

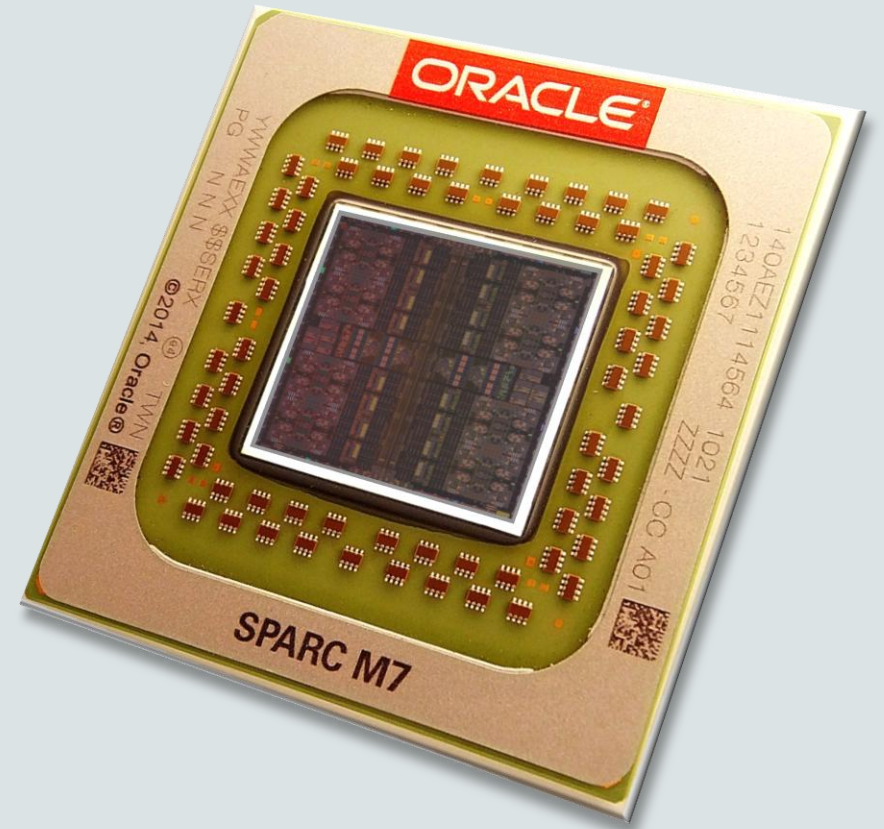
ORACLE®

# Software in Silicon in the Oracle SPARC M7 Processor

Kathirgamar Aingaran  
Director, Hardware Development

Sumti Jairath  
Sr. Hardware Architect

David Lutz  
Principal Software Engineer



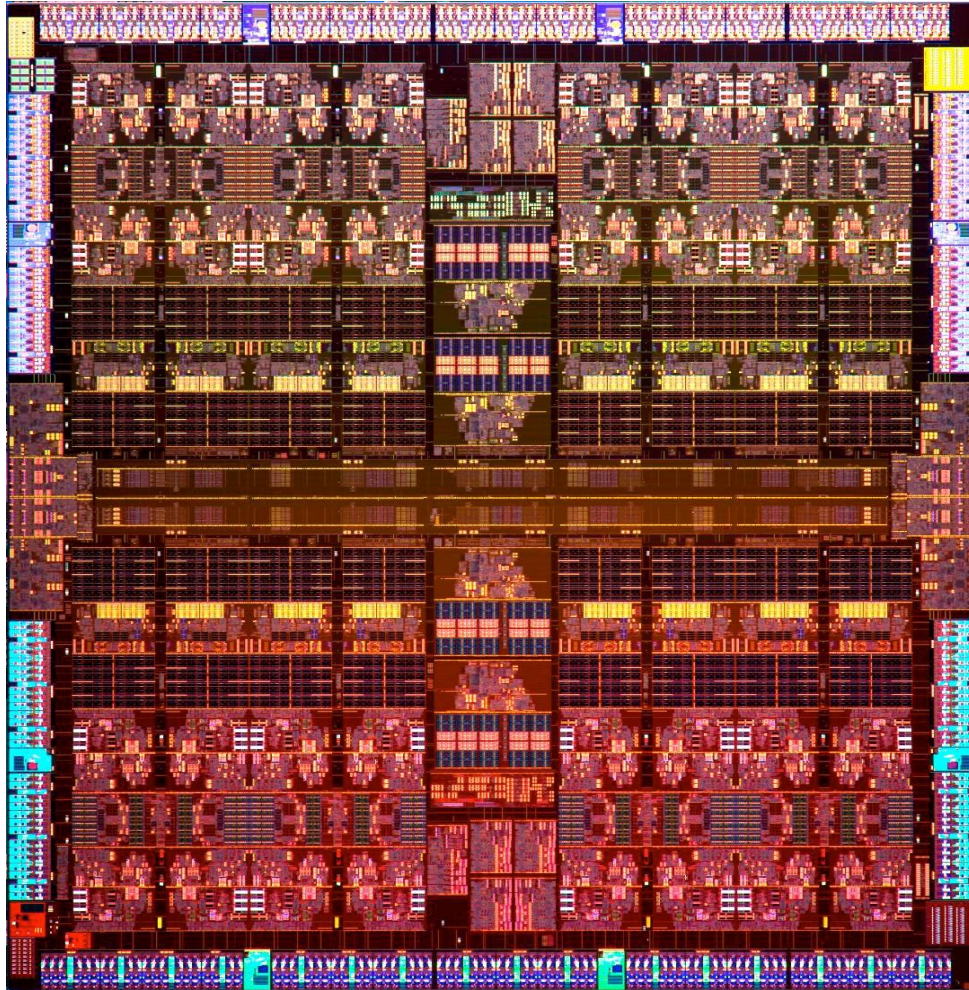
# Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

# Agenda

- 1 Motivation For SW in Silicon
- 2 SW in Silicon Features in SPARC M7
- 3 HW – SW Stack Description
- 4 Oracle In Memory Database Performance Measurements
- 5 Performance Results For Use Cases Beyond Oracle Database
- 6 Conclusion

# Oracle SPARC M7



- 32 core, 4.1GHz SPARC processor
- 64MB L3 Cache
- 8 x DDR4 Ports, ~150GB/s
- Self Governed Cores for Power Management
- SW in Silicon Feature Set
  - Custom Hardware For Targeted Applications
  - Security in Silicon
  - SQL in Silicon
  - Capacity in Silicon
- Same feature set on SPARC T7, S7

# Motivation For Software in Silicon

- **Chip and Server Trends**

- Limitation on frequency scaling
- ILP and TLP scaling close to peak
- Power wall:
  - limit on power a system can deliver and dissipate
- Dark Silicon:
  - limit on simultaneous activity within chip
- Specialized, power efficient, hardware provides advantage over general purpose cores

=> **SW in Silicon**

- **Industry Trends**

- Value shift: Transactions -> Analytics
- Highly data parallel algorithms
- Faster persistent storage
  - Focus on managing, mining, and securing data
- HW-SW Co-Design offers unique opportunities
  - FPGA, On Chip Accelerators, GP-GPU
- Especially for full stack vendors like Oracle

=> **SW in Silicon**

# Agenda

- 1 Motivation For SW in Silicon
- 2 SW in Silicon Features in SPARC M7
- 3 HW – SW Stack Description
- 4 Oracle In Memory Database Performance Measurements
- 5 Performance Results For Use Cases Beyond Oracle Database
- 6 Conclusion

# Feature Candidates for Software in Silicon

- **Analytics**: Simple, data-parallel SQL primitives (marked with a \*)

*Processes:* **SQL:**

*Decode values\**, *Sum aggregation\**

*Hash Joins*

*Bloom Filter Joins\**

*Simple Filters\**

*Range Filters\**

```
select sum(lo_extendedprice*lo_discount) as revenue
```

```
from lineorder, date_dim
```

```
where lo_orderdate = d_datekey and
```

```
d_year = 2012 and
```

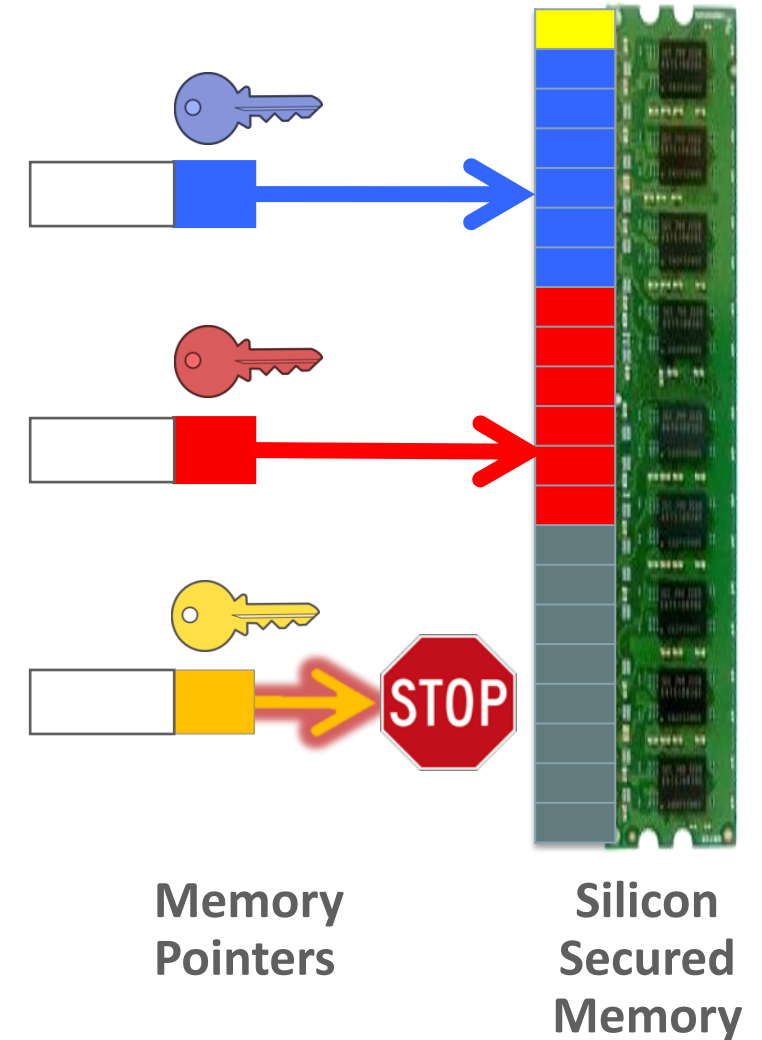
```
lo_quantity between 6 and 25 and lo_discount between 1 and 3
```

- **Inline Decompression**: “free” decompression
- Reduces memory footprint required to fit entire DB in DRAM/NVM
- **Enhanced Security in Memory**
- Added layer of protection for multi-tenant cloud environments



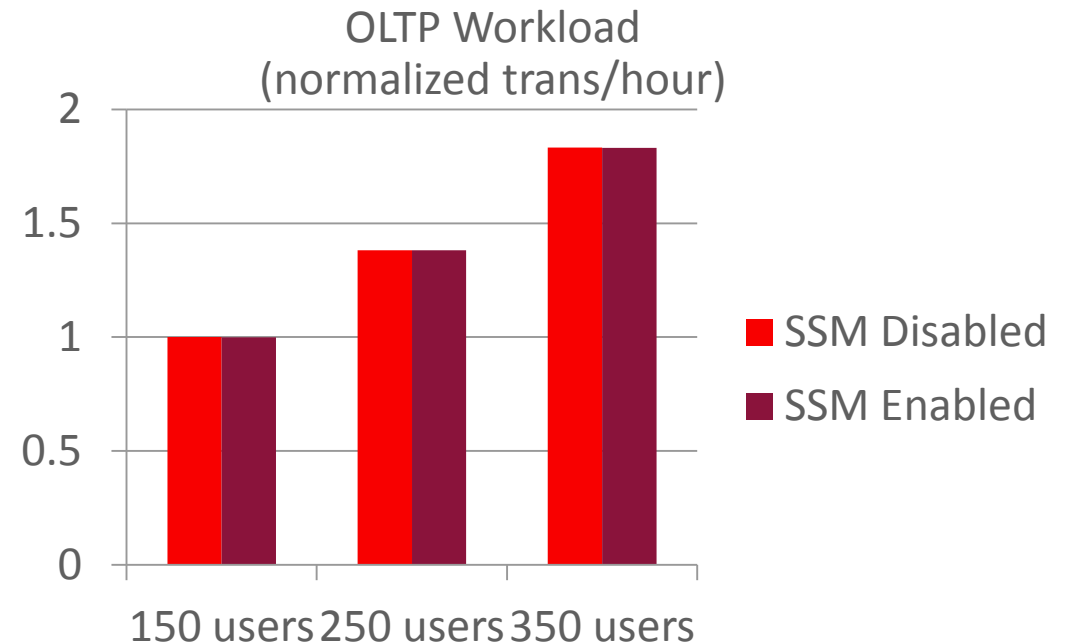
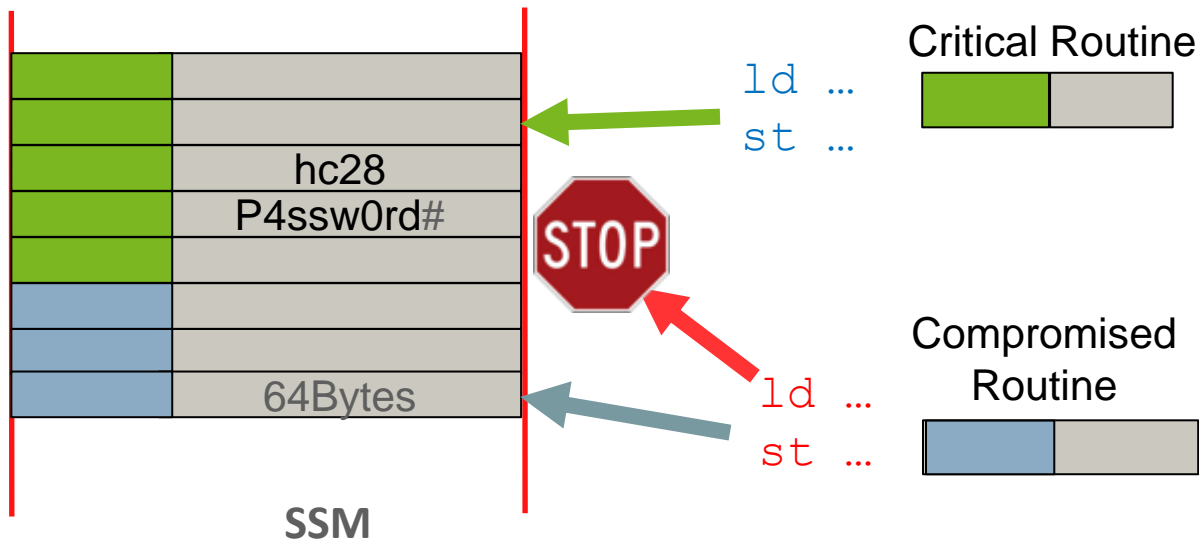
# Security In Silicon

- In-Memory Database
  - Puts terabytes of data in the cloud
  - Better be well protected
- Two types of Silicon enhanced protection
  - Encryption in Silicon
    - HW acceleration of AES, RSA, ECC, SHA-256/512, MD5, Camellia, 3DES
  - Silicon Secured Memory
    - HW colors all data in memory
    - Colors assigned by sys call
    - Data can only be accessed with correct “color” in pointers
    - Support for 14 colors (aka ADI states) at 64B granularity



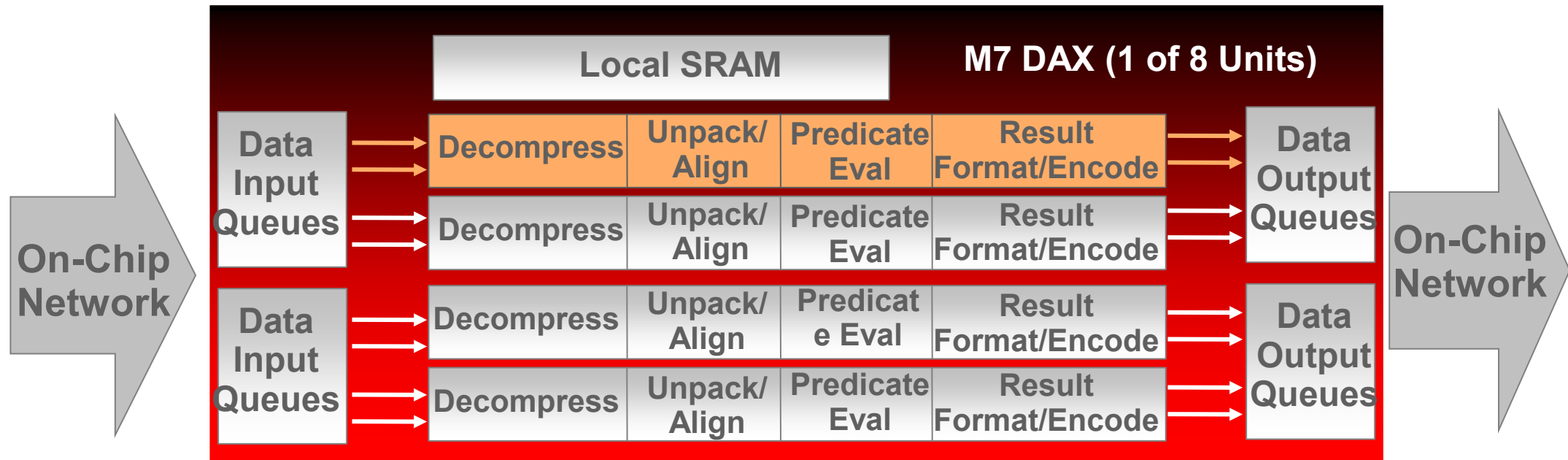
# Silicon Secured Memory in Action

- Heartbleed style buffer overflow attack



- An additional layer on top of all existing security mechanisms. It is not a replacement for encryption
- No performance loss for using Silicon Secured Memory

# SQL in Silicon: Data Analytics Accelerator (DAX)



- Stream Processor for data parallel operations
- DSP-like pipe for efficient filtering operations, typical of first phase of any query
- Cache sparing design with more complex processing in general purpose cores

# DAX Features and Commands

- DAX executes vector commands on entire streams of data
- **Scan:**
  - Does simple equality and range predicates
- **Translate:**
  - Applies Bloom Filter based predicates
- **Select:**
  - Filters input streams, discarding unused data
- **Extract:**
  - Converts memory formats to register friendly formats
  - E.g. Run length expansion (RLE), bit unpacking

- **Usage**

- First step in query plan is usually a WHERE clause  
e.g. `SELECT CAR FROM C  
WHERE C.MAKE = "TOYOTA" ...`
- Further trimming in join steps  
`AND C.DEALERID = D.ID  
AND D.LOC = "San Jose" ...`
- More complicated steps in general purposes cores  
`ORDER BY <distance from me>`

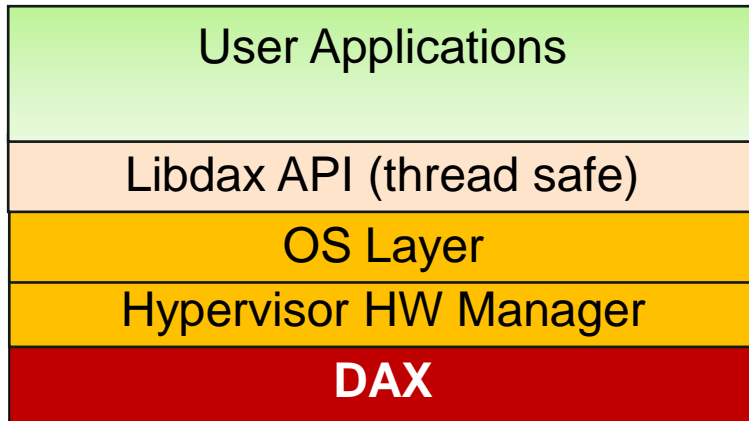
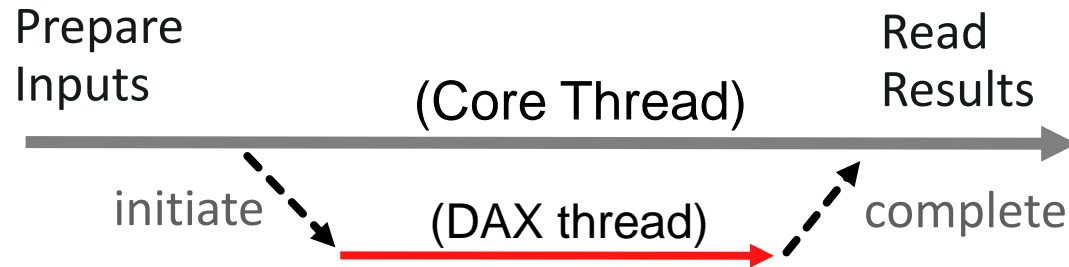
# Capacity in Silicon

- For analytics applications and data warehouses
  - Most data is nicely compressible
  - Most data is immutable
- OZIP
  - Special HW-SW co-designed compression algorithm in Oracle DB 12.1
  - Reduces memory footprint by 2–5x
  - Extremely fast: decompress at close to memory speed, 120GB/s
- Inline decompressors in DAX
  - Decompress data in the path from memory to the general purpose cores.
  - Or decompress inline when running DAX commands  
E.g compressed scan runs filter directly on compressed data

# Agenda

- 1 Motivation For SW in Silicon
- 2 SW in Silicon Features in SPARC M7
- 3 HW – SW Stack Description
- 4 Oracle In Memory Database Performance Measurements
- 5 Performance Results For Use Cases Beyond Oracle Database
- 6 Conclusion

# DAX Programming Model and Task Offload



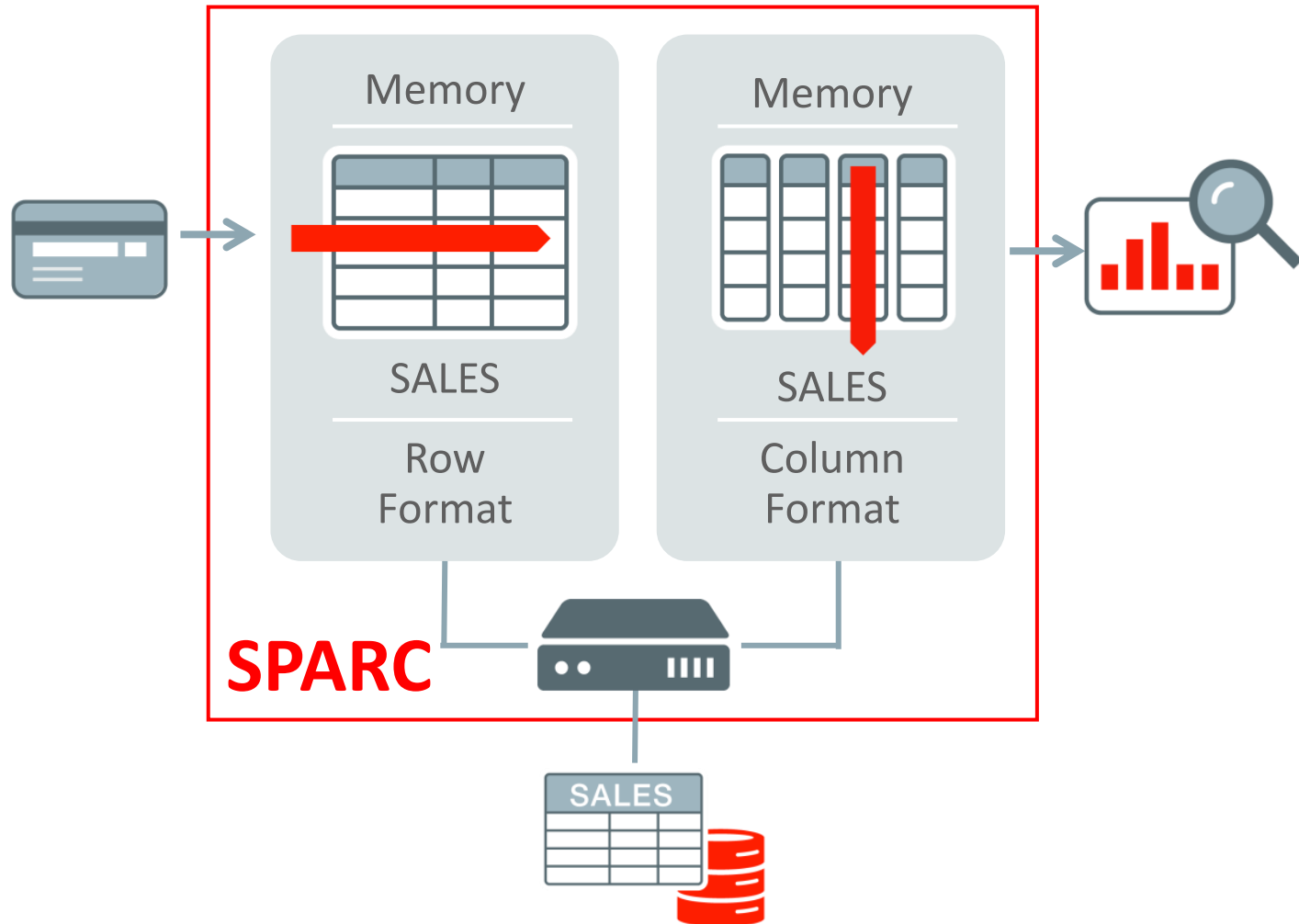
- DAX calls are similar to system calls
- But fork a new “DAX thread” instead of consuming core resources
  - Calling thread continues to execute while DAX function proceeds asynchronously
- Parameters passed through shared memory
  - Results sent back through memory
  - Completion signaling through `mwait()`
- Libdax API hides communication overhead in a simple function call
  - Architectural Invariant
  - Easily called in Java, Scala, Python, C, ...

# Integration with Oracle Database In-Memory

- Full stack integration with the DAX when In-Memory is enabled
- Automatic platform detection
  - Best platform-specific libraries are chosen at run-time
  - DAX enabled libraries are transparently used as appropriate
- Automatic acceleration of database queries
  - No administrative changes required compared to other platforms
  - No application changes required



# Oracle Database In-Memory Dual Format Database



- **BOTH** row and column formats for same table
- Simultaneously active and transactionally consistent
- Analytics & reporting use new in-memory Column format
  - New Analytics Compression means huge amounts of database can now fit in memory
- Automatically detects and uses Software in Silicon on M7
- OLTP uses proven row format

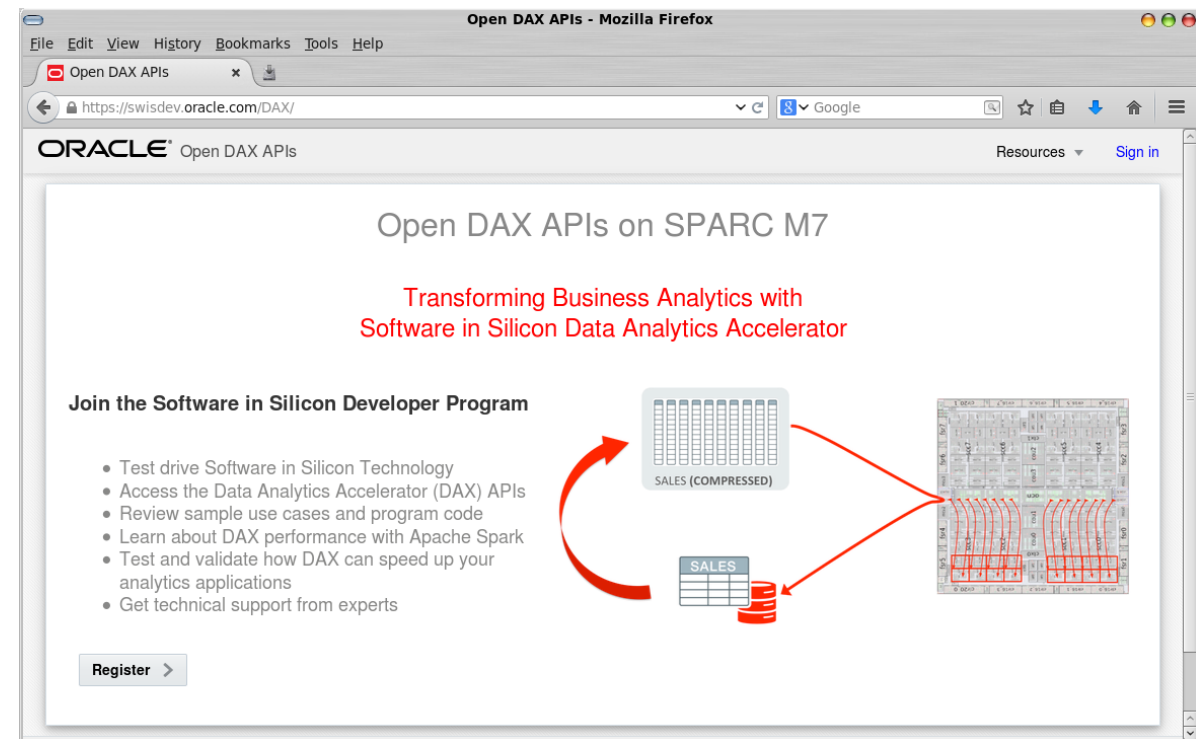
# Oracle Database In-Memory Column Store

- M7 Software in Silicon is **transparently leveraged** based on per-object MEMCOMPRESS compression level and content analysis
- Encoding/compression options and heuristics are specific to each level
- **All levels include Software in Silicon acceleration features**

Compression Level	Description
NO MEMCOMPRESS	Data is populated without any compression
MEMCOMPRESS FOR DML	Minimal compression optimized for Data Manipulation Language performance
MEMCOMPRESS FOR QUERY LOW	Optimized for query performance (default) SPARC M7 enables accelerated RLE processing
MEMCOMPRESS FOR QUERY HIGH	Optimized for query performance as well as space savings SPARC M7 enables OZIP compression with decompress/scan in one op
MEMCOMPRESS FOR CAPACITY LOW	Balanced with a greater bias towards space saving SPARC M7 enables OZIP compression with accelerated decompress
MEMCOMPRESS FOR CAPACITY HIGH	Optimized for space saving

# Libdax Open APIs

- New APIs for SPARC M7/T7/S7 DAX features, available as libdax.so.1
  - Available today on the Software in Silicon Developer Program Portal (see below), will be included in a Solaris 11 support repository update (SRU) later in CY16
  - Provides building blocks for analytics acceleration with any application
  - post and poll commands, debug, log and dtrace probe functions
  - Includes OZIP library so apps can use HW decompression features
- Software in Silicon Developer Portal
  - <https://swisdev.oracle.com>
  - Test drive Software in Silicon technology
  - Learn and share



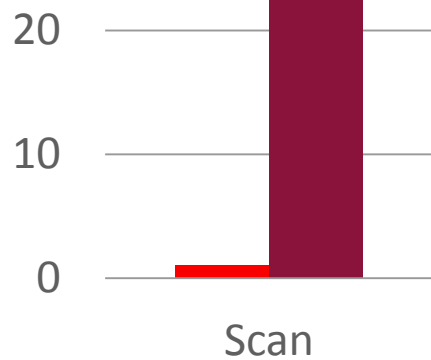
The screenshot shows a Mozilla Firefox browser window titled "Open DAX APIs - Mozilla Firefox". The address bar displays "https://swisdev.oracle.com/DAX/". The page content includes the Oracle logo, the title "Open DAX APIs", and a sub-header "Open DAX APIs on SPARC M7". Below this, a red headline reads "Transforming Business Analytics with Software in Silicon Data Analytics Accelerator". A section titled "Join the Software in Silicon Developer Program" lists several bullet points: "Test drive Software in Silicon Technology", "Access the Data Analytics Accelerator (DAX) APIs", "Review sample use cases and program code", "Learn about DAX performance with Apache Spark", "Test and validate how DAX can speed up your analytics applications", and "Get technical support from experts". A "Register >" button is located at the bottom left of this section. On the right side, a diagram illustrates the data flow: a "SALES" table is connected to a "SALES (COMPRESSED)" table, which in turn is connected to a large data visualization grid. Red arrows indicate the direction of data flow and processing.

# Agenda

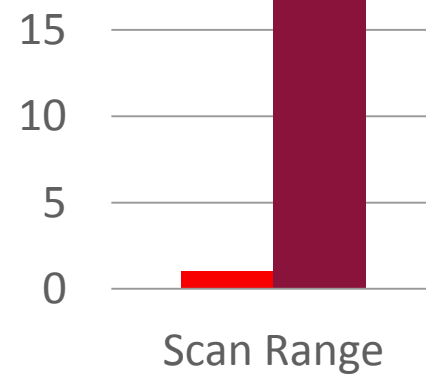
- 1 Motivation For SW in Silicon
- 2 SW in Silicon Features in SPARC M7
- 3 HW – SW Stack Description
- 4 Oracle In Memory Database Performance Measurements
- 5 Performance Results For Use Cases Beyond Oracle Database
- 6 Conclusion

# Timing Traces of Generic vs DAX Accelerated Libraries

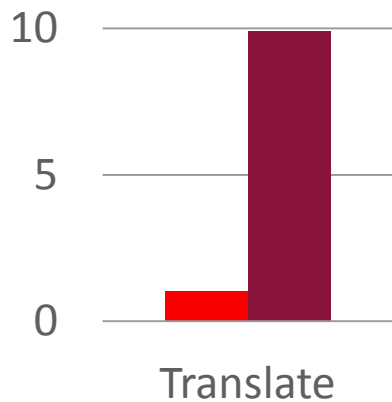
Aggregate of 16 analytic queries based on Scale Factor 1000 Star Schema Benchmark



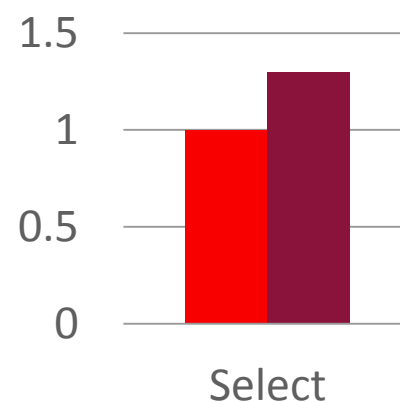
23x speedup of Scan processing



17x speedup of Scan-Range processing



10x speedup of Translate processing (a component of Bloom Filter)

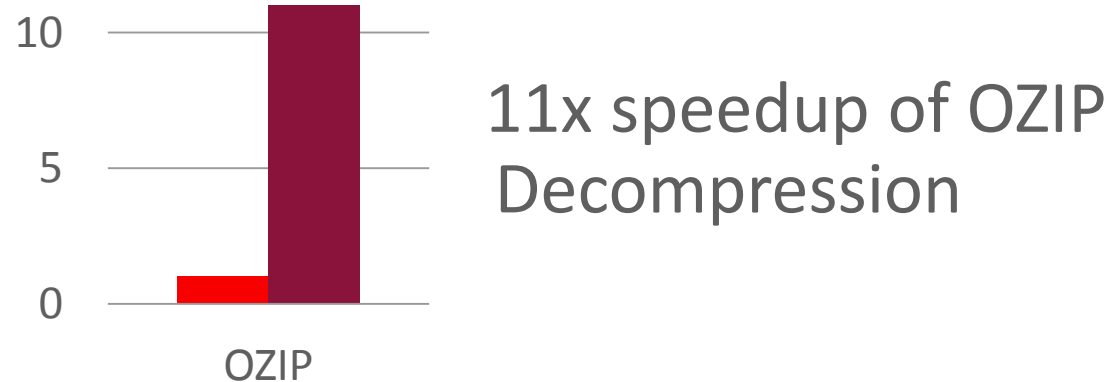
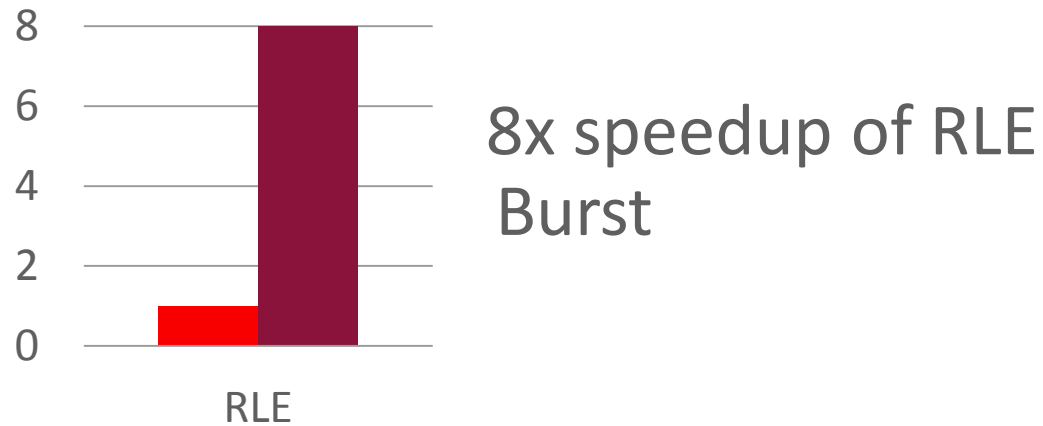


30% speedup of Select processing (a component of Data Projection)

- Includes multiple column widths and encodings with “memcompress for query low”
- For background on Star Schema Benchmark, see: <http://www.tpc.org/tpctc/tpctc2009/tpctc2009-17.pdf>

# Timing Traces of Generic vs DAX Accelerated Libraries

## Decompression Acceleration



- Oracle In-Memory automatically chooses compression algorithm based on MEMCOMPRESS level, performance heuristics, and data content
- Decompression is accelerated by DAX hardware
- Compressed data can also be scanned by the DAX without first decompressing

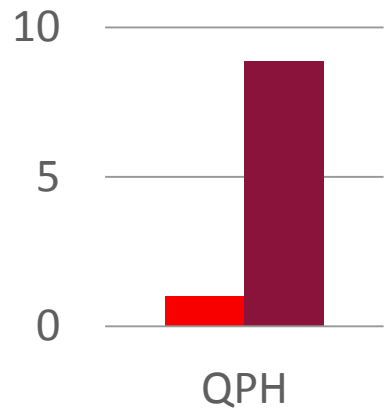
# Performance Gain for Oracle In-Memory

## Generational comparison: SPARC M7 vs SPARC T5

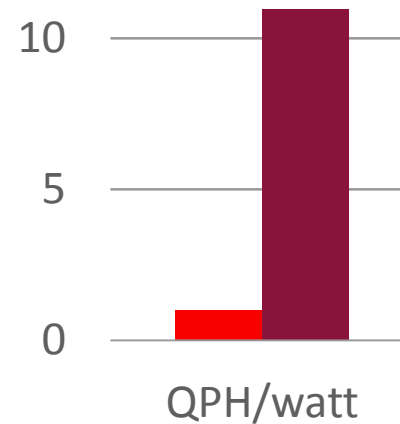
- 4-socket SPARC M7 Logical Domain vs 4-socket SPARC T5-4
- Solaris 11.3, Oracle 12.1.0.2
- Workload based on Scale Factor 1000 Star Schema Benchmark
  - 668 GB uncompressed on-disk
- All tables configured as “inmemory memcompress for capacity low”
  - Enables compression/encodings including full OZIP on both platforms
- Concurrent analytic query streams scaled to 4 processes per core
- Key performance metric: Queries Per Hour (QPH)

# Performance Gain for Oracle In-Memory

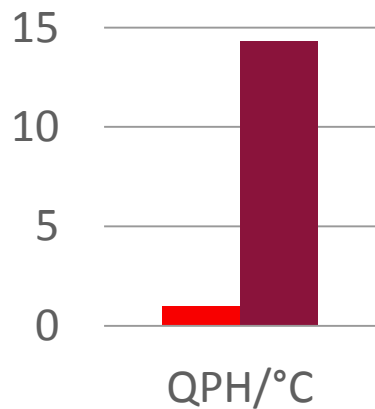
Generational comparison: 4-socket SPARC M7 vs 4-socket SPARC T5



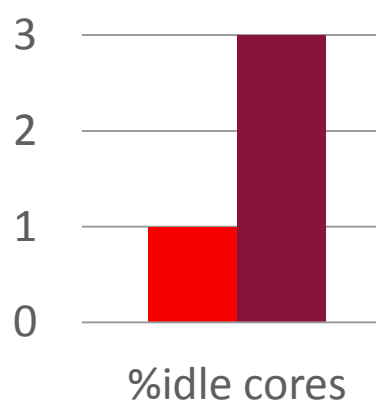
9x throughput gain (Queries per Hour) with M7 Software in Silicon



11x gain in performance per watt (Processor) due to DAX efficiency



14x gain in performance relative to chip temperature



3x reduction in core utilization due to DAX offload – makes resources available for other business processing

- Concurrent analytic query workload with 4 PQ processes per core
- Workload based on Scale Factor 1000 Star Schema Benchmark



# Agenda

- 1 Motivation For SW in Silicon
- 2 SW in Silicon Features in SPARC M7
- 3 HW – SW Stack Description
- 4 Oracle In Memory Database Performance Measurements
- 5 Performance Results For Use Cases Beyond Oracle Database
- 6 Conclusion

# Analytics and Machine Learning (ML) on DAX

Examples of uses and performance

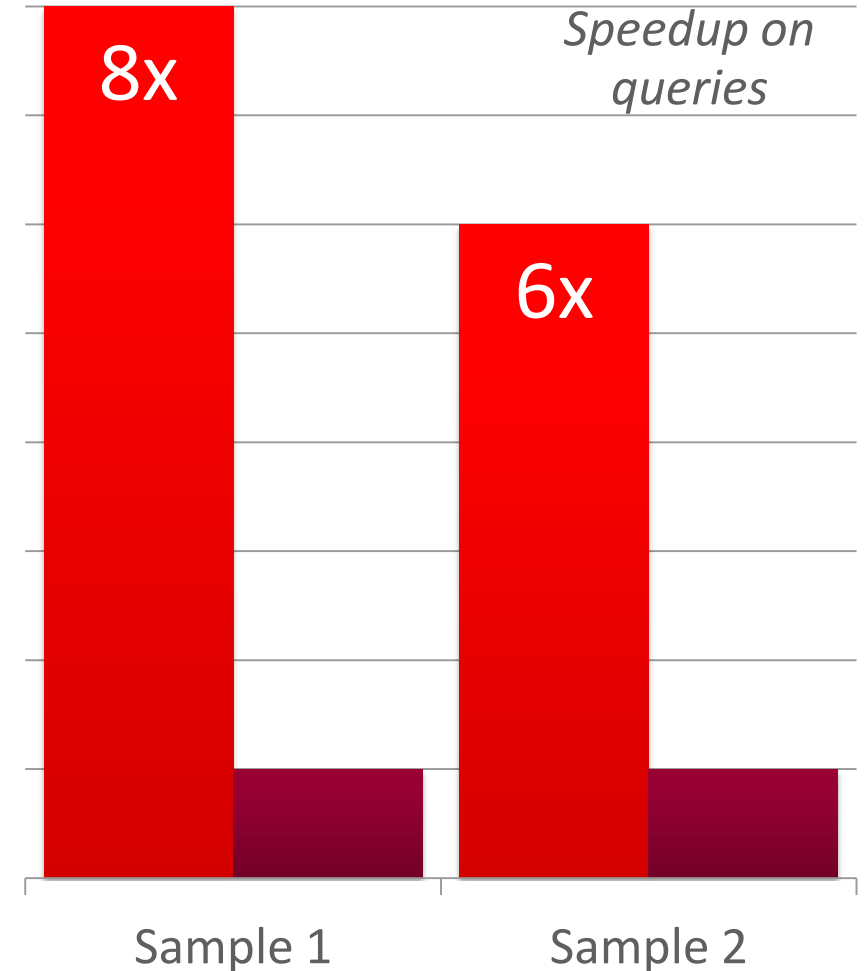
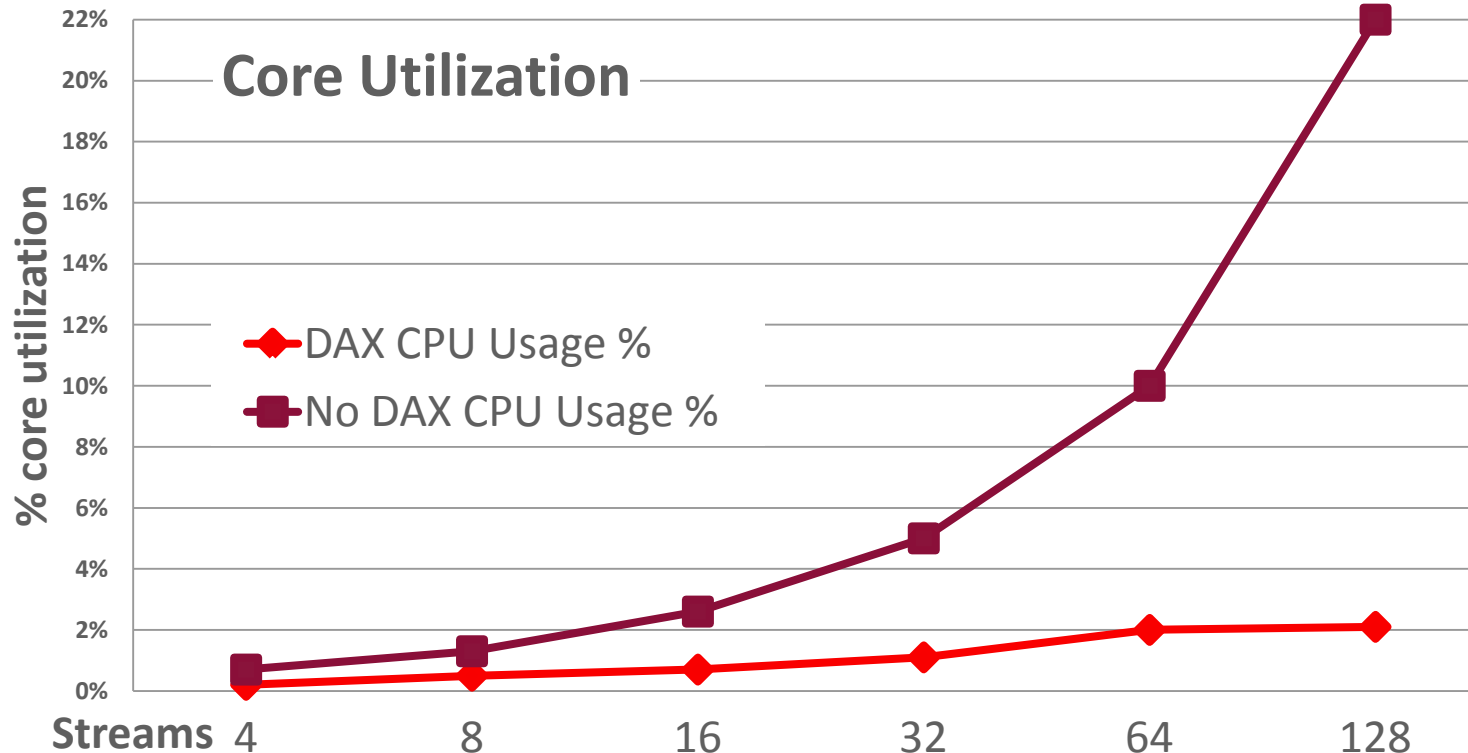
Area	Analytics	DAX Speedup
Machine Learning	K-Nearest Neighbor	4x to 12x
Data Scanning	Top N data in memory	4x to 7x
Data Scanning	SQL on JSON data	4x to 5x
Streaming	Tweet Analysis	5x to 9x

- SPARC DAX Offload acceleration with DAX API
  - Easily used in Java, Scala, Python, C, C++, ...
  - Applicable to a wide variety of algorithms
  - University research is also finding new creative uses of DAX API

# Quartet FS: In-memory Analytical Queries

Queries are 6x to 8x faster with DAX

- Higher query performance with lower core utilization
  - DAX offloads cores, 2x to 8x reduction in core utilization
  - Analytics: [quartetfs.com](http://quartetfs.com)



# Conclusion

- 1 SW in Silicon is custom hardware targeted at specific higher level functions traditionally implemented in software
- 2 Cloud based applications, especially analytics, offer many opportunities for SW in Silicon features
- 3 SW in Silicon in SPARC M7 provides gains in Performance, Power, Security and Memory Capacity
- 4 Oracle database automatically uses these SW in Silicon features. Other applications access through public API
- 5 Oracle is researching tighter HW-SW codesign, targeting deeper gains on a wider set of applications

# Acknowledgments

- The authors wish to acknowledge the contributions of the SPARC M7 architecture, design, firmware, and performance analysis teams, as well as the Oracle Database In-Memory team, for bringing Software in Silicon to fruition, and thank the team members for many helpful suggestions towards this presentation.
- The authors also wish to thank the many Oracle engineers who reviewed this slide set and provided valuable feedback

# Software in Silicon in the Oracle SPARC M7 Processor

Kathirgamar Aingaran  
Sumti Jairath  
David Lutz

**ORACLE**<sup>®</sup>

Copyright © 2016, Oracle and/or its affiliates. All rights reserved. |

ORACLE®