Main-Memory Centric Data Management – Open Problems and Some Solutions

Wolfgang Lehner

Technische Universität Dresden
The Reality: The Petabyte Age

New Realities

- TB disks < $100
- Everything is data
- Rise of data-driven culture
  - CERN’s LHC generates 15 PB a year
  - Sloan Digital Sky Survey (200 GB/night)

The quest for knowledge used to begin with grand theories. Now it begins with massive amounts of data.

Welcome to the Petabyte Age.
The Challenge: The Petabyte Age

The Web is a huge source of information: search engines (Google, Yahoo!) collect and store billions of documents and click streams

- 20 PB processed every day at Google (2008)
- 200 million photos are uploaded to Facebook every day → 2,314 photos/second (2010)
- 60 hours of video uploaded to YouTube every minute (2012)
- By 2015, 1 Zettabyte of data will flow over the internet per day (Cisco Visual Networking Index, June 2011)
- One zettabyte = stack of books from Earth to Pluto 20 times

...and considering sensors/log files/etc., every organisation is able to generate and store massive amounts of data!
How Target Figured Out A Teen Girl Was Pregnant Before Her Father Did

Every time you go shopping, you share intimate details about your consumption patterns with retailers. And many of those retailers are studying those details to figure out what you like, what you need, and which coupons are most likely to make you happy. Target, for example, has figured out how to data-mine its way into your womb, to figure out whether you have a baby on the way long before you need to start buying diapers.

Charles Duhigg outlines in the New York Times how Target tries to hook parents-to-be at that crucial moment before they turn into rampant — and loyal — buyers of all things pastel, plastic, and miniature. He talked to Target statistician Andrew Pole — before Target freaked out and cut off all communications — about the clues to a customer’s impending bundle of joy. Target assigns every customer a Guest ID number, tied to their credit card, name, or email address that becomes a
Main Memory Centric Data Management

What are the main challenges?

Technology perspective

Application perspective
**RAM locality is king**

- Increased CPU calculation speed
- Increased memory bandwidth
- Stagnating memory latency
- Avoid CPU idle time due to missing data
- RAM locality very important
A Look at Current Hardware Trends

Increasing Main Memory Capacity*

„Main Memory“ is the new disk! (?)

Increasing Number of Cores

- CPU/GPU, hybrids
- FPGA (Field Programmable Gate Array)

„Parallelism“ is the name of the game!

Faster Interconnects

* stable RAM will be an additional game changer
Memory Performance Comparison

Motivation for CPU-cache aware data structures, e.g. Vectorwise/X11/Ingres, SAP HANA, Greenplum, Vertica Flexstore, Oracle 11g R2 (PAX)

Is memory the new disk???
- Some characteristics are very similar, e.g. random vs. sequential
- Memory architecture complicates things!

Results for a quad-core i7 2.66GHz, DDR3 1666. 32GB data accessed total.
© Tim Kaldewey

<table>
<thead>
<tr>
<th>Aspect</th>
<th>Memory</th>
<th>Disk</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rand vs. seq</td>
<td>1-2 orders of magnitude</td>
<td>3 orders of magnitude</td>
</tr>
<tr>
<td>Access granularity</td>
<td>Byte addressable in theory, Caches get in the way</td>
<td>1 disk block, usually 4KB</td>
</tr>
<tr>
<td>Writes</td>
<td>Read-modify write (CL)</td>
<td>Read-modify-write (block)</td>
</tr>
<tr>
<td>Concurrency</td>
<td>Parallel memory access for peak performance</td>
<td>Multiple seq. streams random access</td>
</tr>
</tbody>
</table>

© Prof. Dr.-Ing. Wolfgang Lehner | Main-Memory Centric Data Management |
HAEC – Collaborative Research Center
Energy-Adaptive High-Speed Computing Platform

**Optical Interconnect**
- adaptive analog/digital circuits for e/o transceiver
- embedded polymer waveguide
- packaging technologies (e.g. 3D stacking of Si/III-V hybrids)
- 90° coupling of laser

**Radio Interconnect**
- on-chip/on-package antenna arrays
- analog/digital beamsteering and interference minimization
- 100Gb/s
- 100-300GHz channel
- 3D routing & flow management
Project A01
Millimeter-Wave Integrated Circuits for Ultra High-Speed Wireless Board-to-Board Computer Communications

Antennas and Wave Propagation for Adaptive Wireless Backplane Communication

HAEC Box

Network Coding for Wireless and Wired Onboard and Backplane Communication

Sub-THz radio receiver:
Technology: BiCMOS for higher efficiency at target performance
Design approach: positive feedback for \( f_t/2 \) operation

Secure Network Coding for Board-to-Board Communication

25 GHz bandwidth ADC:
Low-resolution for both energy efficiency and fastest operation
...a plea for specialized DB systems

The End of an Architectural Era
(It’s Time for a Complete Rewrite)

Michael Stonebraker
Samuel Madden
Daniel J. Abadi
Stavros Harizopoulos

Nabil Hachem
AvantGarde Consulting, LLC
nhachem@agdba.com

Pat Helland
Microsoft Corporation
phelland@microsoft.com

Implications for the Elephants

- They are selling “one size fits all”
- Which is 30 year old legacy technology that good at nothing
Impact on Database Systems

Implications for the Community

Tons of research projects and startups
Self-made data management platforms

Document Stores
Key Value / Tuple stores

Relational stores

Graph Databases
Multimodel Databases

Database Technology Group
Challenges for Database Systems

“Three things are important in the database world: performance, performance and performance.” Bruce Lindsey

The DBMS Landscape – Performance Needs

Extreme data

OLTP

Data Warehouse

Other apps

Extreme performance

Main-Memory Centric Data Management
Challenges for Database Systems

- Extreme Data
- Extreme Performance
- Extreme Flexibility

Flexibility in Database Systems

+ during deployment time (schema definition)
- during database lifetime (schema evolution)
- during query runtime (scheduling, …)
Flexibility from 10,000 feet

- Applications
  - Querying Web-Tables
  - Role-based object models

- Database System
  - Data comes first, schema comes second
  - Demand flexibility
    - Alternative query model (drill-beyond)
    - Domain-specific data models
    - Statistical operators
  - Storage model-related
  - Record management
  - Logging and persistency
  - HW/SW DB-CoDesign
  - (MV)CC

- Provide flexibility
  - StorageClassMemory
  - MegaCore systems
  - TeraByte-Capacity
  - GPUs
  - FPGAs/FPPAs

- Operating system & hardware

Flexibility is cross-cutting
Some Techniques

• Compression

• Indexing

• Resilience

• (Hardware) Transactional Memory
Compression Overview

Many different compression schemes

Challenges:
- What procedure to apply?
- When to apply compression at all?
Dictionary Encoding

- For each unique value create dictionary entry
- Dictionary can be per-block or per-column
- Column-stores have the advantage that dictionary entries may encode multiple values at once
Frame Of Reference Encoding

- Encodes values as b bit offset from chosen frame of reference
- Special escape code (e.g. all bits set to 1) indicates a difference larger than can be stored in b bits
- After escape code, original (uncompressed) value is written
Early Materialization

Create rows first

Disadvantages:
- Need to construct all tuples
- Need to decompress data
- Poor memory bandwidth utilization
- Loose opportunity for vectorized operation

Select + Aggregate

QUERY:
SELECT custID, SUM(price) 
FROM table 
WHERE (prodID = 4) AND (storeID = 1) AND 
GROUP BY custID
Late Materialization Example

<SQL Query>

```
SELECT custID, SUM(price)
FROM table
WHERE (prodID = 4) AND (storeID = 1) AND
GROUP BY custID
```

<Diagram>

1. ProID = 4, storeID = 1
2. AND
3. Data Source 1
4. Data Source 2
5. custID
6. price
7. AGG (SUM(price))

Main-Memory Centric Data Management
Compression of Intermediates

- Each operator adjusts its output to the requirements of the successive operator
- Compression of base data
- Compression of intermediates
Some Techniques

• Compression
• Indexing
• Resilience
• (Hardware) Transactional Memory
The need for Indexing: Scan vs. Index

- About 40 GB/s scan performance with in-memory Databases
- Real-Time Analytics requires low response time

Graph showing execution time vs. number of threads for Parallel TblScan.

DB Size: 960 GB
4 x 8 (16) cores

- Execution Time [s]:
  - 800
  - 600
  - 400
  - 200
  - 0

# Threads k:
- 1
- 2
- 4
- 8
- 16
- 32
- 64
- 128

- 813 s
- 23 s
- >10^7
- <1 µs

→ Indexing is still necessary for ordering, grouping, point- and range-queries
Index Landscape

KISS-Tree
→ 32bit key index

Buzzard
→ NUMA optimized

Clustered Column Store
→ Scan optimized

Adaptive Radix Tree (ART)
KISS-Tree Overview

Properties
- Specialized version for 32bit keys
- Latch-free updates
- Order-preserving
- 2-3 memory accesses per key

→ Comparable fast to reported order-preserving in-memory indexes for read access
→ BUT:
  High update performance

- Heterogeneous in-memory index structure
- Combination of direct and indirect addressing
- Takes advantage of virtual memory management
- Enables different compression mechanisms
Level 1: Virtual Level

Decimal Key: 327914

\[ \begin{array}{c}
0000 0000 0000 0101 \\
10bit (f_1) \\
\end{array} \quad \begin{array}{c}
0000 0000 11 \\
10bit (f_2) \\
\end{array} \quad \begin{array}{c}
10 1010 \\
6bit (f_3) \\
\end{array} \]

Split key in 3 fragments

- Calculate corresponding 2nd node address from the first fragment
- Direct addressing
- No memory required for level 1

Requirement for Level 2: All nodes have to be stored sequentially in memory
Level 1: Virtual Level

Decimal Key: 327914

0000 0000 0000 0101
16bit ($f_1$)

0000 0000 11 10 1010
10bit ($f_2$) 6bit ($f_3$)

Split key in 3 fragments

Virtual Memory Address

Level 2

4 KB 4 KB 4 KB 4 KB 4 KB 4 KB

Memory Management Unit (MMU)

Page Directory

Physical Memory Address

Operating System

Hardware

Main-Memory Centric Data Management
Level 2: On-demand Level

- Change over to indirect addressing
- 1024 buckets per node containing a compact pointer to the 3rd level node
- 256 MB maximum

Virtual Memory Address

Decimal Key: 327914

16 bit ($f_1$) 0000 0000 0000 0101

10 bit ($f_2$) 0000 0000 11 10 1010

6 bit ($f_3$)

Virtual Memory Address

Level 2

4 KB 4 KB 4 KB 4 KB 4 KB 4 KB ...

4 KB 4 KB

32 bit

$2^{10} = 1024$ pointer to L3 leaf nodes

$2^{16} + 2^{10} = \text{potential L3 leaf nodes of a size between } 2^0 \text{ and } 2^6$
Level 2: On-demand Level

- Allocation of consecutive virtual memory segments for each of the 64 node sizes possible on the third level
- Each segment consists of a maximum of $2^{26}$ blocks of the respective node size

Number of existing elements in L3 leaf node (compression)

Position of L3 leaf node in corresponding memory segment

$\Rightarrow 2^{26}$ blocks of size 1 pre-allocated

$\Rightarrow 2^{26}$ blocks of node size 64 pre-allocated
Level 3: Compression

Decimal Key: 327914

- 16bit ($f_1$)
- 10bit ($f_2$)
- 6bit ($f_3$)

- 64 possible node sizes
- Bitmap indicates which bucket is in use
- Only existing values are stored

Thread-Local Memory Management Subsystem

- Read-copy-updates
- Compare-and-swap
  → No in-place updates possible (lost updates)
**Duplicate Handling**

- Efficient duplicate handling necessary for query processing
  - Scanning a linked list results in random memory accesses

- Page boundaries are a barrier for hardware prefetchers
  - store values sequentially in 4KB blocks
  - blocks grow exponentially until reaching 4KB
  - trade-off between scan performance and memory consumption
Read Performance

Uniform Key Distribution

Single-threaded CSB⁺-Tree: ~3 Million Ops/s
→ KISS-Tree: 18 Million Ops/s with one thread (64M Keys)

Sequential workload

Workloads
8 Byte Payload

Evaluation Hardware
Intel i7-2600 (SMP, 4 cores with Hyper-Threading)
8 MB LLC, 16 GB main memory
Update Performance

Uniform Key Distribution

Sequential workload

Workloads
8 Byte Payload
Evaluation Hardware
Intel i7-2600 (SMP, 4 cores with Hyper-Threading)
8 MB LLC, 16 GB main memory
Some Techniques

- Compression
- Indexing
- Resilience
- (Hardware) Transactional Memory
Motivation: Increasing Error Rates

**Increasing Component Error Rates**
- Decreasing feature sizes (new tech generations)
- Reduced voltage supply
- Static (hard) vs. dynamic (soft) errors
- 8% increase error rate per tech generation [Borkar05]
- 25,000 – 70,000 FIT / Mbit [Schroeder09]

**Increasing System Error Rates**
- Increasing scale
  - # of components (core, transistor)
  - Memory capacities
- Example:
  - Fixed error rate / component

Errors and error-prone behavior will become the normal case
Implicit (silent) vs. Explicit (detected/corrected) Errors
- State-of-the-art: error detection and correction at HW/OS level

State-of-the-Art: Resilient Memory
- ECC / parity bits / memory scrubbing / full data redundancy

ECC Extended Hamming(7+1,4)

State-of-the-Art: Resilient Computing
- Computation redundancy

Such resiliency mechanisms cause „resiliency costs“
Motivation: Resiliency Costs (2)

Resiliency Costs Categories
- Performance overhead (throughput, latency)
- Memory overhead
- Energy consumption
- Monetary HW costs

Resiliency Costs @ OS-Level
- Memory overhead (capacity, bandwidth)
- Computation overhead
- Energy consumption (increased time)

Resiliency Costs @ HW-Level
- Monetary HW costs (Chipset, ECC RAM)
- Energy consumption (time, chip space)
- Computation overhead

Increasing error rates ~ increasing resiliency costs!
Challenge

- Problem: data loss/corruption (explicit/implicit)
- Goal: data stability by data redundancy and error correction

Example (Data Partitioning)

- Table R (a, b, c)
- Data redundancy (synopsis and replicas)

Optimization

- Exploit the multiple replicas (complementary) layouts
- E.g., different sorting orders, partitioning schemes, compression schemes, etc

Test Scheduling
Multiple Replicas
Workload Characteristics
> Resilient Query Processing

**Challenge**

- Problem: missing/invalid tuples (explicit/implicit)
- Goal: reliable query results by error correction / error-tolerant algorithms

**Example (Advanced Analytics)**

- Q: $\Psi_{k=365}(\gamma(\sigma_{a<107}R \bowtie S \bowtie T \bowtie U))$
- Computation redundancy

AR(2): $\hat{y}_t = \phi_1 \cdot y_{t-1} + \phi_2 \cdot y_{t-2}$
Resilient Query Processing (2)

Redundant Query Processing (Triple Modular Redundancy)
- Based on a single query (→ three times execution of an optimal QEP)
- Majority Gate on a single tuple basis
- Drawbacks
  - Assumption of valid raw data (single point of truth)
  - Corrupt raw data are propagated through the model

Alternative
- Based on a single query (→ three different QEPs)
- Majority Gate on a single tuple basis
- Different QEPs based on
  - different operators and
  - in particular on redundant data sources
- Properties
  - no single point of truth
  - data replication is important
Some Techniques

• Compression

• Indexing

• Resilience

• (Hardware) Transactional Memory
Benefit of Lock Elision

Concurrent Execution
- No serialization, no communication if no data conflicts
- Data conflicts are atomically terminated by the underlying TM infrastructure

Example: Hash Table
Main Memory Centric Data Management

What are the main challenges?

Technology perspective  Application perspective
From basic statistics to machine learning and new ways to think about visualization, the Data Science Starter Kit gives you the tools you need to get started with data. If you haven't yet taken the leap, why wait? And if you're already experienced with data, the Starter Kit will push you further. The package includes (8) titles on R, basic statistics and data analysis, Python, machine learning, and visualization.

This includes everything you need to move from analysis, visualization, to management.

A data scientist possesses a combination of analytic, machine learning, data mining and statistical skills as well as experience with algorithms and coding. Perhaps the most important skill a data scientist possesses, however, is the ability to explain the significance of data in a way that can be easily understood by others.
let us map the situation of data analytics to ...
The Five Orders of Ignorance

0th Order Ignorance (0OI)—Lack of Ignorance

- I know something and can demonstrate my lack of ignorance (0OI is knowledge)
- When I have 0OI, I have the answer to the problem.

Mapping to data analytics

- Data Consumption Side
  - no need to look at the data
- Data Cube / Data Space Selection
  - no need to identify, tap, and combine potentially interesting data cubes
- Data Provisioning
  - don’t bother about data analytics
The Five Orders of Ignorance

1st Order Ignorance (1OI) — Lack of Knowledge

- I don’t know something and can readily identify that fact (1OI is basic ignorance)
- When I have 1OI, I have the question.
- Usually, having a good question makes it fairly easy to find the answer

Mapping to data analytics

- Data Consumption Side
  - Reporting!
    - you know the dimension values
    - you are looking specifically for a measure of the specific dimension values
- Data Cube / Data Space Selection
  - no need to actively search for possibly interesting data cubes
    - place where to search your information is well known
- Data Provisioning
  - data sources are well-known and understood; integration/transformation steps exist
The Five Orders of Ignorance

2nd Order Ignorance (2OI) — Lack of Awareness

- I don’t know that I don’t know something
  (not only am I ignorant of something, I am unaware of this fact. I don’t know enough to know that I don’t know enough.)
- Not only do I not have the answer I need, I don’t even have the question.

Mapping to data analytics

- Data Consumption Side
  - Guided Navigation / Explorer-Style
- Data Cube Selection
  - ??? how can I find the 'right' data
- Data Provisioning
  - I don’t know my sources, but I suspect that there is helpful data out there...
3rd Order Ignorance (3OI) — Lack of Process

- I have 3OI when I don’t know a suitably efficient way to find out I don’t know that I don’t know something → lack of process
- If I have 3OI, I don’t know of a way to find out there are things I don’t know that I don’t know

Mapping to data analytics

- Data Consumption Side
  - Data mining („Tell me something interesting!“)
  - Data visualization
- Data Cube Selection
  - I don’t even know how to design my BI entities
- Data Provisioning
  - I don’t know anything about my data sources
Main Memory Centric Data Management

What are the main benefits?

Help business people with an $x^{th}$ Order Ignorance
**Situation in a Typical Organization**

**Corporate EDW(s)**
- IT governed infrastructure
- Well defined with standardized processes, taxonomies, and KPIs
- Top-down control for the enterprise-wide business data

**Dozens of data marts, 100s of local databases, 1000s of spreadsheets**
- Locally defined analytics
- Bottom-up generation and ad-hoc access pattern

Extend DWH Infrastructure

Self-Service/ Agile BI

Data Marts and ‘Shadow’ Databases
~90% of data

EDW
~10% of data

© Prof. Dr.-Ing. Wolfgang Lehner | Main-Memory Centric Data Management | 52
MAD Skills

**Magnetic**
- „Attract data and practitioners“
- Usage of all data source independent of their data quality

**Agile**
- „Rapid iteration: ingest, analyze, productionalize“
- Continuous evolution of the logical and physical structures
- ELT (Extraction, Loading, Transformation)

**Deep**
- „Sophisticated analytics in Big Data“
- Extended algorithmic run-time
- Ad-hoc advanced analytics and statistics
Requirement 1: Balance Performance and Data Volume
- High query performance for interactive analysis / data exploration
- **Challenge:** Huge number of parallel users and ad-hoc query/data schemes

- Batch processing for offline tasks
  (hypothesis generation / consistency checks following complex business semantics)
- **Challenge:** Need to massage massive data volumes with complex business logic

Requirement 2: Support Pull and Push Processing
- Full spectrum of load granularity (batch – trickle feed – stream processing)
- **Challenge:** Need to integrate data streaming systems into DWH infrastructure

Requirement 3: Corporate Memory
- „Expect the unexpected“
- **Challenge:** Manage the organization’s archives and individuals’ memories

- Exist in hybrid environments
- **Challenge:** Provide cloud-enabled DWH environments

> Integrated Information Hub
Situation in a typical Organization

Data is Everywhere !!!

Corporate EDW(s)
- IT governed infrastructure
- well defined with standardized processes and measures
- Top-down control for the enterprise-wide business data

Dozens of data marts,
100s of local databases,
1000s of spreadsheets
- Locally defined analytics
- Bottom-up generation and ad-hoc access pattern

Extend DWH Infrastructure

Self-Service/ Agile BI

Data Marts and ‘Shadow’ Databases
≈90% of data

EDW
≈10% of data

Main-Memory Centric Data Management
Approach: Enable business analysts to

- For business expert users, provide infrastructure to create agile data spaces
- For experienced users, easily run **ad-hoc queries** on complex data warehouses

**Challenges**

- Find the right **input language** for business people (**simplified natural language**)
- Work on **very large and complex** database schemas (hundreds of tables, inheritance patterns)
- **Incrementally** improve meta-model by experience gained from previous ad-hoc queries
- Balance well-structured and IT-governed data flow with ad-hoc-analytics requirements
- **Reduce time to consumption** – no time left for tuning mechanisms
„Drill Beyond“: Extending OLAP using Open Data

- Open World“ approach: Include all external open data sets → open schema

```
select * from nation where nation.gdp > X;
```

Main Problems

- Query segmentation + mapping to external data sources
- Keyword-search over relational data + huge set of open data
- Ambiguities → top-k results + user feedback
- Integration of “open world“ results into local database
The Big Wedding ahead!!!

Data Management

High Performance Computing

Situation in a typical Organization

Data is Everywhere !!!

Corporate EDW(s)
- IT governed infrastructure
- well defined with standardized processes and measures
- Top-down control for the enterprise-wide business data

Dozens of data marts,
100s of local databases,
1000s of spreadsheets
- Locally defined analytics
- Bottom-up generation and ad-hoc access pattern

Extend DWH Infrastructure

Data Marts and 'Shadow' Databases
- 90% of data

Self-Service/ Agile BI
Novel type of applications !!!

*data crunching* meets *number crunching*

- Simulation
- Data Imputation
- Forecasting
- Data Mining (classification, association rules, ...)
- OLAP
- Reporting

computational complexity

data volume
The Netflix Competition

Watch as many movies as you want! For only $7.99 a month.

- Streaming instantly over the Internet to your PC, Mac & TV
- Only $2 more a month to get unlimited DVDs by mail
- Cancel anytime

Questions? 1-866-636-3076 24 hours a day

Unlimited TV episodes & movies instantly.

Connect devices like these to your Netflix account to watch instantly on your TV.

Wii  PS3  XBOX360

Watch as often as you want, anytime you want.

See other devices that stream instantly from Netflix
Earlier this week, Netflix, the online movie rental service, announced it will award $1 million to anyone who can come up with an algorithm that improves the accuracy of its movie recommendation service.

In doing so, the company is putting out a call to researchers who specialize in machine learning—the type of artificial intelligence used to build systems that recommend music, books, and movies. The entrant who can increase the accuracy of the Netflix recommendation system, which is called Cinematch, by 10 percent by 2011 will win the prize.

Recommendation systems such as those used by Netflix, Amazon, and other Web retailers are based on the principle that if two people enjoy the same product, they are likely to have other favorites in common too.

But behind this simple premise is a complex algorithm that incorporates millions of ratings, tens of thousands of items, and ever-changing relationships between users and preferences.
\[ \hat{r}_{ui} = b_{ui} + \frac{1}{\|N(u)\|} \sum_{j \in N(u)} e^{-\beta_{ui} |t_{ui} - t_{uj}|} c_{ij} + \frac{1}{\|R(u)\|} \sum_{j \in R(u)} e^{-\beta_{ui} |t_{ui} - t_{uj}|} ((r_{uj} - \tilde{b}_{uj}) w_{ij}) + \sum_{j \in R(u)} e^{-\gamma_{ui} |t_{ui} - t_{uj}|} ((r_{uj} - \tilde{b}_{uj}) d_{ij}). \]
The NetFlix Competition (4)

Database Technology Group

Alice (1.98)

Bob (1.21)

Michael (2.30)

About Schmidt (2.24)

Lost In Translation (1.92)

Sideways (1.18)

(2.7) (2.3) (2.7)
A simple experiment ...
> ... our test object

color code := user rating

different movies
Phase 1: drop 75% of all pixels
The Experiment ...

Phase 2: Random permutation of rows and columns
The Experiment ...

Phase 3: Determine the latent factors

<table>
<thead>
<tr>
<th>Phase 3</th>
<th>Determine the latent factors</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Value 1</th>
<th>Value 2</th>
<th>Value 3</th>
<th>Value 4</th>
<th>Value 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>99.05</td>
<td>101.3</td>
<td>3.99</td>
<td>1.50</td>
<td>99.96</td>
</tr>
<tr>
<td>98.50</td>
<td>97.45</td>
<td>8.00</td>
<td>4.98</td>
<td>100.0</td>
</tr>
<tr>
<td>923.7</td>
<td>114.7</td>
<td>2.98</td>
<td>1.01</td>
<td>99.37</td>
</tr>
<tr>
<td>97.91</td>
<td>94.18</td>
<td>3.97</td>
<td>2.02</td>
<td>99.35</td>
</tr>
<tr>
<td>103.4</td>
<td>103.7</td>
<td>7.97</td>
<td>4.03</td>
<td>99.60</td>
</tr>
<tr>
<td>101.4</td>
<td>106.6</td>
<td>9.22</td>
<td>2.07</td>
<td>100.4</td>
</tr>
<tr>
<td>100.8</td>
<td>100.8</td>
<td>9.55</td>
<td>2.06</td>
<td>99.55</td>
</tr>
</tbody>
</table>

Main-Memory Centric Data Management
## Phase 4: Reconstruction

<table>
<thead>
<tr>
<th>Phase</th>
<th>Accuracy</th>
<th>Speed</th>
<th>Latency</th>
<th>Throughput</th>
<th>Extra Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>99.06</td>
<td>101.5</td>
<td>8.99</td>
<td>1.50</td>
<td>99.96</td>
<td>100.0</td>
</tr>
<tr>
<td>98.30</td>
<td>92.45</td>
<td>8.00</td>
<td>4.98</td>
<td>100.0</td>
<td>99.71</td>
</tr>
<tr>
<td>96.47</td>
<td>114.7</td>
<td>2.98</td>
<td>1.01</td>
<td>99.37</td>
<td>101.5</td>
</tr>
<tr>
<td>97.91</td>
<td>94.18</td>
<td>3.97</td>
<td>2.02</td>
<td>99.35</td>
<td>101.3</td>
</tr>
<tr>
<td>103.4</td>
<td>103.7</td>
<td>7.97</td>
<td>4.03</td>
<td>99.60</td>
<td>100.8</td>
</tr>
<tr>
<td>108.4</td>
<td>108.4</td>
<td>7.98</td>
<td>2.02</td>
<td>100.1</td>
<td>100.85</td>
</tr>
</tbody>
</table>
Phase 5: Final Result Generation
The Experiment ...
Main Memory Centric Data Management

- requires a holistic picture from system level to end-user experience
- data management layer is still the name of the game
- novel applications will combine the complexity of number-crunching and data-crunching

Huge impact on

- design of analytics infrastructures
- design of database systems
- implementation of analytical applications

**to finally conclude: Recap**

- 5 Orders of Ignorance
  - 0OI - Lack of Ignorance
  - 1OI - Lack of Knowledge
  - 2OI - Lack of Awareness
  - 3OI - Lack of Process
  - 4OI?

4th Order Ignorance (4OI) — Meta Ignorance

I have 4OI when I don’t know about the Five Orders of Ignorance.

... we hopefully passed that stage!

The quest for knowledge used to begin with grand theories. Now it begins with massive amounts of data.

Welcome to the Petabyte Age.
Main-Memory Centric Data Management – Open Problems and Some Solutions

Wolfgang Lehner

Technische Universität Dresden