

ORACLE®

12 Index NF



Online Operationen

- DROP INDEX ONLINE
- ALTER INDEX UNUSABLE ONLINE
- ALTER INDEX [VISIBLE | INVISIBLE]

ORACLE

Partitioning

- ALTER TABLE ... MOVE PARTITION wird online-fähig
→ globaler Index wird weiter gepflegt
- Asynchrone Pflege eines globales Index während DROP | TRUNCATE einer Partition möglich
- Partieller Index für partitionierte Tabellen
 - Local & Global

ORACLE

3 | Copyright © 2012, Oracle and/or its affiliates. All rights reserved. |

Multiple Indices

auf der gleichen Spaltenmenge

- Möglich, solange sie unterschiedliche Charakteristika haben
 - B-Tree vs. Bitmap
 - Unique vs. Non-Unique
 - Partitioning Strategie

ORACLE

4 | Copyright © 2012, Oracle and/or its affiliates. All rights reserved. |

Oracle Text & Spatial

- Index-Performance für SDO_GEOMETRY
- RDF Semantic
- Near Real-Time Indexing für Text
- Forward Index
- ...

ORACLE

5 | Copyright © 2012, Oracle and/or its affiliates. All rights reserved. |

Optimizing Transaction and Query Performance

Row Format Databases versus Column Format Databases

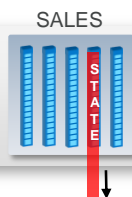
Row



- **Transactions run faster on row format**

- Insert or query a sales order
- Fast processing few rows, many columns

Column



- **Analytics run faster on column format**

- Report on sales totals by state
- Fast accessing few columns, many rows

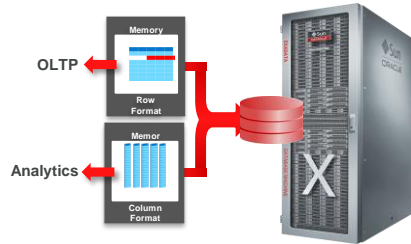
ORACLE

6 | Copyright © 2012, Oracle and/or its affiliates. All rights reserved. |

Oracle In-Memory Design Goals

Target Specification

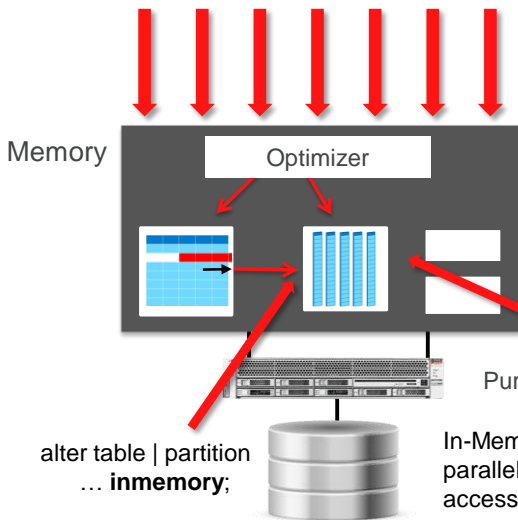
- Integration in existing Infrastructure
 - High Availability
 - Backup & Recovery and Disaster Recovery
- Applications should profit without any change
 - Transparent Usage
 - OLTP & Analytics should benefit
 - No Data Migration
- Ease of Use
 - Simple Definition through SQL



ORACLE

7 | Copyright © 2012, Oracle and/or its affiliates. All rights reserved. |

DB12c In-Memory Database Option



- Ease of Use ? ✓
- Integration in existing Infrastructure ? ✓
- Applications should profit without any change ? ✓
- Is it transactionally consistent ? ✓
- Is it really fast ? ✓

Converts join processing into fast column scans

Scans use super fast SIMD vector instructions

No Analytic Index Overhead

alter table | partition
... inmemory;

In-Memory queries are
parallelized across servers to
access local column data

....
Bloom Filter

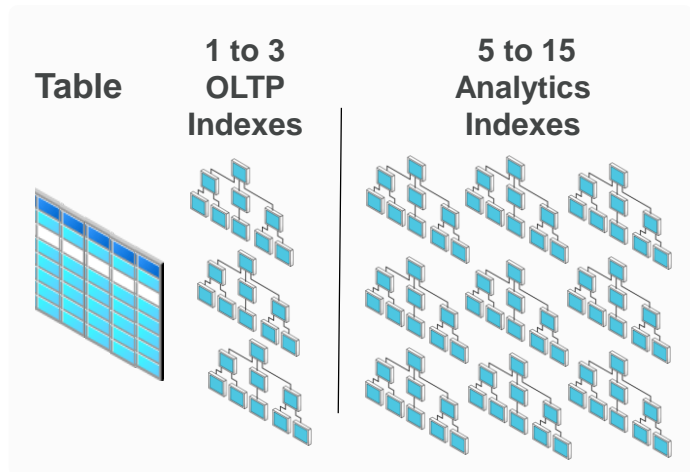
Pure in-memory
columnar processing:
no logging

ORACLE

8 | Copyright © 2012, Oracle and/or its affiliates. All rights reserved. |

OLTP is Slowed Down by Analytic Indexes

- Most Indexes in OLTP (e.g. ERP) databases are only used for analytic queries
- Indexes work well for predictable access patterns both in-memory and on-disk
- Inserting one row into a table requires 10-20 analytic indexes to be updated: **Slow!**

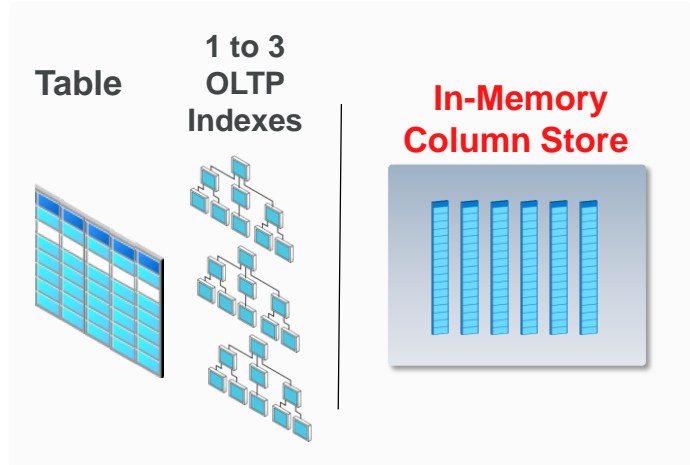


ORACLE

9 | Copyright © 2012, Oracle and/or its affiliates. All rights reserved. |

Column Store Replaces Analytic Indexes

- Analytic queries **100X** faster
- OLTP & batch: **2X - 3X** faster
- No Analytic Index Overhead
- Replaced by Column Store: Near Zero Overhead Updates
- Less Tuning & Administration



ORACLE

10 | Copyright © 2012, Oracle and/or its affiliates. All rights reserved. |

ORACLE®