SAP HANA: Delivering A Data Platform for Enterprise Applications on Modern Hardware

HANA Platform Team
Anil K. Goel and Shel Finkelstein, SAP
BECAUSE WE CAN!!
Agenda

Who is SAP?
Why did SAP decide to build HANA?
What is SAP HANA™?
How are customers using HANA?
Where is HANA going next?
Conclusion
Agenda

Who is SAP?
Why did SAP decide to build HANA?
What is SAP HANA?
How are customers using HANA?
Where is HANA going next?
Conclusion
Who was SAP (before HANA)?
74% of the world’s transaction revenue touches an SAP system.
How Did the SAP Use Database Before HANA?

See “The SAP Transaction Model: Know Your Applications”, SIGMOD 2008 Industrial Talk

- Database was mainly a dumb store …
  - Retrieve/Store data (Open SQL, no stored procedures)
  - Transaction commit, with locks held very briefly
  - Operational utilities

- … because SAP kept the following in the application server:
  - Application logic
  - Business object-level locks
  - Queued updates
  - Data buffers
  - Indexes

With the HANA platform, computation-intensive data-centric operations are moved to the Database
Simplify Technology Stack with the SAP HANA Platform

Insight to Action

Applications

Analytics

SAP HANA Platform
Mobile First, Cloud, Open Platform

Mobile First Experience

- Applications
- Analytics

SAP HANA Platform

Partners & Startups
Agenda

Who is SAP?

Why did SAP decide to build HANA?

What is SAP HANA?

How are customers using HANA?

Where is HANA going next?

Conclusion
### DRAM Price/GB

<table>
<thead>
<tr>
<th>Year</th>
<th>Price/GB</th>
</tr>
</thead>
<tbody>
<tr>
<td>2013</td>
<td>$5.50</td>
</tr>
<tr>
<td>2010</td>
<td>$12.37</td>
</tr>
<tr>
<td>2005</td>
<td>$189</td>
</tr>
<tr>
<td>2000</td>
<td>$1,107</td>
</tr>
<tr>
<td>1995</td>
<td>$30,875</td>
</tr>
<tr>
<td>1990</td>
<td>$103,880</td>
</tr>
<tr>
<td>1985</td>
<td>$859,375</td>
</tr>
<tr>
<td>1980</td>
<td>$6,328,125</td>
</tr>
</tbody>
</table>

In-Memory Computing

Yes, DRAM is 125,000 times faster than disk, but DRAM access is still 10-80 times slower than on-chip caches.

Using Intel Ivy Bridge for approximate values. Actual numbers depends on specific hardware.
Software Advances: Build for In-Memory Computing
Reduce Memory Access Stalls

- **In-Memory Computing**: It is all data-structures (not just tables)
- **Parallelism**: Take advantage of tens, hundreds of cores
- **Data Locality**: On-chip cache awareness
Enterprise Workloads are Read Dominated

Workload in Enterprise Applications consists of:
- Mainly read queries (OLTP 83%, OLAP 94%)
- Many queries access large sets of data
Evolution of traditional business models is happening faster and faster

→ Interactive
→ Real-time
→ Complex

From gridlocks to smart cities of the future
From transactions to 1:1 engaging relationships
From unpredictability to 360º view of risk and reward
From generic treatments to personalized medicine
From mass production to 3-D manufacturing

Business transformation
The explosive pace of tech trends
Creating a ripple effect that compels us to re-think how we conduct businesses

1 billion people in social networks
Rewires business and personal boundaries

Data doubling every 18 months, 80% with locations
Creates new opportunities and risks for value creation

Personalization Squared
What if machine learning starts to anticipate your interests & desires?

More devices than people
Require fresh thinking designed for an “always-on” world
Leading Businesses Across Industries are Showing the Way

Transforming Business with Database & Technology

From Mass Retail to 1:1 interaction Experiences

From Manufacturing to Full Services Experiences

From Passive Commerce Enabler to Active Market-Maker
Agenda

Who is SAP?
Why did SAP decide to build HANA?

What is SAP HANA?
How are customers using HANA?
Where is HANA going next?

Conclusion
One Global Database Platform Team
SAP HANA Collaborative Research

- Students and alumni: [http://scn.sap.com/docs/DOC-26824](http://scn.sap.com/docs/DOC-26824)

University collaborators at PhD level include:
- TU Dresden, Prof. Wolfgang Lehner
- University of Mannheim, Prof. Guido Moerkotte
- TU München, Prof. Alfons Kemper & Prof. Thomas Neumann
- ETH Zürich, Prof. Donald Kossmann
- EPFL, Prof. Anastasia Ailamaki
- HPI, Prof. Hasso Plattner
- DHBW Mannheim, Prof. Carsten Binnig
- TU Ilmenau, Prof. Kai-Uwe Sattler
- TU Karlsruhe, Prof. Peter Sanders
- University of Heidelberg, Prof. Michael Gertz
- University of Toronto, Prof. Ryan Johnson & Nick Koudas
- Conversations with others in-progress
SAP HANA Database
Multi-Engine for Different Application Needs
SAP HANA Technology & Features
Combined in one DBMS Platform

In-memory DBMS
- Exploit SSD/disk for spilling, aging/archiving, durability/fault-tolerance

Standard RDBMS features
- SQL, stored procedures
- ACID, MVCC with snapshot isolation, logging and recovery

Focus on column store
- Late materialization and decompression
- Row store capability, e.g. for system catalogs

High Performance
- Efficient compression techniques
- Parallelization at multiple levels
- Scanning operations co-optimized with hardware

Reduced TCO and administration
- Avoid indexes, aggregates and materialized views, with exceptions (like primary key indexes)
In-Memory Computing – Data Structures

<table>
<thead>
<tr>
<th>Order</th>
<th>Country</th>
<th>Product</th>
<th>Sales</th>
</tr>
</thead>
<tbody>
<tr>
<td>456</td>
<td>France</td>
<td>corn</td>
<td>1000</td>
</tr>
<tr>
<td>457</td>
<td>Italy</td>
<td>wheat</td>
<td>900</td>
</tr>
<tr>
<td>458</td>
<td>Italy</td>
<td>corn</td>
<td>600</td>
</tr>
<tr>
<td>459</td>
<td>Spain</td>
<td>rice</td>
<td>800</td>
</tr>
</tbody>
</table>

**SAP HANA: column order**

**Typical Database**

```
SELECT Country, SUM(sales) FROM SalesOrders
WHERE Product = 'corn'
GROUP BY Country
```
SAP HANA: Dictionary Compression

Column „Name“ (uncompressed)

<table>
<thead>
<tr>
<th>Value IDs</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Miller</td>
</tr>
<tr>
<td>1</td>
<td>John</td>
</tr>
<tr>
<td>5</td>
<td>Millman</td>
</tr>
<tr>
<td>0</td>
<td>Zsuwalski</td>
</tr>
<tr>
<td>1</td>
<td>Baker</td>
</tr>
<tr>
<td>2</td>
<td>John</td>
</tr>
<tr>
<td>3</td>
<td>Johnson</td>
</tr>
<tr>
<td>4</td>
<td>Jones</td>
</tr>
<tr>
<td>5</td>
<td>Miller</td>
</tr>
<tr>
<td>5</td>
<td>Millman</td>
</tr>
</tbody>
</table>

Value-ID array
One element for each row in column

Column „Name“ (dictionary compressed)

Dictionary

<table>
<thead>
<tr>
<th>Value IDs</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Baker</td>
</tr>
<tr>
<td>1</td>
<td>John</td>
</tr>
<tr>
<td>2</td>
<td>Johnson</td>
</tr>
<tr>
<td>3</td>
<td>Jones</td>
</tr>
<tr>
<td>4</td>
<td>Miller</td>
</tr>
<tr>
<td>5</td>
<td>Millman</td>
</tr>
</tbody>
</table>

Value ID, implicitly given by sequence in which values are stored

point into dictionary

sorted
Additional Compression Technologies

1) Prefix Coding

<table>
<thead>
<tr>
<th>Uncompressed</th>
<th>Prefix Coded</th>
</tr>
</thead>
<tbody>
<tr>
<td>44444444353110</td>
<td>84353110</td>
</tr>
</tbody>
</table>

- value
- Number of occurrences

2) Run Length Coding

<table>
<thead>
<tr>
<th>Uncompressed</th>
<th>Run length encoded</th>
</tr>
</thead>
<tbody>
<tr>
<td>55544444332222</td>
<td>5432</td>
</tr>
</tbody>
</table>

- value
- start position

3) Cluster Coding

<table>
<thead>
<tr>
<th>Uncompressed</th>
<th>N=6, cluster size=4</th>
</tr>
</thead>
<tbody>
<tr>
<td>444444444444333333331000000</td>
<td>44433311000</td>
</tr>
</tbody>
</table>

- Cluster coded
- Bit vector 110101 Bit 1=1 if cluster i was replaced by single value

4) Sparse Coding

<table>
<thead>
<tr>
<th>Uncompressed</th>
<th>Sparse Coded</th>
<th>Bitvector</th>
</tr>
</thead>
<tbody>
<tr>
<td>44433100000</td>
<td>433100000</td>
<td>11100000011110</td>
</tr>
</tbody>
</table>

- Removes the value v which occurs most often. Bit vector indicates positions of v in the original sequence.

5) Indirect Coding

<table>
<thead>
<tr>
<th>Uncompressed</th>
<th>block wise dictionary coding of Value-IDs</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 126 576 55 126 2 2 55 881 212 3 19 461 792 45 13 911 28 28 13 13 911 911</td>
<td></td>
</tr>
</tbody>
</table>

- Compressed (conceptual)
- Block 2 is not compressed
- Block size=8
- Dictionary for Block 1
- Dictionary for Block 3
SAP HANA Column Store

**Column Main: Read-optimized store for immutable data**
- High data compression
- Efficient compression methods (dictionary and run-length, cluster, prefix, etc.)
  - Dictionary values for main are sorted in same order as data
- Heuristic algorithm orders data to maximize secondary compression of columns
- Compression works well, speeding up operations on columns (~ factor 10)

**Column Delta: Write-optimized store for inserts, updates and deletes**
- Less compression of data
- Data is appended to delta to optimize write performance
- Unsorted dictionary on delta helps speed write performance
- Delta is merged with main periodically, or when thresholds are exceeded
  - Delta merge for a table partition is done on-line, in background
  - Enables highly efficient scan of Main again
SAP HANA: Multi-Core Parallelization

Core 1

processed by

Col A
1000032
67867868
2345
8986757
234123
21
2342343
78787
9999993
13427777
23423
123123123
1212
2009
454544711

Core 2

processed by

Col B
4545
76
6347264
435
3434
1252
342555
3333333
8789
4525323
6767312
789976
20002
2346098
78787

Col C
2500
21
78675
3432423
89089
562356

processed by

Core 3

Core 4

processed by
Parallelization at All Levels

- Multiple user sessions
- Concurrent operations within a query (… T1.A … T2.B…)
- Data partitioning on one or more hosts
- Horizontal segmentation, concurrent aggregation
- Multi-threading at Intel processor core level
- Vector processing
Parallel Aggregation Execution

n Aggregation Threads
• Each thread fetches a small part of the input relation.
• Aggregate part and write results into a small hash-table.
  – If the entries in a hash-table exceed a threshold, the hash-table is set aside, and aggregation in that thread continues on a new empty hash-table.

m Merge Threads
• Merge thread aggregate a specific range and write results into a private part hash-table.
• The final result is obtained by concatenating all the part results.
Parallel Join Execution

- Like aggregations, joins can be computed using hash-tables.
- **Build Phase**: Parallel computation of part hash-tables on the smaller table, Table A
- **Probe Phase**: Probing of the larger table, Table B, against the part hash-tables:
  - A set of hash-maps is created in parallel.
  - Local hash-maps are compared with part hash-maps.
Parallel OLAP Execution

- Column processing, with late materialization and decompression
- Distribution and parallelism at many levels
  - Dimension tables may be replicated
- Generalized calculation operations

Single Instruction Multiple Data (SIMD)

**Scalar processing**
- traditional mode
- one instruction produces one result

**SIMD processing**
- with Intel® SSE
- one instruction produces multiple results

### Diagram

**SOURCE**
- X
- Y
- DEST
- XopY

**SSE/2/3 OP**
- X4
- X3
- X2
- X1
- Y4
- Y3
- Y2
- Y1
- X4opY4
- X3opY3
- X2opY2
- X1opY1

**DEST**
- 127
- 0
SAP HANA Use of SIMD

SIMD-Scan: Ultra Fast in-Memory Table Scan using on-Chip Vector Processing Units, VLDB 2009, by Intel, SAP and HPI

- 3.2B Codeword Scans / Second / Core  
  - A “codeword” is a compressed integer value
- 12.5M Aggregates / Second / Core
- 1.5M Inserts / Second (Load)

- Use vector-based processing (SIMD)
- Leverage data locality
- Act on compressed data, not word aligned
- Cache line (64 Bytes) aligned data
- Hyper-threading
Vectorizing Database Column Scans with Complex Predicates
VLDB 2013 ADMS Workshop

- AVX2-Scan consistently 30% faster than SIMD-Scan.
- Throughputs between 4 and 10 billion codewords per second with peaks of 17 billion.
Two AVX2 instructions are key to achieving the parallelization of the Scan with in-list predicate algorithm:

- Vector-vector shift instruction: The new vector-vector shift instruction allows shifting each word of the AVX register with an independent value. We use it to convert the value into word where only the bit at the index that equals the value is set to 1.
- Gather instruction: The new gather instruction loads elements from memory based on a base address and offsets for each data element. We use it to gather the different chunks of the predicates relevant to vectorized comparison.
HANA Core Platform

ONE platform for simple and efficient data processing

SAP HANA Core Engine

Client Interfaces / Session Layer

Query Compilation + Execution / DDL / DML

Row Store

Column Store

Data Persistence

Non-Relational Stores (GIS, Graph, Flexible, …)

Modeling

Operations & Administration

Data Quality

Federation

Data On Disk

Replication

Big Data

Streaming / CEP

© 2014 SAP AG or an SAP affiliate company. All rights reserved.
Petabyte Test; see HANA Performance White Paper

The System
Single Instance of SAP HANA
100 Servers / 4,000 Cores / 100 TB RAM

The Data
1 Petabyte of Raw Data
10 Years / 1.2 Trillion rows of ECC SD Data

Customer
1,000,000 across 26 countries
Material
1,000,000 across 10 groups

Sales Organization
999
Division
9
Value Type
10

Distribution Channel
9
Currency
256
Unit of Measure
256

SD Fact Tables
1,200,000,000,000 rows
61 columns

The Queries
Representative BI queries with varying complexity from moderate to very complex
18 Distinct Queries
• General Reporting
• Iterative Drill-Down
• Ranking
• Year-Over-Year Analysis
Ranging in Complexity
• Multiple Joins
• In-Lists
• Sub Queries
• Correlated Sub Queries
• Union All

All queries were parallelized across all nodes in the cluster.

The Results
Throughput: 60 Streams running 112,602 queries per hour, representing > 5,000 concurrent BI users
Response Time: See below for individual queries.
SAP HANA Technology Innovation

- Column and row store
- Insert only on change
- Minimal projections
- Text Retrieval & Exploration
- No aggregates
- Analytics on historical data
- Multi-core/parallelization
- Multi-threading within nodes
- In-memory Compression
- Active/passive & data aging
- SQL interface on columns & rows
- Single and multi-tenancy
- Dynamic Extensibility
- Reduction of tiers / layers
- Partitioning
- SQL
- Map reduce
- Group Key
- In-memory Apps
- Bulk load
- Geo-data

Insert only on change
Minimal projections
Text Retrieval & Exploration
No aggregates
Analytics on historical data
Multi-core/parallelization
Multi-threading within nodes
In-memory Compression
Active/passive & data aging
SQL interface on columns & rows
Single and multi-tenancy
Dynamic Extensibility
Reduction of tiers / layers
Partitioning
SQL
Map reduce
Group Key
In-memory Apps
Bulk load
Geo-data
Agenda

Who is SAP?
Why did SAP decide to build HANA?
What is SAP HANA?
How are customers using HANA?
Where is HANA going next?
Conclusion
SAP Runs SAP
Building the Foundations

**Mobile**
- P&L to Go
- Management Contract Cockpit
- Board Transparency App

**Cloud for People**
- fully integrated with on-premise HCM

**Cloud for Customer**
- fully integrated with on-premise CRM

**Ariba**
- fully integrated with on-premise ERP

**Travel on Demand**
- fully integrated with on-premise ERP

**BW on HANA**
- 4,500+ business users
  - SAP’s mainstream capabilities for financial management reporting representing single point of truth

**CRM on HANA**
- 15,000+ business users
  - Big Bang Implementation – All Users, All Geographies

**ERP on HANA**
- 65,000+ business users
  - Mission critical ERP system - All users, All geographies

Powered by **SAP HANA**
Customer Study: Dunning Run

Dunning run determines all open and due invoices
• Analyzes Days Sales Outstanding and history and determines action

Running SAP customer’s queries on 250M records
• Before HANA: 20 min
• After HANA: 1.5 sec

Speed-up from factors including:
• In-memory column store
• Parallel stored procedures
High Performance Application: Customer Engagement Intelligence

“We saw an opportunity with **SAP Audience Discovery & Targeting** to influence our customers’ buying behavior and reduce product return rates. We sell over 1 million products to 8 million customers and estimate that **decreasing our return rate by only 1% can lead to 7-digit Euro savings** on the bottom line. SAP Audience Discovery & Targeting enables our marketing teams to **uncover hidden trends driving product returns** across different customer groups and build **high precision marketing campaigns** to target these groups and influence their return behavior.”

--Michael Künzl, VP IT Systems, Home Shopping Europe (HSE24)
Moving Complex Application Logic Close to the Data

Application logic deals with unit/measure pairs
- Calculation engine handles selection and aggregation for heterogeneous units

Data-driven conversion during query execution enables “what if” simulations
- Quantity unit conversion for stock reports, logistics, etc.
- Planning with disaggregation

<table>
<thead>
<tr>
<th>Product Group</th>
<th>Product</th>
<th>Price</th>
<th>Quantity</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Beverages</strong></td>
<td>Cola</td>
<td>1.50 € / bottle</td>
<td>5 bottles</td>
<td>7.50 €</td>
</tr>
<tr>
<td></td>
<td>Milk</td>
<td>3.00 $ / quart</td>
<td>2 quarts</td>
<td>6.00 $</td>
</tr>
<tr>
<td><strong>Bev Total</strong></td>
<td></td>
<td>X</td>
<td>12.9 liters</td>
<td>11.88 €</td>
</tr>
<tr>
<td><strong>Vegetables</strong></td>
<td>Onions</td>
<td>1.00 € / kg</td>
<td>3 kg</td>
<td>3.00 €</td>
</tr>
<tr>
<td></td>
<td>Tomatoes</td>
<td>2.00 € / kg</td>
<td>5 kg</td>
<td>10.00 €</td>
</tr>
<tr>
<td><strong>Veg Total</strong></td>
<td></td>
<td>X</td>
<td>7 kg</td>
<td>13.00 €</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td>X</td>
<td>*</td>
<td>24.88 €</td>
</tr>
</tbody>
</table>
Top Ten Sales & Opportunity Planning (S&OP) Value Drivers Enabled by SAP HANA

1. Unlimited Simulation: What-if changes by anyone, anytime, on the full detailed demand-supply-finance model!
2. Scenario Comparison: Compare and promote scenario plans without compromising speed and model size.
3. Instant Financial Plan Impact: Change your plans and immediately see the effect on costs and gross and net profit across the entire business.
4. Real-time Alerts: No more waiting for the alert to show up, it’s already there.
5. Real-time Customizable Analytics: See the impact of changes as you click refresh in state-of-art analytics!
6. Real-time Constrained Planning: Change demand and see the impact based on supply constraints – without having to get a cup of coffee.
7. Load in the Details: Bring in detailed data from Advanced Planning and Optimization (APO) and any other systems without pre-aggregating to simplify data integration among other things.
8. Calculate at the Detailed Levels: Get accurate results by computing financial and supply calculations at low levels. Even seemingly simple calculations like revenue will dramatically improve.
9. Drill Down, all the Way Down: Work at the aggregate level (e.g., by Family) and easily drill down to specific areas within the same view in Excel or analytics (and change the detailed mix).
10. Drive Execution: We keep and compute details by product, resource, customer, component, etc. So the data is ready for use in APO, other planning systems and your ERP—to drive even more value!

See S&OP Blog for more information.
1. Simplify the Core Business Processes To Uncover More Value

**Simplified Financials**
- Improved federal tax calculation
  - Improved federal tax calculations (PIS / Cofins) and adhering to Brazilian government’s legal requirements within the stated deadline
  - Petróleo Brasileiro S.A

**Simplified Manufacturing**
- 16x improvements results in multi-million dollar savings
  - Multi-million savings through 16x improvement in delivery of critical material list for manufacturing by implementing BW on HANA
  - Large Auto Manufacturer

**Simplified Inventory**
- 0.5% increase in monthly revenue
  - 0.5% increase in monthly revenue through improving the fill-rate of outstanding sales orders by enabling ad-hoc inventory allocation
  - Under Armour
Foster More Breakthroughs
To Enable New Business Models

**CONNECTED CAR**
3% customer fuel and cost reduction

Made possible through predictive maintenance based on sensor data

*Pirelli*

**PERSONALIZED MEDICINE**
1000x faster tumor data analysis

Makes it possible to improve cancer treatment with new patient therapies

*Charite*

**IN-MOMENT PROMOTION**
Real-time promotion led up to 0.25% possible margin improvement

On slow and non-moving products

*Liverpool*
Create More Experiences
To Share The Fruits Of Innovation With Everyone

TRANSFORM
USER
EXPERIENCE

Transform user experiences of ERP to extend its value to everyone in enterprise

190 SAP Fiori apps on HANA

TRANSFORM
DEVELOPER
EXPERIENCE

Sales Pipeline & Commission App for 5000 employees, reduce codes by 97%

SAP River used by NetApp

TRANSFORM
SPORTS FAN
EXPERIENCE

Enrich fan experiences, improve player performance, and maximizes digital ad revenue

Hoffenheim

TRANSFORM
CONSUMER
EXPERIENCE

Smart vending machine & Precision Marketing

TSM
Agenda

Who is SAP?
Why did SAP decide to build HANA?
What is SAP HANA?
How are customers using HANA?

Where is HANA going next?

Conclusion
Continuing Challenges of Emerging Hardware

- Challenge 1: Parallelism: Take advantage of tens, hundreds, thousands of cores
- Challenge 2: Large memories & data locality/NUMA
  - Yes, DRAM is 125,000 times faster than disk…
  - But DRAM access is still 10-80 times slower than on-chip caches
HANA Platform On-Going Architectural Evolution

Data models
- Flexible schemas, graph functionality, geospatial, time series, historical data, Big Data, external libraries

Resource and workload management
- Memory, threads, scheduling, admission control, service level management, data aging

Application services
- XS Engine, CDS and River

Continuing performance improvements
- Hardware advances, NUMA, improved modularization and architecture

Cloud and multi-tenancy
Co-innovating the Next Big Wave in Hardware Evolution

Multi-Core and Large “Memory” Footprints

Storage Class Memories / Non-Volatile Memory
- Leverage as DRAM and/or as persistent storage

On-Board DIMMs
- Very high density, byte-addressable
- DRAM like (< 3X) latency and bandwidth; similar endurance
- Compete with disk on cost/bit by 2020

Extreme Speed Network Fabric/Interconnects
- Inter-socket NUMA gets worse while inter-host NUMA gets better
- Inter-socket and Inter-host latencies converge

Exploiting Dark Silicon for Database Hardware Acceleration
- Also exploit GPUs for specific use cases, such as regression analysis
Agenda

Who is SAP?
Why did SAP decide to build HANA?
What is SAP HANA?
How are customers using HANA?
Where is HANA going next?

Conclusion
SAP HANA: Delivering A Data Platform for Enterprise Applications on Modern Hardware

HANA is a new database platform built for modern hardware

HANA is designed for both existing and new enterprise applications

SAP and partners ship applications on HANA that help businesses run better

- More complex business models
- Faster answers
- Current data
- Simpler administration
- Better user experience
More Information about SAP HANA

About SAP HANA
HANA Developer Center
HANA Marketplace
HANA Academy
HANA Product and Solutions Center
HANA on Public Cloud
Customer Stories
Customer Reviews
SAP HANA: Delivering A Data Platform for Enterprise Applications on Modern Hardware

Questions?

Anil K. Goel, anil.goel@sap.com
Shel Finkelstein, shel.finkelstein@sap.com

Jobs at SAP:  http://jobs.sap.com

© 2014 SAP AG or an SAP affiliate company. All rights reserved.