *Analytic SQL Queries for database*	✔	✔
Java Analytics *Java Code that probes data SQL-like code*	✔	✔
Machine Learning (ML) *Automatically finding hidden patterns & correlations*	✔	✔
Graph *Automatically find patterns in Social networks, etc.*		✔
NoSQL *Fast access to semi-structured & unstructured data*		✔
Spatial *Fast throughput on geometric data*		✔

Feature generation & Selection for ML is often SQL

ORACLE

Java & Database are SPARC's Design Targets

Innovations differentiate SPARC from the generic computing

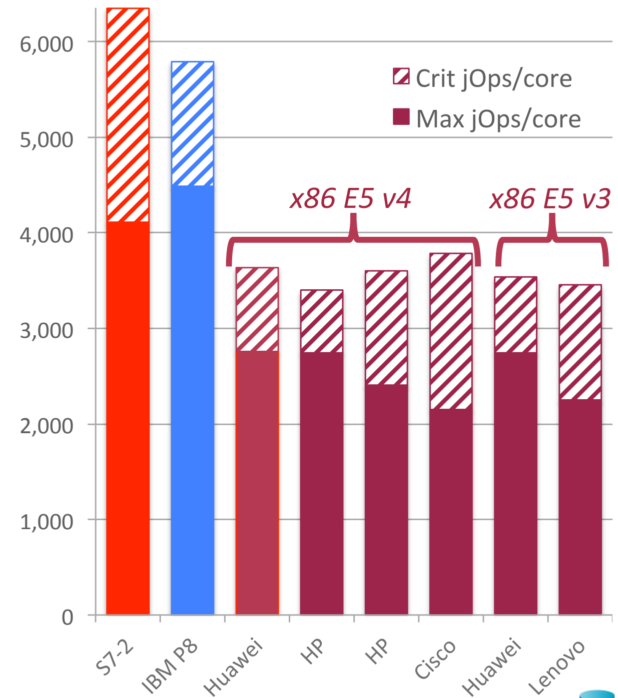
- Oracle **believes** the Enterprise requires ***deep innovations*** to **make servers significantly better** for **Cloud** and **Analytics**
  - SPARC, DB, & Java co-engineered to create unique chip innovations
    - Breakthroughs of 70% to 10x or more by working across the stack boundaries
  - Oracle revolutionizing the Cloud
    - Dramatic innovations in security, analytics, and efficient VMs

SPARC drastically reduces hardware & licensing cost

SPECjbb2015 MultiJVM SPARC S7

SPARC S7 core 1.5x to 1.9x faster than x86 E5 v4 core on Max jOps/core

	Processor	chip, core	Max jOPS	Crit jOPS
SPARC S7-2	**4.27GHz SPARC S7**	**2,16**	**65,790**	**35,812**
IBM S812LC	2.9GHz Power8 10c	1,10	44,883	13,032
Huawei RH2288Hv3	2.2GHz E5 v4 22c	2,44	121,381	38,595
Cisco C220	2.2GHz E5 v4 22c	2,44	94,667	71,951
Huawei RH2288Hv3	2.3GHz E5 v3 18c	2,36	98,673	28,824
Lenovo x240 M5	2.3GHz E5 v3 18c	2,36	80,889	43,654



- Innovations in SPARC processor design continue to improve Java and JVM performance
  - x86 performance is flat generation to generation
    - Trade off between tuning for Max jOps or Crit jOps on x86

Java(JVM) is the Language of the Cloud & FOSS

Java & the underlying JVM is the design target for SPARC

Big Data & DevOps	Description	Written to
Oracle Fusion	Enterprise Apps	Java
Apache Spark	In-memory Analytics	JVM
Apache Cassandra	NoSQL database	Java
Apache Hadoop	Disk to disk Map Reduce	Java
Apache HDFS	filesystem	Java
Apache Lucene	Doc text Indexing	Java
Apache Solr	text/doc/web search	Java
Jersey/Grizzly (REST)	REST Interface	Java
Apache Hive	SQL on HDFS	Java
Neo4j	Graph	Java
Scala	Language	JVM

Big Data & DevOps	Description	Written to
Akka (REST)	REST for Big Data	JVM
Apache Accumulo	key/value store	Java
Apache Flink	Batch &Stream Processing	Java(JVM)
Apache Kafka	Streaming Message Broker	JVM
Apache log4j	Log event manager	Java
Apache Samza	Stream processing	JVM
Apache Storm	Stream processing	Java & Clojure
Apache Yarn	Cluster job manager	Java
Apache Zookeeper	Config & group services	Java
H20.ai	ML Apps libraries	Java, Python,R
Apache Hbase	NoSQL database	Java

JVM = Java Virtual Machine - Runtime

SPARC core is Fastest for FOSS – x86 much slower

1.6x to 2.0x faster per core means you need spend lots more on x86 than SPARC

- FOSS (Free & Open-Source Software)
  - Apache Spark — SPARC DAX >15x faster
  - Apache Cassandra — SPARC core 1.7x faster
  - Apache Hadoop — SPARC core 1.6x faster
- SPARC Advantages:
  - SPARC fastest at JVM & highest efficiency, GC improvements
  - Chip design innovations each generation
  - SPARC scales with multi-threading: SPARC 8 threads vs. x86 2-threads
  - SPARC has 2x greater memory bandwidth

Much of FOSS runs on JAVA or JVM

Oracle & FOSS: Transactions & Analytics

Transactions now getting enriched with live Analytics

	Transactions	SQL Query Analytics	NoSQL	ML Machine Learning	Graph Analytics	Streaming
Oracle	Fusion & Oracle DB	Oracle 12c In-memory	Oracle NoSQL	Oracle Advanced Analytics	Oracle Spatial & Graph	Oracle Stream Analytics
		Oracle Big Data SQL (merge data from various sources)				
FOSS	DevOps & Database	Spark SQL	Cassandra NoSQL & MongoDB	Spark MLlib	Spark GraphX	Spark Streaming & Kafka…

- SPARC processors accelerate every type of Transactional & Analytical workload
  - Hadoop components: Hive(SQL), HBase(NoSQL), Mahout(ML), Giraph(Graph),…
  - ML = machine learning/statistics

Cassandra NoSQL 2.2.6 (Yahoo Cloud Serving Benchmark)

SPARC core is 1.7x faster than E5 v3 core

Cassandra	Chip, core	Processor	Ave Ops/s	K Ops/s per core	SPARC per core Advantage
SPARC S7-2	**2,16**	**4.27 SPARC S7**	**66,955**	**4.2K**	**1.7x**
E5 v3 Haswell	2,36	2.3 E5 v3	90,184	2.5K	1.0x

- SPARC S7 *1.7x faster per core* than E5 v3 Haswell
  - SPARC's advantages on Java/JVM benefit Cassandra
    - YCSB – Yahoo Cloud Serving Benchmark – 300M
      - Mixed Load Workload A (50% Read/50% Update)

Oracle's NoSQL on YCSB (Yahoo Cloud Serving Benchmark)

ORACLE NOSQL DATABASE

SPARC S7 core is 1.9x faster per core than E5 v4 (Broadwell)

NoSQL YCSB	Chip, core	Processor	Ops/s	K Ops/s per core	SPARC per core Advantage	NoSQL Version
SPARC S7-2	**2,16**	**4.27 SPARC S7**	**340,766**	**21.3k**	**1.9x**	**4.0**
2-chip E5 v4 Broadwell	2,44	2.2 E5 v4	502,273	11.4k	baseline	4.0

- Oracle NoSQL 4.0
- SPARC S7 1.9x faster per core than E5 v4 Broadwell
  - E5 v3 Haswell core same performance as E5 v4 Broadwell core
  - YCSB (Yahoo Cloud Serving Benchmark)
    - Mixed Load: 95% Read/5% Update

"per core = (server performance)/(server core count)"

SPARC is Faster than x86 for Hadoop

SPARC core secure is 1.4x faster than unsecure x86 v2
SPARC core unsecure is 1.6x faster than unsecure x86 v2

Terasort 10TB	Chip, core	Processor	Perf GB/min Per chip	Perf GB/min Per core	Hadoop	SPARC M7 Core Advantage
Oracle T7-4 1node	**4, 128**	**4.13 SPARC M7**	**35.2**	**0.90**	**Open source**	**(unsecure baseline)**
Oracle T7-4 1node	**4, 128**	**4.13 SPARC M7**	**32.2**	**1.02**	**Open source**	**Secure AES-256-GCM baseline**
Ivy Bridge EX	1, 12	2.8 E5 v2	8.2 est	0.67	Cloudera 5.3	Estimate 1.35x (1.5x) est

- SPARC M7 secure 3.9x faster per chip than *estimated unsecure x86 E5 v2 Ivy Bridge*
  - E5 v3 Haswell less bandwidth per core than E5 v2 Ivy Bridge
- SPARC M7 secure 4.3x faster per chip than unsecure IBM Power8 6c

Unpredictable Amazon AWS vs. Oracle's OPC

AWS's Unreliable QoS is a hidden cost

- Most AWS x86 processors are slow & older
  - AWS uses lower performance chips, Oracle uses fastest x86 chips at each generation
  - AWS vCPU x86 = ½ core = 1 hyperthread (mostly E5 v2, some E5 v3)
  - OPC OCPU x86 = 1 core = 2 hyperthreads (mostly E5 v3, new E5 v4)
  - OPC OCPU SPARC = 1 core = 8 HW threads (SPARC M7 & SPARC T7)

  Same Cost
- AWS Unpredictable run-to-run performance variation
  - Noisy neighbor often hurts network, storage, & CPU performance
  - x86 core in AWS are slower and more variable than bare-metal x86

SPARC CMT Threads designed to scale
x86 Hyperthreading have well-known scaling issues

Required Benchmark Disclosure Statement

Must be in SPARC S7 & M7 Presentations with Benchmark Results

- Additional Info: http://blogs.oracle.com/bestperf
-
- SPEC and the benchmark name SPECjEnterprise are registered trademarks of the Standard Performance Evaluation Corporation. Results from www.spec.org as of 7/6/2016. SPARC S7-2, 14,400.78 SPECjEnterprise2010 EjOPS (unsecure); SPARC S7-2, 14,121.47 SPECjEnterprise2010 EjOPS (secure) Oracle Server X6-2, 27,803.39 SPECjEnterprise2010 EjOPS (unsecure); IBM Power S824, 22,543.34 SPECjEnterprise2010 EjOPS (unsecure); IBM x3650 M5, 19,282.14 SPECjEnterprise2010 EjOPS (unsecure).
- SPEC and the benchmark name SPECjbb are registered trademarks of Standard Performance Evaluation Corporation (SPEC). Results from http://www.spec.org as of 6/29/2016. SPARC S7-2 (16-core) 65,790 SPECjbb2015-MultiJVM max-jOPS, 35,812 SPECjbb2015-MultiJVM critical-jOPS; IBM Power S812LC (10-core) 44,883 SPECjbb2015-MultiJVM max-jOPS, 13,032 SPECjbb2015-MultiJVM critical-jOPS; SPARC T7-1 (32-core) 120,603 SPECjbb2015-MultiJVM max-jOPS, 60,280 SPECjbb2015-MultiJVM critical-jOPS; Huawei RH2288H v3 (44-core) 121,381 SPECjbb2015-MultiJVM max-jOPS, 38,595 SPECjbb2015-MultiJVM critical-jOPSHP ProLiant DL360 Gen9 (44-core) 120,674 SPECjbb2015-MultiJVM max-jOPS, 29,013 SPECjbb2015-MultiJVM critical-jOPS; HP ProLiant DL380 Gen9 (44-core) 105,690 SPECjbb2015-MultiJVM max-jOPS, 52,952 SPECjbb2015-MultiJVM critical-jOPS;; Cisco UCS C220 M4 (44-core) 94,667 SPECjbb2015-MultiJVM max-jOPS, 71,951 SPECjbb2015-MultiJVM critical-jOPS; Huawei RH2288H V3(36-core) 98,673 SPECjbb2015-MultiJVM max-jOPS, 28,824 SPECjbb2015-MultiJVM critical-jOPs; Lenovo x240 M5 (36-core) 80,889 SPECjbb2015-MultiJVM max-jOPS,43,654 SPECjbb2015-MultiJVM critical-jOPS; SPARC T5-2 (32-core) 80,889 SPECjbb2015-MultiJVM max-jOPS, 37,422 SPECjbb2015-MultiJVM critical-jOPS; SPARC S7-2 (16-core) 66,612 SPECjbb2015-Distributed max-jOPS, 36,922 SPECjbb2015-Distributed critical-jOPS; HP ProLiant DL380 Gen9 (44-core) 120,674 SPECjbb2015-Distributed max-jOPS, 39,615 SPECjbb2015-Distributed critical-jOPS; HP ProLiant DL360 Gen9 (44-core) 106,337 SPECjbb2015-Distributed max-jOPS, 55,858 SPECjbb2015-Distributed critical-jOPS; HP ProLiant DL580 Gen9 (96-core) 219,406 SPECjbb2015-Distributed max-jOPS, 72,271 SPECjbb2015-Distributed critical-jOPS; Lenovo Flex System x3850 X6 (96-core) 194,068 SPECjbb2015-Distributed max-jOPS, 132,111 SPECjbb2015-Distributed critical-jOPS.
- SPEC and the benchmark names SPECfp and SPECint are registered trademarks of the Standard Performance Evaluation Corporation. Results as of October 25, 2015 from www.spec.org and this report. 1 chip resultsSPARC T7-1: 1200 SPECint_rate2006, 1120 SPECint_rate_base2006, 832 SPECfp_rate2006, 801 SPECfp_rate_base2006; SPARC T5-1B: 489 SPECint_rate2006, 440 SPECint_rate_base2006, 369 SPECfp_rate2006, 350 SPECfp_rate_base2006; Fujitsu SPARC M10-4S: 546 SPECint_rate2006, 479 SPECint_rate_base2006, 462 SPECfp_rate2006, 418 SPECfp_rate_base2006. IBM Power 710 Express: 289 SPECint_rate2006, 255 SPECint_rate_base2006, 248 SPECfp_rate2006, 229 SPECfp_rate_base2006; Fujitsu CELSIUS C740: 715 SPECint_rate2006, 693 SPECint_rate_base2006; NEC Express5800/R120f-1M: 474 SPECfp_rate2006, 460 SPECfp_rate_base2006.
- Two-tier SAP Sales and Distribution (SD) standard application benchmarks, SAP Enhancement Package 5 for SAP ERP 6.0 as of 5/16/16:SPARC M7-8, 8 processors / 256 cores / 2048 threads,SPARC M7, 4.133 GHz, 130000 SD Users, 713480 SAPS, Solaris 11, Oracle 12cSAP, Certification Number: 2016020, SPARC T7-2 (2 processors, 64 cores, 512 threads) 30,800 SAP SD users, 2 x 4.13 GHz SPARC M7, 1 TB memory, Oracle Database 12c, Oracle Solaris 11, Cert# 2015050. HPE Integrity Superdome X (16 processors, 288 cores, 576 threads) 100,000 SAP SD users, 16 x 2.5 GHz Intel Xeon Processor E7-8890 v3 4096 GB memory, SQL Server 2014, Windows Server 2012 R2 Datacenter Edition, Cert# 2016002 . IBM Power System S824 (4 processors, 24 cores, 192 threads) 21,212 SAP SD users, 4 x 3.52 GHz POWER8, 512 GB memory, DB2 10.5, AIX 7, Cert#201401. Dell PowerEdge R730 (2 processors, 36 cores, 72 threads) 16,500 SAP SD users, 2 x 2.3 GHz Intel Xeon Processor E5-2699 v3 256 GB memory, SAP ASE 16, RHEL 7, Cert#2014033. HP ProLiant DL380 Gen9 (2 processors, 36 cores, 72 threads) 16,101 SAP SD users, 2 x 2.3 GHz Intel Xeon Processor E5-2699 v3 256 GB memory, SAP ASE 16, RHEL 6.5, Cert#2014032. SAP, R/3, reg TM of SAP AG in Germany and other countries. More info www.sap.com/benchmarks.

ORACLE

Backup Slides

Oracle's Big Data SQL

Oracle Database runs in-memory with DAX and combines data from NoSQL data source

Query	Database Query	Modified to also access NoSQL	Change in time
a1	32.8s	34.0s	1.04x
b1	18.4s	19.6s	1.07x
b3	14.6s	17.7s	1.21x
q11x	14.5s	15.5s	1.07x
q12x	16.2s	17.4s	1.07x

- Oracle's Big Data SQL
  - Same SQL accesses both Database & NoSQL
  - SPARC T7-2 for Oracle Database with DAX & second SPARC T7-2 for Oracle's NoSQL
    - 3TB in-memory database, single-threaded query

```
SELECT c_name, c_mktsegment,
         lo_orderpriority, lo_ordtotalprice
FROM  t1, lineorder, date_dim, customer
WHERE lo_orderdate = d_datekey
     and c_email = ekey
     and lo_custkey = c_custkey
     and d_daynuminweek = 1
     and lo_quantity = 1
     and lo_orderpriority = '1-URGENT'
     and lo_shipmode = 'RAIL'
     and lo_discount = 1
     and c_region = 'ASIA'
     and c_mktsegment = 'BUILDING'
     and lo_ordtotalprice
          BETWEEN 1700000 and 1900000;
```

NoSQL accessed by adding: t1 and ekey

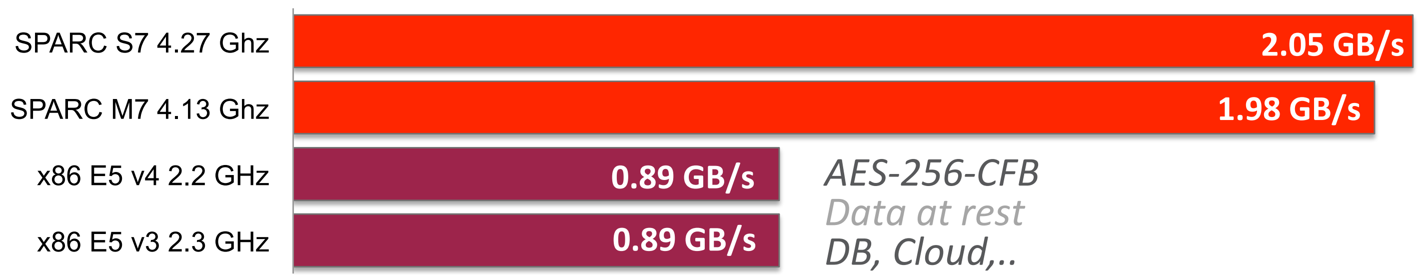
ORACLE

FIPS 140-2 Level 1 Validation Solaris 11.3:
Kernel Cryptographic Framework Module
& Userland Cryptographic Framework Module

Security Kernel SPARC S7 Performance

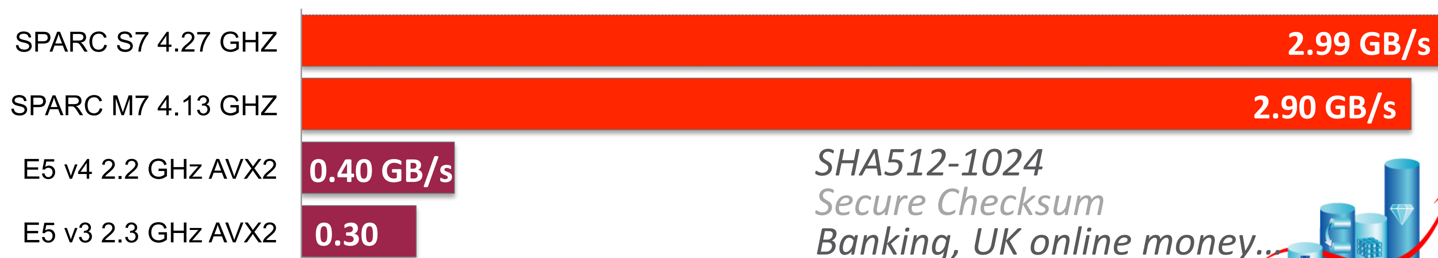
AES: SPARC S7 core is 2.3x faster than x86 E5 v4 core (AES-NI)
SHA: SPARC S7 core is 7.5x faster than x86 E5 v4 core

AES-CFB per core
S7 **2.3x faster than**
E5 v4 Broadwell



AES-256-CFB
Data at rest
DB, Cloud,..

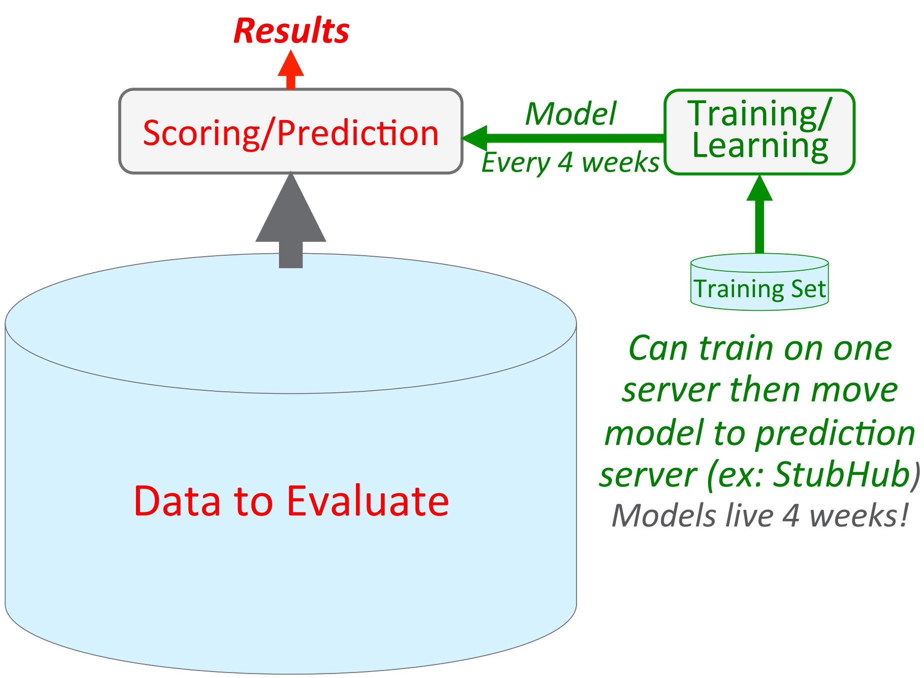
SHA512 per core
S7 **7.5x faster than**
E5 v4 Broadwell



SHA512-1024
Secure Checksum
Banking, UK online money...

Machine Learning (ML):
Scoring/Prediction versus Training/Learning Characteristics

Prediction/Scoring operates on huge amounts of data with low compute intensity



	ML Score/ Prediction	ML Learn/ Train
% of activity	Most Data	*Initial
Computation	O(n^2) Matrix-vector	O(n^3) Matrix-matrix
Data	O(n^2)	O(n^2)
Compute Intensity (Compute/Data)	Low constant	O(n)
SPARC Advantage Due to Memory Bandwidth & design	3x to 6x per core	Up to 1.3x per core

Initial and then occasionally update models

GPUs Only Accelerate Some Compute Kernels

Detached GPU memory bandwidth is bottleneck for most computations

Routine Level	Operation	Flops	Mem	Compute Intensity	Tesla K80 peak	Max Possible Gflops @12GB/s block=100 PCIe	GPU % peak	Max Possible Gflops @80GB/s block=100 Nvlink (future)	GPU % peak
BLAS1 (vector)	y = ax + y	2n	3n	2/3	935 Gflops	1 Gflops	**0.1%**	7 Gflops	**0.7%**
BLAS2 (matrix-vector)	y = Ax + y	2n^2 + 2n	n^2	2	935 Gflops	3 Gflops	**0.3%**	20 Gflops	**2.1%**
BLAS3 (matrix-matrix)	C = AB + C	2n^3	4 n^2	n / 2	935 Gflops	75 Gflops	**8.0%**	500 Gflops	**53.5%**
FFT	C = FFT(A)	n log(n)	2n	Log(n)/2	935 Gflops	239 Gflops 1M 1D-FFT	**25%**	935 Gflops 1M 1D-FFT	**Near peak**

Only 6 BLAS3 routine types in Nvidia Library: gemm, syrk, syr2k, trsm, trmm, symm
Can write codes using BLAS2/1 in GPUs but resulting compute intensity must be large to accelerate

Integrated Cloud

Applications & Platform Services