



Migration von Oracle auf HANA – Was bedeutet das?

Peter Heintzen, Bereichsleiter
28.06.2014

DOAG
SIG Oracle & SAP

Facts & Figures

Geschäftsform ▶ Inhabergeführte AG

Hauptsitz: ▶ Ratingen

Gründung ▶ 1984

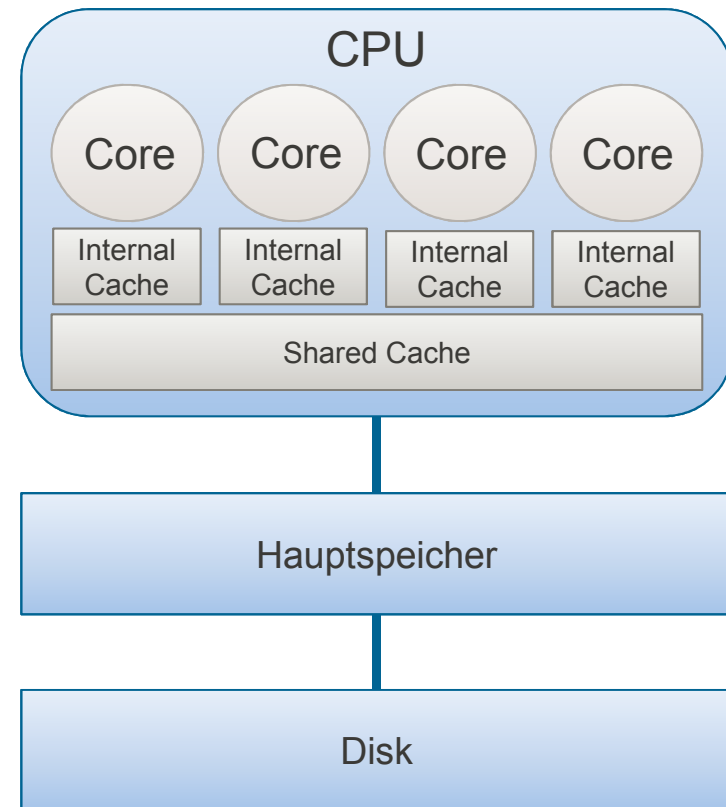
Beschäftigte ▶ 180 Mitarbeiter

Beteiligungen ▶ MT-ifs GmbH (Ratingen)
MT-ifs Sarl (Luxemburg)

**Danke an Oracle und SAP, daß sie
gemeinsam einen Markt öffnen zu
innovativen Business Szenarien und
Investitionen.**

Technologische Möglichkeiten verbindet bestehende Konzepte neu in InMemory

- Multicore CPUs, Hyper-Threading ermöglichen eine Massive Parallelverarbeitung
- Big Data hat Modelle, wie Map Reduce, populär gemacht und reifen lassen, die eine Teilung und Parallelisierung von Aufgaben ermöglichen.
- Großer Hauptspeicher ist verfügbar und bezahlbar geworden.



Was sind typische Szenarien, die Kunden mit HANA verbessern wollen?

Warum HANA?

Data Governance in Echtzeit

Investitionsschutz besonders für SAP basierte Systeme

Verbesserung der Datenqualität

SQL basierte Anwendungen

- ✓ Operationales Reporting und Dashboards mit Daten in "Echtzeit"
- ✓ Echtzeit Risikoanalysen
- ✓ Visualisierung von On-the-Fly-Analysedaten und -Prognosen
- ✓ Erfüllung der Erwartungshaltung an Antwortzeitverhalten mobiler Geräte

Warum HANA?

Data Governance in Echtzeit

Investitionsschutz besonders für SAP basierte Systeme

Verbesserung der Datenqualität

SQL basierte Anwendungen

- ✓ Verfügbarkeit von zukünftigen Produkt- / Modul-Innovationen nur im Zusammenspiel mit HANA
- ✓ Es existiert ein Migrationspfad für alle SAP Produkte und Lösungen zu HANA
- ✓ Zu erwartende zügige Zertifizierung von neuen Releases auf HANA
- ✓ Tiefe SAP Integration, z.B. Cubes optimiert und getuned für SAP BW

Warum HANA?

Data Governance in Echtzeit

Investitionsschutz besonders für SAP basierte Systeme

Verbesserung der Datenqualität

SQL basierte Anwendungen

- ✓ Vermeidung von Datenkopien, Summationstabellen oder zusätzlichen Daten-Cubes durch Berechnung dieser in Echtzeit (Fast Path Access).
- ✓ Nach Möglichkeit keine Datenredundanz – “Single point of truth”
- ✓ Weniger ETL Prozesse

Warum HANA?

Data Governance in Echtzeit

Investitionsschutz besonders für SAP basierte Systeme

Verbesserung der Datenqualität

SQL basierte Anwendungen

- ✓ Konsolidierung weiterer Anwendungen im Zusammenspiel mit SAP Modulen
- ✓ Aufgelöstes SAP Datenmodell
- ✓ Übergreifende Selects sind möglich (“Integration by SQL”)

SAP HANA-Architektur

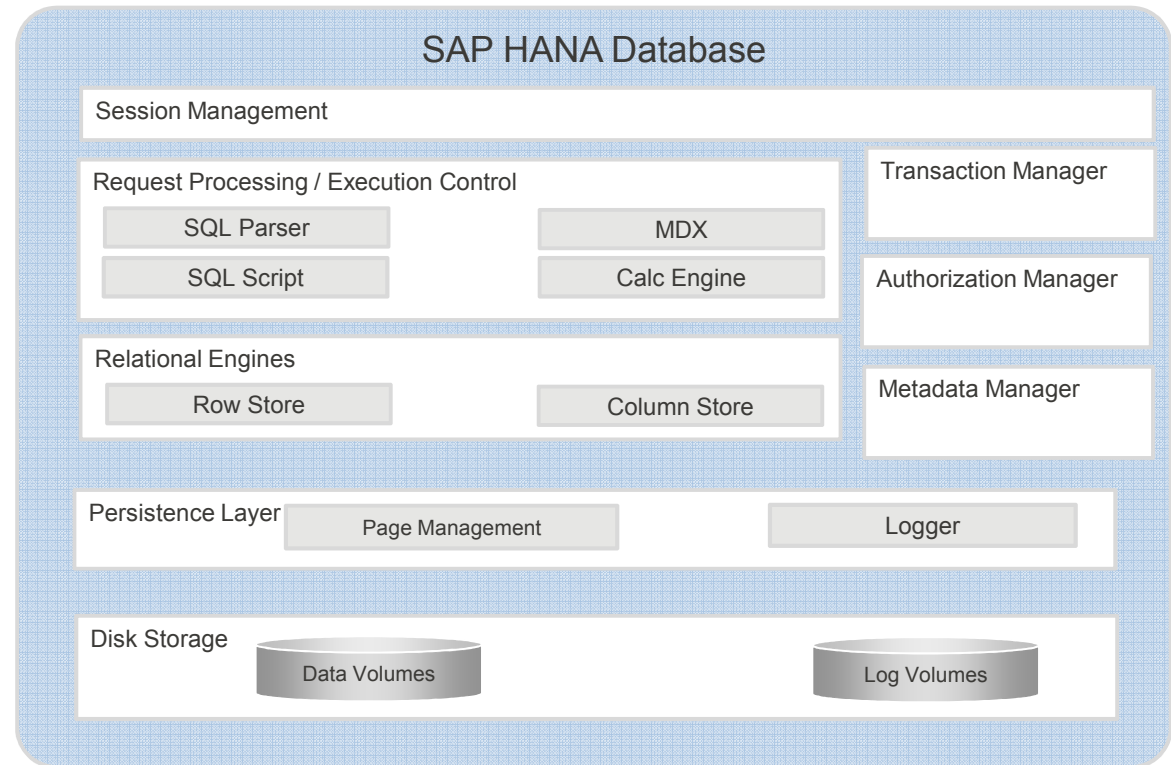
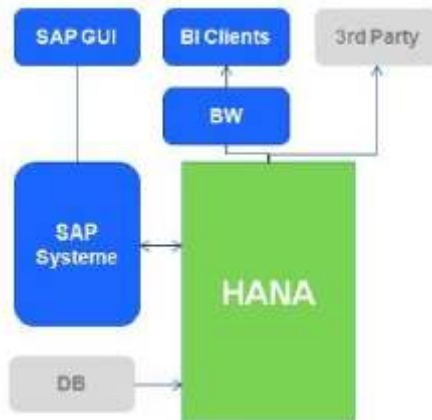
SAP HANA – High Performance Analytic Appliance

Datenbank

- InMemory Datenbank
- Hybrider Ansatz von zeilen- und spaltenorientierter Datenverwaltung (Transaktion vs. Analyse)
- MPP (massively parallel processing)
- Standardzugriffsmöglichkeiten

Appliance

- Multi-Core & 64-Bit Unterstützung



Wie einfach ist der Weg zu HANA?

Datenbankmigration auf HANA – Jetzt ist alles schnell?

- Eine 1:1 Datenbankmigration ohne Veränderung der Datenmodelle, SQL Statements etc. bringt meistens in Summe eine Verbesserung in Prozenten, nicht in Faktoren.
- Bei SAP Modulen bzw. SAP BW sieht es besser aus, da Development schon entsprechende erste Performance Massnahmen eingearbeitet hat im Zusammenspiel mit HANA.
- Die Anwendung ist getuned und ausgelegt auf ein RDBMS mit möglichst hoher Parallelität im IO. Jetzt ist hohe Parallelität in der Verarbeitung durch die Cores gefragt.
- Weitere Fragen:
 - Was ist mit Datenbanklogik in PLSQL?
 - Was ist mit Security?
 - Systemarchitektur / NFRs?
 - Lohnen alle Daten in die HANA mitzunehmen?

Ein Wechsel von einem RDBMS zu HANA ist ein Architekturwechsel

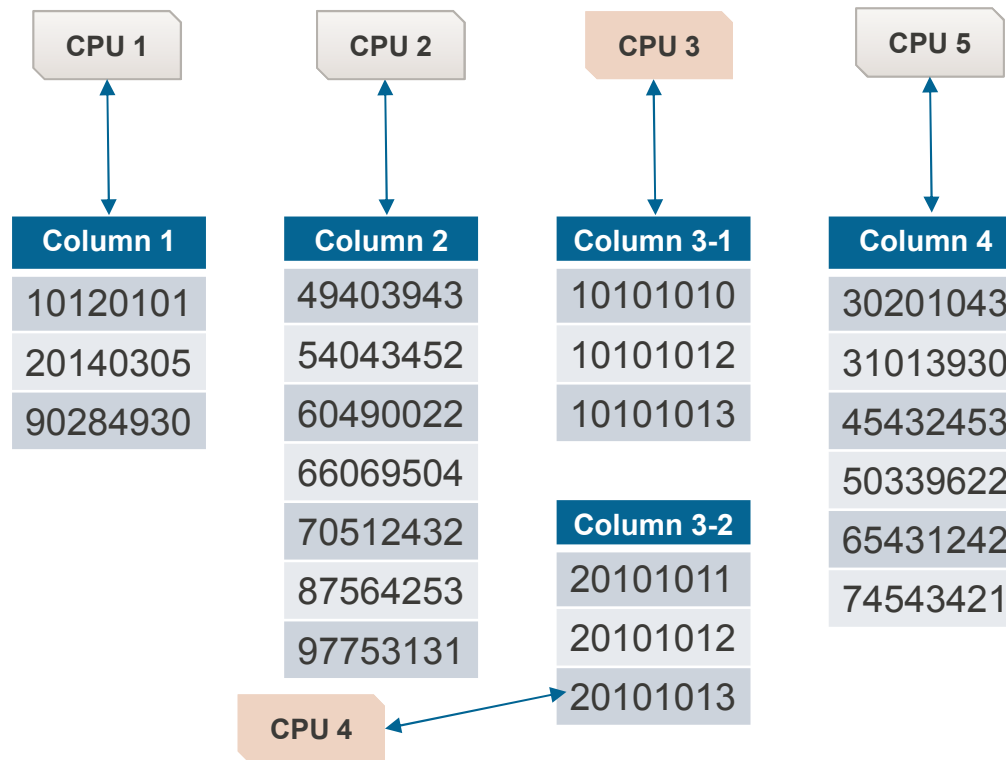
- Methodisches Vorgehen klassisch mit Analyse, Design, Implementierung, Test
- In der Analyse Delta ermitteln, was man vielleicht nicht mehr zur Verfügung hat oder anders tun muss.
- Wenn ich Faktoren will muss in der Zielarchitektur gedacht werden !
- Datenmodell
 - Voraggregierte Views entfernen
 - Bei Spaltenorientierter Speicherung macht Denormalisierung öfters Sinn. Kompression eliminiert die Doppelten.

Ein Wechsel von einem RDBMS zu HANA ist ein Architekturwechsel

- Daten
 - Welche Daten muss ich dafür auf die HANA mitnehmen? Welche Daten sind z.B. nur Compliance-relevant und werden vorgehalten?
 - Wie groß wird der Datenhotspot (ca. 92% der Zugriffe)?
 - Wo lege ich die restlichen Daten ab? Mögliche Beispiele:
 - Warm: Nearline Storage, Kalt: Sybase IQ
 - Verbleiben auf Oracle DB und über Smart Data Access Tabellen als Extern definieren
 - Spaltenorientierte Speicherung besteht immer aus zwei Teilen; ein Main und einem Delta Bereich, die zusammen eine logische Einheit bilden. Im Delta Bereich werden die Datenveränderungen in einem anderen Format hinterlegt - Schnell für Veränderungen, nicht optimal für Analysen. Über Merge werden die Daten in den Column Store eingefügt.
 - Bei Bulk Loads oder Imports Merge explizit abschalten und danach wieder einschalten

SQL Performance durch Partitionierung

SAP HANA Performance (Massively Parallel Processing)



Quelle: Hasso-Plattner-Institut

Parallelisierung durch Zerlegung in SQLScript

```
create procedure "_SYS_BIC"."HANA_DEV/MARA_TEST" ( in IV_MANDT INTEGER ,
out mara_ids "_SYS_BIC"."HANA_DEV/MARA_TEST/tabletype/mara_ids" , out
out_mat_roh "_SYS_BIC"."HANA_DEV/MARA_TEST/tabletype/out_mat_roh" , out
out_mat_halb "_SYS_BIC"."HANA_DEV/MARA_TEST/tabletype/out_mat_halb" )
language SQLSCRIPT sql security definer default schema "HANA_DEV" reads sql
data as
```

```
BEGIN
```

```
-- Query 1
```

```
mara_ids = select "MATNR","ERSDA","ERNAM","MTART"
              from "HANA_DEV"."MARA"
              WHERE "MANDT" = :IV_MANDT
              AND "MTART" = 'HALB'
              OR "MTART" = 'ROH'
              ORDER BY "ERSDA";
```

```
-- Query 2
```

```
out_mat_roh = SELECT "MTART",
                    count(*) as cnt FROM :mara_ids
                    WHERE "MTART" = 'ROH'
                    GROUP BY "MTART";
```

```
-- Query 3
```

```
out_mat_halb = SELECT "MTART",
                     count(*) as cnt FROM :mara_ids
                     WHERE "MTART" = 'HALB'
                     GROUP BY "MTART";
```

```
END;
```

Beispiel für SQLScript Struktur
nicht für Parallelisierung

SELECT-Statement ohne Parallelisierung

```
ET_AGENCIES = SELECT A.AGENCYNUM, T.NAME, T.POSTCODE,  
T.CITY, T.COUNTRY, A.PAYMENTSUM, A.CURRENCY  
FROM (SELECT TOP 5 B.AGENCYNUM, SUM (B.LOCCURAM) AS  
PAYMENTSUM, B.LOCCURKEY AS CURRENCY  
FROM : IT_CONNECTIONS AS C INNER JOIN SBOOK AS B ON  
B.CARRID = C.CARRRID AND B.CONNID = C.CONNID  
WHERE B.MANDT = :IV_MANDT AND B.CANCELLED <> 'X'  
GROUP BY B.AGENCYNUM, B.LOCCURKEY  
ORDER BY SUM(B:LOCCURAM) DESC ) AS A  
INNER JOIN STRAVELAG AS T ON  
T.AGENCYNUM = A.AGENCYNUM WHERE T.MANDT = :IV_MANDT;
```

Quelle: Buch „ABAP Entwicklung für SAP HANA“

SELECT-Statement mit Parallisierung

```
LT_AGENCIES = SELECT TOP 5 B.AGENCYNUM,  
SUM (B.LOCCURAM) AS PAYMENTSUM, B.LOCCURKEY AS CURRENCY  
FROM : IT_CONNECTIONS AS C INNER JOIN SBOOK AS B ON  
B.CARRID = C.CARRRID AND B.CONNID = C.CONNID  
WHERE B.MANDT = :IV_MANDT AND B.CANCELLED <> 'X'  
GROUP BY B.AGENCYNUM, B.LOCCURKEY  
ORDER BY SUM(B:LOCCURAM) DESC;
```

```
ET_AGENCIES = SELECT A.AGENCYNUM, T.NAME, T.POSTCODE, T.CITY,  
T.COUNTRY, A.PAYMENTSUM, A.CURRENCY  
FROM :LT_AGENCIES AS A INNER JOIN STRAVELAG AS T  
ON T.AGENCYNUM = A.AGENCYNUM  
WHERE T.MANDT = :IV_MANDT;
```

Quelle: Buch „ABAP Entwicklung für SAP HANA“

Oracle PLSQL und SAP HANA SQLScript

Oracle PLSQL

```
CREATE OR REPLACE PROCEDURE cursor_procl IS
  v_isbn VARCHAR(20);
  cnt INTEGER;
  CURSOR c_cursor1(v_isbn VARCHAR(20))
    is SELECT book_id FROM books WHERE book_id= v_isbn;
BEGIN
  OPEN c_cursor1('123');
  FETCH c_cursor1 INTO v_isbn;

  IF c_cursor1%NOTFOUND THEN
    msg_proc('Cursor not found');
  ELSIF c_cursor1%ROWCOUNT>1 then
    cnt := c_cursor1%ROWCOUNT;
  END IF;

  CLOSE c_cursor1;
END;
```

SAP HANA SQLScript

```
CREATE PROCEDURE cursor_procl LANGUAGE SQLSCRIPT
  AS
  v_isbn VARCHAR(20);
  cnt INTEGER;
  CURSOR C_CURSOR1(v_isbn VARCHAR(20))
    FOR SELECT book_id FROM books WHERE book_id = v_isbn;
BEGIN

  OPEN C_CURSOR1('123');
  FETCH C_CURSOR1 INTO v_isbn;

  IF C_CURSOR1::NOTFOUND THEN
    CALL msg_proc('Cursor not found');
  ELSEIF C_CURSOR1::ROWCOUNT > 1 then
    cnt := C_CURSOR1::ROWCOUNT;
  END IF;

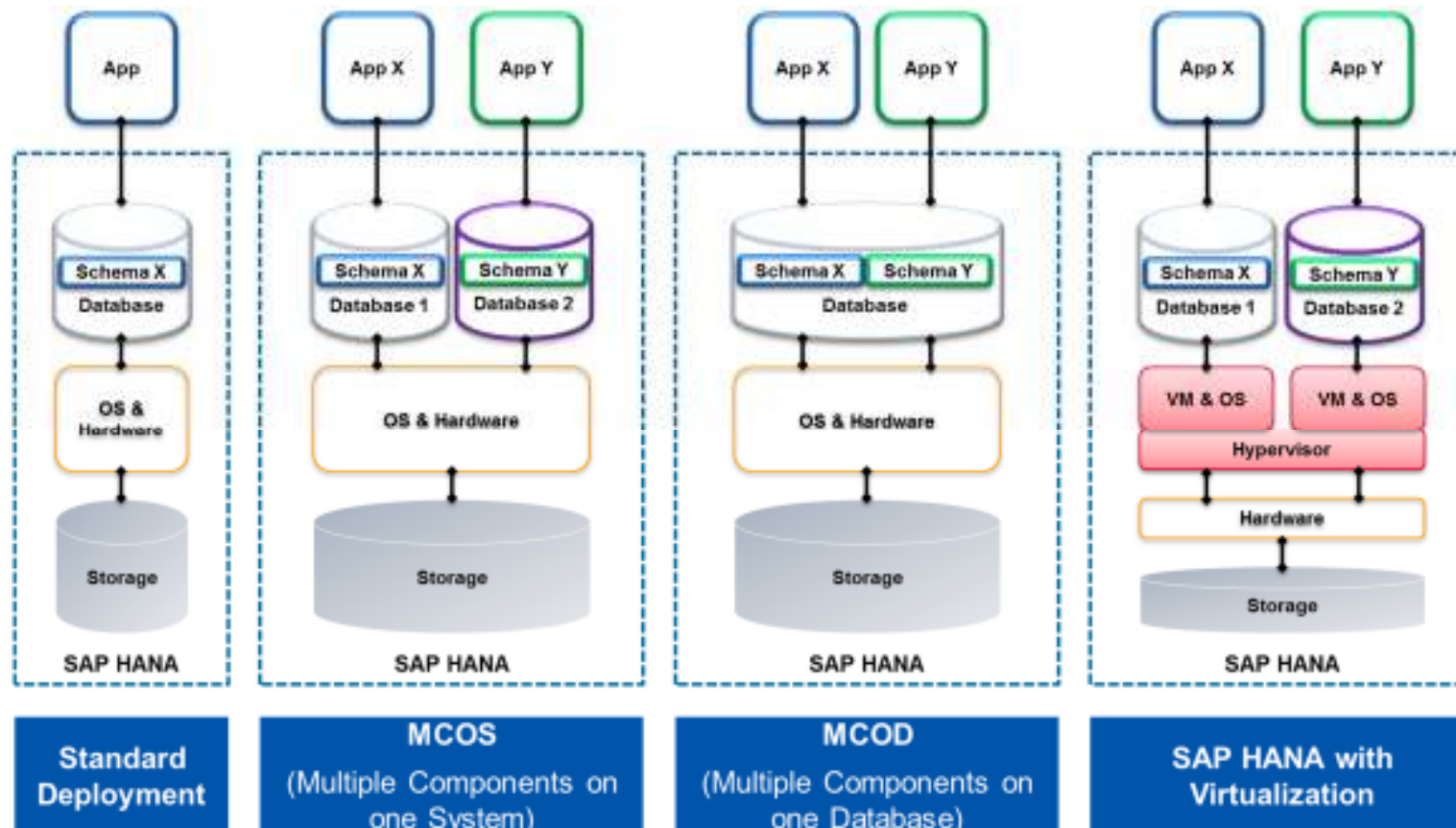
  CLOSE C_CURSOR1;
END;
```

Quelle: lspirer.de

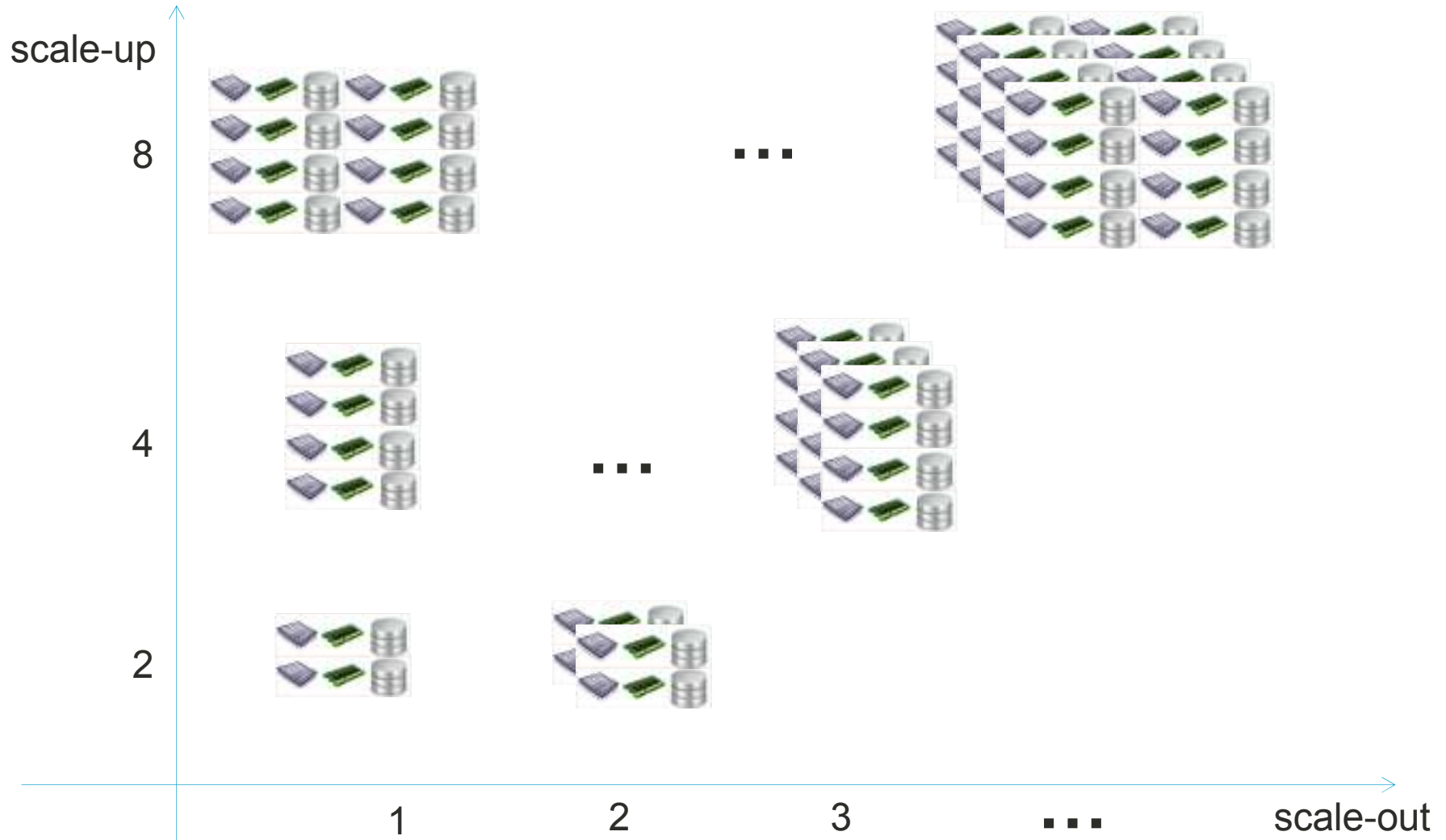
Ein Wechsel von einem RDBMS zu HANA ist ein Architekturwechsel

- PLSQL->SQLScript
 - Es ist möglich Code weitestgehend zu migrieren, da die Sprachen erst einmal eine sehr ähnliche Syntax haben.
 - Es gibt erste Tools (z.B. Ispirer), die eine automatische Codemigration anbieten.
 - In der Praxis läuft es oft auf eine Re-Implementierung hinaus, da wir in beiden Welten doch auf unterschiedliche Limitationen bzgl. Trigger etc. treffen.

Deployment-Varianten



SAP HANA Deployment Options



Mein Fazit – HANA wird den Weg in die installierte SAP Basis finden





Vielen Dank.

Peter Heintzen

Bereichsleiter Integration Services

Telefon: +49 (0) 21 02 309 61-563

Telefax: +49 (0) 21 02 309 61-10

E-Mail: peter.heintzen@mt-ag.com

www.mt-ag.com